Performance Comparison in Net Thrust Prediction for a *Scramjet* Engine

Ângelo de Carvalho Paulino[1], Pedro Paulo Batista de Araújo[2], Roberto Yuji Tanaka[2] and Angelo Passaro[1,2]
[1]Aeronautics Institute of Technology (ITA), São José dos Campos/SP - Brazil
[2]Institute for Advanced Studies (IEAv), São José dos Campos/SP - Brazil

***Abstract*— Surrogate selection for hypersonic propulsion remains challenging because method performance may vary across the operational envelope. This work compares a previously trained deep neural network (DNN), evaluated for a fixed geometry, with six interpolation and fitting methods for predicting net thrust (F_{net}) over a feasible scramjet performance envelope subject to the Korkegi criterion: bilinear Q_1, nodal Q_2 and Q_3, complete quadratic P_2, weighted least-squares P_2, and Gaussian radial basis function interpolation. At four central validation points, the DNN achieved the lowest root mean square percentage error (RMSPE, 2.41%) and root mean square error (RMSE); at four edge points, several methods achieved lower RMSE values. No method dominated across regions and error criteria. The results support a complementary, region-aware use of the DNN and classical methods/estimators based on domain location, nodal support, and point coverage rather than aggregate metrics alone.**

***Keywords*— Scramjet, net thrust, nodal interpolation, deep neural network.**

I. INTRODUCTION

Conceptual design and multi-objective optimization of aerospace vehicles, especially air-breathing hypersonic configurations such as scramjets, require the repeated evaluation of aerodynamic and propulsive performance parameters [1]. However, high-fidelity Computational Fluid Dynamics (CFD) simulations remain financially and computationally prohibitive for extensive iterative optimization loops [2], [3]. This computational bottleneck is even more severe in propulsion problems involving hypersonic reactive flows [4], [5].

To overcome this limitation, the literature has established surrogate models as efficient approximations for expensive functions [6], [7]. Traditionally, response surfaces have been constructed using classical local and global approximation methods, including Radial Basis Functions (RBFs), thereby replacing repeated high-cost evaluations within optimization procedures [8], [9]. More recently, deep-learning approaches have received increasing attention. Deep Neural Networks (DNNs) have emerged as alternatives for mapping nonlinear relationships with low-cost inference, thereby reducing the computational burden of repeated evaluations in optimization problems [10]–[12].

Â. de C. Paulino, angeloacp@fab.mil.br; P. P. B. Araújo, araujo.projects@gmail.com; R. Y. Tanaka; A. Passaro. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

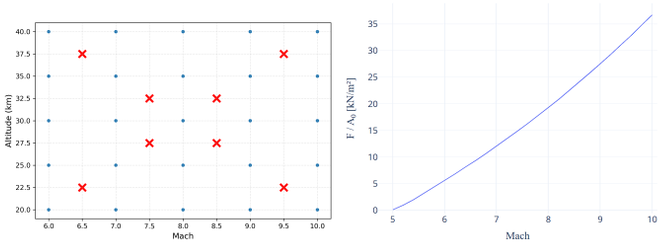
Despite the increasing adoption of DNNs, the unrestricted replacement of classical interpolation methods by neural models still requires caution [13]. In realistic design scenarios, such as scramjets, the feasible search space is not only nonlinear but also bounded by operational and physical constraints [14], [15]. One such constraint is the semi-empirical Korkegi criterion [16], applied to avoid boundary-layer separation induced by shock–boundary-layer interaction. Imposing this limit acts as a feasibility filter that generates a discrete envelope of feasible performance, in which the optimal geometry may change abruptly between neighboring flight conditions [1]. Such changes may introduce local irregularities or discontinuities in the derivatives of the resulting performance surface.

Recent work has shown that favorable global error metrics may conceal regional degradation in DNN predictions when the Mach–altitude domain is non-uniformly represented [13]. Previous scramjet studies have compared surrogate models for design and optimization, including Artificial Neural Networks (ANNs), global polynomial response surfaces, and Kriging models [17], as well as Kriging-based combustor optimization coupled with NSGA-II [18]. However, these studies focused on global surrogate models over design spaces rather than on local nodal interpolation across an operational envelope. To the authors' knowledge, the literature still lacks a direct region-wise comparison between DNN predictions and local mathematical interpolators on a feasibility-constrained scramjet performance surface, explicitly accounting for local nodal support, edge-specific support adaptations, and feasibility-induced irregularities.

This work evaluates methods for estimating the net thrust F_{net} at intermediate points of the operational grid defined by the freestream Mach number, M_0, and geometric altitude, Z (km). The DNN is evaluated for a fixed scramjet geometry selected as the configuration yielding the largest F_{net} at Mach 8 and 30 km altitude. For the local interpolation and fitting methods, the discrete grid available in the SCRJTTD dataset [13] is used to construct the nodal performance surface. For each pair (M_0, Z), the largest F_{net} value among feasible candidate geometries is selected, subject to Korkegi = 1, which indicates non-violation of the Korkegi feasibility criterion [1]. The resulting nodal surface therefore represents a discrete envelope of feasible performance.

The evaluated intermediate points lie between the discrete grid conditions, as shown in Fig. 1a, and

include both central points and points near the domain edges. This distinction is relevant because higher-order local methods depend on the availability and spatial distribution of neighboring nodes and may therefore behave differently near the domain boundaries. In addition, the expected physical behavior of F_{net} as a function of M_0 is nonlinear [13] (Fig. 1b), motivating the evaluation of methods beyond bilinear interpolation. The adopted polynomial methods are described through the relationship between approximation bases, nodal matrices, and interpolation functions, following the finite-element formulation presented by Passaro [19]. In addition, a local weighted least-squares quadratic fit, Gaussian RBF interpolation [20], and the external prediction of a previously trained DNN [13] are considered.



(a) Evaluated intermediate points, marked with red "x", versus available datapoints (blue dots). (b) Expected physical behavior of F_{net}. Source: adapted from [13].

Fig. 1. Evaluation scenario.

The contribution of this work is methodological: it provides a region-aware comparison between a previously trained DNN and classical interpolation and fitting methods over a feasibility-constrained scramjet performance envelope. By separating central and edge conditions and accounting for nodal support and point coverage, the analysis evaluates both predictive accuracy and regional variations in DNN reliability. The classical methods/estimators are thus used both as alternative estimators and as diagnostic references. The mathematical formulations and comparison metrics are presented next.

II. MATHEMATICAL FORMULATION

A. Discrete nodal support surface

For each discrete condition $(M_{0,i}, Z_j)$, the nodal value used by the interpolation or estimation methods is defined as the largest feasible F_{net} among the dataset candidates. Let:

$$\mathcal{C}_{ij} = \{k : M_{0,k} = M_{0,i},\ Z_k = Z_j,\ \text{Korkegi}_k = 1\} \tag{1}$$

be the set of feasible candidates associated with the discrete cell $(M_{0,i}, Z_j)$. In Eq. (1), k is the candidate index in the dataset, $M_{0,k}$ and Z_k are, respectively, the Mach number and altitude of that candidate, and $\text{Korkegi}_k = 1$ identifies the feasibility condition used in the selection. When $\mathcal{C}_{ij} \neq \varnothing$, the nodal value is defined as (Eq. (2)):

$$F^*_{ij} = F^*(M_{0,i}, Z_j) = \max_{k \in \mathcal{C}_{ij}} F_{\text{net},k}. \tag{2}$$

where F^*_{ij} is the best nodal value for F_{net} selected in cell $(M_{0,i}, Z_j)$, and $F_{\text{net},k}$ is the net thrust of candidate k. This rule reproduces the selection implemented in the code: Korkegi $= 1$ is retained, and the largest F_{net} is selected in each cell, corresponding to the condition-specific optimal geometry. Since this geometry may vary from cell to cell, the surface F^* represents a performance envelope and may exhibit derivative discontinuities between neighboring cells; this feature must be considered when interpreting interpolation errors. From these nodal values, each method m estimates:

$$\widehat{F}_{(m)}(M_0^*, Z^*) \tag{3}$$

at intermediate points of the domain. In Eq. (3), (M_0^*, Z^*) is the query point and $\widehat{F}_{(m)}$ is the estimate produced by method m. The altitude Z is used to compute P_0 and T_0 from the standard atmosphere [21], without interpolation of these quantities. The methods described below operate only on the nodal values of F_{net}.

B. Standard matrix nodal form

Consider a polynomial basis written as a column vector, as in Eq. (4):

$$\mathbf{p}(\xi,\eta) = \begin{bmatrix} P_1(\xi,\eta) & P_2(\xi,\eta) & \cdots & P_n(\xi,\eta) \end{bmatrix}^T \in \mathbb{R}^{n\times 1} \tag{4}$$

where ξ and η are local coordinates, $P_n(\xi,\eta)$ is the n-th basis function, and n is the number of basis terms. Since $\mathbf{p}(\xi,\eta)$ is defined as a column vector, its transpose $\mathbf{p}(\xi,\eta)^T$ is used to form the inner product with the coefficient vector of the approximation $\mathbf{a}$ (Eq. (5)):

$$F(\xi,\eta) = \mathbf{p}(\xi,\eta)^T\mathbf{a}, \qquad \mathbf{a} \in \mathbb{R}^{n\times 1} \tag{5}$$

where $F(\xi,\eta)$ is the approximation of the quantity of interest at the local point (ξ,η). Evaluating the basis at the n nodal points defines the nodal matrix:

$$[P_n] = \begin{bmatrix} \mathbf{p}(\xi_1,\eta_1)^T \\ \mathbf{p}(\xi_2,\eta_2)^T \\ \vdots \\ \mathbf{p}(\xi_n,\eta_n)^T \end{bmatrix} \in \mathbb{R}^{n\times n} \tag{6}$$

In Eq. (6), (ξ_i,η_i) is the i-th nodal point in the reference system, and each row of $[P_n]$ contains the polynomial basis evaluated at that point. The vector of nodal values $\mathbf{F}_n$, Eq. (7), satisfies:

$$\mathbf{F}_n = [P_n]\mathbf{a}. \tag{7}$$

where $\mathbf{F}_n = [F_1, F_2, \ldots, F_n]^T$ collects the F_{net} values at the nodal points used by the method. When $[P_n]$ is square and nonsingular, one obtains:

$$\mathbf{a} = [P_n]^{-1}\mathbf{F}_n. \tag{8}$$

The nonsingularity condition in Eq. (8) ensures that the nodal values uniquely determine the basis coefficients. Substituting Eq. (8) into Eq. (5) gives Eq. (9):

$$F(\xi,\eta) = \mathbf{p}(\xi,\eta)^T [P_n]^{-1} \mathbf{F}_n. \tag{9}$$

Now, defining:

$$\mathbf{N}(\xi,\eta)^T = \mathbf{p}(\xi,\eta)^T [P_n]^{-1}, \tag{10}$$

the nodal form is obtained by combining Eq. (9) and Eq. (10):

$$F(\xi,\eta) = \mathbf{N}(\xi,\eta)^T \mathbf{F}_n. \tag{11}$$

In Eqs. (10) and (11), $\mathbf{N}$ is the vector of interpolation functions. For exact nodal interpolations, these functions satisfy $N_i(\xi_j,\eta_j) = \delta_{ij}$, where δ_{ij} is the Kronecker delta: $\delta_{ij} = 1$ when $i = j$ and $\delta_{ij} = 0$ otherwise. This condition means that the i-th interpolation function is equal to one at its own node and zero at all other nodes, according to the finite-element nodal formulation [19]. In the method notation adopted in this work, this exact nodal form is used by methods "a" (Q_1 interpolation), "b" (Q_2 interpolation), "c" (Q_3 interpolation), and "d" (complete quadratic P_2 interpolation). Method "e" denotes the local weighted least-squares quadratic fit; it preserves the same algebraic structure through generalized shape functions, but without the strict nodal property. Method "f" denotes Gaussian RBF interpolation, which can also be written in generalized nodal form, as shown next.

C. Summary of the methods

The chosen methods are summarized in Table I. Throughout the text, methods "a"–"f" denote six interpolation or estimation procedures applied to the nodal surface $F^*(M_0, Z)$, while "DNN" denotes the prediction obtained from the previously trained neural model. The polynomial methods use local coordinates (ξ,η) obtained by affine mapping or by centering and scaling the grid around the target point. The Q_1, Q_2, and Q_3 bases are tensor-product bases, whereas P_2 is a complete quadratic basis of total degree two [19].

Method "a" uses the four vertices of the rectangular cell containing the target point and combines them with the standard bilinear weights. Method "b" extends this idea to a 3×3 grid and determines a tensor-product Q_2 polynomial from nine nodal values; near edges, the grid is shifted asymmetrically to remain inside the available domain. Method "c" uses the analogous 4×4 grid and a Q_3 tensor-product basis, increasing polynomial order and nodal support when enough neighboring points are available. Method "d" uses the complete quadratic space P_2 and thus requires six geometrically unisolvent support points; in practice, the selected grid must yield a full-rank nodal matrix so that the quadratic polynomial is uniquely determined. Method "e" uses the same P_2 basis, but solves an overdetermined weighted least-squares problem with at least nine local nodes. The weights are based on inverse squared distance to the target point, increasing the influence of nearby nodes and making the method a local fit rather than an exact nodal interpolator.

TABLE I. Summary of the evaluated methods.

Method	Basis/model	Support	Main characteristic
a	$Q_1 = \langle 1, \xi, \eta, \xi\eta \rangle$	4 cell vertices	Bilinear nodal
b	$Q_2 = \langle \xi^i \eta^j \rangle_{i,j=0}^2$	3×3 grid	Biquadratic tensor-product
c	$Q_3 = \langle \xi^i \eta^j \rangle_{i,j=0}^3$	4×4 grid	Bicubic tensor-product
d	$P_2 = \langle 1, \xi, \eta, \xi^2, \xi\eta, \eta^2 \rangle$	6 nodes	Complete quadratic
e	P_2 WLS	≥ 9 local nodes	Weighted quadratic local fit
f	$\phi(r) = \exp[-(r/\varepsilon)^2]$	available nodes	Gaussian RBF
DNN	$f_\theta(\mathbf{x})$	n/a	External DNN prediction

Method "f" uses a Gaussian RBF, $\phi(r) = \exp[-(r/\varepsilon)^2]$, where ϕ is the radial kernel, r is the normalized Euclidean distance between the query point and a support node in the (M_0, Z) plane [20]. The shape parameter $\varepsilon > 0$ controls the radial decay, that is, how rapidly the influence of a support node decreases with distance; smaller values produce more localized functions, whereas larger values produce smoother and more global influence. In this work, ε is scaled by a constant factor of 1.5 relative to the median nearest-neighbor distance to balance localization and smoothness. Tikhonov regularization is employed to improve the numerical conditioning of the RBF linear system [20], [22]. Specifically, regularization is triggered whenever the condition number of the system exceeds the maximum threshold of 10^{12}. To obtain acceptable conditioning while limiting regularization, the candidates $[0, 10^{-12}, 10^{-10}, 10^{-8}, 10^{-6}]$ are tested sequentially.

Finally, the DNN prediction is treated as an external model evaluation: the trained mapping f_θ is evaluated for the fixed geometry under analysis, where f denotes the mapping function and θ represents its trained weights and biases. No retraining is performed within the present comparison, and the architecture and parameters are used exactly as in [13]. The prediction is given by Eq. (12):

$$\widehat{F}_{(\mathrm{DNN})} = f_\theta(h_3, y_0, y_1, y_2, T_0, P_0, M_0). \tag{12}$$

where $\widehat{F}_{(\mathrm{DNN})}$ is the predicted net thrust and $f_\theta(\cdot)$ is the DNN mapping parameterized by the trained weights and biases θ. The input arguments comprise the combustor height (h_3), the shock intensities at the first, second, and third ramps (y_0, y_1, and y_2), and the freestream properties given by the Mach number (M_0), pressure (P_0), and temperature (T_0). The flight altitude is converted into T_0 and P_0 through the standard atmosphere model [21]; thus, altitude does not appear explicitly as an input of

the mapping in Eq. (12). The fixed geometry adopted in this study corresponds to the configuration selected for yielding the largest F_{net} at $M_0 = 8$ and $Z = 30$ km, a condition located at the center of the investigated domain. Predictions for this fixed geometry are evaluated at each of the eight points marked in Fig. 1a. None of these points was included in the DNN training set.

III. PERFORMANCE METRICS

For each point i and method (m), the signed error is defined in Eq. (13) as:

$$e_{i,(m)} = \widehat{F}_{i,(m)} - F_{\text{ref},i}. \tag{13}$$

where $\widehat{F}_{i,(m)}$ is the value estimated by method (m) at point i, and $F_{\text{ref},i}$ is the reference value at the same point. The values $F_{\text{ref},i}$ come from the computational optimization software used in [1], which is based on reliable physical models accounting for Mach number and flight altitude [13]. Positive values indicate overestimation; negative values indicate underestimation. The pointwise absolute error is defined in Eq. (14) as:

$$AE_{i,(m)} = |e_{i,(m)}|. \tag{14}$$

where $AE_{i,(m)}$ expresses the absolute discrepancy at point i for method (m). The pointwise absolute percentage error is defined below, Eq. (15):

$$APE_{i,(m)} = 100\frac{AE_{i,(m)}}{F_{\text{ref},i}}, \qquad F_{\text{ref},i} > 0. \tag{15}$$

so that $APE_{i,(m)}$ measures the absolute error relative to the magnitude of the reference value. If, for any given datapoint, $F_{\text{ref},i} = 0$, the percentage error is mathematically undefined. Thus, the point remains in the absolute metrics but is excluded from percentage statistics. At $M_0 = 6.5$, $Z = 37.5\,\text{km}$, the reference value is assigned as $F_{\text{ref},i} = 0$ by the optimization software because the evaluated solution violates the Korkegi criterion. This value is therefore an infeasibility flag rather than a physically meaningful zero-thrust operating condition. For a set $S_{(m)}$ of valid points for a method, the following aggregate metrics are used:

$$\text{RMSE}_{(m)} = \sqrt{\frac{1}{|S_{(m)}|}\sum_{i\in S_{(m)}} e^2_{i,(m)}}, \tag{16}$$

$$\text{RMSPE}_{(m)} = \sqrt{\frac{1}{|S^+_{(m)}|}\sum_{i\in S^+_{(m)}} \left(APE_{i,(m)}\right)^2}. \tag{17}$$

In Eqs. (16)–(17), $|S_{(m)}|$ is the number of points evaluated by method (m), $S^+_{(m)}$ is the subset of points with $F_{\text{ref},i} > 0$, $\text{RMSE}_{(m)}$ is the global root mean square error, and $\text{RMSPE}_{(m)}$ is the root mean square percentage error. Limitations of RMSE in model comparisons are discussed in the literature [23]. Since methods "b" and "c" are not applied to the same set of points, comparing their aggregate indicators must account for point coverage.

IV. RESULTS AND DISCUSSION

Table II summarizes point coverage by method. Methods "a", "d", "e", "f", and DNN are evaluated at the eight intermediate points. Method "b" is applied only to the four edge points, whereas method "c" is applied only to the four central points. This coverage difference arises from the method-specific grid definitions used in the calculation cell and must be considered when comparing aggregate metrics. As mentioned previously, the reference values used in the metrics of this section were obtained with the computational optimization software employed in [1].

TABLE II. Point coverage by method.

Method	Edge	Central	Total
a	4	4	8
b	4	0	4
c	0	4	4
d	4	4	8
e	4	4	8
f	4	4	8
DNN	4	4	8

Figure 2 directly compares the estimated and reference normalized values. The point $(M_0, Z) = (9.5, 22.5\ \text{km})$ dominates the absolute scale because it concentrates the largest reference value and the largest absolute discrepancies. This figure shows that not all methods yield close agreement with the reference, especially at this point. By contrast, the estimates are more closely clustered at $(M_0, Z) = (6.5, 37.5\ \text{km})$.

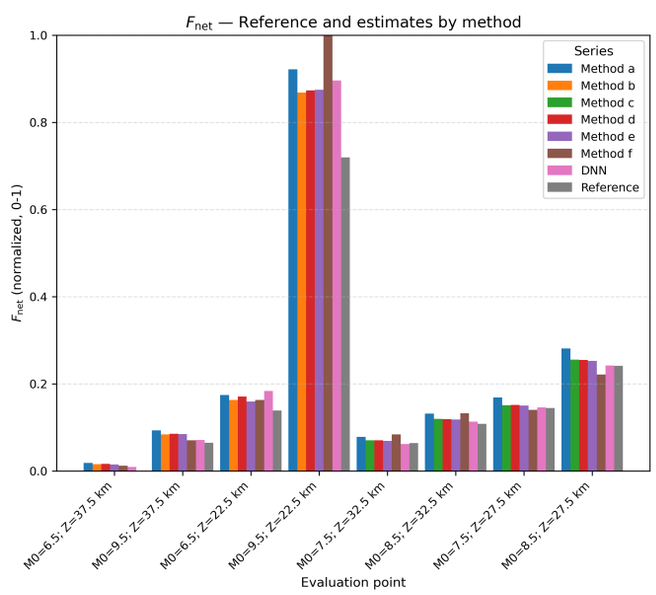


Fig. 2. Normalized reference net thrust and estimated values per method.

Figure 3 summarizes $\text{RMSPE}_{(m)}$ by point class. At the edge points, methods "b", "e", "f", and DNN produce similar values, ranging from 21.90% to 22.62%, whereas method "a" yields the largest error (32.53%). At the

central points, the DNN achieves the lowest $\text{RMSPE}_{(m)}$ (2.41%). This central advantage may be related to the proximity of these points to the condition $(M_0, Z) = (8, 30\text{ km})$ used to select the fixed geometry, although the present results do not establish this relationship causally.

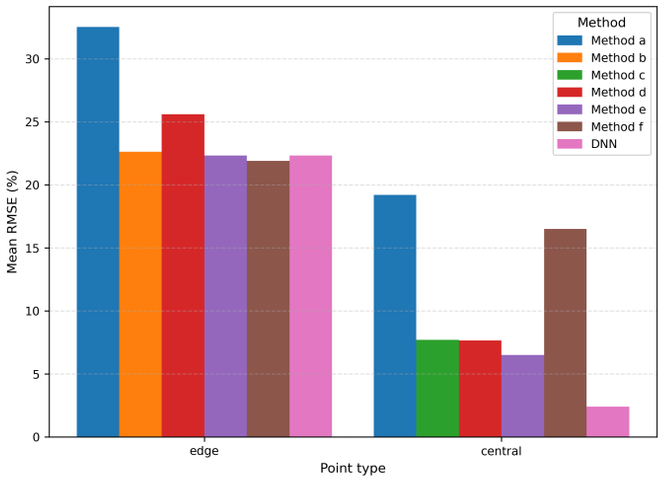


Fig. 3. Root mean square percentage error by point class.

Table III reports both percentage and absolute errors by point class. Methods "b" and "c" are evaluated on different subsets: method "b" is restricted to the edge points, whereas method "c" is restricted to the central points. Their metrics must therefore be interpreted according to this difference in coverage. In addition, the edge-point RMSE includes the condition with $F_{\text{ref}} = 0$, whereas the corresponding RMSPE excludes that condition because its percentage error is undefined.

TABLE III. RMSE and RMSPE by point class.

Method	**Edge** $\text{RMSPE}_{(m)}$	**Central** $\text{RMSPE}_{(m)}$	**Edge** $\text{RMSE}_{(m)}$	**Central** $\text{RMSE}_{(m)}$
a	32.53	19.19	1166.02	304.31
b	22.62	–	856.90	–
c	–	7.70	–	115.01
d	25.60	7.65	895.62	111.71
e	22.33	6.51	889.27	95.94
f	21.90	16.50	1581.86	213.15
DNN	22.33	2.41	1023.06	33.07

Method "a" produces the largest RMSPE in both point classes and the largest central RMSE, indicating that the bilinear basis is insufficient to represent the local variation of the performance surface. Method "b" reduces both edge metrics relative to method "a", although its support must be shifted asymmetrically near the domain boundaries. In the central subset, method "c", the highest-order polynomial interpolator considered, produces results comparable to method "d", but it cannot be assessed at the edge points.

The comparison between methods "d" and "e" illustrates the effect of replacing exact nodal interpolation with local approximation. Method "d" constructs a complete P_2 interpolant from six unisolvent support points and is therefore sensitive to the selected stencil. Method "e" uses an overdetermined weighted least-squares fit and yields lower RMSPE and RMSE in both point classes.

Method "f" yields the lowest edge RMSPE (21.90%) but the largest edge RMSE (1581.86). This contrast must be interpreted in light of the different sets used by the two metrics: the zero-reference edge point contributes to RMSE but is excluded from RMSPE. Consequently, the percentage result alone does not represent the method's behavior over all edge conditions. Because Gaussian RBF interpolation depends on the shape parameter, regularization, and coordinate scaling [20], [22], this result also motivates a sensitivity analysis of each.

The DNN exhibits the strongest regional contrast. It achieves the lowest central RMSPE (2.41%) and RMSE (33.07), but its edge RMSE increases to 1023.06, exceeding the corresponding values obtained by methods "b", "d", and "e". Thus, its relative advantage is confined to the central subset and does not extend uniformly to the domain boundaries. One possible contributing factor is the variation of the combustor-entrance temperature T_3 at the edge conditions, whereas the DNN was trained using a fixed value of T_3. This hypothesis should be assessed by extending the training data over a range of T_3 values.

Wall-clock time was not measured; computational cost is compared qualitatively in terms of online operations. Method "a" uses four-node bilinear interpolation; "b" and "c" solve 9- and 16-node polynomial systems; "d" solves a six-node P_2 system after stencil checks; "e" performs weighted least squares; and "f" solves a dense RBF system with conditioning and regularization tests. DNN inference requires a single forward pass, excluding training. Thus, "a" has the lowest nominal workload among classical methods/estimators and "f" the highest; DNN cost depends on architecture and hardware. All seven predictors avoid repeating the metaheuristic optimization used to generate the reference solutions in [1], which involved 20 runs of each metaheuristic per problem, each with a stopping criterion of 10^4 objective-function calls, and thus require orders of magnitude fewer iterative evaluations.

V. CONCLUSION

This work compared a previously trained DNN with classical local interpolation and fitting methods over a feasible scramjet performance envelope constrained by the Korkegi criterion. Its main scientific contribution is a region-aware comparison that distinguishes central and edge behavior and relates prediction errors to nodal support and point coverage, allowing local estimators to serve as diagnostic references for DNN reliability. No method dominated all error criteria; performance depended on the metric, domain location, and local nodal structure.

At the central points, the DNN achieved the lowest $\text{RMSPE}_{(m)}$ and $\text{RMSE}_{(m)}$ among the methods evaluated at those points, indicating better predictive performance near the Mach-altitude condition used to select the fixed geometry. This advantage did not extend to the edges: the DNN edge $\text{RMSE}_{(m)}$ (1023.06) exceeded

those of methods "b" (856.90), "d" (895.62), and "e" (889.27). Thus, aggregate or central-region metrics may conceal regional degradation, as observed at the evaluated high-Mach, low-altitude conditions.

Taken together, these findings support a complementary, region-aware strategy that exploits the DNN's central-region accuracy and low-cost inference while using local estimators as alternative predictions and reliability diagnostics near domain boundaries. This interpretation is limited by the eight intermediate validation points and the fixed geometry used for DNN evaluation. The results provide evidence of region-dependent behavior at the sampled points but do not demonstrate generalization throughout the feasible envelope or to other geometries. Future work should increase the number and spatial coverage of validation points, consider additional geometries and variations in T_3, and examine hybrid strategies combining DNN predictions with local interpolation diagnostics. Alternative neural architectures may also be investigated to determine whether they improve the accuracy and efficiency of nonlinear scramjet performance prediction, including self-attention/transformer networks [24], Interpolating Neural Networks (INNs) [25], and Kolmogorov–Arnold Networks (KANs) [26], [27].

REFERENCES

[1] P. P. B. Araújo, R. Y. Tanaka, C. A. Silva, A. Passaro, P. G. P. Toro, "Multi-objective optimization of a hypersonic airbreathing vehicle," *Physics of Fluids*, vol. 36, no. 2, p. 026120, 2024.

[2] G. Catalani, S. Agarwal, X. Bertrand, F. Tost, M. Bauerheim, J. Morlier, "Neural fields for rapid aircraft aerodynamics simulations," *Scientific Reports*, vol. 14, no. 1, p. 25496, 2024.

[3] M. Schouler, A. Belme, P. Cinnella, "Comparison of multi-fidelity surrogate models for multi-objective aerodynamic optimization in turbomachinery under extreme cost imbalance," *Advanced Modeling and Simulation in Engineering Sciences*, vol. 12, no. 1, p. 35, 2025.

[4] M. Krügener, J. F. Zapata Usandivaras, M. Bauerheim, A. Urbano, "Coaxial-injector surrogate modeling based on Reynolds-averaged Navier–Stokes simulations using deep learning," *Journal of Propulsion and Power*, vol. 38, no. 5, pp. 783–798, 2022.

[5] S. Nasir, S. Sahibzada, F. S. Malik, "Physics-informed neural networks for predicting combustion dynamics in rocket engine chambers," *International Journal of Innovative Research in Computer Science and Technology*, vol. 13, no. 4, pp. 41–44, 2025.

[6] M. K. I. Zim, A. Singh, L. Sandhu, "Surrogate modeling with artificial neural networks for efficient optimization," in *2025 IEEE International Conference on Intelligent Signal Processing and Effective Communication Technologies*, 2025, pp. 1–6.

[7] M. Alswaitti, K. Siddique, S. Jiang, W. Alomoush, A. Alrosan, "Dimensionality reduction, modelling, and optimization of multivariate problems based on machine learning," *Symmetry*, vol. 14, no. 7, p. 1282, 2022.

[8] D. Cao G. Bai, "DNN-based surrogate modeling-based feasible performance reliability design methodology for aircraft engine," *IEEE Access*, vol. 8, pp. 229 201–229 218, 2020.

[9] J.-X. Leng, Y. Feng, W. Huang, Y. Shen, Z.-G. Wang, "Variable-fidelity surrogate model based on transfer learning and its application in multidisciplinary design optimization of aircraft," *Physics of Fluids*, vol. 36, no. 1, p. 017131, 2024.

[10] X. Liu, S. Yang, H. Sun, Z. Wang, X. Guan, Y. Gu, Y. Wang, "Review of deep learning-based aerodynamic shape surrogate models and optimization for airfoils and blade profiles," *Physics of Fluids*, vol. 37, no. 4, p. 041304, 2025.

[11] K. Shukla, V. Oommen, A. Peyvan, M. Penwarden, N. Plewacki, L. Bravo, A. Ghoshal, R. M. Kirby, G. E. Karniadakis, "Deep neural operators as accurate surrogates for shape optimization," *Engineering Applications of Artificial Intelligence*, vol. 129, p. 107615, 2024.

[12] D. Pereira, F. Afonso, F. Lau, "End-to-end deep-learning-based surrogate modeling for supersonic airfoil shape optimization," *Aerospace*, vol. 12, no. 5, p. 389, 2025.

[13] Â. d. C. Paulino, P. P. B. d. Araújo, R. Y. Tanaka, A. Passaro, "Effects of non-uniform data coverage on deep neural network-based scramjet performance prediction," *Journal of Aerospace Technology and Management*, vol. 18, 2026. [Online]. Available: https://doi.org/10.1590/jatm.v18.1439

[14] V. Esfahanian, M. J. Izadi, H. Bashi, M. Ansari, A. Tavakoli, M. Kordi, "Aerodynamic shape optimization of gas turbines: a deep learning surrogate model approach," *Structural and Multidisciplinary Optimization*, vol. 67, p. 2, 2024.

[15] S.-H. Qin A.-M. Yang, "Aerodynamic optimization of a coaxial rotor system using a deep learning-based multi-fidelity surrogate model," *Engineering Applications of Computational Fluid Mechanics*, vol. 19, no. 1, p. 2528120, 2025.

[16] R. H. Korkegi, "Comparison of shock-induced two- and three-dimensional incipient turbulent seperation," *AIAA Journal*, vol. 13, no. 4, pp. 534–535, 1975.

[17] R. R. Shenoy, T. G. Drozda, E. L. Axdahl, R. A. Baurle, "Numerical investigation and optimization of a flushwall injector for scramjet applications at hypervelocity flow conditions," in *JANNAF Propulsion Meeting*, no. NF1676L-31988. Dayton, OH, United States: NASA Langley Research Center, Jun. 2019, NTRS Document ID: 20200002676. [Online]. Available: https://ntrs.nasa.gov/citations/20200002676

[18] J. Yu, Z. Zhou, J. Peng, T. Tang, W. Yang, Y. Yang, H. Wang, "Optimization design of combustion chamber configuration parameters for scramjet engines based on surrogate models," *Chinese Journal of Theoretical and Applied Mechanics*, vol. 56, no. 11, pp. 3359–3370, 2024.

[19] A. Passaro, "Método dos elementos finitos: Aula 7 – parte II," Material didático da disciplina TE-294, PG-CTE, 2026.

[20] M. D. Buhmann, *Radial Basis Functions: Theory and Implementations*, ser. Cambridge Monographs on Applied and Computational Mathematics. Cambridge: Cambridge University Press, 2003, no. 12.

[21] National Oceanic and Atmospheric Administration, National Aeronautics and Space Administration, United States Air Force, "U.S. Standard Atmosphere, 1976," U.S. Government Printing Office, Washington, DC, Tech. Rep. NOAA-S/T 76-1562; NASA-TM-X-74335, 1976, NTRS Document ID: 19770009539. [Online]. Available: https://ntrs.nasa.gov/citations/19770009539

[22] G. E. Fasshauer, *Meshfree Approximation Methods with MATLAB*, ser. Interdisciplinary Mathematical Sciences. Singapore: World Scientific, 2007, vol. 6.

[23] T. Chai R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)? – arguments against avoiding RMSE in the literature," *Geoscientific Model Development*, vol. 7, no. 3, pp. 1247–1250, 2014.

[24] Z. Niu, G. Zhong, H. Yu, "A review on the attention mechanism of deep learning," *Neurocomputing*, vol. 452, pp. 48–62, 2021.

[25] C. Park, S. Saha, J. Guo, H. Zhang, X. Xie, M. A. Bessa, D. Qian, W. Chen, G. J. Wanger, J. Cao, T. J. R. Hughes, W. K. Liu, "Unifying machine learning and interpolation theory via interpolating neural networks," *Nature Communications*, vol. 16, no. 1, p. 8753, 2025.

[26] I. A. R. Oliveira, D. A. S. Mendes, H. F. C. Velho, E. S. Pereira, "Solving backward heat transfer by physics-informed machine learning," *Computational and Applied Mathematics*, vol. 45, p. 421, 2026.

[27] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljačić, T. Y. Hou, M. Tegmark, "KAN: Kolmogorov–Arnold networks," in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: https://proceedings.iclr.cc/paper_files/paper/2025/hash/afaed89642ea100935e39d39a4da602c-Abstract-Conference.html