SCALING LAWS FOR ORDER FLOW GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Limit order books capture financial markets at their most granular level, and autoregressive generative models have emerged as a powerful approach to modeling their dynamics, yet their scaling behavior remains poorly understood. We study how the size of state-space models (Mamba-3) and cumulative training-token exposure affect cross-entropy (CE) on S&P 500 order flow, examining both logged training CE from 2022–2025 and forward-time test CE on January 2026 data. We analyse 32 training lineages across 12 model sizes, spanning over two orders of magnitude (2.63–293.28 million parameters), with 249 January checkpoints evaluated on the same 487 tickers. Under the realised training budgets, January endpoint CE improves sharply across the smallest models and then varies non-monotonically with size. An additive parameter–exposure law predicts held-out lineages and sizes within the observed range more accurately when restrictive exponent bounds are relaxed. Most whole-size prediction error comes from extrapolating to the smallest model, while coefficient estimates remain sensitive to size coverage and fitting choices. We also evaluate generated continuations with LOB-Bench, comparing observed and generated event and book-state distributions after processing generated events through a matching engine. We relate the resulting discrepancy scores to CE at matching checkpoints. Using activity-derived hardware-cost estimates, we compare IsoFLOP quadratics with the additive surface at fixed compute budgets. The results show where the additive law describes order-flow prediction and where its fitted exponents depend on the experiment and fitting choices.

1 INTRODUCTION

Limit-order-book (LOB) data is an abundant high-resolution source of sequential data outside natural language. LOBSTER reconstructs the evolving book and its associated message stream from exchange data (Huang & Polak, 2011). Once tokenised, these streams support autoregressive models that can score observed events and generate market continuations. Training larger models raises a practical question: how should parameters and training exposure be varied to improve prediction of future order flow?

Language-model scaling laws provide a starting point (Kaplan et al., 2020; Hoffmann et al., 2022), but their accuracy for order-flow models remains an empirical question. Overlapping market windows, temporal variation, model-size-dependent training horizons, and a small number of independent training runs all affect what an observed loss surface can identify. Many checkpoints from one training trajectory do not substitute for independently trained models. Likewise, next-token cross-entropy and the distributional quality of generated continuations measure different properties.

We study these questions in an S&P 500 Mamba-3 size–exposure sweep. Our primary outcome is January 2026 cross-entropy, with equal weight per ticker. Repeated evaluations are averaged within checkpoints, and checkpoint continuations are grouped into training lineages. We compare completed training budgets and show interrupted runs separately. We compare additive fits, reduced models, fixed-reference exponent fits, and interpolation within observed support. The additive surface predicts much of the observed size–exposure range reasonably well; extrapolation to the smallest model remains unreliable. We distinguish prediction quality from sensitivity of the fitted exponents.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

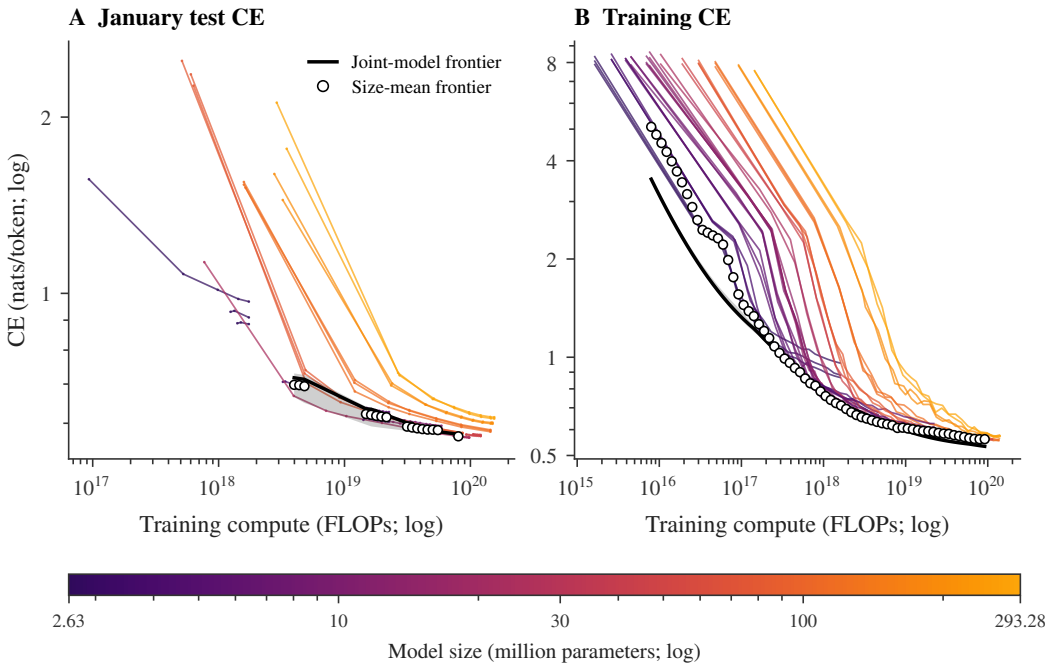

Figure 1: Mamba-3 January test CE and logged training CE against activity-derived training compute. Colours encode model size; each thin line is one training lineage. Training curves show means in fixed log-compute bins. Solid black curves minimise Equation 2 over profiled model sizes supported at that budget, using $D = C/q_i$; open circles mark the observed size-mean minima. Shading gives pointwise 95% bands from 2,000 lineage-bootstrap draws, each repeating the size selection and minimisation.

We report the trajectories, fitted curves, uncertainty and prediction errors needed to assess these scaling relationships. Hardware-cost estimates also allow a comparison between local IsoFLOP fits and the joint parameter–token surface.

Our main contribution is the identification of an empirical scaling law for numerical event sequences, extending the study of scaling beyond natural language and images to a different data modality: tokenised limit-order-book events. We quantify how predictive cross-entropy depends jointly on model size and cumulative training-token exposure, and provide an explicit fitted law with numerical coefficients (Equation 3). This relationship provides a quantitative reference for institutions planning larger order-flow models, helping them assess choices of model size and training-token budget and, together with compute-cost estimates, plan the required computational resources.

2 Background

Books, events and prediction. A limit order book records outstanding buy and sell orders, ranked by price and arrival time. New submissions add liquidity, cancellations remove it, and executions consume orders when prices match. The best bid and ask are the highest buy price and lowest sell price; the ask minus the bid is the spread. At these prices, queue imbalance is $(v_b - v_a)/(v_b + v_a)$, where v_b and v_a are the resting bid and ask volumes. The book is a state, whereas order flow is the sequence of events that changes it. We encode event fields as discrete tokens and train an autoregressive model to predict them from preceding tokens and book information. The representation and the alignment needed for causal book conditioning are described in Appendix A.

Scaling variables and allocation. We distinguish the total parameter count N, cumulative training-token exposure D, and training compute C. Exposure counts token occurrences, including repeated visits to the same data. The additive law in Equation 2 describes loss as an empirical offset E plus separate parameter and exposure terms. Its positive exponents α and β describe the decline

of those terms when the other variable is held fixed. They are different from exponents describing how an optimal parameter count or token budget grows with compute. The fitted offset should not be interpreted as the market's irreducible entropy. At fixed compute, a model with a higher cost per token receives less exposure: for a measured size N_i, $D = C/q_i$, where q_i is its estimated cost per token. Our allocation comparisons therefore evaluate the fitted surface on measured sizes, each at its corresponding exposure.

Likelihood and generated fidelity. Teacher-forced CE scores token predictions along observed event histories. A rollout instead repeatedly samples an event, applies it to a matching engine, and uses the resulting book and sampled history to predict the next event. The matching engine maintains outstanding orders and applies market rules; generated tokens alone do not define a consistent evolving book. Prediction errors can accumulate through this feedback, so a lower CE need not imply better agreement for every generated market statistic. We assess this agreement with LOB-Bench feature-distribution distances. The generation evaluation uses a separate protocol from the multi-ticker CE evaluation.

3 Experimental design

Corpus and representation. The models train on 488 S&P 500 tickers from January 2022 to December 2025. Inputs contain 500-message windows encoded as 26 tokens per message, with a book-conditioned message-generation architecture. The architecture uses Mamba-3 SISO blocks, with two message layers, two book layers and six fused layers. The book layers sit either side of the book projection, before the streams merge. Appendix A details the encoding and topology, with the raw-record-to-token diagram (Figure 5) and the dual-stream architecture (Figure 6). Kernel weights use Muon at base learning rate 10^{-2}, named SSM parameters use Adam, and remaining parameters use AdamW. The SSM base rate is approximately $0.512/d_{\mathrm{model}}$. Muon and AdamW use weight decay 0.005, dropout is zero, and the learning-rate schedule has 1% linear warmup followed by cosine decay to 5% of its peak. Appendix B describes the schedule, continuation behaviour and intended width-scaling prescription. Architecture and representation follow the selective state-space and tokenised order-flow lines of work (Gu & Dao, 2023; Dao & Gu, 2024; Lahoti et al., 2026; Nagy et al., 2023).

Model size and cumulative exposure. We use total parameter counts throughout. Nominal run labels serve as identifiers: the smallest label corresponds to 2.63 million parameters. Each local update processes 128 sequences across the workers, giving 1,664,000 token occurrences per step. Workers take ten local updates between cross-node parameter averages; gradients are averaged within each four-GPU node. Thus $D = 1{,}664{,}000s$ at cumulative step s, including updates before and after checkpoint restores. Repeated token occurrences count towards D. Terminal budgets were chosen using the per-size hardware-cost estimates to obtain overlapping trajectories for roughly two or three IsoFLOP slices. The resulting horizons vary with model size; endpoint comparisons therefore reflect both size and the assigned exposure. Node counts also vary from 2 to 32, reducing the local batch from 64 to 4 sequences per node while keeping 128 sequences across workers. Width therefore changes local optimisation dynamics as well as model capacity; this sweep does not isolate their effects.

Coverage and completion. The training plan specifies 34 runs across 12 sizes. A run is complete when training reaches its intended budget. The 32 completed lineages contribute 249 January CE checkpoints; 2 interrupted runs are shown in the trajectory figure. The endpoint is the first evaluated checkpoint reaching the intended budget; small overshoots arise from checkpoint spacing. Post-budget observations do not extend the endpoint or redefine the late-budget windows.

A training lineage comprises an initial run and its checkpoint continuations. No two completed lineages share an initial run; Appendix D documents seed-label changes across restores.

January test CE. The primary outcome is forward-time test CE in nats per token over 487 training-universe tickers. Every evaluation covers the same tickers, with one result per ticker. For checkpoint

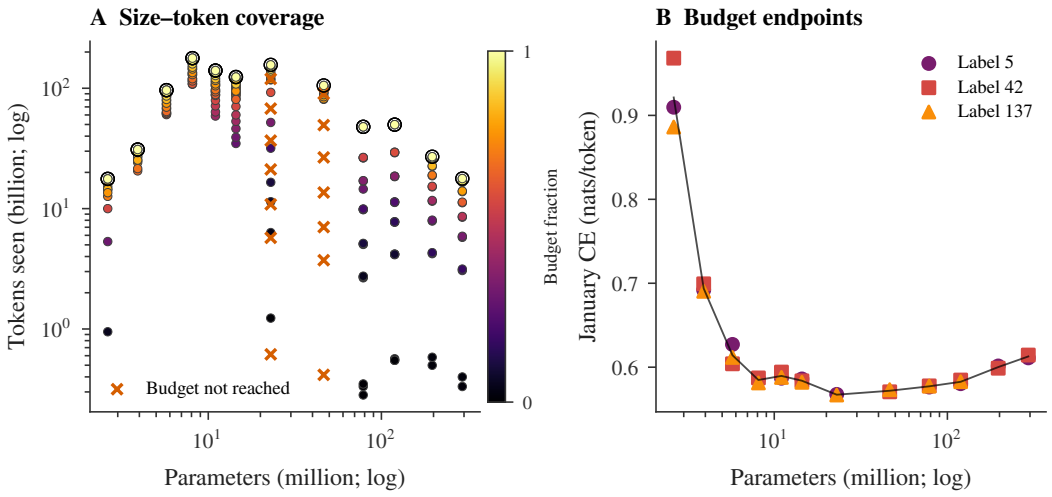

Figure 2: Experimental coverage. Left: checkpoints coloured by budget fraction, with endpoints circled. Crosses mark 14 checkpoints from two interrupted runs, excluded from fitting. Right: endpoint CE by model size; the line connects size means. Markers distinguish runs labelled 5, 42, and 137, corresponding to random seeds 5, 42, and 137, respectively; Appendix D explains the seed-label changes in resumed runs.

c, evaluation replicate e, and ticker t, we calculate

$$\widehat{L}_c = \frac{1}{|\mathcal{E}_c|} \sum_{e \in \mathcal{E}_c} \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \widehat{L}_{cet}. \tag{1}$$

Repeated evaluations contribute one checkpoint mean; independent replication is counted by training lineage. CE is averaged over the scored token positions throughout each window. Equal ticker weights do not imply uniform sampling across trading days; the sampled-window lists are unavailable. We analyse logged training CE separately. The logging code averages ten local losses; Appendix B describes this summary and its limits.

4 Fitting and Uncertainty

Functional form and weights. We fit

$$L(N, D) = E + a\left(\frac{N}{N_0}\right)^{-\alpha} + b\left(\frac{D}{D_0}\right)^{-\beta}, \qquad N_0 = 10^7, \quad D_0 = 10^{10}, \tag{2}$$

where $a, b > 0$ and $E \geq 0$. The parameter E represents the asymptotic cross-entropy floor of the fitted law as model size and training-token exposure increase. It is motivated by the irreducible uncertainty in predicting the next token given the available context, but the fitted value need not equal the true conditional entropy of the data. The reference fit constrains $E \leq \min L$ and $\alpha, \beta \in [0.1, 2]$. We minimise Huber loss on log residuals with threshold 10^{-3}, assigning equal total weight to each training lineage and equal weights to selected checkpoints within it. Optimisation uses up to nine deterministic starting points, analytic gradients checked against numerical derivatives, and convergence checks. A boundary estimate reflects the imposed limit on that parameter.

We fit budget endpoints and the final 10%, 25%, 40%, and 50% of intended budgets. Each selection includes every checkpoint in its budget window, regardless of loss. Late checkpoints provide comparisons near the end of cosine decay. We compare the additive law with parameter-only and data-only models, a fixed data exponent $\beta = 0.28$, and fixed $(\alpha, \beta) = (0.34, 0.28)$ reference exponents from Hoffmann et al. (2022).

Identification and prediction. We examine residuals against size and budget fraction, one-dimensional objective profiles with other parameters refitted, wider exponent bounds, a raised floor

cap, robust-loss thresholds, squared log loss, and size exclusions. Prediction tests omit an entire training lineage or an entire model size. All continuations of an omitted lineage stay together. Appendix Table 9 maps each fitting target to its objective and weights. Interpolation uses piecewise-linear predictions in $(\log N, \log D)$ only inside the training fold's convex hull; outside-hull predictions from parametric models are labelled extrapolations. We report aggregate prediction errors descriptively and do not treat overlapping cross-validation folds as independent replications.

Bootstrap. For endpoints and the final-quarter fit, we draw 2,000 bootstrap samples by resampling training lineages within each observed size. Each draw retains the lineage's complete selected trajectory and multiplicity; it does not resample tickers or individual checkpoints. The size grid is fixed, with at least two distinct training lineages at each size. We count failed fits and boundary estimates, and compare quantiles from the first and second halves of the draws. This bootstrap conditions on the observed corpus, evaluation protocol, and size grid; it does not quantify corpus shift or model misspecification.

5 RESULTS

Budget endpoints. At the budget endpoints, mean CE is 0.9214 at 2.626M parameters, reaches its lowest observed size mean of 0.5672 at 22.949M parameters, and is 0.6128 at 293.283M parameters (Figure 2). Training exposure differs across these endpoints, so their comparison combines changes in size and token budget.

Table 1: Additive fits under five budget selections. Every parameter-exponent estimate reaches the upper bound of 2; these values are constraint-limited. RMSE is weighted equally by run.

Selection	Points	Runs	α	β	E	RMSE
Endpoint	32	32	2.000	1.444	0.5661	0.0176
Final 10%	64	32	2.000	0.925	0.5585	0.0179
Final 25%	131	32	2.000	0.855	0.5575	0.0186
Final 40%	175	32	2.000	0.656	0.5503	0.0188
Final 50%	188	32	2.000	0.942	0.5614	0.0195

Sensitivity to checkpoint selection. The parameter exponent reaches the reference upper bound in every budget selection. The endpoint data exponent is 1.444, while the final-quarter estimate is 0.855. Widening the exponent bounds to $[0.01, 5]$ gives $(\alpha, \beta) = (2.675, 0.994)$, whereas excluding the two smallest sizes gives $(0.555, 0.167)$ under the reference bounds. The exponent estimates therefore depend strongly on the fitted size range and constraints. Figure 9 shows the remaining residual structure.

Bootstrap and held-out prediction. In the final-quarter lineage bootstrap, the parameter exponent reaches a bound in 100.0% of successful draws. The data exponent has median 0.919 and a percentile interval of $[0.488, 1.570]$. All 2,000 fits converge. These percentiles measure variation under the fitted constraints and size grid. The 2,000 draws improve the numerical precision of intervals based on only two or three independent training lineages per size.

Table 2: Final-quarter held-out RMSE (nats/token), weighting runs equally. Bounds apply to all free exponents. The final column uses the same 74 of 131 held-out-size observations inside each training fold's convex hull; the all-points columns include extrapolation.

Bounds	Model	Lineage out	Size out: all	Size out: within
$[0.1, 2]$	Additive	0.0262	0.0518	0.0163
$[0.1, 2]$	Parameter only	0.0289	0.0345	0.0176
$[0.1, 2]$	Fixed β	0.0228	0.0495	0.0172
$[0.01, 5]$	Additive	0.0180	0.0487	0.0093
$[0.01, 5]$	Parameter only	0.0245	0.0241	0.0148
$[0.01, 5]$	Fixed β	0.0192	0.0491	0.0139

Prediction depends on bounds and domain. Under the reference bounds, fixing β improves leave-lineage-out error and the parameter-only model improves whole-size error when extrapolations are included. On held-out-size observations within the common size–exposure support, the additive model has RMSE 0.0163, versus 0.0176 for parameter-only. Widening the bounds reduces additive RMSE to 0.0180 for omitted lineages and 0.0093 for supported omitted-size observations, the lowest among the three models compared. All fits with wider bounds converge; the error differences should be interpreted descriptively.

The smallest omitted size contributes 86.1% of the reference surface's run-weighted whole-size squared error; the two extreme sizes together contribute 93.0%. Extrapolation to the smallest model remains difficult with wider bounds. The initial cap therefore restricts the exponent estimate, while most prediction error arises outside the training fold's observed size–exposure range.

Consolidated compute trajectories. Figure 1 overlays all model sizes on a common compute axis. January CE uses 249 checkpoints; training CE uses 10,136 training-loss summaries from the same 32 completed lineages. We derive the fitted frontier from Equation 2 by substituting $D = C/q_i$ and minimising the prediction across profiled model sizes supported at that budget. The January curve uses the full-trajectory surface and its bootstrap fits; training CE has a separate fit with the same functional form, bounds and run-balanced log-Huber objective. The bands come from resampling entire lineages, checking support and selecting the predicted minimum in each draw. Appendix H reports the fit parameters, direct frontier alternatives and a separate held-out model comparison.

Trajectories and uncertainty. Appendix Figure 8 shows the size-specific Mamba-3 trajectories with a full-trajectory additive fit and pointwise 95% lineage-bootstrap bands. The surface fits all 249 January checkpoints with equal total weight per run. Fixed-compute slices interpolate these trajectories wherever they bracket a budget. Its parameter exponent is 2.000, its data exponent is 0.941, and run-balanced RMSE is 0.0344 nats/token. Across 2,000 paired lineage draws, 0 fits fail and 100.0% reach the parameter-exponent bound. The bands measure uncertainty in the fitted mean curve; new-run prediction also includes the residual variation.

IsoFLOP versus the parametric surface. Using the activity-derived q_i directly, we construct nine fixed-compute slices by interpolation within each trajectory. Both raw-loss and log-loss quadratics pass the initial curvature and bracketing checks at 3 budgets; only 2 pass after every size deletion. For slices passing the initial checks, the raw-quadratic and surface choices on the same profiled size grid differ by factors ranging from 1.00 to 1.59 (Figure 4). Their fitted CE curves differ by RMSE 0.007–0.053 nats/token on those grids. The paired bootstrap repeats curvature, bracketing and all size deletions for each draw; Appendix G reports the corresponding failure counts. These slices combine checkpoints at different schedule phases. The two slices that pass every size deletion agree on 22.949M parameters under the reference surface; the 1.59-fold discrepancy comes entirely from the unstable slice. With wider exponent bounds, the full-trajectory surface selects 14.446M at all three initially accepted budgets (Appendix Table 16). Allocation therefore also depends on the surface constraints. The rejected slices have positive curvature but fail bracketing; slices with only five sizes cannot meet the five-size requirement after a deletion.

Generative fidelity. Figure 3 reports LOB-Bench (Nagy et al., 2025) distances for the completed training cohort. The evaluation uses GOOG January continuations: 250 observed conditioning messages followed by 250 generated messages, across 3,136 sequences per evaluation. We average 21 feature-wise Wasserstein-1 distances after standardising the pooled observed and generated samples; Appendix E defines the metrics and protocol. The mean over complete evaluations, with equal lineage weights, is 0.250 at 2.626M parameters, 0.184 at 119.280M, and 0.205 at 293.283M: generation discrepancy does not decrease monotonically with size. These are trajectory averages under different exposure histories, not matched-budget endpoints. Only 5 of 16 checkpoint evaluations at the largest size contain every feature, limiting interpretation of that mean. CE associations use 185 checkpoints with both evaluations and remain descriptive because checkpoints repeat within lineages.

Fixed compute estimates. At size N_i we use the dmon-derived cost q_i directly: $C_{ij} = q_i D_{ij}$. We compare raw-loss and log-loss quadratics with the full-trajectory surface on identical supported size sets, following the sampling-width, centring, and asymmetry concerns of Czech et al. (2026).

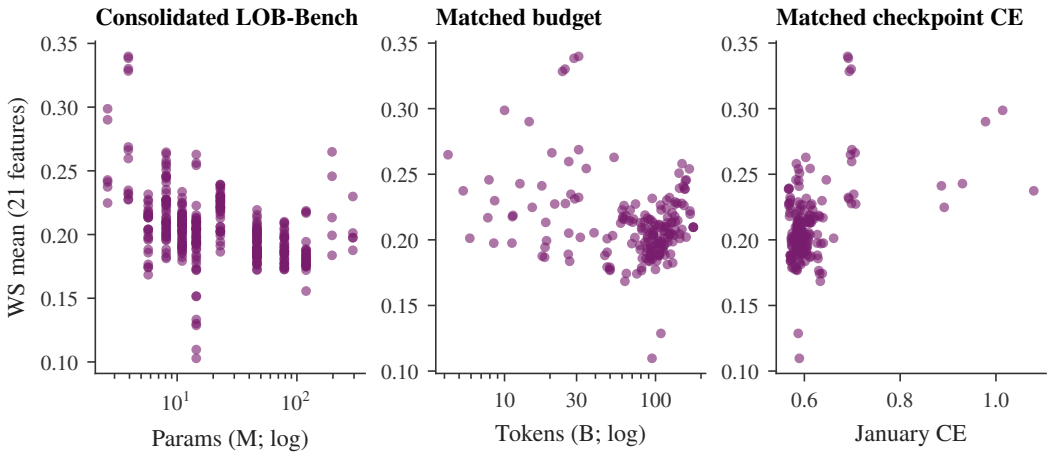

Figure 3: LOB-Bench Wasserstein scores for completed training lineages against model size, cumulative token exposure and January CE. Means require all 21 feature scores. The latter two panels use checkpoints with both generation scores and January CE.

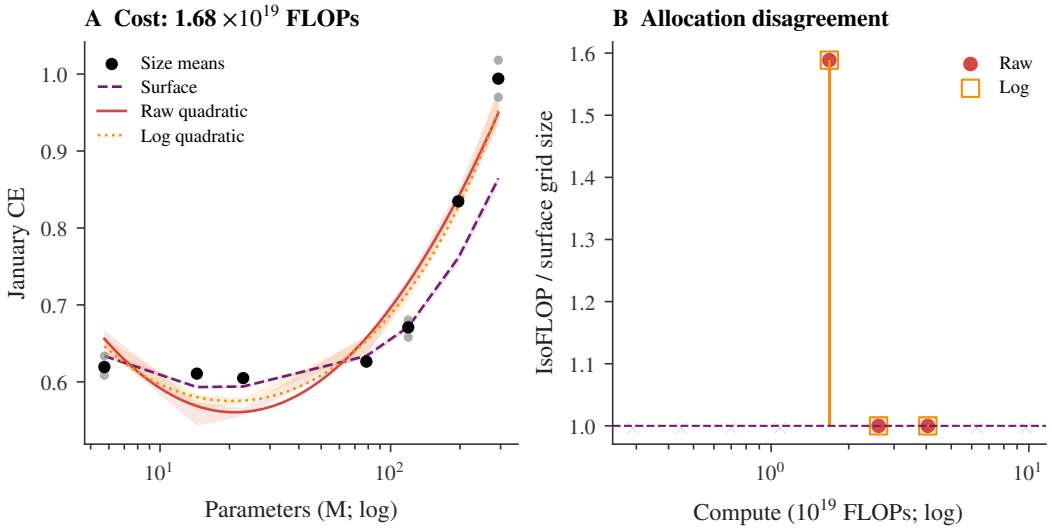

Figure 4: IsoFLOP versus additive fits using fixed activity-derived costs. A: the middle of nine budgets selected from compute coverage, with lineage observations (grey), size means (black), a global surface (purple dashed) and raw/log quadratics (red/orange). Shading shows pointwise 95% bootstrap bands evaluated at observed sizes. B: the ratio of quadratic-selected to surface-selected sizes on the same discrete grid, with paired bootstrap percentile intervals. Crosses along the bottom mark unbracketed slices; they are not zero ratios. IsoFLOP intervals condition on initial validity; deletion stability is reported separately.

These analyses hold the hardware-cost estimates fixed; Appendix F gives the measurements and calculations.

5.1 Validation loss

We complement training and January test CE with fixed-sample validation under the 2022–2025 protocol: the same 30,720 windows across 488 tickers, with 500 messages and 26 token positions per window (399.36M scored positions per checkpoint). Unlike changing training batches, valida-

tion inputs remain fixed; CE weights tickers equally. Scoring tokens do not contribute to training exposure D, and this evaluation does not measure future-market generalisation.

Fitting range. We restrict the primary scaling-law fit to models with at least 5.736M parameters, motivated by three observations. First, the two smallest models, with 2.626M and 3.911M parameters, received only 17.66B and 30.97B training tokens, compared with 96.46B for the next model size. Second, both exposures fall below the mean of 82.63B tokens across the twelve model sizes. Third, their endpoint validation CE remains substantially higher across all evaluated runs: the size means are 0.9145 and 0.6838 nats per token, compared with 0.6185 at 5.736M. These observations indicate poorer prediction of the tokenised order-flow data under the realised training budgets and motivate examining a restricted model-size range. We retain the excluded observations and full-range fits as sensitivity analyses.

Validation scaling. Using Equation 2's objective and reference bounds on final-quarter checkpoints, we obtain

$$\boxed{\widehat{L}_{\mathrm{val}}(N, D) = 0.5278 + 0.0451 \left(\frac{N}{10^7}\right)^{-0.7582} + 0.0626 \left(\frac{D}{10^{10}}\right)^{-0.5410}}. \tag{3}$$

Run-balanced RMSE is 0.0040 nats/token. Across 2,000 successful lineage-bootstrap fits, 95% percentile intervals are $[0.5464, 1.1720]$ for α and $[0.4297, 0.8484]$ for β, conditional on the selected domain.

For applications prioritising approximate arrival times, prices and quantities, exact ns/us components may be less relevant or harder to predict. Table 3 omits their loss contributions without assuming pure noise: inputs, weights and the 26-position denominator remain unchanged, so lower ablated CE does not itself imply improved prediction.

Table 3: Validation exponents for 5.736–293.283M models: 26 endpoints and 114 final-quarter checkpoints.

Scored loss	Endpoint (α, β)	Final 25% (α, β)
Original	(0.784, 0.712)	(0.758, 0.541)
Remove ns	(0.747, 0.691)	(0.736, 0.542)
Remove ns and us	(0.709, 0.699)	(0.704, 0.568)

Table 4 compares checkpoint selections; identical final-25/40/50% observations do not provide independent robustness evidence.

Table 4: Primary validation fits after excluding the two smallest sizes. Model range: 5.736–293.283M parameters. Final 25%, 40%, and 50% select identical available observations.

Selection	Points	Runs	α	β	E	RMSE
Endpoint	26	26	0.784	0.712	0.5344	0.0039
Final 10%	56	26	0.783	0.645	0.5327	0.0039
Final 25%	114	26	0.758	0.541	0.5278	0.0040
Final 40%	114	26	0.758	0.541	0.5278	0.0040
Final 50%	114	26	0.758	0.541	0.5278	0.0040

Table 5 tests fitting and scoring sensitivity.

Appendix J details coverage, scoring and held-out prediction.

6 Related Work

Financial foundation models include general time-series models (Ansari et al., 2024; Woo et al., 2024; Das et al., 2024; Liu et al., 2024; Goswami et al., 2024; Garza et al., 2023; Rasul et al., 2024)

Table 5: Final-quarter validation sensitivity analyses restricted to 5.736–293.283M parameters. Every row uses the same 114 checkpoints from 26 completed lineages.

Setting	Points	α	β	RMSE
Reference	114	0.758	0.541	0.0040
Wider exponent bounds	114	0.758	0.541	0.0040
Relaxed floor bound	114	0.758	0.541	0.0040
Remove ns	114	0.736	0.542	0.0039
Remove ns and us	114	0.704	0.568	0.0021
Sample-weighted CE	114	0.768	0.538	0.0037
Log squared error	114	0.836	0.482	0.0038

and the finance-specific Kronos (Shi et al., 2025). LOBSTER provides billions of finer-grained order messages (Huang & Polak, 2011). LOB modelling spans CNNs (Zhang et al., 2019; Briola et al., 2025), GANs (Coletta et al., 2021; Cont et al., 2023), diffusion (Backhouse et al., 2025; Berti et al., 2025; Wang & Ventre, 2026) and recurrent networks (Hultin et al., 2023); LOB-Bench compares generated distributions (Nagy et al., 2025). Among autoregressive models, LOBS5 combines tokenised LOBSTER data with S5 (Nagy et al., 2023; Smith et al., 2023); MarketGPT and TradeFM use Transformers (Wheeler & Varner, 2024; Kawawa-Beaudan et al., 2026); ByteGen uses lossless byte representations (Li & Chen, 2025); and MarS and LOBERT compress messages into single tokens for longer effective contexts, using a simulation engine and a BERT-style encoder, respectively (Li et al., 2025a; Linna et al., 2025).

The closest prior scaling experiments are MarS and TradeFM. MarS reports an order-model sweep from 2M to 1.02B parameters trained on 32B tokens (Li et al., 2025a). TradeFM fits forward-time test-loss frontiers at three scales, approximately 125M, 250M and 500M parameters, including repeated exposure over four epochs (Kawawa-Beaudan et al., 2026). Our analysis adds replicated experiments across model widths, joint parameter–exposure fits, held-out prediction within and beyond the observed support, sensitivity to fitting constraints, and generation scores at matching checkpoints. Differences in representation, corpus and evaluation limit direct comparison of exponents across these studies.

Kaplan et al. (2020) established language-model power-law scaling, and Hoffmann et al. (2022) introduced an additive parameter–data law with an irreducible-loss floor. Scaling has also been studied for images, video and mathematics (Henighan et al., 2020), time series (Edwards et al., 2024; Yao et al., 2025), world models (Pearce et al., 2024), music (Qu et al., 2024), motion (Baniodeh et al., 2025) and genomes (Nguyen et al., 2024). Related architectures include selective state-space models (Gu & Dao, 2023; Dao & Gu, 2024), gated recurrences (De et al., 2024) and xLSTM (Beck et al., 2026). Methodological work highlights narrow uncertainty intervals (Besiroglu et al., 2024), parameter-counting conventions (Pearce & Song, 2024), output-layer compute and optimisation choices (Porian et al., 2024), reporting standards (Li et al., 2025b), and structural biases in IsoFLOP parabola fits (Czech et al., 2026).

7 Conclusion

Under the realised training budgets, Mamba-3 endpoint CE improves sharply across the smallest sizes and then varies non-monotonically. The additive parameter–exposure surface predicts held-out lineages and supported held-out sizes better with wider exponent bounds. These results support using the fitted relationship within its observed range, with separate checks on coefficient sensitivity and extrapolation. The two deletion-stable IsoFLOP slices agree with the reference surface, but their allocation changes when its exponent bounds are relaxed. The findings apply to the studied corpus, training budgets and evaluation protocol; their validity across future market periods remains untested.

AI USE STATEMENT

Generative AI tools (OpenAI Codex) assisted with data collation; statistical diagnostics, bootstrap and held-out prediction code; synthetic checks; interpretation of analysis outputs; and revisions to the text, tables, figures and LaTeX source. The analysis uses existing training and evaluation results; synthetic examples appear only in labelled methodological checks. The authors remain responsible for the final text, analysis and claims.

REPRODUCIBILITY STATEMENT

Section 3 defines the corpus, checkpoint accounting, and January CE average. Appendix A specifies the representation and architecture, and the subsequent appendices document training runs, fitting diagnostics, generation metrics and compute measurements. The accompanying reproducibility archive contains source snapshots, derived tables, executable analysis scripts, figures, and a mapping from numerical claims to calculations. The analysis uses existing experiments and requires no new neural-network training, inference or evaluation. Raw market messages and model checkpoints are not redistributed in this package.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=gerNCVqqtR.

Alfred Backhouse, Kang Li, Jakob Foerster, Anisoara Calinescu, and Stefan Zohren. Painting the market: generative diffusion models for financial limit order book simulation and forecasting. *arXiv preprint arXiv:2509.05107*, 2025.

Mustafa Baniodeh, Kratarth Goel, Scott Ettinger, Carlos Fuertes, Ari Seff, Tim Shen, Cole Gulino, Chenjie Yang, Ghassen Jerfel, Dokook Choe, Rui Wang, Benjamin Charrow, Vinutha Kallem, Sergio Casas, Rami Al-Rfou, Benjamin Sapp, and Dragomir Anguelov. Scaling laws of motion forecasting and planning – technical report, 2025. URL https://arxiv.org/abs/2506.08228.

Maximilian Beck, Kajetan Schweighofer, Sebastian Böck, Sebastian Lehner, and Sepp Hochreiter. xlstm scaling laws: Competitive performance with linear time-complexity, 2026. URL https://arxiv.org/abs/2510.02228.

Leonardo Berti, Bardh Prenkaj, and Paola Velardi. Trades: Generating realistic market simulations with diffusion models. *arXiv preprint arXiv:2502.07071*, 2025.

Tamay Besiroglu, Ege Erdil, Matthew Barnett, and Josh You. Chinchilla scaling: A replication attempt. *arXiv preprint arXiv:2404.10102*, 2024.

Antonio Briola, Silvia Bartolucci, and Tomaso Aste. Hlob–information persistence and structure in limit order books. *Expert Systems with Applications*, 266:126078, 2025.

Andrea Coletta, Matteo Prata, Michele Conti, Emanuele Mercanti, Novella Bartolini, Aymeric Moulin, Svitlana Vyetrenko, and Tucker Balch. Towards realistic market simulations: a generative adversarial networks approach. In *Proceedings of the Second ACM International Conference on AI in Finance*, pp. 1–9, 2021.

Rama Cont, Mihai Cucuringu, Jonathan Kochems, and Felix Prenzel. Limit order book simulation with generative adversarial networks. *Available at SSRN 4512356*, 2023.

Eric Czech, Zhiwei Xu, Yael Elmatad, Yixin Wang, and William Held. Problems with chinchilla approach 2: Systematic biases in isoflop parabola fits, 2026. URL https://arxiv.org/abs/2603.22339.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality, 2024. URL https://arxiv.org/abs/2405.21060.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Soham De, Samuel L. Smith, Anushan Fernando, Aleksandar Botev, George Cristian-Muraru, Albert Gu, Ruba Haroun, Leonard Berrada, Yutian Chen, Srivatsan Srinivasan, Guillaume Desjardins, Arnaud Doucet, David Budden, Yee Whye Teh, Razvan Pascanu, Nando De Freitas, and Caglar Gulcehre. Griffin: Mixing gated linear recurrences with local attention for efficient language models, 2024. URL https://arxiv.org/abs/2402.19427.

Thomas DP Edwards, James Alvey, Justin Alsing, Nam H Nguyen, and Benjamin D Wandelt. Scaling-laws for large time-series models. *arXiv preprint arXiv:2405.13867*, 2024.

Azul Garza, Cristian Challu, and Max Mergenthaler-Canseco. Timegpt-1. *arXiv preprint arXiv:2310.03589*, 2023.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: a family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2023. URL https://arxiv.org/abs/2312.00752.

Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Ruihong Huang and Tomas Polak. Lobster: Limit order book reconstruction system. *Available at SSRN 1977207*, 2011.

Hanna Hultin, Henrik Hult, Alexandre Proutiere, Samuel Samama, and Ala Tarighati. A generative model of a limit order book using recurrent neural networks. *Quantitative Finance*, 23(6):931–958, 2023.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Maxime Kawawa-Beaudan, Srijan Sood, Kassiani Papasotiriou, Daniel Borrajo, and Manuela Veloso. Tradefm: A generative foundation model for trade-flow and market microstructure. *arXiv preprint arXiv:2602.23784*, 2026.

Aakash Lahoti, Kevin Y. Li, Berlin Chen, Caitlin Wang, Aviv Bick, J. Zico Kolter, Tri Dao, and Albert Gu. Mamba-3: Improved sequence modeling using state space principles, 2026. URL https://arxiv.org/abs/2603.15569.

Junjie Li, Yang Liu, Weiqing Liu, Shikai Fang, Lewen Wang, Chang Xu, and Jiang Bian. Mars: a financial market simulation engine powered by generative foundation model. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=Yqk7EyT52H.

Margaret Li, Sneha Kudugunta, and Luke Zettlemoyer. (mis) fitting: A survey of scaling laws. *arXiv preprint arXiv:2502.18969*, 2025b.

Yang Li and Zhi Chen. Bytegen: A tokenizer-free generative model for orderbook events in byte space. *arXiv preprint arXiv:2508.02247*, 2025.

Eljas Linna, Kestutis Baltakys, Alexandros Iosifidis, and Juho Kanniainen. Lobert: Generative ai foundation model for limit order book messages. *arXiv preprint arXiv:2511.12563*, 2025.

Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: generative pre-trained transformers are large time series models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Peer Nagy, Sascha Frey, Silvia Sapora, Kang Li, Anisoara Calinescu, Stefan Zohren, and Jakob Foerster. Generative ai for end-to-end limit order book modelling: A token-level autoregressive generative model of message flow using a deep state space network. In *Proceedings of the fourth ACM international conference on AI in finance*, pp. 91–99, 2023.

Peer Nagy, Sascha Yves Frey, Kang Li, Bidipta Sarkar, Svitlana Vyetrenko, Stefan Zohren, Ani Calinescu, and Jakob Nicolaus Foerster. LOB-bench: Benchmarking generative AI for finance - an application to limit order book data. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=CXPpYJpYXQ.

Eric Nguyen, Michael Poli, Matthew G. Durrant, Brian Kang, Dhruva Katrekar, David B. Li, Liam J. Bartie, Armin W. Thomas, Samuel H. King, Garyk Brixi, Jeremy Sullivan, Madelena Y. Ng, Ashley Lewis, Aaron Lou, Stefano Ermon, Stephen A. Baccus, Tina Hernandez-Boussard, Christopher Ré, Patrick D. Hsu, and Brian L. Hie. Sequence modeling and design from molecular to genome scale with evo. *Science*, 386(6723):eado9336, 2024. doi: 10.1126/science.ado9336. URL https://www.science.org/doi/abs/10.1126/science.ado9336.

NVIDIA Corporation. *NVIDIA GH200 Grace Hopper Superchip*. NVIDIA Corporation, March 2025. URL https://dam-cdn.nvd.orangelogic.com/AssetLink/tkno03fa645gb7f33w60rf200d330420.pdf. Datasheet, document 3773000.

NVIDIA Corporation. *NVIDIA DCGM Documentation: Profiling*, n.d. URL https://docs.nvidia.com/datacenter/dcgm/latest/learn/modules/profiling.html. Accessed September 25, 2026.

Tim Pearce and Jinyeop Song. Reconciling kaplan and Chinchilla scaling laws. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=NLoaLyuUUF. Reproducibility Certification.

Tim Pearce, Tabish Rashid, Dave Bignell, Raluca Georgescu, Sam Devlin, and Katja Hofmann. Scaling laws for pre-training agents and world models. *arXiv preprint arXiv:2411.04434*, 2024.

Tomer Porian, Mitchell Wortsman, Jenia Jitsev, Ludwig Schmidt, and Yair Carmon. Resolving discrepancies in compute-optimal scaling of language models. *Advances in Neural Information Processing Systems*, 37:100535–100570, 2024.

Xingwei Qu, Yuelin Bai, Yinghao Ma, Ziya Zhou, Ka Man Lo, Jiaheng Liu, Ruibin Yuan, Lejun Min, Xueling Liu, Tianyu Zhang, Xinrun Du, Shuyue Guo, Yiming Liang, Yizhi Li, Shangda Wu, Junting Zhou, Tianyu Zheng, Ziyang Ma, Fengze Han, Wei Xue, Gus Xia, Emmanouil Benetos, Xiang Yue, Chenghua Lin, Xu Tan, Stephen W. Huang, Jie Fu, and Ge Zhang. Mupt: A generative symbolic music pretrained transformer, 2024. URL https://arxiv.org/abs/2404.06393.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, Marin Biloš, Sahil Garg, Anderson Schneider, Nicolas Chapados, Alexandre Drouin, Valentina Zantedeschi, Yuriy Nevmyvaka, and Irina Rish. Lag-llama: Towards foundation models for probabilistic time series forecasting, 2024. URL https://arxiv.org/abs/2310.08278.

Yu Shi, Zongliang Fu, Shuo Chen, Bohan Zhao, Wei Xu, Changshui Zhang, and Jian Li. Kronos: A foundation model for the language of financial markets, 2025. URL https://arxiv.org/abs/2508.02739.

Jimmy T.H. Smith, Andrew Warrington, and Scott Linderman. Simplified state space layers for sequence modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Ai8Hw3AXqks.

Zhuohan Wang and Carmine Ventre. Difflob: Diffusion models for counterfactual generation in limit order books, 2026. URL https://arxiv.org/abs/2602.03776.

Aaron Wheeler and Jeffrey D Varner. Marketgpt: Developing a pre-trained transformer (gpt) for modeling financial time series. *arXiv preprint arXiv:2411.16585*, 2024.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Forty-first International Conference on Machine Learning*, 2024.

Qingren Yao, Chao-Han Huck Yang, Renhe Jiang, Yuxuan Liang, Ming Jin, and Shirui Pan. Towards neural scaling laws for time series foundation models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=uCqxDfLYrB.

Zihao Zhang, Stefan Zohren, and Stephen Roberts. Deeplob: Deep convolutional neural networks for limit order books. *IEEE Transactions on Signal Processing*, 67(11):3001–3012, 2019.

A MESSAGE REPRESENTATION AND MODEL TOPOLOGY

This appendix describes how order messages and book states enter the model.

A.1 FROM A MARKET MESSAGE TO TOKENS

Each LOBSTER record contains categorical fields (event type and direction), numerical fields (price, size, and timestamp), and an order identifier. Following the field-based representation of Nagy et al. (2023), we encode each message as a fixed block of 26 tokens. Figure 5 shows how the raw fields expand into this block. The order identifier supports a preprocessing lookup of reference-order attributes; it is not itself predicted as a token.

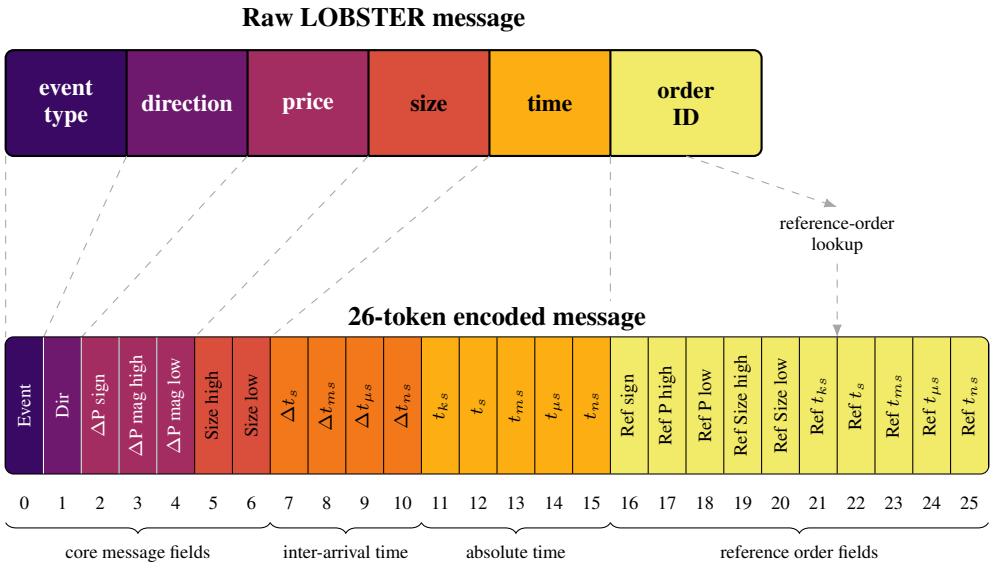

Figure 5: Mapping from a raw LOBSTER record to a 26-token message. Inferno colours identify field groups and their token expansions. Positions 0–6 encode the event, direction, relative price, and size; 7–10 encode inter-arrival time; 11–15 encode absolute time; and 16–25 encode reference-order attributes. Labels and boundaries identify the fields independently of colour. The order ID is a lookup key rather than a token field.

Field encoding. Prices use a sign and two base-1000 magnitude digits: $p = s(1000p_h + p_l)$, where the decoded sign $s \in \{-1, +1\}$ and each digit lies in $0, \ldots, 999$. The representation accommodates magnitudes up to 999,999 relative-price units. Sizes use two base-100 digits, $q = 100q_h + q_l$, representing 0–9,999 shares. Inter-arrival time contains a seconds component followed by millisecond, microsecond, and nanosecond components. Absolute time contains five base-1000 components, from kiloseconds through nanoseconds. Reference fields repeat the price, size, and absolute-time layouts for the referenced order in cancellation, deletion, and execution events. Absolute-time tokens remain part of the prediction target even though timestamps and inter-arrival times are related.

Vocabulary. The vocabulary contains 2,112 IDs, partitioned by field family. IDs 0–3 encode mask, hidden, missing, and start values, respectively. IDs 4–1003 encode time digits, 1004–1007 event types, 1008–1107 size digits, 1108–2107 price digits, 2108–2109 signs, and 2110–2111 directions. Primary and reference fields share these digit pools. Thus vocabulary size and message length are distinct: a 500-message window contains 13,000 token positions drawn from the same vocabulary.

A.2 BOOK-CONDITIONED MESSAGE GENERATION

The model processes message tokens and aligned book states in separate streams before combining them (Figure 6). Let L denote the number of messages, $T = 26L$ the token-sequence length, H

the model width, and d_{book} the width of the input book-state representation. The message stream embeds vocabulary IDs into H dimensions and applies two sequence layers. The book stream applies one layer at d_{book}, projects to H, and applies one further layer at H. The labels "pre" and "post" refer to this book projection: both layers occur before the streams are merged. After feature-wise concatenation, a dense projection maps $2H$ to H, six fused sequence layers process the combined representation, and an output projection produces vocabulary logits. This gives ten sequence layers across the two branches and fused stack: $2 + 1 + 1 + 6 = 10$.

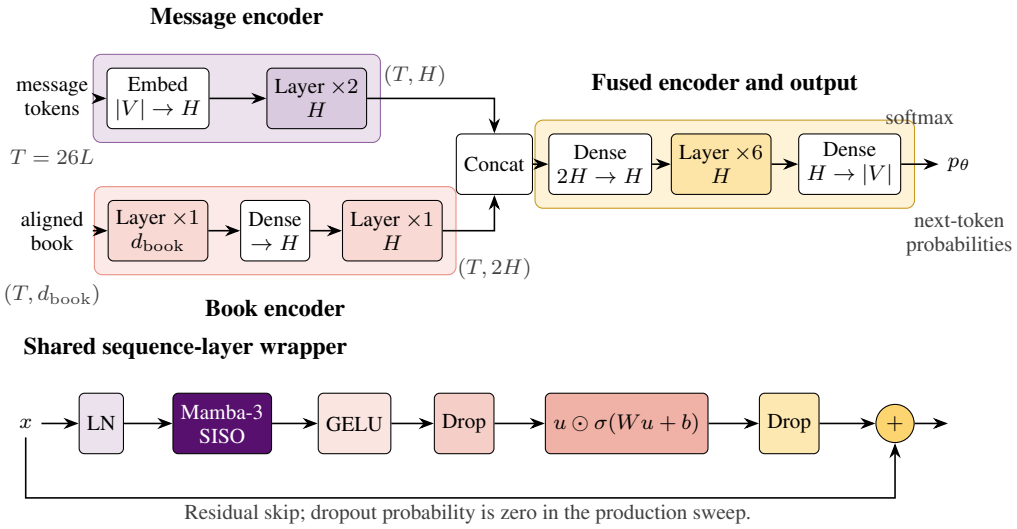

Figure 6: Dual-stream architecture, following the wrapper of Nagy et al. (2023). The message (purple) and book (orange) streams feed the fused stack (gold). Sequence layers use the wrapper shown below: pre-normalisation, a Mamba-3 SISO core, GELU, a multiplicative gate, and a residual connection. Dropout is disabled. Tensor lengths refer to aligned token positions.

Shared sequence layer. Sequence layers use pre-normalisation and the `half_glu1` activation wrapper. With dropout disabled, a layer computes

$$u = \text{GELU}(\text{Mamba3}(\text{LN}(x))), \qquad y = x + u \odot \sigma(Wu + b).$$

This describes the outer wrapper; the Mamba-3 core has its own internal operations.

Event-boundary alignment. Causal prediction of event e_t requires conditioning on the preceding book state B_{t-1}, then applying e_t to obtain B_t for the next prediction. The wrapper defines how the two streams combine, but the training dataloader is unavailable, so the book/target alignment cannot be checked from the accompanying implementation.

B LEARNING-RATE SCHEDULE AND OPTIMISER RATIONALE

Schedule and budget alignment. Training is configured for one epoch truncated at the target horizon, a global batch of 128 sequences, one gradient-accumulation step, a 1% linear warmup and cosine decay. The scheduler includes the target index S, giving $T = S + 1$ steps and $w = \lfloor 0.01T \rfloor$ warmup steps. For a parameter group with peak rate $\eta_{\max}$, the normalised learning rate is

$$\frac{\eta(s)}{\eta_{\max}} = \begin{cases} s/w, & s < w, \\ \rho + \dfrac{1-\rho}{2}\left[1 + \cos\left(\pi \min\left\{\dfrac{s-w}{T-w}, 1\right\}\right)\right], & s \geq w, \end{cases} \qquad \rho = 0.05. \tag{4}$$

The configuration value `lr_min=0` invokes the implementation's 5%-of-peak default; it does not imply decay to zero. Resumed runs retain optimiser state and the schedule counter, including across changes in worker topology. The learning-rate logs agree with this schedule at the sampled steps, but the production executable is not available for every job.

Local updates and logged loss. Training uses ten local updates between cross-node parameter averages, with gradients averaged across the four GPUs within each node. The local-update loop applies the optimiser and advances the step counter once per update. Thus $D = 1{,}664{,}000s$ already counts all ten updates; no additional factor of ten applies. After each group of updates, the implementation averages parameters across nodes without averaging optimiser states or applying an outer momentum update. Node counts range from 2 to 32, while sequences per node range from 64 to 4, maintaining 128 sequences across the workers. This changes the local batch and communication pattern across model sizes.

The logging routine returns the mean of ten local-step losses, each averaged within a node, and rank zero writes the scalar. The implementation does not average losses across nodes, so we report this quantity as logged training CE rather than global-batch CE. Cross-entropy includes all 26 token positions, including time fields. Training plots include positive steps through the target horizon; January trajectories also include the first evaluated checkpoint that reaches it, allowing for checkpoint rounding.

Parameter groups and intended width scaling. The intended learning-rate prescription follows the width-transfer motivation of maximal-update parameterisation (μP). Muon kernels use a common base rate of 10^{-2}; the SSM Adam base rate is approximately $0.512/d_{\text{model}}$, with no weight decay. The intended prescription keeps the AdamW anchor rate independent of width for embeddings, biases and normalisation parameters. Muon and AdamW weight decay is 0.005.

The production μP implementation is no longer available. Configurations specify the SSM base rate and `lr_factor=1`, while the source snapshot derives the auxiliary base from these values. The AdamW anchor value and its mapping to effective parameter rates remain unspecified, as do the reference width, initialisation and forward multipliers needed to reproduce the full μP implementation.

Why the schedule matters for scaling. Hoffmann et al. (2022) matched decay horizons to training budgets when constructing compute-optimal scaling comparisons. This motivates comparing endpoints or common late-budget fractions: an intermediate checkpoint has received a different amount of decay from a run trained to end at the same token count. Porian et al. (2024) identify output-layer accounting, warmup duration and size-dependent optimiser tuning as important factors; in their experiments, careful decay alignment changes loss but is not necessary for the Chinchilla-like allocation relation to emerge. Our fixed-compute slices interpolate trajectories at different schedule fractions; the runs were not separately annealed to each slice budget.

Scope of width transfer. The width-scaling prescription was intended to transfer optimiser settings across model sizes. Its finite-width behaviour also depends on parameter roles, schedule phase

Table 6: Learning-rate settings by model size. T is the inclusive schedule length, w the number of warmup steps, and the last column the SSM Adam base rate. Muon uses peak rate 0.01 throughout; cosine decay ends at 5% of each peak.

N (M)	d_{model}	T	w	SSM base
2.63	64	10603	106	0.008000
3.91	128	18605	186	0.004000
5.74	192	57969	579	0.002667
8.10	256	106910	1069	0.002000
11.00	320	84336	843	0.001600
14.45	384	74487	744	0.001333
22.95	512	93990	939	0.001000
46.43	768	63408	634	0.000667
78.54	1024	28729	287	0.000500
119.28	1280	29960	299	0.000400
196.57	1664	16211	162	0.000308
293.28	2048	10671	106	0.000250

and local-update settings. These experiments vary the local batch alongside width, so their effects cannot be separated from the scaling curves alone.

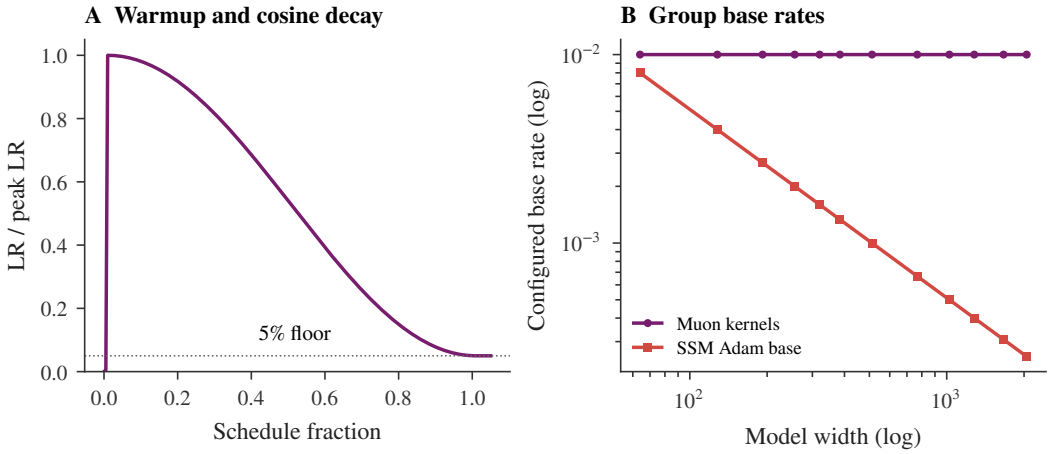

Figure 7: Learning-rate schedule and Muon and SSM base rates. Panel A shows the schedule multiplier; panel B shows base rates by width. Effective AdamW rates are unspecified.

C Training runs and January trajectories

There are two or three completed lineages per model size. Figure 8 shows their January CE trajectories as training exposure increases.

Table 7: Training coverage by parameter count. CE and exposure summarise budget endpoints; the CE-point and endpoint columns exclude interrupted runs.

Params (M)	Planned	Complete	Roots	CE points	Mean CE	D (B)
2.626	3	3	3	11	0.9214	17.66
3.911	3	3	3	12	0.6938	30.97
5.736	3	3	3	26	0.6143	96.46
8.100	3	3	3	30	0.5847	177.90
11.003	3	3	3	30	0.5896	140.34
14.446	3	3	3	30	0.5839	123.95
22.949	3	2	2	18	0.5672	156.40
46.429	3	2	2	16	0.5719	105.51
78.539	3	3	3	21	0.5772	47.81
119.280	3	3	3	21	0.5826	49.85
196.572	2	2	2	18	0.6001	26.99
293.283	2	2	2	16	0.6128	17.77

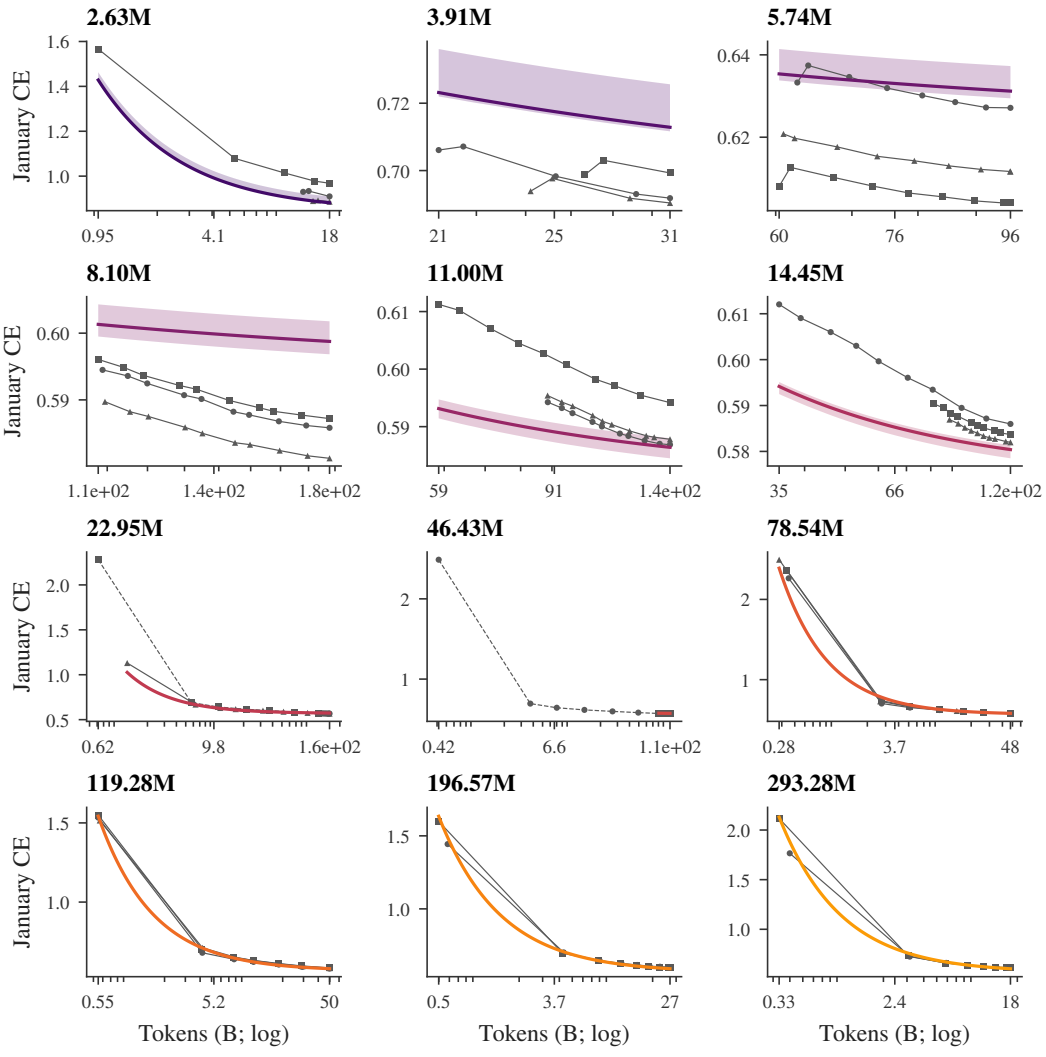

Figure 8: Mamba-3 January CE trajectories at all 12 model sizes. Grey points and thin lines show evaluations; marker shapes distinguish seed labels and dashed observation lines mark interrupted runs. Inferno curves show the full-trajectory additive fit to completed lineages, with pointwise 95% bands from 2,000 lineage-bootstrap draws. Axes vary by panel to show departures from the fitted shape; intervals describe fitted-curve variation rather than new-run prediction error.

D FITTING PROCEDURES AND DIAGNOSTICS

Table 8: Complete additive coefficients in the normalised coordinates of Equation 2. Wider-bound rows are point-fit sensitivities; the plotted bootstrap bands condition on reference bounds.

Selection	Bounds	E	a	b	α	β
endpoint	reference	0.566115	0.018652	0.111844	2.0000	1.4439
endpoint	wide	0.566871	0.008681	0.088942	2.6365	1.0985
tail25	reference	0.557521	0.018753	0.088531	2.0000	0.8548
tail25	wide	0.566871	0.008363	0.077125	2.6750	0.9943
trajectory	reference	0.565222	0.019217	0.063726	2.0000	0.9408
trajectory	wide	0.566871	0.009232	0.062704	2.6196	0.9456

Robust objective. The log-Huber threshold 10^{-3} corresponds to about 0.0006 nats at CE 0.6. It limits the influence of large log residuals: 88.5% of final-quarter residuals lie in its linear branch, so the reference fit behaves largely like absolute log error. RMSE is a separate descriptive scoring rule, not the training objective. The squared-log and threshold sensitivities in Table 9 and the accompanying tables address this choice.

Small-model parameter counts. Parameter count and model width satisfy $N(d) = 65.84375d^2 + 7438.5625d + 1{,}880{,}159$ to numerical precision. The constant term is 71.6% of the smallest model's count. This algebraic decomposition motivates examining fixed overhead when extrapolating in total parameters; it does not assign the term to a particular module or establish the cause of the prediction error.

Temporal scope. January is held out from neural-model training, but its CE values are used to fit the scaling relationship. Omitting runs or sizes within January does not test whether that relationship transfers to another month.

Repeated January evaluations. Eight checkpoints have two evaluations; CE is 0.0103–0.0138 nats higher in the evaluation with the higher job ID in every pair. Both evaluations cover the same 487 tickers. The cause of this shift is unresolved because the window lists, sampling caps and evaluator versions are unavailable. We average the two evaluations equally, as specified in Section 3; they do not count as independent training runs.

Table 9: Fitting targets and weights. All joint surfaces use Equation 2. Huber loss acts on log residuals with threshold 10^{-3}; squared log denotes squared log residuals.

Analysis	Observations and weights	Objective
January endpoints / late windows	Equal runs, then checkpoints in each selection	Huber
January full trajectories	All 249 checkpoints; equal runs, then checkpoints	Huber
Training trajectories	Log-compute bin means; equal runs, then bins	Huber
Main compute frontiers	Minimum of trajectory-surface predictions over eligible sizes	No additional fit
IsoFLOP quadratics	Interpolated lineage means; equal sizes	Squared raw or log CE
Direct frontier alternatives	Observed size-mean minima; equal compute budgets	Squared log
Full-trajectory model comparison	Same observations for all models; equal runs	Squared log

Resumed and incomplete runs. The nominal 23M seed-42 and 46M seed-5 runs stop below budget and contribute the marked, unfitted trajectories. Two nominal 6M runs switch seed labels when training resumes: seed 5 resumes the run labelled 137, and seed 137 resumes the run labelled

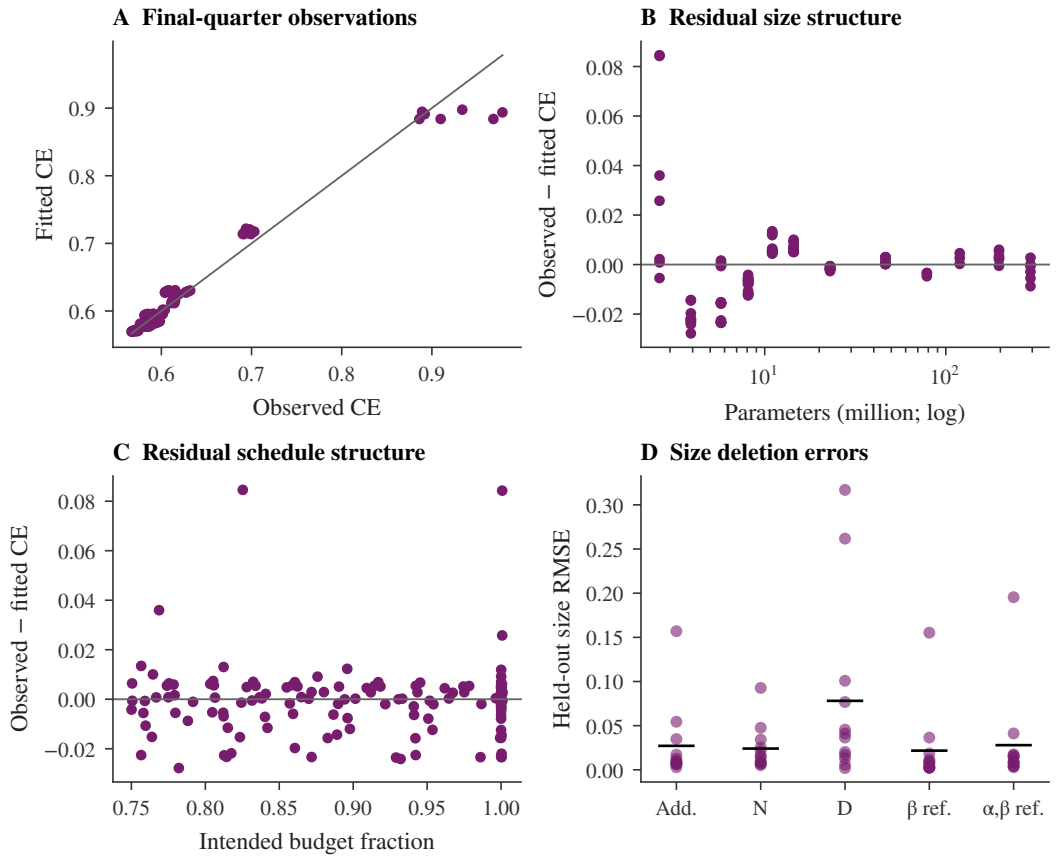

Figure 9: Additive-law adequacy on the final-quarter selection. The observation–prediction comparison and residuals show departures from a smooth monotone surface. Each point in panel D is the run-weighted RMSE for one omitted size; bars average those size errors. Table summaries average across all held-out runs.

Table 10: Final-quarter fitting sensitivity. Every row refits the stated observations and constraints. RMSE provides a common error measure across loss definitions.

Change	α	β	E	RMSE
Wider bounds	2.675	0.994	0.5669	0.0131
Narrower bounds	1.000	0.762	0.5329	0.0470
Raised floor cap	2.000	0.855	0.5575	0.0186
Huber 0.01	2.000	1.131	0.5632	0.0189
Huber 0.0001	2.000	0.843	0.5575	0.0188
Squared log error	2.000	1.556	0.5649	0.0165
Exclude smallest size	1.259	0.243	0.4790	0.0070
Exclude two smallest sizes	0.555	0.167	0.3960	0.0053
Exclude largest size	2.000	2.000	0.5669	0.0169
Exclude relabelled lineages	2.000	1.238	0.5669	0.0188

5. They have distinct initial ancestors and remain separate lineages; Table 10 reports the sensitivity to excluding both.

Numerics. The optimiser uses normalised coordinates with reference sizes $N_0 = 10^7$ and $D_0 = 10^{10}$. Amplitudes are log-parameterised with $\log a, \log b \in [-40, 8]$; the floor may reach zero. The raised-floor-cap sensitivity allows $E \leq 2 \max L$, compared with $E \leq \min L$ in the reference fit. An analytic Huber gradient is checked by finite differences away from its transition points. Multistart

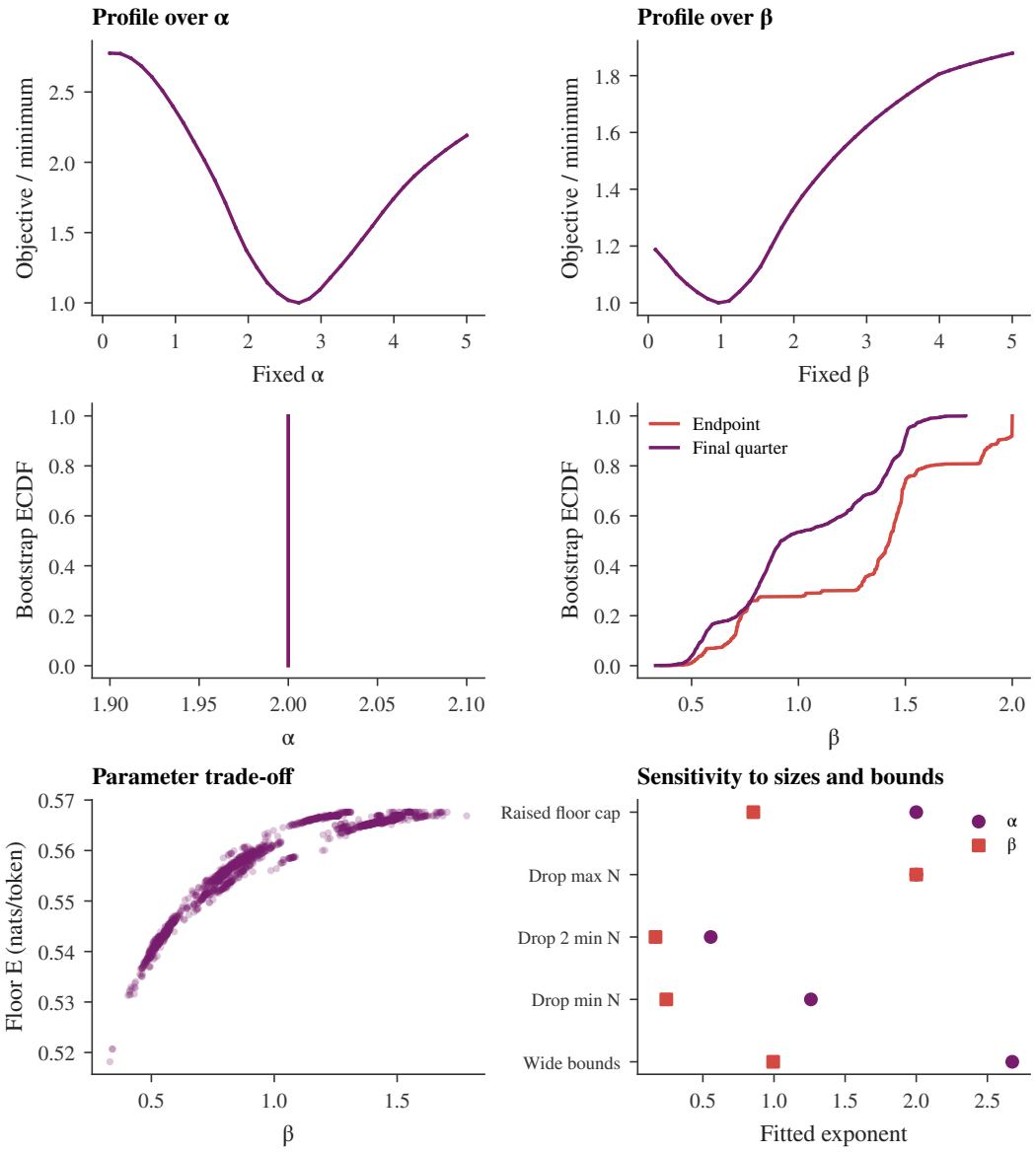

Figure 10: Exponent identification and sensitivity. Profiles use wider exponent bounds and are objective diagnostics, not likelihood-based confidence intervals. Bootstrap ECDFs retain boundary estimates. The floor–data-exponent scatter exposes parameter trade-offs. The final panel refits after changing constraints or size support.

results include convergence status, iteration count, projected-gradient norm, objective, and every fitted parameter. Fits require both solver success and a projected-gradient norm below 10^{-4}. We select the converged start with the lowest objective. Fits with no converged start are counted as failures and excluded from bootstrap quantiles.

Predictive summaries. Tables report the square root of mean squared error after averaging checkpoints within each held-out run and then averaging runs equally. Interpolation is summarised only where it returns a within-hull prediction; its coverage is reported alongside the error. Errors on the same within-hull observations are also reported for both parametric models and interpolation, allowing comparison at equal coverage. Whole-size deletion tests differ from ordinary replicate interpolation, and observations outside the remaining convex support are marked as extrapolations.

Table 11: Mamba lineage bootstrap: 2,000 draws per selection. Boundary percentages use successful draws; failed fits are excluded from quantiles.

Selection	Param.	Median	95% percentiles	Bound (%)	Failed
endpoint	α	2.000	[2.000, 2.000]	100.0	0
endpoint	β	1.426	[0.520, 2.000]	7.6	0
tail25	α	2.000	[2.000, 2.000]	100.0	0
tail25	β	0.919	[0.488, 1.570]	0.0	0
trajectory	α	2.000	[2.000, 2.000]	100.0	0
trajectory	β	0.939	[0.927, 0.961]	0.0	0

Table 12: Final-quarter prediction restricted to the same within-hull observations for every model. Coverage is observations with a prediction / all held-out observations; folds share experiments.

Deletion	Model	Coverage	Runs	RMSE
lineage	additive	129/131	32	0.0257
lineage	parameter only	129/131	32	0.0281
lineage	fixed beta	129/131	32	0.0222
lineage	interpolation	129/131	32	0.0198
size	additive	74/131	20	0.0163
size	parameter only	74/131	20	0.0176
size	fixed beta	74/131	20	0.0172
size	interpolation	74/131	20	0.0512

E LOB-BENCH EVALUATION

Rollout protocol. Each rollout evaluation uses GOOG in January 2026, with 3,136 sequences, 250 conditioning messages and 250 generated messages per sequence, 26-token encoding and 500-level book input. The conditioning segment supplies observed history; generated events evolve the simulated book through the matching engine. We compare features of generated continuations with those of the corresponding observed continuations, pooling samples for each feature. This single-instrument evaluation is separate from the 487-ticker CE evaluation. The per-job sampling temperature, token-filter settings, simulator and scorer revisions, and invalid-event handling are unavailable.

Distances and normalisation. We use LOB-Bench metrics (Nagy et al., 2025). For each feature, let X and Y contain observed and generated samples. We pool them and apply a common standardisation $z = (x - \mu_{X\cup Y})/s_{X\cup Y}$, using the sample standard deviation. Wasserstein-1 is then

$$W_1(X', Y') = \int_{-\infty}^{\infty} |F_{X'}(z) - F_{Y'}(z)|\, dz, \tag{5}$$

where primes denote these standardised observations. Its scale therefore depends on both distributions. The supporting KS score is $\sup_z |F_X(z) - F_Y(z)|$. The score labelled L1 is the histogram total-variation distance $\frac{1}{2}\sum_k |p_k - q_k|$, with separately normalised observed and generated bin counts. The unconditional scores use pooled Freedman–Diaconis bins for continuous features and unique pooled values for discrete features. KS and L1 lie in $[0, 1]$; Wasserstein-1 is nonnegative and has no such upper bound. Lower values indicate closer distributions.

Features and aggregation. Table 13 lists the 21 features. Each feature score is the metric's point estimate, rather than a bootstrap percentile or the mean of bootstrap draws. For metric m, we report $\overline{d}_m = 21^{-1}\sum_{f=1}^{21} d_{mf}$ only if all 21 feature scores are finite. Completeness is checked independently for Wasserstein, KS and L1.

Cohort and missing scores. The generation evaluation uses checkpoints from the completed CE training cohort, with the same architecture, through each run's budget endpoint. The size panel includes eligible generation scores whether or not the checkpoint has a CE evaluation. Every incomplete Wasserstein vector in this cohort lacks the log inter-arrival-time distance; some also lack

Table 13: The 21 generation features. Paired ask/bid entries count separately.

Feature family	Count	Definition
Spread; queue imbalance	2	Best ask minus best bid; $(v_b - v_a)/(v_b + v_a)$ at the best prices.
Log inter-arrival time; log time to cancel	2	Natural logs of event gaps and matched order-to-cancellation durations; zero durations replaced by 10^{-9} seconds.
Ask/bid volume	4	Resting volume at the best price and summed over the first ten price levels, on each side.
Submission/cancellation depth	4	Absolute distance of order price from the midpoint, separated by side and event type.
Submission/cancellation ticks	4	Rounded, nonnegative tick distance from the corresponding best quote; cancellations use the preceding book state.
Execution volume rate	1	Executed volume in one-second bins, converted to a per-minute rate with edge-bin adjustments.
Order-flow imbalance (OFI)	4	Mean signed best-quote volume change over 100 events, unconditional and conditioned on the next midpoint moving up, staying fixed or moving down.

log time to cancel. These incomplete vectors result from non-finite values during scoring. Table 14 shows how feature completeness varies with model size; the means include only checkpoints with complete scores. Generation is evaluated at 463 checkpoints from 32 completed training lineages at 12 sizes. Of these, 365 have all 21 finite Wasserstein scores, and 185 also have a January CE evaluation. The inter-arrival score is missing at 98 checkpoints; 21 of these also lack the cancellation-time score.

Table 14: Generation coverage across completed training lineages. Complete means all 21 Wasserstein features are finite; CE matched additionally requires a January CE evaluation of the same checkpoint. The last column averages complete checkpoint scores within each lineage, then represented lineages equally.

Params (M)	Checkpoints	Lineages	Complete	CE matched	Mean WS
2.626	11	3	6	6	0.250
3.911	12	3	12	12	0.275
5.736	26	3	26	26	0.204
8.100	49	3	49	30	0.219
11.003	53	3	53	28	0.206
14.446	54	3	54	30	0.195
22.949	42	2	36	14	0.222
46.429	74	2	58	16	0.193
78.539	63	3	34	5	0.189
119.280	45	3	27	8	0.184
196.572	18	2	5	5	0.216
293.283	16	2	5	5	0.205

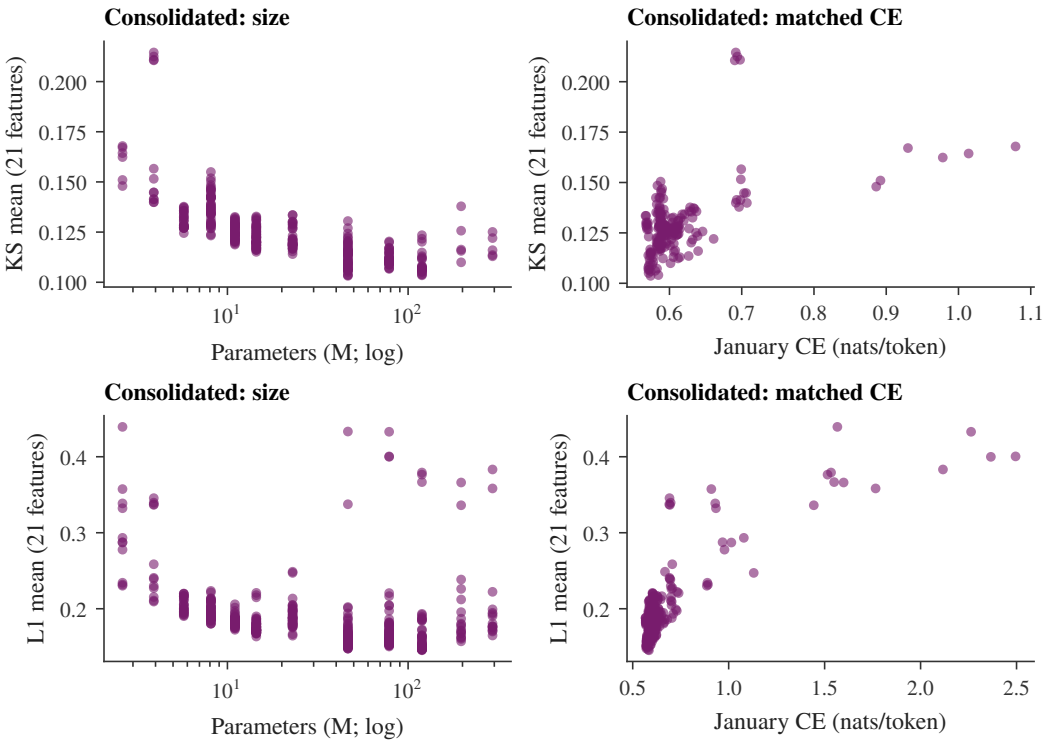

Figure 11: LOB-Bench KS and L1 scores. Means require all 21 features; CE panels use checkpoints with both generation scores and January CE. Multiple checkpoints from one lineage share the same training history.

F HARDWARE-COST CALIBRATION

Table 15: Mamba-3 hardware calibration. N is the parameter count and q the integrated activity-derived FLOPs/token estimate (millions). Busy throughput is in TFLOP/s per GPU; peak percentages are rounded relative to 989 TFLOP/s. Each node has four GPUs, and every batch configuration processes 1,664,000 tokens/step.

N (M)	d_{model}	GPUs	BS/GPU	q (M)	Busy	Peak (%)
2.63	64	8	16	98.4	80	8.1
3.91	128	8	16	157.4	82	8.3
5.74	192	16	8	233.9	88	8.9
8.10	256	16	8	272.6	88	8.9
11.00	320	16	8	419.5	106	10.7
14.45	384	16	8	458.9	103	10.4
22.95	512	32	4	625.1	105	10.6
46.43	768	32	4	1,144.2	122	12.3
78.54	1024	32	4	1,808.7	133	13.5
119.28	1280	64	2	2,891.5	168	17.0
196.57	1664	128	1	5,552.1	220	22.2
293.28	2048	128	1	8,701.4	268	27.1

Hardware measurements. Table 15 reports the hardware calibration used to construct the compute axis. Measurements are made at steady-state training speed. The calibration is fixed across each training trajectory; its transfer to the full workload remains an assumption.

What the estimator measures. We use GPU samples with SM activity at least 50% and convert tensor, FP32 and FP16 activity percentages to throughput using GH200 calibration constants of

989, 67 and 134 TFLOP/s, respectively. We aggregate GPUs by timestamp and divide the summed throughput samples by the profile's token total T_{profile}:

$$q_i = \frac{10^{12} \sum_t \sum_g \widehat{f}_{g,t}\,(1\,\text{s})}{T_{\text{profile}}}. \tag{6}$$

The counters measure fractions of active cycles (NVIDIA Corporation, n.d.); multiplying them by peak rates produces activity-derived estimates. The BF16 calibration constant is 989 TFLOP/s, compared with the rounded 990 TFLOP/s in the March 2025 GH200 datasheet (NVIDIA Corporation, 2025). The last table column reports busy throughput relative to the BF16 peak.

Compute along a trajectory. Compute is $C = q_i D$ at measured size N_i. The uncertainty intervals condition on these fixed costs because the raw traces needed to reassess profile uncertainty are unavailable.

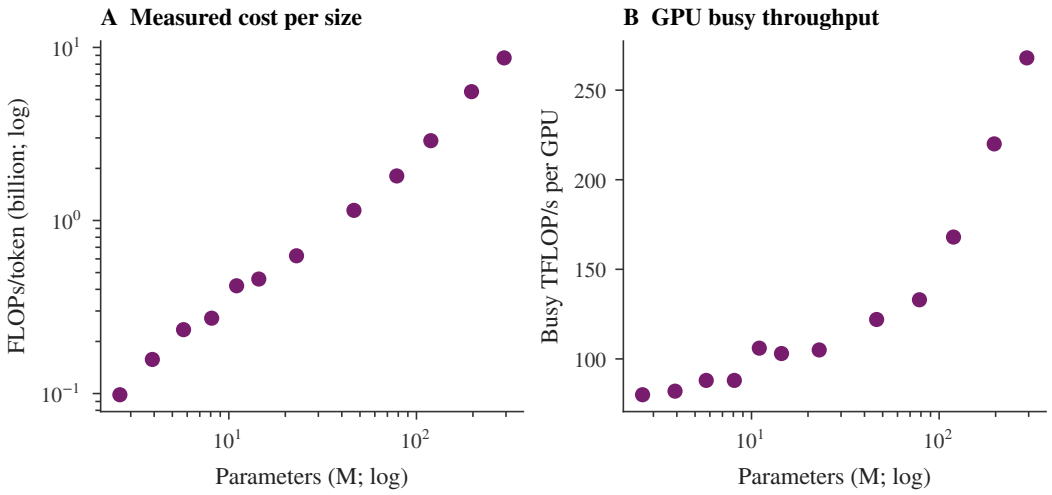

Figure 12: Mamba-3 hardware-cost and busy-throughput estimates by model size. Each point represents one profile; compute analyses use these per-size costs directly.

G IsoFLOP construction, support and uncertainty

For every lineage and measured size, we linearly interpolate January CE against log compute, only between its first and last evaluated checkpoints. The bracketing checkpoints and interval widths are included with the interpolation results. At each budget we average available lineages equally within size, then fit a quadratic with equal size weights. A size contributes when at least one lineage brackets the budget; the consolidated frontier uses the stricter requirement of two lineage observations per size. Nine budgets are spaced geometrically over the coverage envelope containing intervals with at least five measured sizes; their selection does not use CE or fitted curvature. Some lineages do not bracket a given budget and therefore contribute no interpolated observation there.

We fit both L and $\log L$ quadratics in $\log N$. An accepted vertex requires at least five sizes, positive curvature, and at least two sizes on each side of the minimum. All criteria are reevaluated after deleting each size. Czech et al. (2026) show why broad or off-centre sampling and asymmetric loss surfaces can bias parabolic optima; the bracketing and deletion checks test sensitivity to these sampling choices.

The full-trajectory additive surface is refitted on 2,000 complete-lineage resamples using the same draw identities as the late-budget analysis. Each draw uses the same lineage multiplicities across all budgets. Every draw repeats interpolation, averaging, both quadratic fits and all size deletions. If no sampled lineage at a size covers a budget, that size is absent from that draw, and the support checks are rerun on the reduced set. No valley rows, checkpoints or tickers are resampled as independent

training experiments. Pointwise curve and vertex intervals use only initially valid fits; their denominator and the stricter deletion-stable count are both reported. We compare bootstrap quantiles from the first and second halves of the draws to assess Monte Carlo variation.

Continuous quadratic vertices and discrete-grid allocation comparisons answer different questions. For the latter, both methods select the minimum predicted loss from the same set of profiled sizes, using $D = C/q_i$ at each size. The parametric surface fits all 249 January checkpoints; each slice interpolates the subset of trajectories that bracket its compute budget. The endpoint and late-window fits assess sensitivity to training phase. Intervals exclude uncertainty in the hardware profiles, interpolation between sparse checkpoints, corpus shift and model misspecification. Available sizes and schedule phase change across budgets, limiting comparison with independently trained matched-compute endpoints.

Table 16: Initially accepted raw-loss IsoFLOP slices. Both surfaces use all January trajectories and the same measured-size grid. Costs are fixed; these point comparisons vary the exponent bounds only.

$C/10^{19}$	Deletion stable	Quadratic (M)	Surface (M)	Wide surface (M)
1.68	No	22.949	14.446	14.446
2.61	Yes	22.949	22.949	14.446
4.06	Yes	22.949	22.949	14.446

Table 17: IsoFLOP diagnostics conditional on fixed activity-derived costs. A valid vertex requires positive curvature, at least five sizes and two sizes on each side. The deletion check repeats these criteria after removing each size. Bootstrap columns count draws that pass the initial checks and those that pass every size deletion; all draws contribute to the denominators. C is in 10^{19} FLOPs and N^* in millions.

C	Loss	Sizes	N^*	Deletion	Valid draws	Stable draws
0.29	raw	5	–	no	0/2000	0/2000
0.29	log	5	–	no	0/2000	0/2000
0.45	raw	6	–	no	0/2000	0/2000
0.45	log	6	–	no	0/2000	0/2000
0.70	raw	5	–	no	0/2000	0/2000
0.70	log	5	–	no	0/2000	0/2000
1.08	raw	5	–	no	0/2000	0/2000
1.08	log	5	–	no	0/2000	0/2000
1.68	raw	7	21.12	no	1407/2000	0/2000
1.68	log	7	20.58	no	1407/2000	0/2000
2.61	raw	7	30.06	yes	1641/2000	743/2000
2.61	log	7	29.63	yes	1641/2000	743/2000
4.06	raw	8	23.34	yes	2000/2000	1948/2000
4.06	log	8	23.00	yes	2000/2000	1948/2000
6.30	raw	5	–	no	0/2000	0/2000
6.30	log	5	–	no	0/2000	0/2000
9.78	raw	5	–	no	0/2000	0/2000
9.78	log	5	–	no	0/2000	0/2000

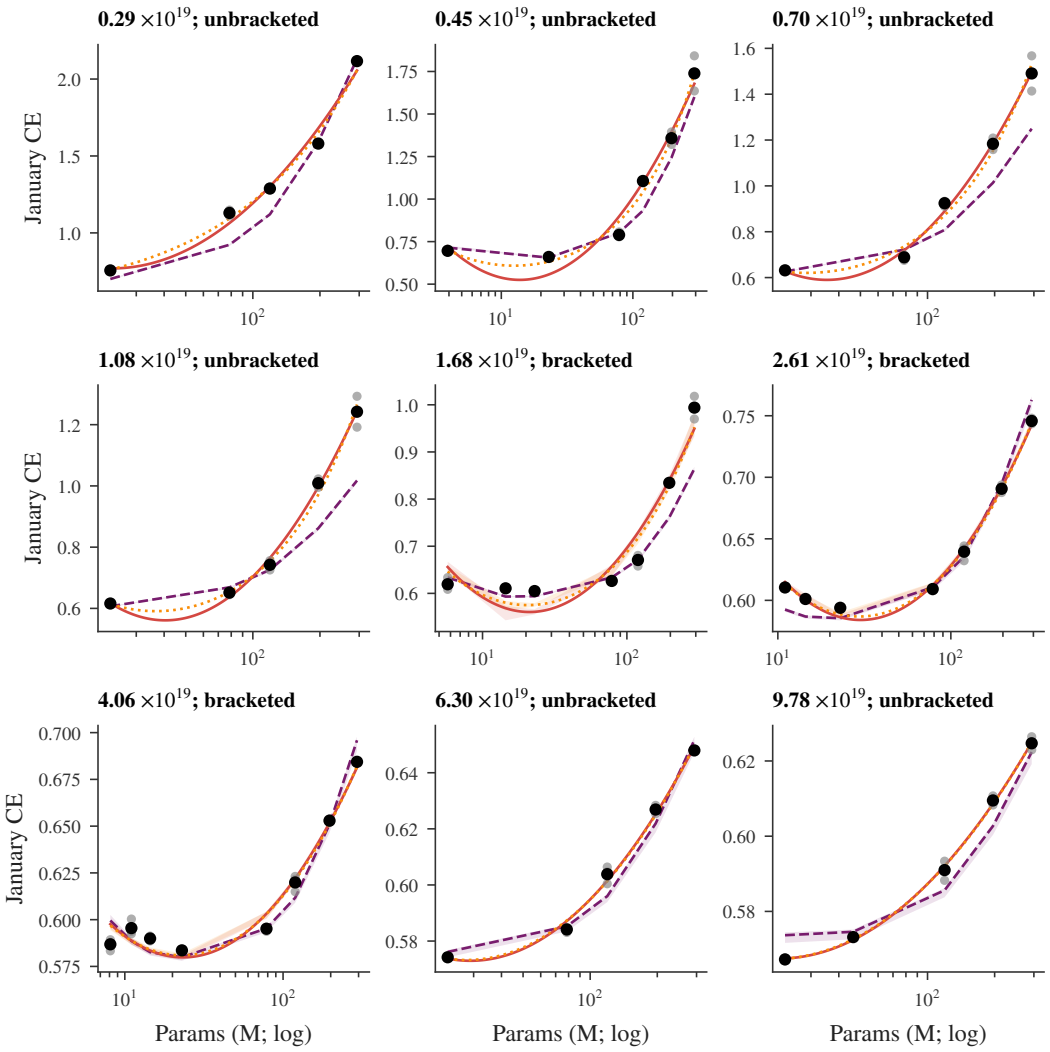

Figure 13: All nine fixed-profile CE slices, including unbracketed quadratics. Observations, fits and pointwise bands use the conventions of Figure 4. Titles report the initial raw-quadratic bracketing result; Table 17 records deletion and bootstrap failures. Curves failing the bracketing checks remain visible; their vertices are excluded from allocation comparisons.

H Consolidated trajectories and model-derived compute frontiers

We express each trajectory in compute coordinates $C = q_i D$, using the activity-derived cost q_i for its model size. January CE is shown at each evaluated checkpoint. Training CE uses the local-loss summaries described in Appendix B. For the plots and frontier calculation, we average training losses within 64 fixed bins equally spaced in log compute over the combined training range. Each bin is placed at the geometric mean of its compute coordinates; binning does not depend on loss.

For each outcome, an 80-point geometric compute grid spans the observed range. Within a lineage, we interpolate loss linearly against log compute between its observed endpoints. At each budget, we average losses equally across available lineages of a given size. A size requires at least two lineage observations, and a budget requires at least five sizes to contribute to the frontier. The observed frontier is the minimum of these size means.

Frontier of the joint surface. The main figure uses the joint size–exposure surface in Equation 2, expressed in compute coordinates:

$$\widehat{L}_{\text{joint}}(C) = \min_{N_i \in \mathcal{S}(C)} \left[\widehat{E} + \widehat{a}(N_i/N_0)^{-\widehat{\alpha}} + \widehat{b} \left(\frac{C}{q_i D_0} \right)^{-\widehat{\beta}} \right], \tag{7}$$

where $\mathcal{S}(C)$ contains the profiled model sizes that satisfy the support criteria at budget C. The relation $C = q_i D$ determines compute from size and exposure. The January curve uses the same full-trajectory log-Huber fit and bootstrap coefficients as the size-specific panels and parametric IsoFLOP comparison. We fit training CE to the log-compute bin means using the same functional form, normalisation, exponent bounds and run-balanced log-Huber objective. The floor allows curvature, and the minimising model can change as sizes enter or leave the available range.

Table 18: Full-trajectory fits of Equation 2 using the log-Huber objective in Section 4. The January fit and 2,000 resamples are shared with the size panels and IsoFLOP analysis. Training CE is fitted to log-compute bin means using the same objective and paired lineage draws. RMSE is run-balanced and in nats/token.

Outcome	E	α	β	RMSE	Failed draws
January	0.565	2.000	0.941	0.0344	0
Training	0.500	1.330	0.671	0.2839	0

We resample entire training lineages within each size, using the same 2,000 draws as the trajectory analysis. Resampling is restricted to compute-grid budgets supported in the original sample; repeated lineages contribute according to their sampled multiplicity. Each draw fits the surface, checks support and selects the predicted minimum. The bootstrap unit is the training lineage, rather than individual tickers, checkpoints or compute-grid rows. Curve bands are pointwise 2.5th–97.5th percentiles of supported model-derived minima. The analysis tables give the number of valid draws at each budget, fit failures, bound hits and coefficient quantiles for each half of the bootstrap. These bands condition on the observed size grid, smoothing convention and fixed activity-derived costs. They exclude uncertainty from selection of the observed minimum, interpolation, schedule effects and model misspecification. The January parameter exponent remains at its upper bound.

Direct frontier alternatives. For descriptive comparison, we regress the observed size-mean minima on C using $a_C(C/10^{19})^{-\eta}$ and $E_C + a_C(C/10^{19})^{-\eta}$. These regressions use equal weights on squared log residuals across the supported compute grid. The floor model has $0 \leq E_C \leq \min L_{\text{frontier}}$ and $\eta \in [0.01, 3]$; numerical convergence is checked using six deterministic starts, analytic derivatives and projected gradients. Exponent-range sensitivities use $[0.001, 6]$ and $[0.1, 1]$. Each lineage draw recomputes the size means, minimum and regression to construct the floor-model bands. A floor permits curvature on log–log axes, whereas the floor-free form is necessarily a straight line. These direct fits optimise the selected minima, while the joint surface fits individual trajectories; a smaller frontier residual alone is not evidence of better prediction of a new run. We interpret the fitted floors as empirical offsets.

Table 19: Fits evaluated against the same observed frontier on the compute grid. Direct regressions fit the minima, while the joint surface fits all trajectories with equal run weights. Frontier residuals therefore reflect different fitting targets and do not measure held-out prediction. Joint curves follow Equation 2 without a separate compute exponent.

Outcome	Curve	E	η	Frontier log RMSE
January	Power only	0.000	0.071	0.0103
January	Floor + power	0.516	0.410	0.0050
January	Joint surface	0.565	–	0.0167
Training	Power only	0.000	0.231	0.2486
Training	Floor + power	0.546	0.629	0.0323
Training	Joint surface	0.500	–	0.1830

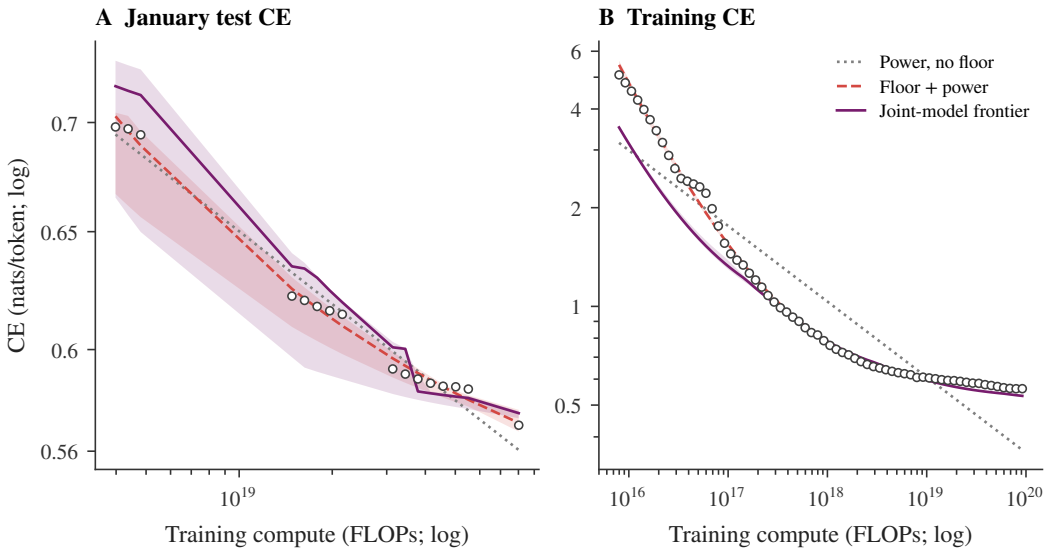

Figure 14: Frontier fits on the same supported compute budgets. Open circles show observed size-mean minima. The dotted curve is a direct floor-free power law; dashed curves add a fitted floor. Solid curves derive the frontier from Equation 2. Shading shows pointwise 95% lineage-bootstrap bands for the floor and joint models. Because the models fit different targets, frontier residuals alone cannot rank their predictive usefulness.

Prediction on individual trajectories. To compare functional forms on a common target, we fit compute-only power, compute-only floor-plus-power, and joint size–exposure models to the same individual observations with a run-balanced squared-log objective. We hold out entire lineages or sizes, keeping all related observations together. We distinguish interpolation from extrapolation using a common convex hull in $(\log N, \log D)$; held-out sizes use their profiled costs, with all their CE observations excluded from fitting. These scores measure full-trajectory prediction conditional on fixed activity-derived costs. Table 9 distinguishes this squared-log comparison from the primary log-Huber fits and direct frontier regressions. Residual structure motivated this exploratory comparison. The joint form improves held-out prediction relative to these compute-only alternatives. Residual errors and the coupling of size with schedule still limit its use for choosing training allocations.

Table 20: Held-out prediction of individual trajectories. All three models use the same run-balanced squared-log objective. Entries are RMSE in nats/token with equal run weights, including extrapolation. This comparison uses separate fits from the direct frontier regressions and the main log-Huber analysis.

Outcome	Model	Leave lineage out	Leave size out
January	Compute power	0.2104	0.2174
January	Compute + floor	0.5054	0.8347
January	Joint size/exposure	0.0356	0.0493
Training	Compute power	0.9970	0.9958
Training	Compute + floor	0.9039	1.0984
Training	Joint size/exposure	0.3407	0.3440

I Supplementary S&P 500 Transformer experiments

These supplementary experiments use a Transformer core with the same event-token representation. Their training corpus spans January 2023–December 2025, whereas the Mamba-3 corpus begins in January 2022. They therefore do not isolate an architecture effect under a matched training setup.

Runs and training corpus. The January Transformer evaluation covers 234 checkpoints from 31 training lineages at 11 sizes. Each lineage includes any resumed training and has a January CE evaluation at or beyond its target horizon. All evaluations cover the same 487 tickers as the Mamba-3 evaluation.

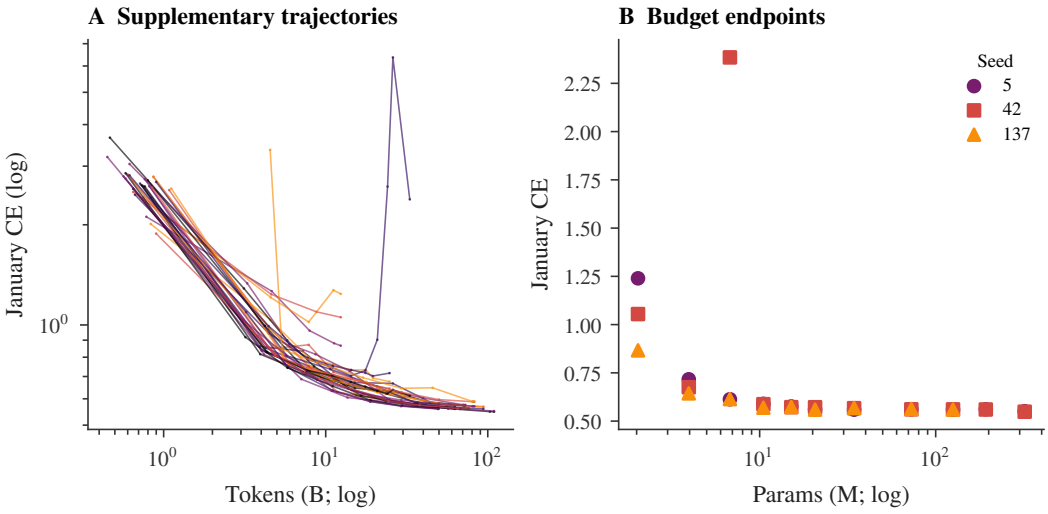

Figure 15: Supplementary Transformer January CE curves and budget endpoints. All observations are shown, including large loss excursions; no loss-threshold filter is applied.

Observed curves and fitting limitations. The largest endpoint CE is 2.3844 nats/token for run `4M-s42` at step 19840. The evaluation metrics do not explain this excursion. The final-quarter additive fit uses 47 observations. Leave-lineage-out RMSE is 0.7690 for the additive law, 0.7683 for the parameter-only model, and 0.7654 for the data-only model. The corresponding whole-size RMSE values are 0.7689, 0.7688, and 0.7616. Large predictive errors and constraint-limited coefficients prevent interpreting these supplementary fits as a reliable architecture-specific allocation law.

Table 21: Separate Transformer January additive fits using all eligible observations, including loss excursions. Exponent bounds are the same as in the main analysis.

Selection	Points	Runs	α	β	RMSE
endpoint	31	31	1.857	0.100	0.3218
tail10	41	31	1.848	0.100	0.3219
tail25	47	31	1.850	0.100	0.7660
tail40	65	31	1.747	0.190	0.5704
tail50	86	31	1.750	0.100	0.5097

Table 22: Transformer lineage bootstrap: 2,000 draws per selection. Boundary percentages use successful draws; failed fits are excluded from quantiles.

Selection	Param.	Median	95% percentiles	Bound (%)	Failed
endpoint	α	1.854	[0.392, 2.000]	27.6	0
endpoint	β	0.100	[0.100, 2.000]	92.6	0
tail25	α	1.848	[0.914, 2.000]	29.1	0
tail25	β	0.100	[0.100, 2.000]	77.9	0

Supplementary hardware profiles. Table 23 reports 11 completed Transformer hardware profiles, including parameter counts, worker topology and rounded cost estimates.

Table 23: Supplementary Transformer hardware profiles. Units match Table 15. Estimates are rounded; the token totals used in the profile integrations are unavailable.

N (M)	d_{model}	GPUs	BS/GPU	q (M)	Busy	Peak (%)
2.04	64	8	16	156.0	172	17.4
3.96	128	8	16	242.0	215	21.7
6.78	192	8	16	351.0	239	24.1
10.50	256	8	16	479.0	273	27.6
15.12	320	8	16	616.0	297	30.1
20.65	384	8	16	754.0	320	32.3
34.40	512	16	8	1,032.0	328	33.1
72.71	768	16	8	1,719.0	382	38.6
125.45	1024	16	8	2,528.0	422	42.7
192.60	1280	32	4	3,374.0	404	40.9
320.36	1664	32	4	5,156.0	463	46.9

J LOCAL VALIDATION PROTOCOL AND SUPPORTING RESULTS

Evaluation sample and interpretation. The local evaluation uses the frozen 2022–2025 validation protocol, with 30,720 windows across 488 tickers. The same window identities, tokenisation, aggregation, and variant-specific scoring masks are used for every checkpoint. The 399,360,000 scored positions per evaluation are token occurrences, not independent samples; overlapping or dependent observations limit the interpretation of the sample size. Logged training CE instead averages ten local losses over changing training batches and is not a cross-node average over a fixed corpus. Fixed validation inputs improve comparability without establishing future-market generalisation or, from the sampling recipe alone, absence of overlap with every historical training stream.

Cohort audit and fitting range. The local archive contains 132 evaluated checkpoints from 33 labelled runs. One observation belongs to a run that did not reach its intended budget and is excluded, leaving 131 checkpoints from 32 completed lineages. Excluding all 17 checkpoints of the 2.626M and 3.911M sizes leaves the 114-point, 26-lineage, ten-size cohort used for every result below. The full-range final-quarter fit gives $(\alpha, \beta) = (1.942, 1.273)$, retained here as a sensitivity comparison. The fitting-domain choice is exploratory and was made after inspecting validation losses; uncertainty conditional on this domain does not account for that selection.

The mean endpoint training exposure of 82.63B cited in Section 5.1 gives equal weight to the twelve original model sizes, rather than to individual runs. Short exposure is not confined to the smallest models: the retained 196.572M and 293.283M models reach 26.99B and 17.77B tokens and have lower CE. All six excluded small-model trajectories decrease over their available late checkpoints. These facts do not establish failed runs or isolate the effects of capacity, exposure and optimisation.

Checkpoint coverage. Training exposure remains $D = 1{,}664{,}000s$, including exposure before checkpoint restores; evaluation tokens do not contribute to D. Endpoint selection requires reaching the intended budget. The 78.539M and 119.280M models each have three local validation observations, one endpoint per run, rather than three training stages. The final 25%, 40%, and 50% selections contain the same 114 available points; their identical fits reflect limited earlier coverage, not independent robustness evidence. Observed training exposures span 13.84B–177.90B tokens, but this is not a fully observed rectangular size–exposure design. Table 4 in the main text reports the corresponding fits.

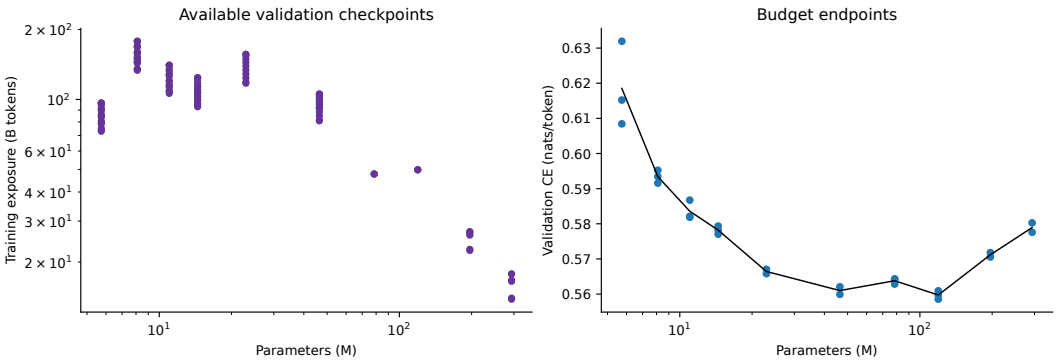

Figure 16: Validation coverage and endpoint CE restricted to 5.736–293.283M parameters. The right-hand line connects size means. Endpoint comparisons combine changes in size and training exposure.

Scoring fine-time components. For applications concerned primarily with approximate event arrival, price and quantity, the finest temporal components may be less relevant or harder to predict. This motivates testing dependence on their contribution to CE, without assuming that they are pure noise or unimportant to all high-frequency applications. Removing ns excludes three scored positions, one each for inter-event interval, event time and reference time; removing ns and us excludes six. Coarser timing, prices and quantities remain scored. Inputs, weights and cumulative training

exposure are unchanged, and removed loss contributions are zeroed while retaining the original 26-position denominator. These scores are not renormalised per retained token and are not likelihoods of newly trained, coarser-resolution models. A lower score partly follows mechanically from omitting non-negative loss terms.

Fitting sensitivity. The same objective, bounds and run weighting as Equation 2 are used. Wider exponent bounds are $[0.01, 5]$; the relaxed-floor analysis permits E up to twice the largest observed loss. The sample-weighted variant weights windows directly rather than weighting tickers equally. All rows retain the same 114 checkpoints and do not reintroduce the excluded sizes. The results are reported in Table 5 in the main text.

Bootstrap and held-out prediction. All 2,000 within-size, whole-lineage bootstrap fits converge. The medians and 95% percentile intervals are 0.737 $[0.546, 1.172]$ for α and 0.581 $[0.430, 0.848]$ for β. Boundary rates are 0.0% for α, 0.1% for β, and 0.0% for the floor. Resampling whole lineages preserves within-trajectory dependence, but two or three lineages per size limit independent replication. Neither the fixed validation sample nor the model-size grid is resampled.

For held-out prediction, all selected observations from an omitted lineage or size are excluded before refitting. Every reported fold converges. Whole-size errors include extrapolation beyond the retained training-fold support and are not pure interpolation errors. They do not assess the excluded small sizes or future-market data.

Table 24: Held-out prediction within the restricted size cohort. Whole-size errors include extrapolation beyond a fold's retained support.

Holdout	Model	RMSE	Failed folds
Lineage	additive	0.0044	0/26
Lineage	parameter only	0.0084	0/26
Lineage	wide additive	0.0044	0/26
Size	additive	0.0043	0/10
Size	parameter only	0.0086	0/10
Size	wide additive	0.0043	0/10