/E/

In-Context Memory Along Airfoil Polars

FoilCORE

6394-parameter causal surrogate; teacher-forced benchmarks against NeuralFoil

Avneh Singh Bhatia* Joshua Selvaraj

*Lead author and lead researcher.

Exea Labs

avnehb@gmail.com

Abstract

Fast lift and drag estimates along airfoil polars matter for design sweeps, but high-fidelity tools are slow at scale. We show that a *causal* sequence model with only 6394 trainable parameters can match or beat two public NeuralFoil [15, 16] checkpoints (`xxsmall` and `xxxlarge`) on the same withheld BigFoil-derived test rows [11] when each method sees identical preprocessing and the decoder is fed ground-truth past coefficients along the polar (teacher forcing). On the first 4096 rows of that split, pooled mean C_ℓ MAE is 0.051 54 and C_d MAE is 0.002 99, below both baselines at much smaller width. Source polar tables are not bundled with the paper; access is described in Section 3.

FoilCORE walks the polar in angle-of-attack order. Geometry and $(\alpha, Re, M, C_\ell, C_d)$ tokens embed at width d=8 [3]. A masked causal trunk, path nonlinearities, gated outer products [4], and a two-output head predict the next step, so the curve is an ordered sweep rather than unrelated (α, Re) samples.

Greedy decoding with K=0 raises validation MAE versus teacher forcing; Section 7.4 describes a NeuralFoil `xxxlarge` warmstart for open-loop use. Scope and caveats are in Section 9.

Keywords: airfoil polars; sequence models; low-rank mixing; small neural surrogates; benchmarking.

1 Introduction

Aerodynamic coefficient prediction is a computational bottleneck in aerospace and wind-energy applications. Established tools ranging from coupled viscous integral solvers [7] to Reynolds-averaged Navier–Stokes (RANS) codes and experimentation remain informative but comparatively expensive during large-scale parametric surveys.

What we test.

(1) Sequential order.
Causal ordering improves parameter efficiency for reconstructing the polar on matched rows and preprocessing compared with independent pointwise regression.

(2) Mixing without width.
Gated outer products [4] add multiplicative interactions without growing trunk width.

(3) Near stall.
Under teacher forcing, errors stay minimal in a narrow band around peak lift even at extreme compression.

NeuralFoil alongside this work. NeuralFoil [15, 16] is the de facto benchmark for neural airfoil analysis surrogates. Section 7.4 explains how to attach NeuralFoil `xxxlarge` coefficients for open-loop sweeps.

Contributions.

1. Compact causal surrogate: architecture through training (Tables 2–3).
2. Ordered-polar motivation plus withheld-row benchmarks versus NeuralFoil and tail diagnostics.
3. Teacher forcing versus greedy rollout, NeuralFoil-`xxxlarge` coefficient prefixes, and latency sketches.
4. Qualitative AF104K (held-out airfoil) coordinate hold-out (Figure 21).

How headline numbers are computed. Main tables use the same withheld BigFoil test rows [11] and preprocessing for both models, with teacher forcing (true past (C_ℓ, C_d) in the decoder). Tabulated runs use the first 4096 rows in fixed disk order (Appendix D). Autoregressive tests use $K{=}0$; Section 7.4 covers longer prefixes for deployment-style sweeps.

2 Related Work

2.1 Surrogate models, corpora, and panel lineage

Panel and integral methods remain standard for economical polar estimation [7]. Learned regressors map airfoils or flows to forces [14, 6, 9]. Libraries such as NeuralFoil [15, 16] show that panel-style polars can be fit at scale; we use those checkpoints as *pointwise* baselines and, in Section 7.4, as coefficient anchors for open-loop decoding. Public datasets support systematic evaluation [2, 5]. Physics-informed networks add PDE-based losses [12]. Broader surrogate practice is reviewed by [8]. Our labels are tabulated polars from the same panel-style pipeline as the training data (XFOIL-style conventions [7]). We score models on stored curves; we do not rerun XFOIL inside the figure scripts.

2.2 Causal polar models versus pointwise MLPs

Most learned surrogates still pose each (α, Re) row as an isolated regression: a vanilla MLP over geometry and scalars does not, by itself, tie station α_i to α_{i-1}. Yet viscous polars are ordered—boundary-layer state carries forward along the sweep, so coefficients trace a trajectory in α. Foil-

CORE mirrors that traversal with causal masking (token t attends only backward along the sequence). Path nonlinearities and gated outer products add multiplicative mixing without MLP-scale trunk width.

2.3 Architectural precedents and baseline contrasts

Transformer attention [18] mixes tokens. FoilCORE keeps width small ($d = 8$ [3]), adds path residuals and gated outer products [4], and uses the same low-rank second-order flavour as factorization machines [13]. Tokens form an $(\alpha, Re, M, C_\ell, C_d)$ sequence after a geometry prefix; Tables 5–10 use teacher forcing.

3 Problem Setup and Data

3.1 Task Definition

Each example is an airfoil polar curve. The input contains geometry tokens and polar tokens. The geometry tokens encode a resampled airfoil surface. The polar tokens contain normalized operating-condition and coefficient values:

$$[\alpha,\ Re,\ M,\ C_\ell,\ C_d].$$

The model outputs two values at polar positions:

$$\hat{y}_t = [\hat{C}_{\ell,t},\ \hat{C}_{d,t}].$$

Training and reported evaluation use a next-token teacher-forced setup: the model is run on the embedded sequence, and predictions at position t are matched to polar targets at the following position where appropriate in the sequence-loss implementation. The paper reports only the polar head.

3.2 Input Representation

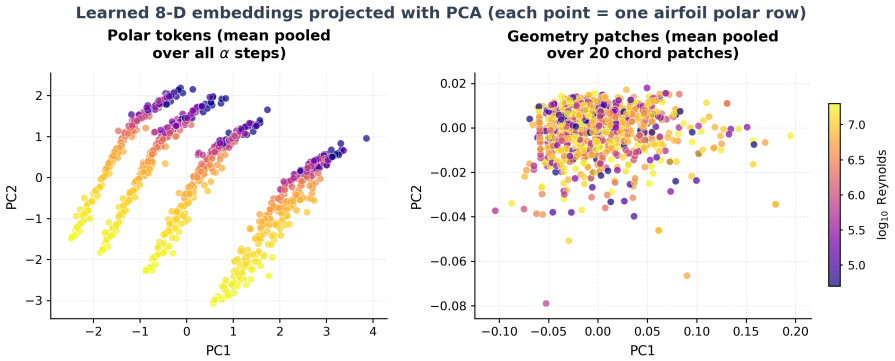
The polar input dimension is $D_{\text{polar}} = 5$: angle of attack, Reynolds number, Mach number, lift coefficient, and drag coefficient. These values are z-scored using training statistics stored in the checkpoint. For the evaluated checkpoint, the stored means are

$$(0.4164,\ 6.7327{\times}10^6,\ 0.2027,\ 0.3132,\ 0.04247),$$

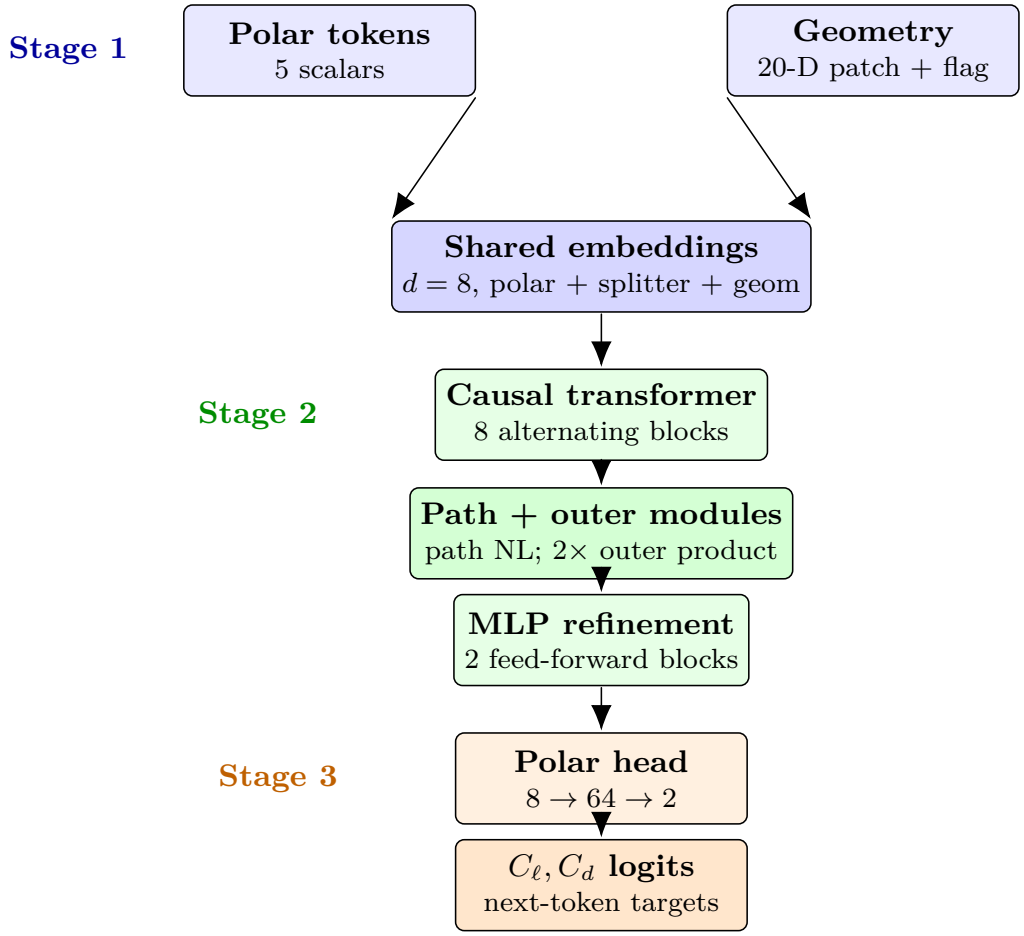
and the stored standard deviations are

$$(10.9011,\ 6.2945{\times}10^6,\ 0.1842,\ 0.9216,\ 0.05757).$$

The geometry is resampled to 200 coordinate points by arc length, chord-normalized to $x \in [0, 1]$, flattened, and grouped into $N_{\text{geom}} = 20$ patches of 10 (x, y) pairs each. Thus each raw geometry patch has $D_{\text{geom}} = 20$ scalar coordinate values. A binary surface flag is appended before projection, giving $D_{\text{geom}} + 1 = 21$ geometry inputs per patch. The first half of patches use surface flag 0 and the second half use surface flag 1, following the repository's half-perimeter heuristic. A sinusoidal positional encoding is added to polar tokens using angle of attack and to geometry tokens using chord-position index. A learned splitter embedding separates the geometry prefix from the polar sequence.



(a) PCA of mean-pooled 8-D polar and geometry embeddings; colour encodes $\log_{10} Re$ and each marker is one record from the withheld test split.



(b) Symbolic pipeline: embeddings, causal trunk with path residuals and gated outer inserts, shallow MLPs, and polar decoder head.

Figure 1: **(a)** Mean-pooled eight-dimensional embeddings coloured by $\log_{10}$ Re. **(b)** Stage 1–3 stack: embed, causal trunk with path and outer blocks, shallow MLPs, polar head (Sections 3–4).

Figure 1(**a**) pools $\mathbb{R}^8$ tokens by PCA, coloured by $\log_{10}\mathrm{Re}$; panel (**b**) sketches Stages 1–3 (Sections 3–4).

3.3 Dataset Description

The benchmark is a withheld *test* split of 47 104 polar-curve rows (airfoil polars and chord-normalized coordinates) from the BigFoil compilation [11], curated by Mike Quayle.[1] Regenerated paper metrics use the first 4096 rows in fixed on-disk order (default row cap in the scripts; reproducible shortcut rather than a stratified draw). Every tabulated row requires successful FoilCORE teacher-forced inference and both NeuralFoil baselines on mutually valid polar lengths.

Training (329 719 rows), validation (94 205), and test splits follow polar-curve records; empirical ranges are in Table 1. Raw splits are not bundled with the artifact (Appendix D); leakage and coverage caveats appear in Section 9.

3.4 Preprocessing

Coordinates are parsed from JSON strings, resampled to 200 points by arc length, chord-normalized, and grouped into 20 flattened patches. Polar arrays are parsed from JSON strings and converted to token arrays $[\alpha, Re, M, C_\ell, C_d]$. Polar quantities are z-scored using the checkpoint's stored training statistics. Variable-length polar sequences are padded in batches, with masks marking valid polar positions. At evaluation time, geometry tokens are visible, the splitter is inserted, and all valid polar tokens are embedded for teacher-forced prediction.

3.5 Evaluation Metrics

The main metrics are per-airfoil mean absolute errors in C_ℓ and C_d, averaged across evaluation rows, with distribution summaries (mean through maximum). Stall-region metrics retain points within $\pm 3^\circ$ of each airfoil's empirical peak C_ℓ along the stored α sequence (global maximum). The 3° half-width is a fixed benchmark choice in the withheld evaluator so every model sees the same angular slice around $C_{\ell,\max}$; reading coefficients in a narrow α band near maximum lift follows standard airfoil polar practice [1]. Symmetric polars with multiple peaks follow the stored global maximum, which can blur "stall" semantics. Rows are excluded if the peak is the final polar point or fewer than two points lie in the window. Of 4096 valid evaluation rows, 2929 have a valid stall subset.

We also report paired win rates (fraction of rows where FoilCORE beats each NeuralFoil baseline), threshold coverage (e.g. C_d MAE < 0.005), peak-C_ℓ error, and stall-angle error.

4 Model Architecture

FoilCORE has three stages, as in Figure 1(**b**): embed polar and geometry tokens, run a causal transformer trunk with path blocks and outer products, then decode with a narrow two-layer polar head. Section 2.2 contrasts this with pointwise MLPs; Section 6.3 ablates the path block before the outers.

[1] Raw BigFoil tabular extracts used to regenerate paper metrics are not redistributed with this manuscript or companion code; see [11] for the public catalog and licensing. For bespoke tabular extracts prepared for this study, contact Mike Quayle. For broader context on other polar corpora see [2, 17]. Withheld labels follow viscous-inviscid panel-style polar conventions common in low-order airfoil analysis [7].

Table 1: Dataset scale and empirical ranges (numeric columns parsed from manifest statistics for the splits used in this study). Train/validation counts follow the dataset split report packaged with the code release.

Quantity	Train	Val
Polar-curve rows	329 719	94 205
Value ranges on the training split (full manifest scan)		
α range (°)		−20.00–20.00
Re: min 4.0100×10^4; max 2.0000×10^7		
C_ℓ range		−2.207–3.175
C_d range		0.0000–0.4104
Same ranges on the validation split (first 50k rows scanned)		
α (°)		−20.00–20.00

4.1 High-Level Architecture

FoilCORE keeps d_{model} small: causal depth and masking encode sweep order; path nonlinearities and gated outer products supply multiplicative mixing without needing to widen the trunk. The evaluated model uses $d_{\text{model}} = 8$ [3], eight causal self-attention blocks, two outer-product expansion modules [4], two MLP refinement blocks, dropout 0.05, residual connections, and a two-output polar decoder. Conceptually:

$$\begin{aligned} &\text{embedding} \to \text{causal attention blocks} \to \text{path + gated outers} \\ &\quad \to \text{MLP refinement} \to \text{polar decoder (two-layer head).} \end{aligned} \tag{1}$$

4.2 Embedding Layer

The polar projection is

$$e_t^p = W_p x_t^p + b_p + \text{PE}(\alpha_t),$$

where $x_t^p \in \mathbb{R}^5$ and $W_p \in \mathbb{R}^{8\times 5}$. The geometry projection is

$$e_j^g = W_g[x_j^g; s_j] + b_g + \text{PE}(u_j),$$

where $x_j^g \in \mathbb{R}^{20}$, $s_j \in \{0, 1\}$ is the surface flag, and u_j is the chord-position index. The splitter token is a learned vector in $\mathbb{R}^8$.

4.3 FoilCORE Trunk

The main trunk uses eight standard causal self-attention blocks. Each block contains two attention heads, a residual connection, dropout, and LayerNorm. The causal mask prevents token t from attending to later sequence positions. After each attention block, FoilCORE applies the selected featurewise path module.

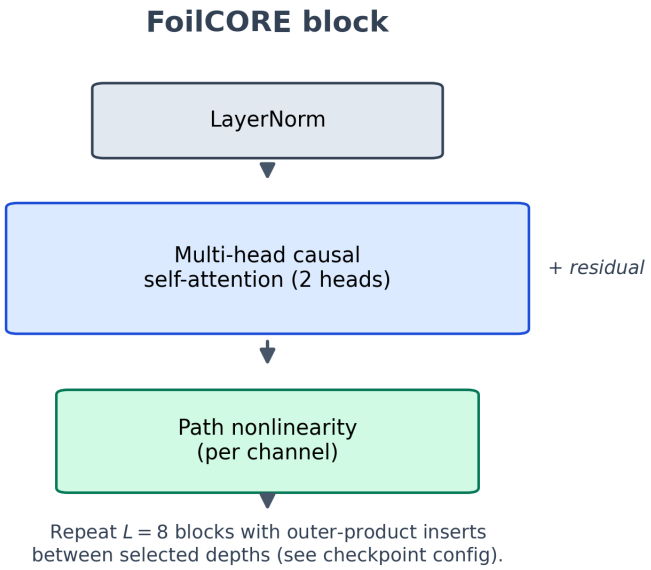


Figure 2: Single FoilCORE block. The evaluated trunk uses standard two-head causal self-attention followed by a learned featurewise path nonlinearity.

4.4 Path Nonlinearity

The evaluated checkpoint uses the `full` path mode. For feature i, the path transform is

$$\begin{aligned}\phi_i(x_i) = \; & \arccos(u_i) + c_i \sin(d_i x_i)^3 \\ & + f_i x_i + g_i (x_i - h_i)^2 + m_i,\end{aligned} \tag{2}$$

where $u_i = \mathrm{clip}(b_i x_i, -1+\epsilon, 1-\epsilon)$. We evaluate $\arccos(u_i)$ with `torch.atan2`$(\sqrt{1-u_i^2}, u_i)$ on the clipped domain for numerical stability. The coefficients $b_i, c_i, d_i, f_i, g_i, h_i, m_i$ are learned per channel (seven scalars each). The evaluated model has 12 path modules (672 parameters).

4.5 Outer-Product Expansion

After the attention/path stack, FoilCORE applies two multiplicative outer-product expansion modules [4]. The form is related to second-order maps in factorization machines [13]. A linear map needs a wide hidden state to rebuild multiplicative mixes of channels; gated outers add about $\mathcal{O}(K d_{\text{model}})$ parameters for small K. For hidden vector h, each module forms

$$\tilde{h} = h + \sum_{k=1}^{K} \mathrm{softmax}(v)_k \, \mathrm{GELU}(h\, w_k),$$

with $K = 32$ learned outer channels. This layer captures multiplicative interactions with only $2K = 64$ parameters per module. The residual form is enabled in the main model.

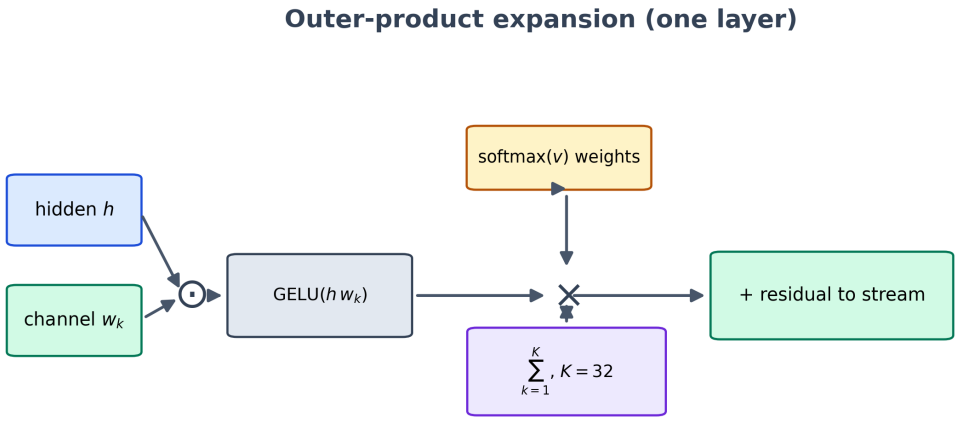
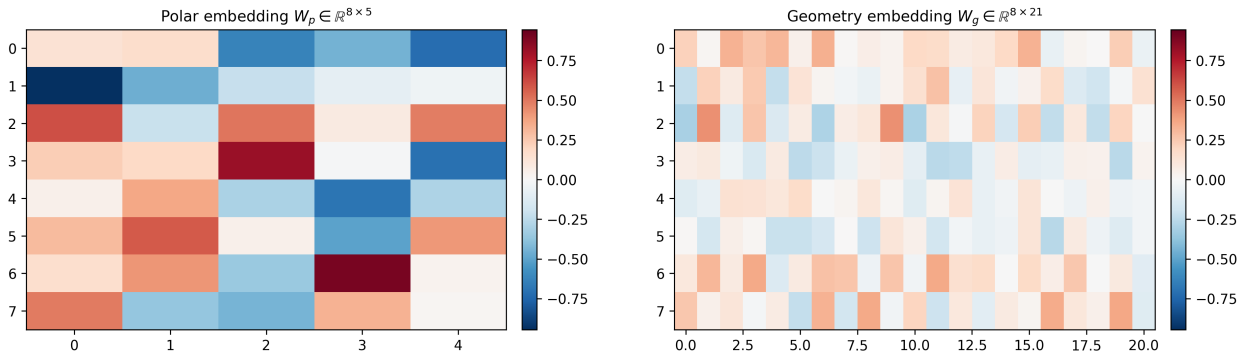


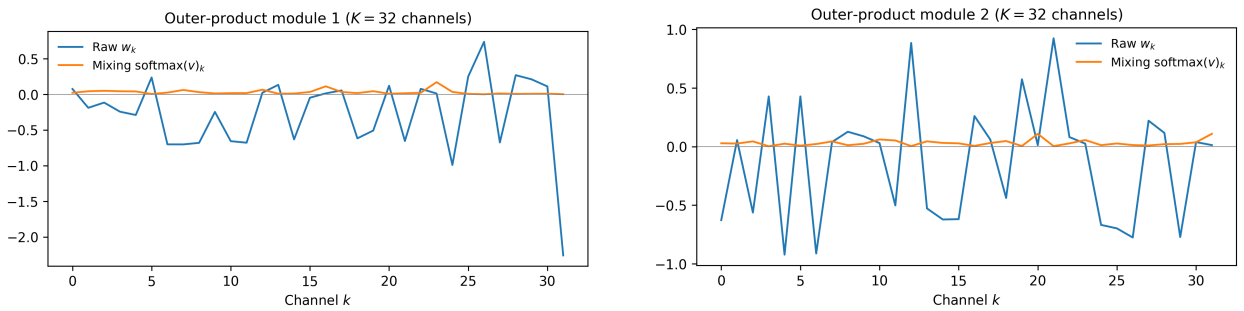
Figure 3: Outer-product expansion module. Learned scalar channels produce low-cost multiplicative feature interactions, followed by softmax channel mixing and a residual connection.



(a) Polar projector W_p.

(b) Geometry projector W_g.

Figure 4: Learned affine embedding kernels (heatmap of matrix entries). Colors are symmetric about zero to emphasize cancellations.



(a) Residual outer-product module #1.

(b) Residual outer-product module #2.

Figure 5: Outer-product kernels w and normalized mixing weights softmax(v) for each gated channel.

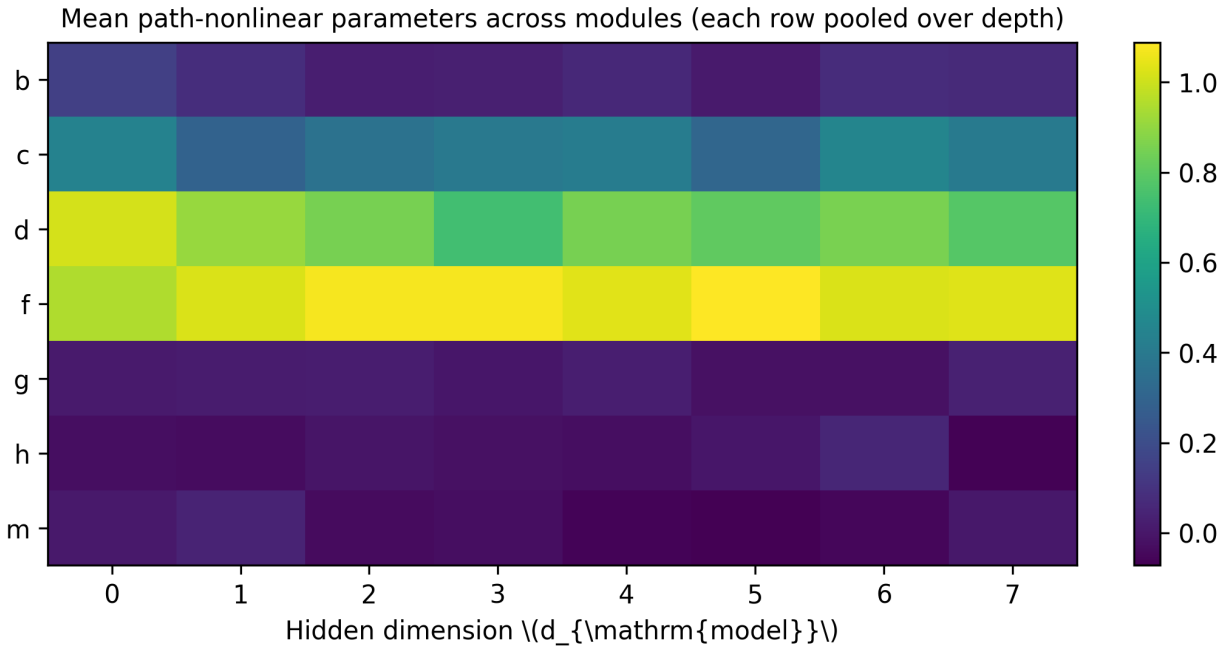


Figure 6: Mean absolute path-nonlinear tensors averaged across instantiated `PathNonlinearity` modules; rows correspond to (b, c, d, f, g, h, m) in (2).

The two MLP refinement blocks each use hidden width 64, GELU activation, dropout, residual connection, and LayerNorm. Weight matrices map the eight-token stream into a widened bottleneck and back again:

$$\text{Linear}(8, 64) \to \text{GELU} \to \text{Dropout}(0.05) \to \text{Linear}(64, 2),$$

yielding logits for incremental updates to (C_ℓ, C_d).

4.6 Parameter Breakdown

Table 2 provides a complete accounting of the model's 6,394 parameters.

Table 2: FoilCORE parameter breakdown ($d = 8$).

Component	Details	Parameters
Polar embedding	Linear($5 \to 8$)	48
Geometry embedding	Linear($21 \to 8$)	176
Splitter embedding	Learned vector	8
Attention blocks (8)	$8 \times$ (MultiheadAttention + LN)	2,432
Path nonlinearities (12)	$12 \times$ PathNonlinearity	672
Outer-product expansions (2)	$2 \times$ OuterProductExpansion	128
MLP refinement (2)	$2 \times$ MLPBlock	2,224
Polar decoder	Linear($8 \to 64 \to 2$)	706
Total		**6,394**

5 Training Objective

5.1 Loss Function

Training uses a weighted MAE in normalized polar space:

$$\mathcal{L} = |\hat{C}_{\ell,z} - C_{\ell,z}| + 31\,|\hat{C}_{d,z} - C_{d,z}|.$$

The angle-of-attack, Reynolds, and Mach dimensions are input-only in the polar loss and have zero loss weight. This weighting is stored in the training code and reflected in the checkpoint-generation configuration.

5.2 Rationale for Drag Weighting

Drag is numerically smaller than lift and can be underfit if both channels are weighted equally after normalization. A strong drag term is especially important near stall, where drag changes quickly and where engineering decisions are sensitive to drag rise. The chosen $31\times$ normalized C_d weight prioritizes drag accuracy without removing the C_ℓ term.

5.3 Optimization Details

The matching recorded training configuration uses AdamW [10] with learning rate 3×10^{-4}, weight decay 10^{-4}, batch size 900, gradient clipping at 1.0, 10 warmup epochs, and ReduceLROnPlateau after warmup. The best checkpoint used here is epoch 22 with recorded validation loss 1.623 27.

Table 3: Training hyperparameters for the evaluated checkpoint.

Setting	Value
Optimizer	AdamW
Learning rate	3×10^{-4}
Weight decay	10^{-4}
Warmup	10 epochs
Schedule	ReduceLROnPlateau after warmup
Batch size	900
Checkpoint epoch	22
Checkpoint validation loss	1.62327
Polar loss weights	[0, 0, 0, 1, 31]
Gradient clipping	1.0
Mixed precision	Full precision on CPU and Metal (MPS). Mixed precision (AMP) enabled only when training on CUDA.

Mixed precision activates only under CUDA (`torch.cuda.amp`); CPU and Metal runs stay in FP32 exactly as surfaced in Table 3.

6 Experiments

Unless stated otherwise, numbers use the first 4096 withheld test rows (Appendix D for larger exports). We report teacher-forced benchmarks matched to NeuralFoil, validation-split autoregressive stress tests, and path-mode ablations.

Teacher-forcing metrics use bundled evaluation utilities on rows where FoilCORE and both NeuralFoil baselines succeed.

6.1 Baseline Models and Checkpoints

NeuralFoil comparisons [15, 16] load official pretrained `torch` checkpoints from the public release. We call the public Python API without fine-tuning on the training split. Each row supplies chord-normalized coordinates and pointwise (α, Re); NeuralFoil's released models do not take a per-sample Mach argument in `get_aero_from_coordinates`, whereas FoilCORE receives ground-truth M each polar step. NeuralFoil is a strong *pointwise* surrogate and diagnostic baseline on the benchmark rows.

FoilCORE results use the primary FoilCORE checkpoint trained on polar rows drawn from the training split with validation diagnostics on the validation split, following the optimisation recipe in Section 5. Where tables pair the two approaches, preprocessing enforces mutually valid polar lengths and identical subsets of airfoils.

6.2 Experimental Setup

All ablation studies are run via a unified pipeline and logged to a structured manifest. Training defaults are held constant unless the specific variable is under study: 120 epochs, batch 64, AdamW (lr $= 3 \times 10^{-4}$, wd $= 10^{-4}$), cosine annealing, dropout $p = 0.05$, $d = 8$, 8 layers, path mode "full", 2 outer-product layers, 60% train fraction.

6.3 Path-Mode Ablation

We compare two nonlinear trunk configurations at identical width ($d = 8$) and depth (eight causal blocks):

Table 4: Path-mode ablation with stall-region metrics. All numbers are CPU teacher-forced on the first 4096 rows of the withheld test split. Stall region = $\pm 3^\circ$ around peak C_ℓ. p95 = 95th percentile of per-airfoil MAE.

Path Mode	Params	Global MAE		Stall MAE		Stall MAE p_{95}	
		C_ℓ	C_d	C_ℓ	C_d	C_ℓ	C_d
PathNonlinearity (baseline)	6394	**0.0515**	**0.002 99**	**0.060**	**0.0031**	**0.11**	**0.008**
GELU path (no custom nonlin)	5722	0.2042	0.004 10	0.218	0.0033	0.49	0.0087

Replacing the nonlinear path with elementwise GELU substitutes increases pooled C_ℓ MAE from 0.051 to 0.204 (GELU-path ablation checkpoint; `path_mode = gelu`). Within the $\pm 3^\circ$ stall band, stall-average C_ℓ MAE moves from 0.060 to 0.218 and ninety-fifth percentile stall C_ℓ MAE from 0.11 to 0.49 (Table 4).

6.4 Outer-product depth and dropout

Each outer module adds $\mathcal{O}(K)$ parameters with $K = 32$. The primary checkpoint stacks two modules (64 parameters each) between attention blocks and uses dropout $p = 0.05$ (Table 3).

6.5 Embedding-width sweeps and structural grids

Trunk width d can be changed at training time (even width, two attention heads); the public checkpoints use $d = 8$ [3]. Other widths (e.g. $4, 6, 12$) use the same driver; see the README for flags and for saving checkpoints with exported CPU teacher-forcing metrics.

6.6 Training Dynamics

Figure 7 shows train and validation loss and the learning-rate schedule for the reproduced baseline. The checkpoint used in tables is epoch 22 (validation loss 1.623).

7 Results

7.1 Global Performance

On the 4096-row teacher-forced CPU benchmark, FoilCORE reports mean C_ℓ MAE 0.051 54 and mean C_d MAE 0.002 99. Under identical rows and preprocessing, NeuralFoil `xxsmall` yields 0.0853 / 0.013 07 and `xxxlarge` yields 0.0779 / 0.012 30. With far fewer parameters (Table 2), the narrow sequential model matches or beats both sizes on pooled MAE. Tail behaviour and autoregressive rollouts are in Sections 7.5 and 7.4.

Figure 7: Train and validation loss with learning-rate drops (reproduced baseline; best epoch 22).

Table 5: Main teacher-forced results on the withheld test split (matched rows, shared preprocessing).

Model	Params	C_ℓ MAE	C_d MAE	Stall C_ℓ	Stall C_d	$C_\ell\, p_{95}$	$C_d\, p_{95}$
FoilCORE	6,394	0.0515	0.00299	0.0446	0.00205	0.0863	0.00588
NeuralFoil xxsmall	22,000	0.0853	0.01307	0.0895	0.01455	0.2333	0.04994
NeuralFoil xxxlarge	1,500,000	0.0779	0.01230	0.1012	0.01729	0.2383	0.04682

C_ℓ/C_d MAE: global teacher forcing (matched rows). *Stall C_ℓ/C_d*: $\pm 3^\circ$ stall window (§3). Columns p_{95}: ninety-fifth percentile of per-airfoil MAE on the global curves.

7.2 Lift/drag scatter, residual bias, and error versus Reynolds

Aggregating every valid teacher-forced polar position on the evaluation prefix quantifies correlation, bias, and AoA-dependent residuals without averaging per airfoil first. Figure 9 shows predicted versus true coefficients; Figure 10 plots mean residuals in α bins with an airfoil-pooled dispersion band; Figure 11 summarises pooled error against Reynolds-number quantile bins.

7.3 Bootstrap intervals and inference tests

Table 6 summarizes bootstrap 95 % intervals for population-mean global and stall-window per-airfoil MAE (matching the pooled JSON behind Tables 5–10; stall-window resampling restricts to rows flagged `has_stall` = true in the evaluator). Exact two-sided p-values for paired t-tests contrast each NeuralFoil baseline versus FoilCORE on paired MAE tuples for global metrics and stall-window subsets; Wilcoxon and Levene entries apply only to *global* per-airfoil MAE.

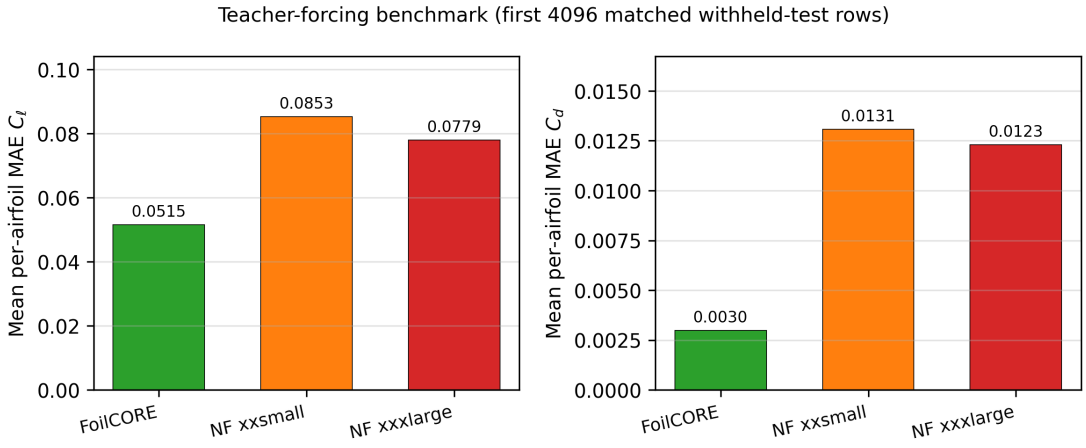


Figure 8: Global mean absolute error for lift (left) and drag (right) on the withheld test split under teacher forcing. FoilCORE attains lower pooled MAE than both NeuralFoil sizes on the matched rows; bootstrap 95 % intervals are in Table 6.

7.4 Autoregressive Rollout on the Validation Split

Aggregated over 94 145 mutually successful airfoils in the archived validation export, switching from teacher forcing to open-loop decoding (ground-truth (α, Re, M) each step, greedy feedback for C_ℓ, C_d when no teacher prefix is set) inflates mean C_ℓ MAE from 0.0544 to 0.0950 and mean C_d MAE from 0.003 06 to 0.005 56 (74.6 % and 81.7 % relative). Stall-window mean C_d MAE rises from 0.002 08 to 0.003 80. Tables 5–10 report teacher-forced metrics for matched comparison with NeuralFoil on the withheld test prefix.

NeuralFoil `xxxlarge` warmstart for open-loop use. In applications that already ship NeuralFoil [15, 16], one open-loop recipe is to *warmstart* FoilCORE: embed the first K polar steps with ground-truth (α, Re, M) and C_ℓ, C_d from `xxxlarge`, then run greedy feedback for the rest. A small driver sets K, α sweeps, and geometry for that prefill. The larger `xxxlarge` model gives a short prefix of coefficients so the narrow causal head does not start cold along the sweep.[2]

Withheld-set tables use $K{=}0$ so autoregressive diagnostics give NeuralFoil the same per-step (α, Re) inputs they were trained on; deployed sweeps typically use $K > 0$.

7.5 Per-airfoil error distributions and residual shapes

Figure 13 pools teacher-forced pointwise residuals $\hat{C} - C$ for C_ℓ and C_d; each panel overlays a Gaussian with the empirical mean and variance of the pooled sample (reference only). Figure 14 highlights non-Gaussian wings in the same C_ℓ residuals on normal-probability axes. Table 14 (Section 7.15) adds a controlled α-embedding noise stress test.

FoilCORE's global C_ℓ p_{95} error is 0.0863, compared with 0.2333 for NeuralFoil `xxsmall` and 0.2383 for NeuralFoil `xxxlarge`. For C_d, FoilCORE's p_{95} is 0.005 88, compared with 0.049 94 and 0.046 82. The sample standard deviation of C_d MAE is also materially smaller: 0.001 85 for

[2]Running both models in sequence duplicates work on prefix rows; fused schedules or distilled teachers are not explored here.

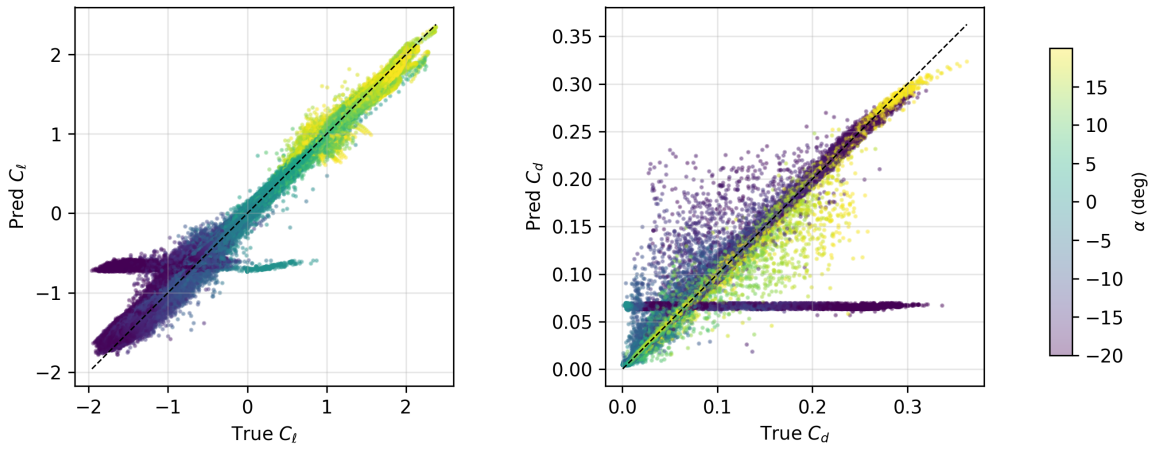


Figure 9: Teacher-forced C_ℓ and C_d at pooled polar steps (density encoded by α). Identity lines correspond to perfect agreement.

FoilCORE versus 0.015 46 and 0.014 75 for the NeuralFoil sizes (about 8.4× wider for `xxsmall`; Table 8).

Heavy tails and worst-case rows. Figure 13 overlays Gaussian curves matched to $\hat{\mu}$ and $\hat{\sigma}$ of the pooled sample; empirical $\hat{C} - C$ mass extends well past both wings for both channels. Figure 14 shows the same C_ℓ residuals on normal probability axes: curvature at the extremes marks heavy tails (sporadic large misses at specific geometry/Re rows and α indices that barely move pooled means). Table 8, stall analogues in Table 10, high percentiles, and threshold coverage (Section 7.8) matter for risk budgeting alongside means. Figure 16: NeuralFoil `xxsmall` puts more mass in large per-airfoil C_ℓ and C_d failures than FoilCORE; FoilCORE still shows a slim far-right tail.

Thickness tertiles (Figure 17; Table 9) smooth over rare high-MAE polars inside each bin; airfoil-level maxima in the evaluation JSON (Appendix B) complement bin averages.

7.6 Lift and drag versus thickness bins

Heuristic $\widehat{t/c}$ is reconstructed from withheld JSON coordinates with the same 200-point resampling and chord normalisation employed during training (§3); upper versus lower arcs follow the 10-patch upper half and 10-patch lower half encoded by surface flags in the dataloader. After rotating inferred leading/trailing extremes onto a chord-aligned frame, spline stations yield $\widehat{t/c}(x)$; $\max|\widehat{t/c}(x)|$ stratifies tertiles enumerated in Table 9. On this excerpt thinner profiles carry slightly higher pooled C_ℓ MAE (≈ 0.0549 in the first tertile versus ≈ 0.050 thereafter) whereas FoilCORE C_d improves monotonically with thickness; NeuralFoil `xxsmall` shows analogous drag ordering but reports higher pooled errors than FoilCORE across tertiles on this slice. Auxiliary tertile metadata retains mean peak camber norms for exploratory filtering beside $\widehat{t/c}$.[3]

[3]Heuristic $\widehat{t/c}$ descriptors are scoped in Section 9.

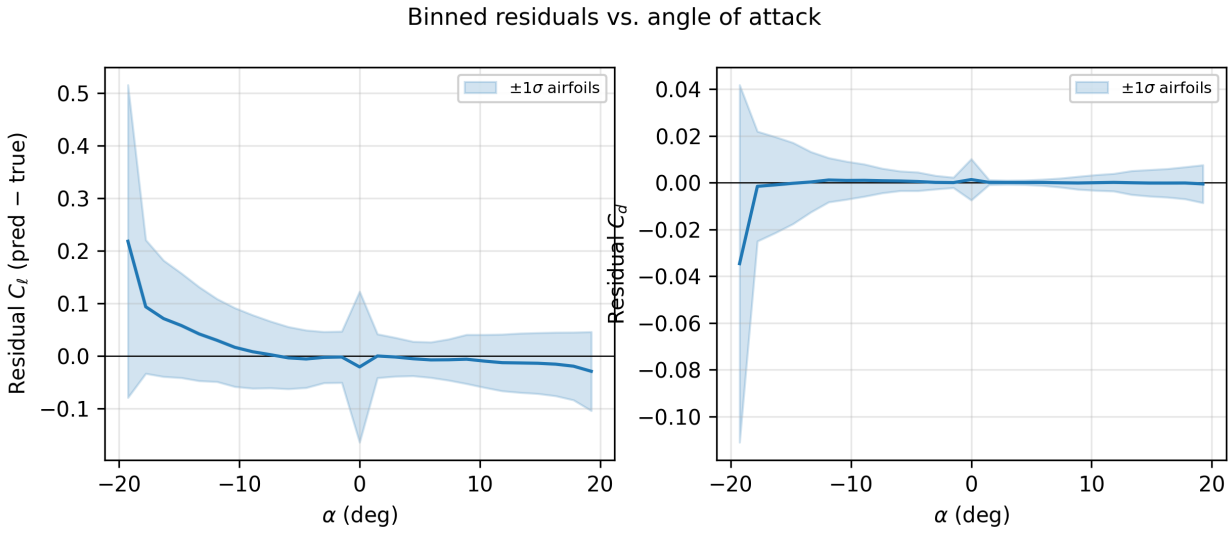


Figure 10: Binned mean residual (predicted minus true) with $\pm 1\sigma$ dispersion of point-wise residuals inside each α bin.

7.7 Stall-Region Results

FoilCORE reaches stall-window C_ℓ MAE 0.044 57 and stall-window C_d MAE 0.002 05. On matched rows under teacher forcing, both stall metrics sit substantially below the NeuralFoil `xxsmall` and `xxxlarge` baselines; the drag channel shows the largest absolute gap (numerical ratios appear in Table 5). Interpreting generously, the narrow head still picks up local structure near stall.

7.8 Row-by-row wins and error-threshold coverage

On the excerpt, FoilCORE achieves strictly lower per-airfoil MAE than NeuralFoil `xxsmall` on 60.8 % of rows for C_ℓ and 82.8 % for C_d; versus `xxxlarge`, the same paired comparison yields 52.4 % for C_ℓ and 75.5 % for C_d. These fractions summarise row-wise mismatches under teacher forcing.

A global C_d MAE below 0.005 is achieved on 86.0 % of FoilCORE rows, compared with 41.1 % for NeuralFoil `xxsmall` and 45.6 % for NeuralFoil `xxxlarge`. In the stall subset, C_d MAE below 0.005 is achieved on 92.5 % of FoilCORE rows, versus roughly 24 % for both NeuralFoil baselines.

7.9 Parameter Efficiency

FoilCORE's exact parameter count is 6,394. The largest components are the eight attention blocks (2,432 parameters), the MLP refinement blocks (2,224), the polar decoder (706), and the path nonlinearities (672). The embedding system contributes only 232 parameters.

7.10 Qualitative coordinate hold-out (AF104K)

Figure 21 shows a linear α sweep at Re=1×10^5 on an AF104K-style supercritical section (`.xy` in the code release): greedy FoilCORE after a three-step NeuralFoil `xxxlarge` coefficient warmstart (Section 7.4), with teacher-forced FoilCORE and both NeuralFoil sizes as references. On this out-of-open-shard geometry the narrow causal head tracks the `xxxlarge` polar closely. The illustration augments bulk withheld metrics; it is not a catalogued family hold-out (Section 9).

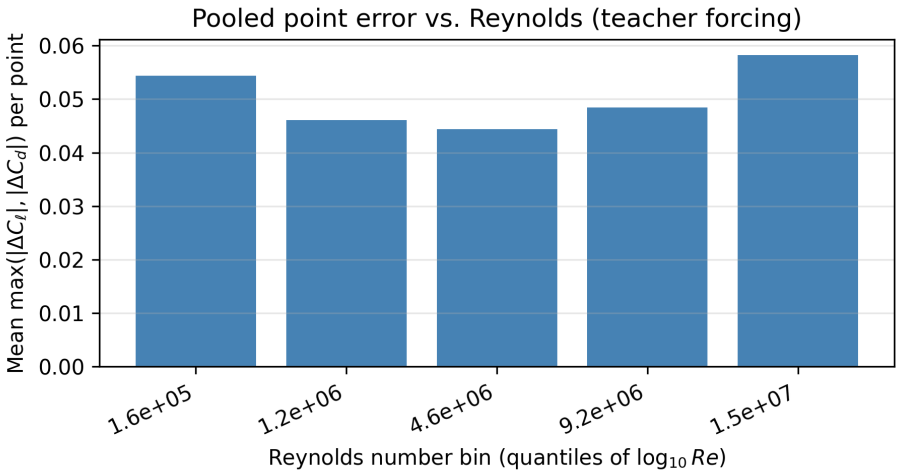


Figure 11: Mean point-wise $\max(|\Delta C_\ell|, |\Delta C_d|)$ versus $\log_{10} Re$ quantile bins (teacher forcing).

7.11 Synthetic perimeter stress

On the post-warmstart tail (greedy decoding after K=3 NeuralFoil-`xxxlarge` anchors at Re=1×10^5), mean absolute gaps in C_ℓ between FoilCORE autoregression and NeuralFoil-`xxsmall` are ≈ 2.07, versus ≈ 0.41 against NeuralFoil-`xxxlarge` ($\approx 0.20\times$ the xxsmall-aligned error). For C_d the same tail gaps are ≈ 10.61 versus ≈ 0.0187 ($\approx 1.8 \times 10^{-3}\times$). `xxsmall` departs strongly in drag on this perimeter; autoregressive FoilCORE stays near the `xxxlarge` polar it was seeded with. Figure 22 overlays teacher-forced FoilCORE, greedy autoregression, and both NeuralFoil traces.

The perimeter is a smoothed closed curve (deterministic noise kernel; seed 42) chord-normalised like training inputs but not drawn from any training row: a deliberate shape shift away from catalogued families, same protocol as Figure 21 (Section 7.4). Warmstart and prefill tie this probe to the `xxxlarge` manifold; it complements distributional tests rather than certifying OOD coverage.

7.12 Error versus angle of attack

Pooled mean absolute error by α is lower for FoilCORE than for both NeuralFoil checkpoints over most of the evaluated range, with the largest separation in C_d near stall angles (Figure 23). Stall-window MAE in Table 5 is computed per airfoil; Figure 23 pools every valid point into shared α bins—same physics, two different cuts through the data.

7.13 Inference latency and NeuralFoil warmstarts

Figure 24 plots CPU throughput versus TensorLoader batch size on the fixed row prefix used during asset regeneration. FoilCORE teacher-forced inference over 4096 rows consumed 3.49 s at batch size 512 (roughly 0.85 ms per row amortised; batch size one amplifies interpreter overhead). NeuralFoil `xxxlarge` averaged 1.5 ms per row on the same host family.
Table 12 complements Figure 24: it reports end-to-end wall time per polar on a single CPU thread (no GPU), which is the relevant regime for Raspberry Pi–class hardware. On the reference laptop-class CPU draw used for manuscript regeneration, NeuralFoil `xxxlarge` requires roughly 2.5× more time per polar than the ultra-compact FoilCORE trunk, while the smaller `xxsmall` checkpoint

Table 6: Bootstrap intervals (top block) pool stored per-airfoil evaluation records (2000 bootstrap means each, RNG seed 0; stall-window columns reuse only `has_stall` rows). Bottom block lists Wilcoxon signed-rank and Levene tests on global contrasts (NF MAE − FC MAE per row) followed by paired two-sided t-tests for global and stall-window MAE contrasts (four MAE contrasts per NeuralFoil size: C_ℓ and C_d).

Bootstrap 95 % intervals for population mean
(2000 resampled means per statistic, RNG seed 0; pooled per-airfoil JSON)

	Global per-airfoil MAE		Stall-window per-airfoil MAE[a]	
Model	C_ℓ	C_d	C_ℓ	C_d
FoilCORE	[0.050 88,0.052 22]	[0.002 94,0.003 05]	[0.043 35,0.045 72]	[0.001 98,0.002 13]
NeuralFoil `xxsmall`	[0.083 07,0.087 62]	[0.012 61,0.013 54]	[0.087 27,0.091 81]	[0.014 09,0.015 00]
NeuralFoil `xxxlarge`	[0.075 78,0.080 22]	[0.011 84,0.012 76]	[0.097 78,0.104 70]	[0.016 57,0.018 02]

Nonparametrics and paired t-tests

(t: two-sided `ttest_rel(NF MAE, FC MAE)` on matched CSV rows; Stall tests use `has_stall` = true only, $n = 2929$)

Statistic	`xxsmall`[b]		`xxxlarge`	
	C_ℓ	C_d	C_ℓ	C_d
Wilcoxon signed-rank (Δ = NF − FC, **global** MAE only)	1.20×10^{-121}	0.00	5.83×10^{-50}	0.00
Levene (NF vs. FC per-airfoil MAE variances, median-centered)	0.00	0.00	0.00	0.00
Paired t global MAE	4.07×10^{-167}	0.00	3.58×10^{-103}	0.00
Paired t stall-window MAE	1.58×10^{-238}	0.00	6.75×10^{-173}	1.67×10^{-290}

[a]Stall-window bootstrap intervals use only rows with `has_stall` = true.
[b]Applies to both NeuralFoil column pairs: two-sided p-values for C_ℓ then C_d (NF vs. FoilCORE on matched rows).

remains faster in raw milliseconds at the cost of the accuracy gap documented in Table 5. Table 13 lists the same three models under deliberately favorable settings for each stack: batched accelerator inference for FoilCORE and default-thread NumPy for NeuralFoil. Absolute milliseconds shift with board, OS, and linkage.

Figures 25–26 vary the NeuralFoil `xxxlarge` warmstart length $K = 0, \ldots, 5$ on a withheld prefix, plotting pooled teacher-forced versus autoregressive MAE and early polar-index error. They extend the recipe in Section 7.4; tabulated withheld aggregates use K=0 for comparability with the pointwise baselines.

7.14 Error along the polar sequence index

Withheld polars omitting masking artefacts contain $\mathcal{O}(70)$ α-sorted sample points ($\approx$ 66 to 82 on matched test rows). Figure 27 pools mean absolute $\hat{C}_\ell$ and $\hat{C}_d$ deviations at ordinal positions 1 to 20, restricted to curves with polar length $L \geq 20$; this diagnoses whether inaccuracies cumulate visibly along causal decoding order under teacher forcing versus remaining roughly flat early in each polar.

Table 7: Physics-oriented sanity diagnostics on pooled points and raw polars (see script for definitions).

Check	Value
Spearman $\rho(\lvert C_{\ell,\text{true}}\rvert, C_{d,\text{true}})$	0.348 ($p = 0.00$)
Max predicted / true C_ℓ	2.350 / 2.380
GT pre-peak C_ℓ non-monotone step fraction	0.0704
Mean upper/lower surface symmetry residual	0.030 62 (chord coords)

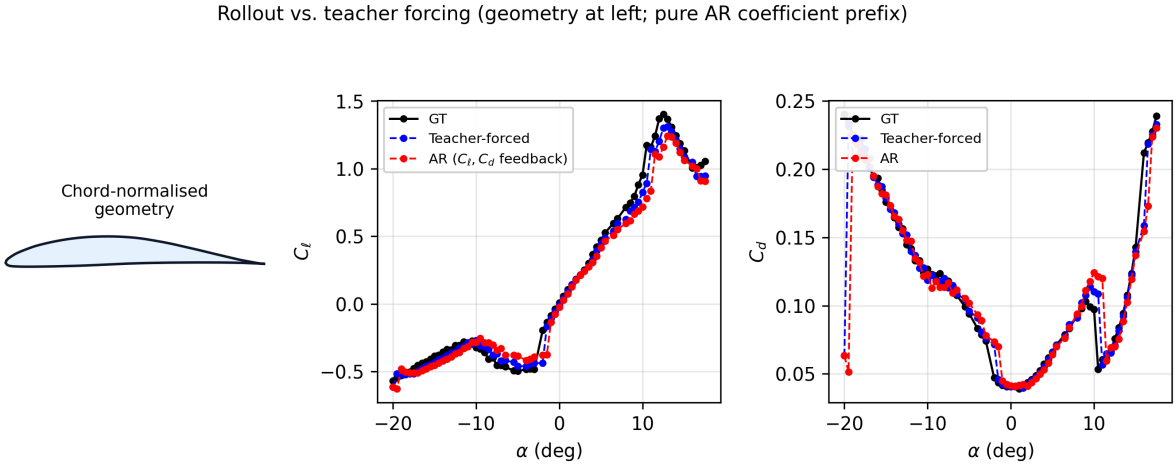


Figure 12: Representative polar for the chord-normalised section shown at left of the figure, comparing teacher forcing to greedy autoregressive C_ℓ, C_d feedback (ground-truth $\alpha/Re/M$ retained at every step). Greedy feedback compounds error along the α sequence relative to teacher forcing; Section 7.4 describes coefficient warmstarts when true coefficients are unavailable.

Table 8: Global per-airfoil error distribution.

Model	Metric	Mean	Median	Std	p25	p75	p95	p99
FoilCORE	Cl	0.0515	0.0479	0.0212	0.0373	0.0615	0.0863	0.1134
FoilCORE	Cd	0.00299	0.00312	0.00185	0.00110	0.00435	0.00588	0.00716
NF `xxsmall`	Cl	0.0853	0.0583	0.0728	0.0313	0.1252	0.2333	0.3239
NF `xxsmall`	Cd	0.01307	0.00742	0.01546	0.00203	0.01877	0.04994	0.06876
NF `xxxlarge`	Cl	0.0779	0.0501	0.0738	0.0221	0.1161	0.2383	0.3240
NF `xxxlarge`	Cd	0.01230	0.00593	0.01475	0.00208	0.01843	0.04682	0.06411

7.15 Angle noise at inference and path-module shapes

Table 14 lists pooled teacher-forced MAE when zero-mean Gaussian noise (standard deviation as multiples of the checkpoint's training σ_α) is injected into α embeddings at inference (regenerated

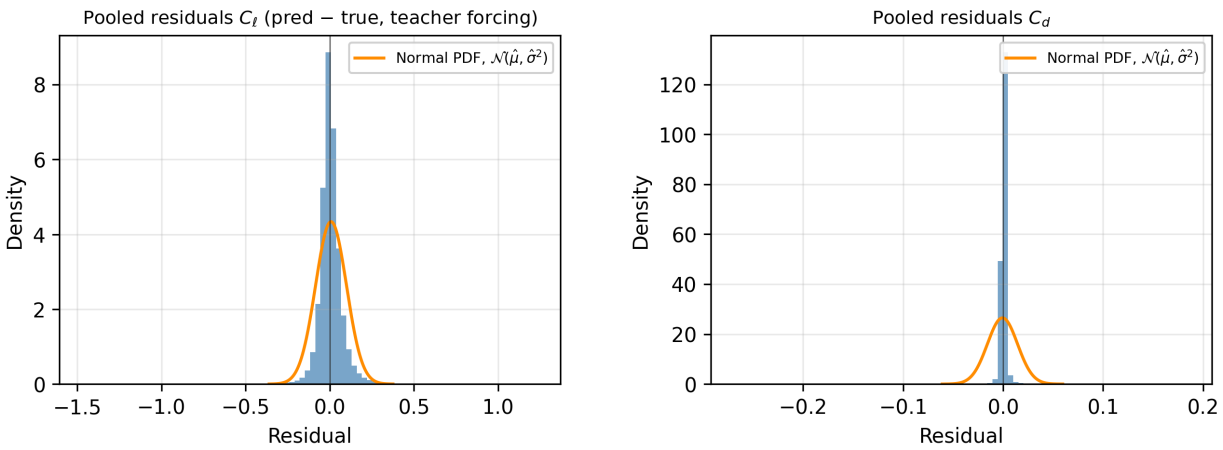


Figure 13: Pooled pointwise residuals (pred minus true; same pooled polar steps as Figures 9–10) with a normal overlay matched to $\hat{\mu}, \hat{\sigma}^2$ computed from the displayed sample. Excess mass in the wings is the heavy-tail story in Section 7.5; see Figure 14.

Table 9: Teacher-forcing mean pooled MAEs within empirical $\widehat{t/c}$ tertiles.

t/c tertile	FoilCORE mean MAE		NF `xxsmall` mean MAE	
Bin	C_ℓ	C_d	C_ℓ	C_d
Thin t/c	0.054 94	0.003 85	0.086 88	0.016 68
Medium t/c	0.049 46	0.002 88	0.086 60	0.013 28
Thick t/c	0.050 22	0.002 24	0.082 42	0.009 23

Empirical tertiles $t/c = 0.110\,00$ and $0.150\,12$; thickness proxy uses training resampler and patch-based upper/lower split.

Table 10: Stall-region per-airfoil error distribution ($\pm 3^\circ$ of peak C_ℓ).

Model	Metric	Mean	Median	Std	p25	p75	p95	p99
FoilCORE	Cl	0.0446	0.0367	0.0324	0.0233	0.0556	0.1048	0.1595
FoilCORE	Cd	0.00205	0.00143	0.00208	0.00093	0.00228	0.00610	0.01049
NF `xxsmall`	Cl	0.0895	0.0785	0.0621	0.0386	0.1282	0.1983	0.2868
NF `xxsmall`	Cd	0.01455	0.01180	0.01208	0.00520	0.01912	0.04058	0.05528
NF `xxxlarge`	Cl	0.1012	0.0797	0.0967	0.0372	0.1365	0.2514	0.5131
NF `xxxlarge`	Cd	0.01729	0.01229	0.02008	0.00527	0.02103	0.04909	0.11086

with the paper asset bundle). Figure 28 plots the first `PathNonlinearity` map on a synthetic grid. Figure 14 shows standardized pooled C_ℓ residuals depart from a normal bridge in the tails—same heavy-tail theme as Section 7.5.

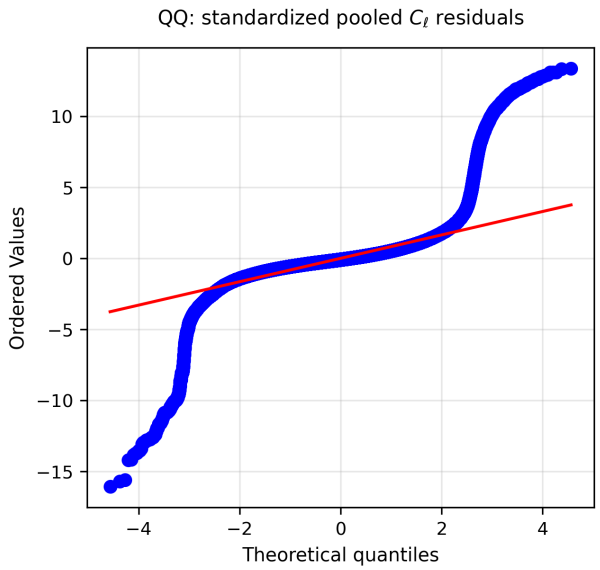


Figure 14: Normal probability plot for standardized pooled C_ℓ residuals under teacher forcing; departures from the diagonal at extreme quantiles mark heavy tails (Section 7.5).

Table 11: Mean per-airfoil C_d MAE on the withheld test split stratified by post-peak C_ℓ drop severity (ground-truth polars).

Stall severity (post-peak C_ℓ drop)	Rows	Mean C_d MAE (FoilCORE)
Mild ($< 20\%$)	1862	0.002 62
Moderate (20−40%)	1005	0.004 22
Deep ($> 40\%$)	62	0.005 35

8 Discussion

On the first 4096 withheld rows, teacher-forced MAE is lower than for both NeuralFoil sizes at the counts in Table 2. Causal order along the polar adds information beyond pointwise regression at the same width.

NeuralFoil is a strong pointwise baseline with a stable public API. FoilCORE targets sequential polar prediction; Section 7.4 combines a short NeuralFoil `xxxlarge` coefficient prefix with greedy continuation.

Autoregressive decoding without prefixes raises error (Section 7.4); longer prefixes trade compute for stability. Warmstart sweeps and throughput are in Figures 25–26 and Section 7.13. Geometry-stratified hold-outs, multi-seed ensembles, and full withheld-grid structural reruns are not reported here.

9 Limitations

What the numbers actually mean. All tables report teacher-forced reconstruction on withheld BigFoil rows with NeuralFoil-matched preprocessing [11]. Inputs must match Section 3 to reproduce

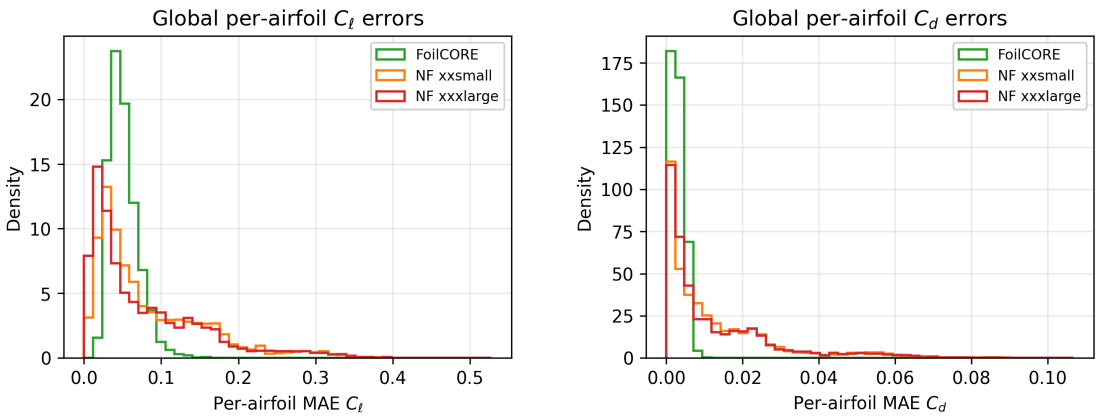


Figure 15: Per-airfoil error distributions (C_ℓ and C_d) under teacher forcing. Histogram mass concentrates in lower MAE buckets for FoilCORE on this excerpt; global C_d MAE dispersion is narrower than for NeuralFoil `xxsmall` (Table 8), with non-negligible far-tail mass for all models (Section 7.5).

the numbers. Geometry-disjoint generalisation is *not* tested: rows repeat, manifests lack family labels, and the 4096-row slice is the first on-disk prefix, not a stratified sample. Lineage-clean splits need external catalog metadata.

Stall metrics drop rows without a usable peak neighbourhood; that is selection bias by construction. Table 1 lists how many rows remain, not why others fail.

Repository autoregressive exports use $K{=}0$ so NeuralFoil stays on its training distribution; deployed sweeps should use the `xxxlarge` warmstart in Sections 8–7.4.

NeuralFoil parameter counts come from published cards; FoilCORE's count is a direct tensor walk (Table 2).

Bootstrap intervals in Table 6 resample one fitted checkpoint, not an ensemble over training rerolls.

Tails. Section 7.5: pooled residuals are not Gaussian in the wings. Means alone miss tail risk; p_{99}, JSON maxima, and Section 7.8 give extremes on the benchmarked prefix.

We did not run an automated XFOIL [7] resweep at matched panel density; figures use stored polar supervision from the training pipeline.

Thickness proxy. $\widehat{t/c}$ tertiles (Table 9; Figure 17) use a simple patch-split heuristic, not published NACA thickness rules. Treat tertile plots as exploratory, not as ground truth for thickness families.

Ethics Statement

This work relies exclusively on simulated aerodynamic quantities and public-style coordinate conventions; it does not involve human subjects, sensitive personal data, or field experiments.

Acknowledgments

Model training was run on AMD Instinct MI300X accelerators that AMD provided free of charge.

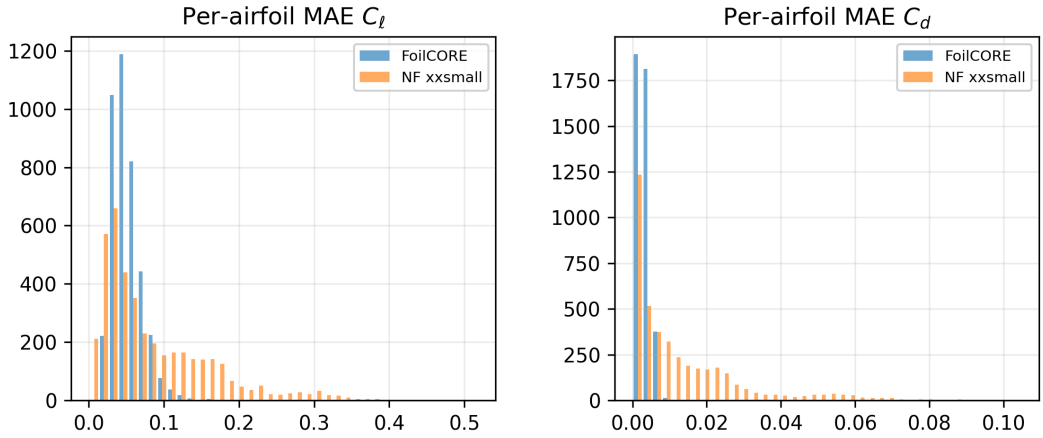


Figure 16: Overlapping per-airfoil MAE histograms for FoilCORE vs. NeuralFoil `xxsmall` under teacher forcing; compare tail mass with Section 7.5.

10 Conclusion

A 6394-scalar causal model (Table 2) matches or beats NeuralFoil `xxsmall` and `xxxlarge` on pooled teacher-forced MAE over the first ∼4096 withheld test rows when past coefficients are supplied to the decoder. Pooled means hide tail risk; heavy tails and row-wise win rates (Sections 7.5 and 7.8) matter for deployment.

Greedy autoregression without prefixes raises error versus teacher forcing; NeuralFoil `xxxlarge` prefixes help in open loop (Section 7.4). Further work could add catalogued geometry hold-outs, multi-seed training ensembles, full withheld-grid structural ablations, and clearer latency numbers for stacked models.

A Full Hyperparameters

Table 3 lists hyperparameters for the evaluated checkpoint.

B Full Tables

Tables 8 and 10 include p99 in addition to p95. The generated JSON also includes maximum per-airfoil MAE, threshold fractions, row-wise paired-comparison rates, peak-C_ℓ error, and stall-angle error.

C Model Details

The exact parameter count is derived from the loaded checkpoint modules. The count includes the polar projection, geometry projection, splitter embedding, attention trunk, all path modules, outer-product layers, MLP refinement layers, and polar decoder. It excludes optimizer state.

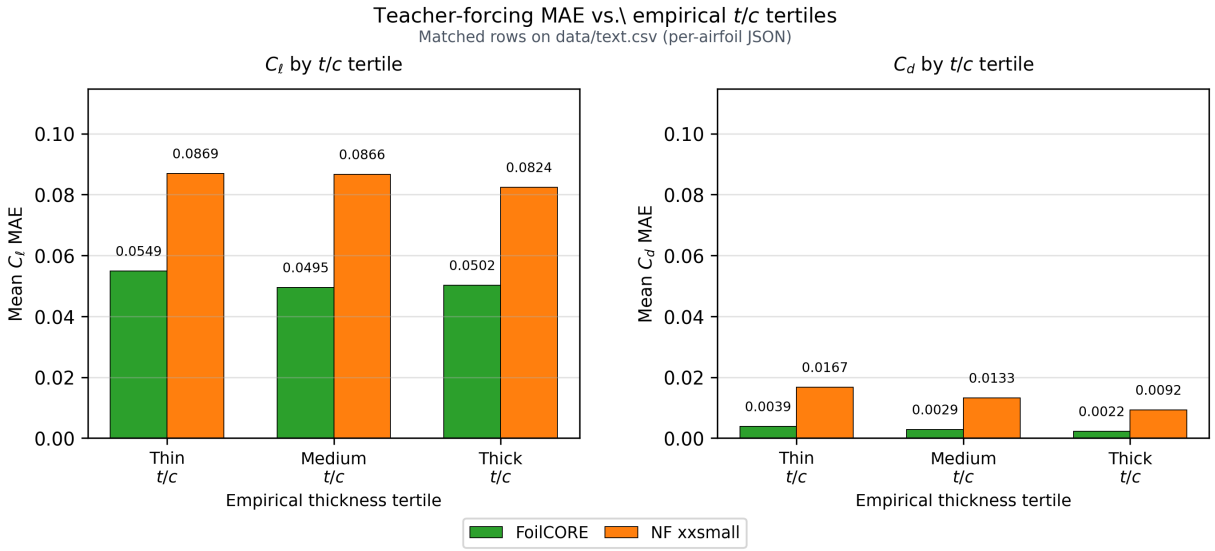


Figure 17: Teacher-forcing mean pooled C_ℓ and C_d MAEs within empirical $\widehat{t/c}$ tertiles (columns match Table 9); bar heights average many polars per bin. Section 7.5 discusses tail risk inside each tertile.

D Data and Software Availability

Source code, preprocessing scripts, and evaluation harnesses ship with the project (see repository metadata). Figures 4–6 are reproducible utilities, not standalone datasets. Combine pretrained checkpoints with third-party data only after checking NeuralFoil licensing.

Regenerating figures and tables. With BigFoil-derived inputs as in Section 3, the companion repository provides Python entry points for figures, tables, metrics, bootstrap summaries, and robustness panels. The README lists checkpoint paths, row caps, batch sizes, and outputs; it also documents withheld-set evaluation against NeuralFoil and α sweeps with NeuralFoil `xxxlarge` rollout prefixes (Section 7.4). Optional width sweeps use the same driver; store exported CPU teacher-forcing metrics next to each checkpoint when comparing widths.

Larger withheld-set metrics (teacher forcing and NeuralFoil baselines) may ship as structured evaluation outputs with the code release where licensing allows; legacy regenerated tables may reference earlier CPU metric snapshots. Raw polar CSVs are excluded from redistribution.

A JSON file with model configuration, training metadata, and normalization tensors ships with the companion bundle.

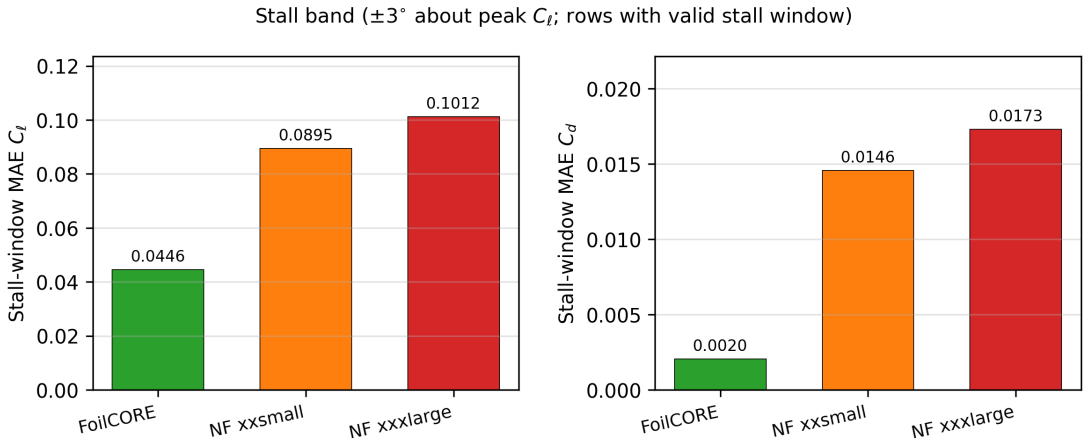


Figure 18: Stall-window C_ℓ and C_d MAE ($\pm 3^\circ$ about peak α at maximum C_ℓ); bar labels coincide with pooled means in Table 5 (including stall-window $C_d \approx 0.002\,05$ for FoilCORE, $\approx 0.014\,55$ for NeuralFoil `xxsmall`, and $\approx 0.017\,29$ for NeuralFoil `xxxlarge`). Drag shows the largest absolute separation on this excerpt.

E High-drag failure excerpts

Table 15: Highest per-airfoil FoilCORE C_d MAE on the benchmarked withheld-test excerpt (five worst cases). Chord-normalised outlines correspond to the indexed evaluation rows referenced in the companion export.

Airfoil (chord norm.)	C_ℓ MAE	C_d MAE
	0.517 24	0.030 36
	0.246 69	0.013 31
	0.160 68	0.010 11
	0.162 87	0.010 10
	0.184 77	0.009 28

Diagnostics. Reynolds numbers here are not uniformly low: Re is about 7.5×10^5–1.8×10^7 (median near 5×10^6 on the withheld prefix). The two worst paired rows (943–942) are thick sections (about 10.8 % chord vertical span by a simple heuristic) at Re $\approx 1.8 \times 10^7$ and 6×10^6, where separation and wake drag are hard for a small model. Other high-error rows are thinner but

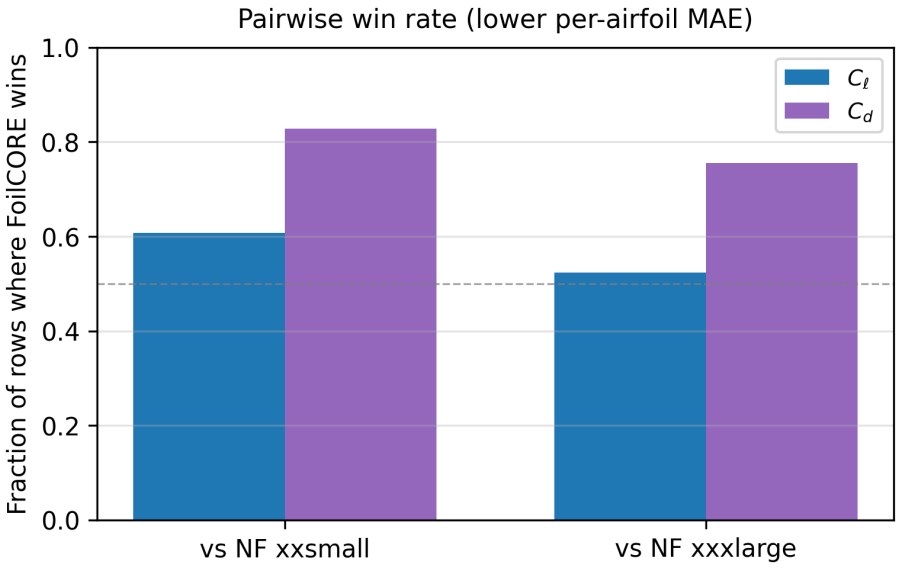


Figure 19: Row-wise paired comparison on the paper excerpt: fraction of airfoils where FoilCORE attains strictly lower per-airfoil MAE than NeuralFoil `xxsmall` (60.8 % for C_ℓ, 82.8 % for C_d) or `xxxlarge` (52.4 %, 75.5 %).

still in the upper tail of drag error; the pattern is high drag and a two-output head, not a single "thin airfoil / low Re" rule.

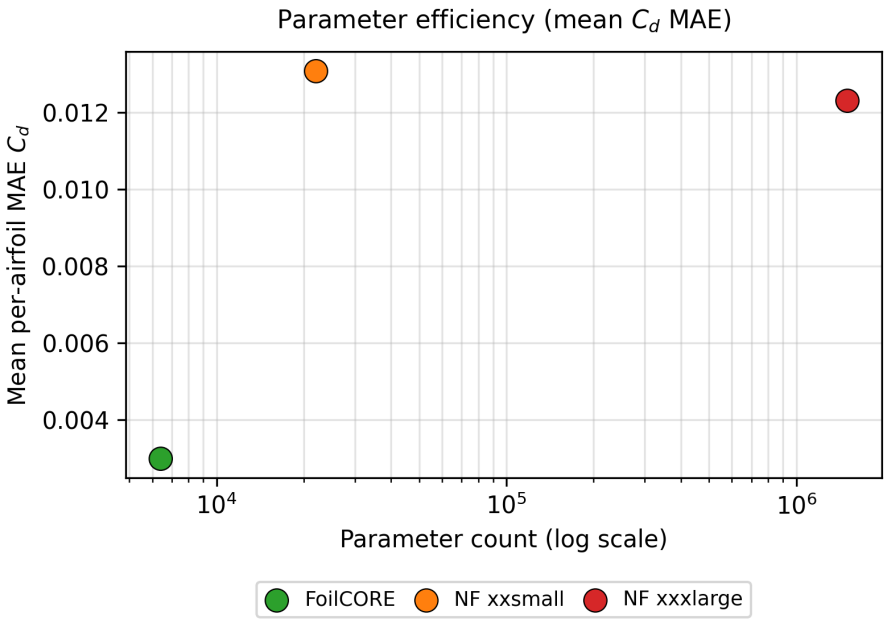
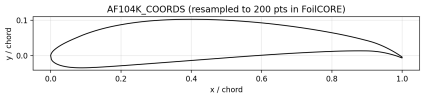


Figure 20: Mean C_d MAE versus parameter count under teacher forcing. FoilCORE sits far to the left of the NeuralFoil checkpoints.

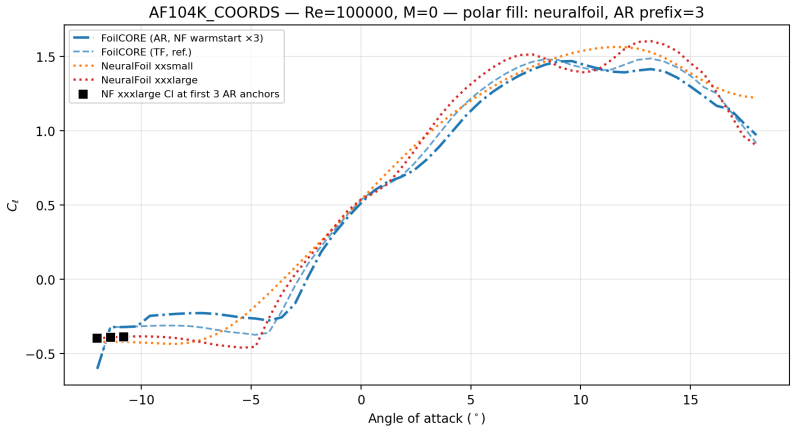
Table 12: CPU polar-level inference latency (batch one curve). FoilCORE uses one teacher-forced forward; NeuralFoil uses `get_aero_from_coordinates` over the full α grid for the same rows. Relative timings are stable across environments; absolute milliseconds shift with hardware and threading.

Model	Params	Time / polar[†]	Rel. time ratio
FoilCORE (TF)	6394	1.362 ± 0.083	1.00×
NeuralFoil `xxsmall`	22 000	0.633 ± 0.040	0.46×
NeuralFoil `xxxlarge`	1 500 000	3.468 ± 0.447	2.55×

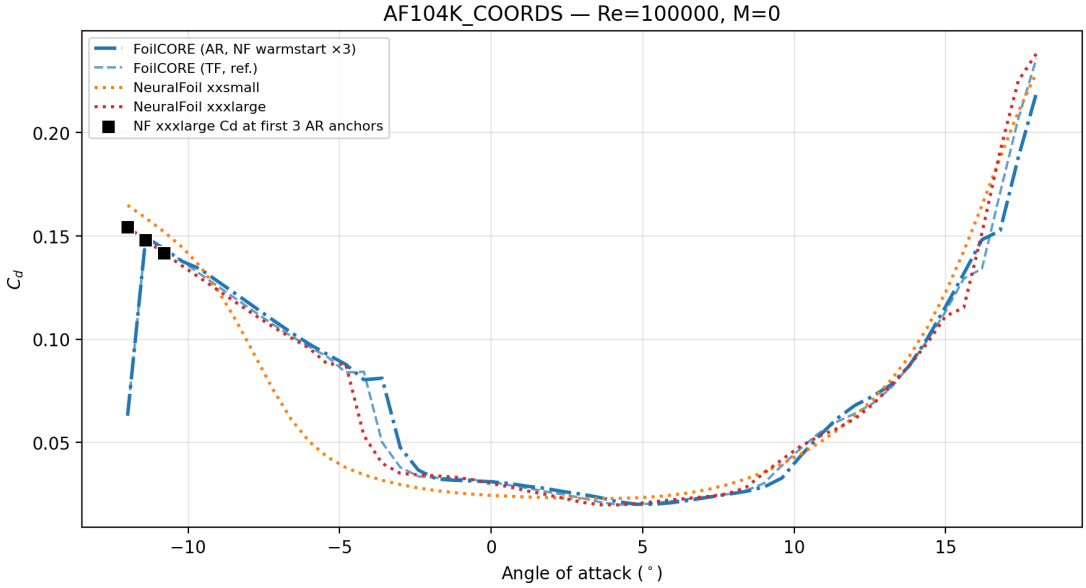
[†] CPU wall time in milliseconds for one full polar (batch 1; PyTorch CPU for FoilCORE; NeuralFoil on CPU NumPy; BLAS threads capped at `1`). Mean polar length ≈ 67.9 stations per curve on this draw. **Ratio column:** values below 1 are faster than FoilCORE; above 1 are slower.



(a) Chord-normalised AF104K section (resampled patch input).

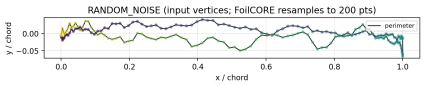


(b) C_ℓ versus α for FoilCORE (TF, dashed; AR with K=3 NF `xxxlarge` seeds, dash-dot) versus NeuralFoil `xxsmall`/`xxxlarge`.

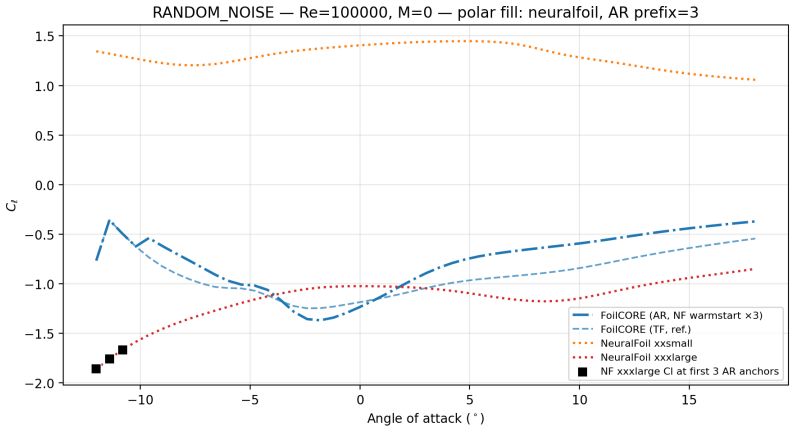


(c) C_d for the same sweep; markers highlight the first K warmstart anchors.

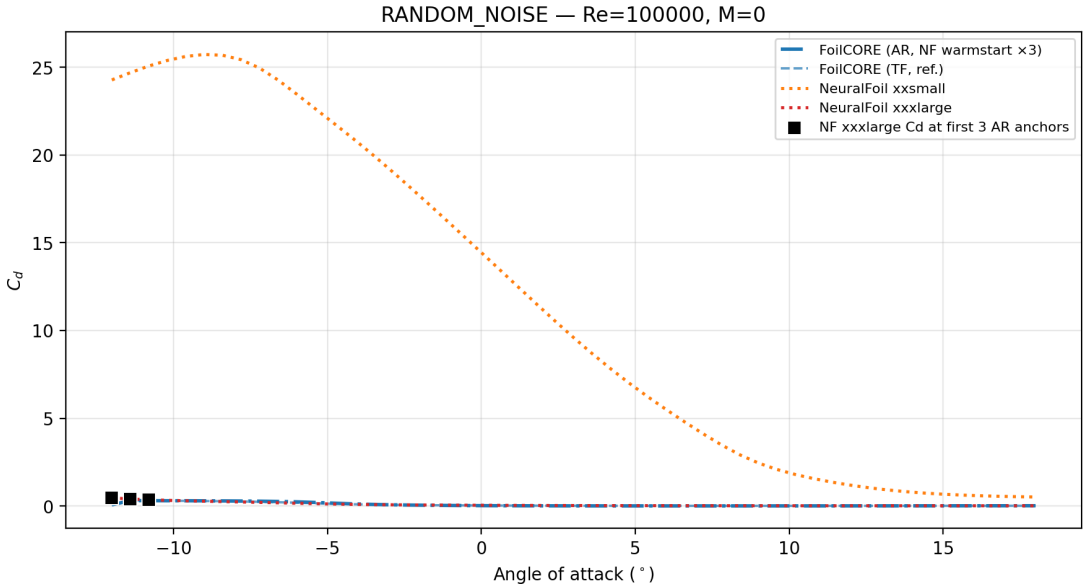
Figure 21: Out-of-training-catalogue AF104K coordinates with NeuralFoil-`xxxlarge` polar prefill (Re=1×10^5).



(a) Synthetic chord-normalised perimeter (FoilCORE resamples to patch resolution).



(b) C_ℓ versus α; markers: first K NF `xxxlarge` anchors.



(c) C_d for the same sweep; xxsmall departs strongly from both FoilCORE traces on this geometry.

Figure 22: Smoothed random-noise perimeter (seed 42) with K=3 NeuralFoil-`xxxlarge` warmstart at Re=1×10^5.

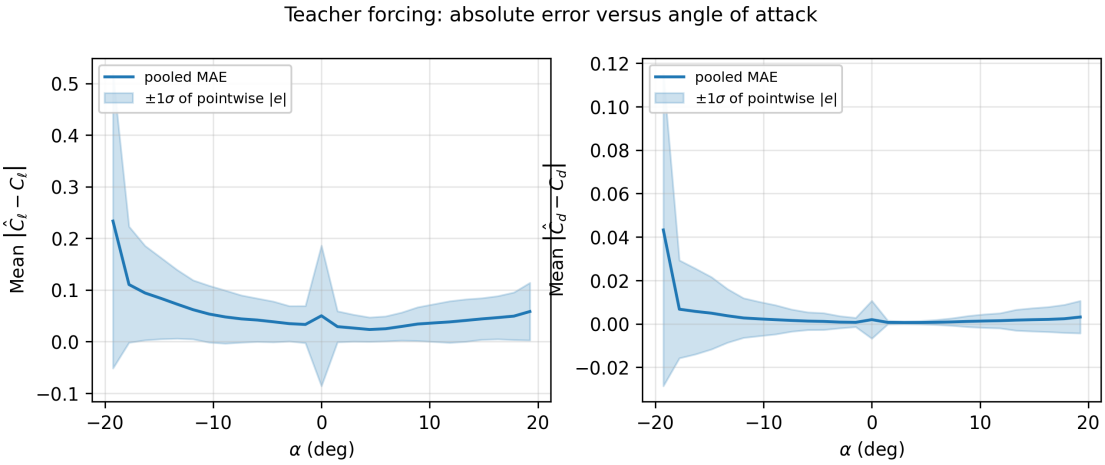


Figure 23: Teacher forcing on the withheld test prefix: pooled mean absolute C_ℓ and C_d errors in α bins (solid) with $\pm 1\sigma$ dispersion of pointwise absolute errors inside each bin (shaded). Labels follow the same panel-derived polar truth as Figure 9; compare bias-oriented Figure 10.

Table 13: Favorable-throughput polar timing. FoilCORE: batched teacher-forced forwards on CUDA if available (else MPS/CPU), per-polar ms = batch wall / batch size; autocast on CUDA only. NeuralFoil: public NumPy API, one call per polar with full α (no GPU); default host BLAS threading.

Model	Params	Time / polar[†]	Rel. time ratio
FoilCORE (TF)	6394	1.268	1.00×
NeuralFoil `xxsmall`	22 000	0.872 ± 2.620	0.69×
NeuralFoil `xxxlarge`	1 500 000	3.470 ± 0.446	2.74×

[†] Milliseconds per polar under favorable settings: FoilCORE batched on the best available accelerator with batch 128; NeuralFoil full-α API on CPU NumPy (BLAS threading left at process defaults). Mean polar length ≈ 67.9 stations per curve on this draw. **Ratio column:** values below 1 are faster than FoilCORE; above 1 are slower.

Table 14: Pooled teacher-forced MAE versus injected α noise (Gaussian, scale relative to training σ_α); same checkpoint as Table 5, prefix length capped as in asset script.

σ_α noise (× train std)	MAE C_ℓ	MAE C_d
0.00	0.053 23	0.003 09
0.01	0.054 28	0.003 36
0.05	0.078 26	0.006 07
0.10	0.152 20	0.010 11

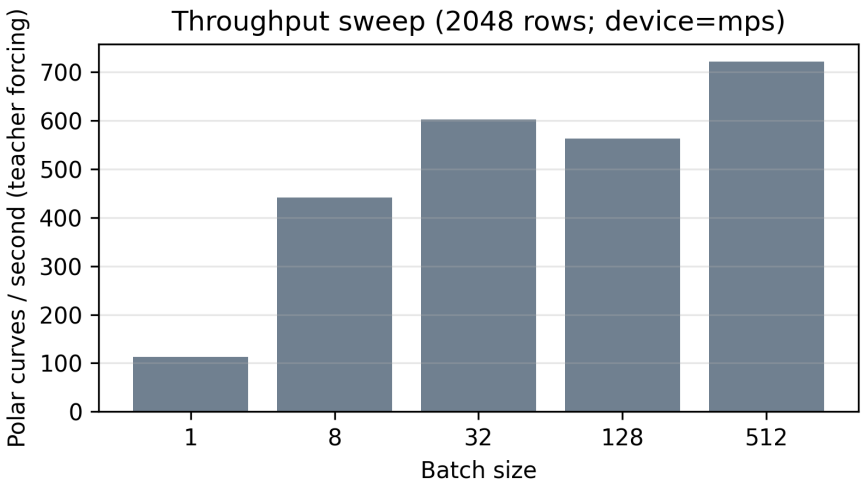


Figure 24: Throughput sweep (curves per second) on a fixed 2048-row prefix as batch size changes; measurements include PyTorch dataloading on the host used for asset generation.

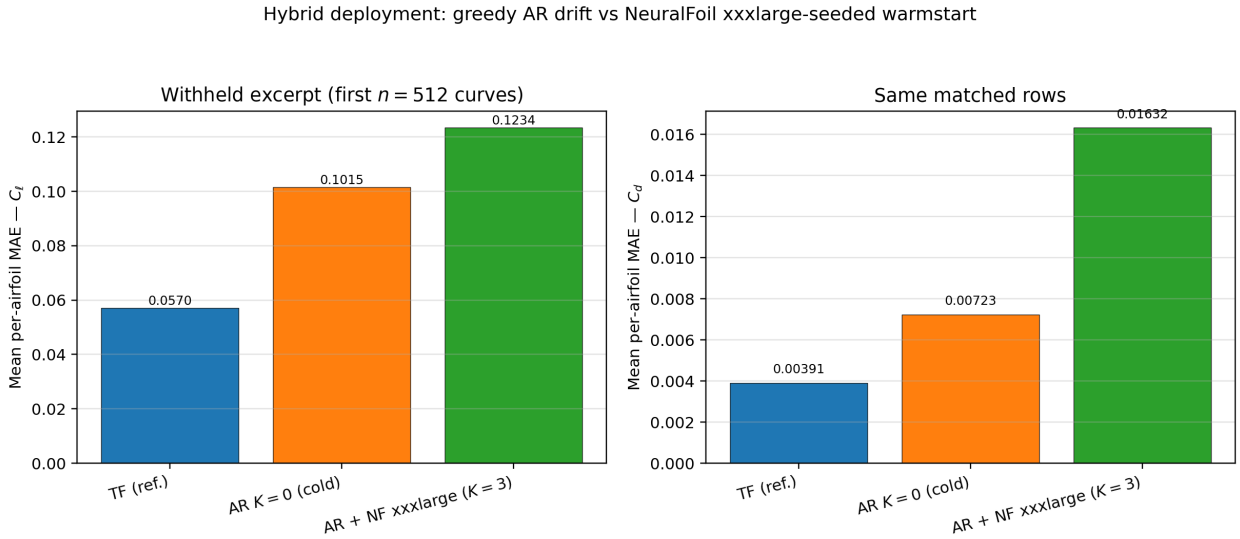


Figure 25: Pooled per-airfoil MAE summary for FoilCORE teacher forcing versus autoregressive decoding with NeuralFoil `xxxlarge` warm-starts of lengths $K = 0, \ldots, 5$ (structured export on a withheld prefix). Baseline K=0 matches Section 7.4; larger K reallocates information between the prefix and greedy tail.

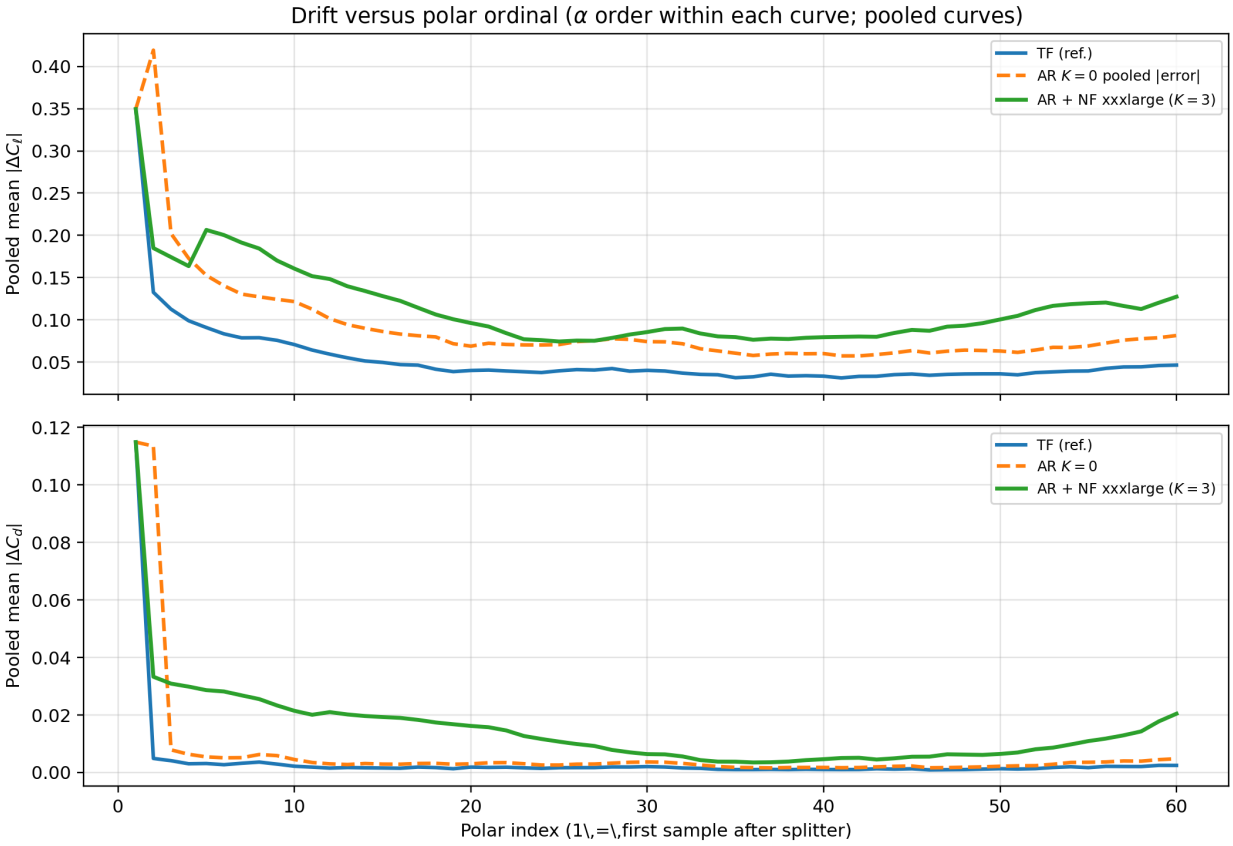


Figure 26: Ordinal-index diagnostics for the same warmstart grid: pooled mean absolute $\hat{C}_\ell$ and $\hat{C}_d$ errors at early polar positions under teacher forcing versus each AR prefix length.

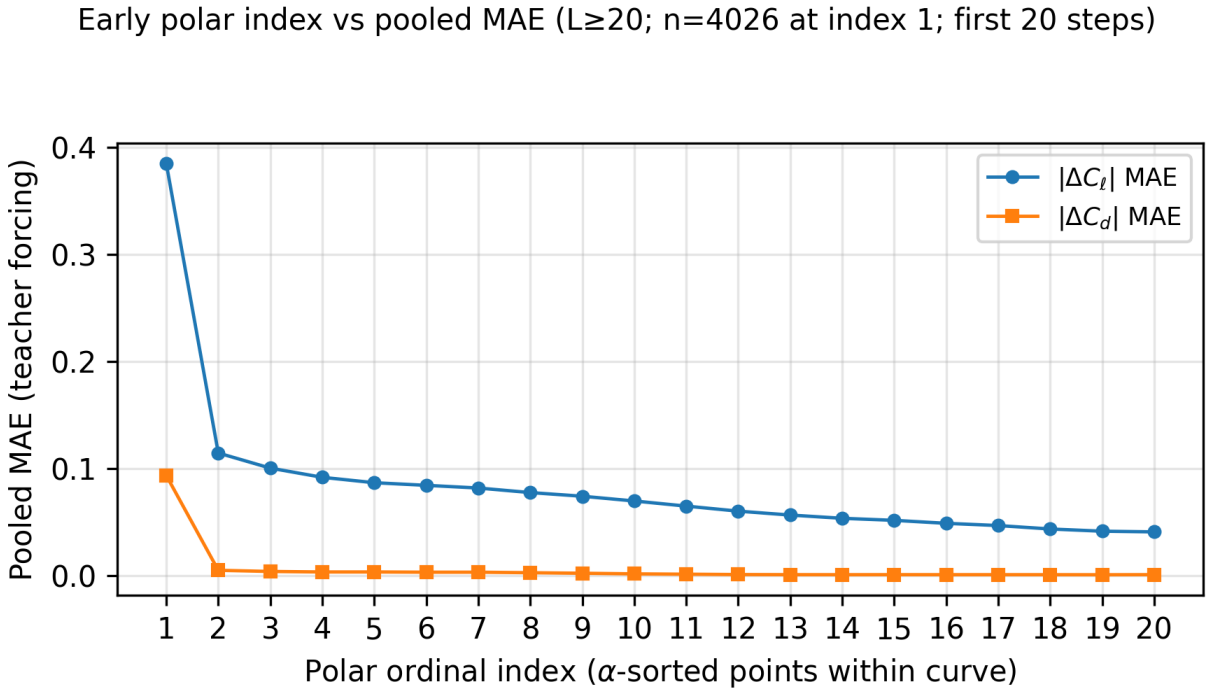


Figure 27: Teacher-forced FoilCORE: pooled mean absolute $\hat{C}_\ell - C_\ell$ and $\hat{C}_d - C_d$ at α-ordinal polar indices 1 to 20 (only curves with $L \geq 20$); the regenerated figure title annotates pooled counts.

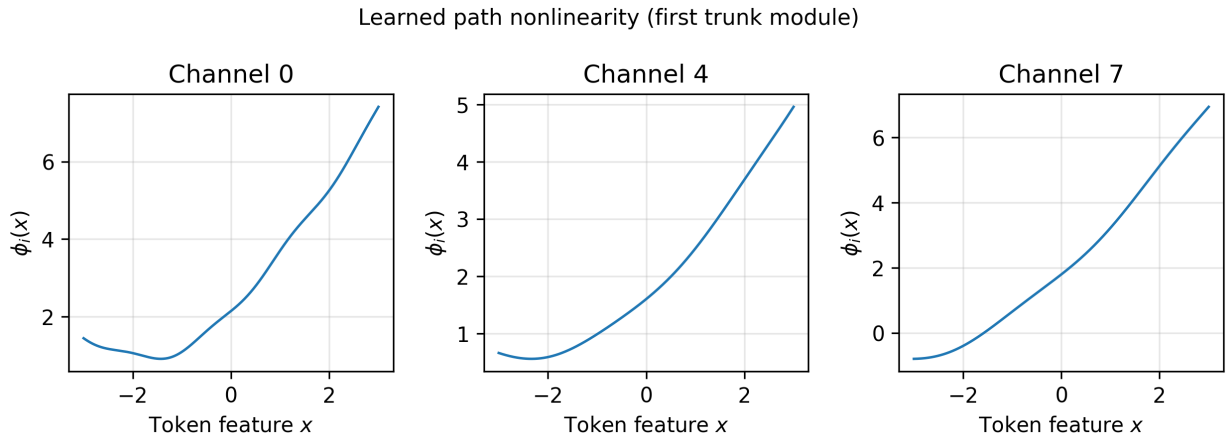


Figure 28: Learned featurewise path shapes $\phi_i(x)$ for three channels in the first trunk path module after training.

References

[1] Ira H. Abbott and Albert E. von Doenhoff. *Theory of Wing Sections, Including a Summary of Airfoil Data*. Dover Publications, New York, 1959. Classic reference for lift polars and stall-related behaviour on canonical airfoil families.

[2] Kajal Agarwal, Venkat Vijaykrishnan, Debadatta Mohanty, and Manikandan Murugaiah. A comprehensive dataset of aerodynamic coefficients of publicly available airfoils. *Data*, 9(5):64, 2024.

[3] Avneh Bhatia. FoilGen2: Learning coupled latent spaces for hybrid and performance-driven airfoil generation. *Engineering Archive* (22 January 2026). https://doi.org/10.31224/6319, 2026.

[4] Avneh Bhatia. Ultra-compact geometry-aware transformers for airfoil polar prediction: FoilForm. *Engineering Archive* (April 2026). https://doi.org/10.31224/6870, 2026.

[5] Felix Bonnet, Joseph Mazari, Paola Cinnella, and Patrick Gallinari. AirfRANS: High fidelity computational fluid dynamics dataset for approximating RANS solutions. In *NeurIPS Datasets and Benchmarks Track*, 2022.

[6] Haiwei Chen, Lun He, Weibin Qian, and Shuai Wang. Multiple aerodynamic coefficient prediction of airfoils using a convolutional neural network. *Symmetry*, 12(4):544, 2020.

[7] Mark Drela. XFOIL: An analysis and design system for low reynolds number airfoils. *Low Reynolds Number Aerodynamics*, pages 1–12, 1989.

[8] Alexander I. J. Forrester, András Sóbester, and Andy J. Keane. *Engineering Design via Surrogate Modelling*. Wiley, 2008.

[9] Xinrong Li, Xiaojia Du, and Joaquim R. R. A. Martins. Machine learning in aerodynamic shape optimization. *Progress in Aerospace Sciences*, 134:100849, 2022.

[10] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. arXiv:1711.05101. https://arxiv.org/abs/1711.05101.

[11] Mike Quayle. BigFoil airfoil polar and geometry catalog. https://www.bigfoil.com/, 2025.

[12] Maziar Raissi, Paris Perdikaris, and George E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[13] Steffen Rendle. Factorization machines. In *IEEE International Conference on Data Mining*, pages 995–1000, 2010.

[14] V. Sekar, Mingfu Zhang, Chang Shu, and B. C. Khoo. Inverse design of airfoil using a deep convolutional neural network. *AIAA Journal*, 57(3):993–1003, 2019.

[15] Peter Sharpe. NeuralFoil: An airfoil aerodynamics analysis tool using physics-informed machine learning. *arXiv:2503.16323*, 2025.

[16] Peter Sharpe. NeuralFoil software repository. https://github.com/peterdsharpe/NeuralFoil, 2025. Accessed 17 May 2026.

[17] University of Illinois at Urbana–Champaign. UIUC Selig airfoil coordinate database. https://m-selig.ae.illinois.edu/ads/coord_database.html, 2026. Public-domain coordinate repository for canonical airfoil shapes.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.