SLIM: SIMPLEX-LATTICE INTERPOLATION MERGING

Anonymous authors
Paper under double-blind review

ABSTRACT

Optimizing merging coefficients for large language models can require many costly benchmark evaluations. We propose **Simplex-Lattice Interpolation Merging (SLIM)**, which constructs a quadratic surrogate of aggregate performance on the coefficient simplex using a classical mixture design. Evaluations of individual experts and equal-weight pairs determine the surrogate with the minimum number of measurements needed to identify a general quadratic on this domain. SLIM then optimizes the surrogate without further target-metric evaluations. Experiments on two model architectures demonstrate accurate prediction of unseen multi-expert mixtures and competitive merge performance under limited evaluation budgets. Matched-budget comparisons show that structured evaluation points improve prediction fidelity over random designs, including those using regularized fitting.

1 INTRODUCTION

Model merging can combine capabilities acquired by multiple trained models without further parameter training (Yang et al., 2026; Song & Zheng, 2026). For large language models (LLMs), reusing publicly available fine-tuned checkpoints can reduce the additional cost of building models with multiple capabilities. We study the widely used setting of merging *experts*: models with the same architecture fine-tuned from a common base. Task Arithmetic (Ilharco et al., 2023) adds a weighted sum of task vectors—the parameter differences between experts and the base—to the base model. Even in this linear setting, the coefficients controlling each expert's contribution affect merge performance.

We call the downstream measure to be improved the *target metric*; here, we use AVG, the unweighted mean of domain scores on a common scale. On generation-based LLM benchmarks, evaluating a candidate requires constructing merged weights, generating responses, and scoring outputs, making searches such as evolutionary merging costly (Akiba et al., 2025; Su & Geiping, 2025). A surrogate that predicts performance from coefficients can support coefficient selection using inexpensive predictions (Li et al., 2025). For example, MAP fits task-wise surrogates to measured scores and uses them to approximate Pareto fronts. However, constructing the surrogate itself incurs an upfront evaluation cost because fitting it requires target-metric measurements from merged models. For LLMs, these measurements remain expensive even when subsequent surrogate optimization is cheap. We therefore ask: *How few target-metric evaluations are needed to construct a surrogate that supports competitive merge performance?*

Quadratic surrogates capture curvature and pairwise interactions with relatively few parameters, a choice also used by MAP (Li et al., 2025). To reduce the required evaluations, we restrict coefficients to the simplex, where they are nonnegative and sum to one. Weight averaging and MetaGPT provide precedents for effective merging within this domain (Wortsman et al., 2022; Zhou et al., 2024). For T coefficients, the restriction reduces the number of independent quadratic parameters from $(T+1)(T+2)/2$ to $T(T+1)/2$. However, the identification minimum does not guarantee stable fitting or accurate prediction, which also depend on evaluation placement and the adequacy of the quadratic approximation. We test whether measurements of individual experts and pairs suffice to predict AVG at unseen mixtures, and whether coefficient selection within the simplex yields competitive merge performance.

We propose Simplex-Lattice Interpolation Merging (SLIM), which constructs a quadratic surrogate from exactly $T(T+1)/2$ target-metric evaluations. SLIM adopts Scheffé's $\{T, 2\}$ simplex-lattice

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.

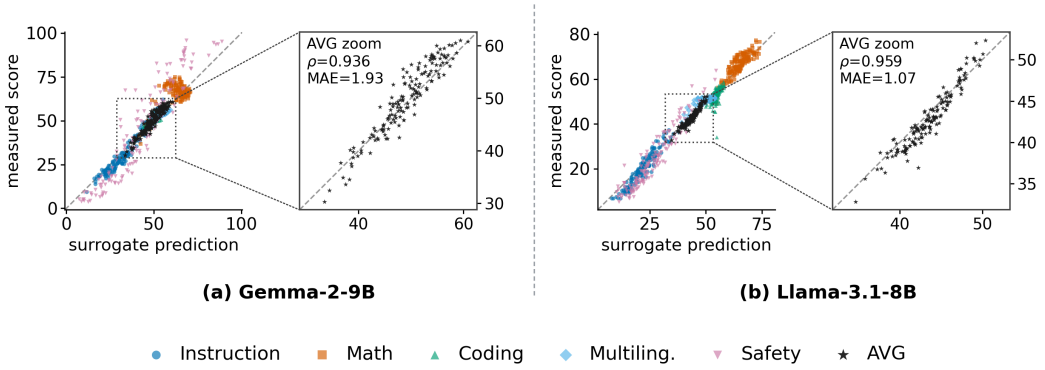

Figure 1: A quadratic surrogate constructed from individual experts and equal-weight pairs accurately predicts AVG at unseen mixtures of all eight experts. For each architecture, 36 construction evaluations are followed by testing on 150 held-out interior vectors sampled from $\mathrm{Dirichlet}(1, \ldots, 1)$. Axes show predicted (horizontal) and measured (vertical) scores; colored markers show domain scores and black stars show AVG. AVG zoom panels report Spearman's ρ and MAE; dashed lines indicate $y = x$.

design from classical mixture experiments (Scheffé, 1958). Its vertices and edge midpoints correspond to individual experts and equal-weight expert pairs. SLIM interpolates their measured AVG values and numerically optimizes the resulting quadratic over the simplex using its analytic gradient, without further target-metric evaluations.

We evaluate SLIM on Gemma-2-9B (Gemma Team et al., 2024) and Llama-3.1-8B (Grattafiori et al., 2024), each with eight experts spanning five domains. The surrogates accurately predict AVG at unseen mixtures involving all eight experts (Figure 1). At the same evaluation budget, the lattice design improves prediction fidelity over random designs, including ridge-regularized fits. We assess selected models on in-distribution (ID) benchmarks used for selection and held-out out-of-distribution (OOD) benchmarks. With 36 evaluations, SLIM exceeds MAP at 45 evaluations in all four architecture-by-distribution settings and remains competitive with evolutionary merging using 150–200 evaluations in three settings (Figure 3). On Llama ID, methods searching the broader coefficient domain achieve higher performance.

Contributions Our contributions are twofold:

- **Predictability:** We show that evaluations of individual experts and equal-weight pairs support accurate quadratic prediction of AVG at unseen mixtures involving all eight experts on both architectures studied.
- **Minimal-cost selection:** We propose SLIM, which selects coefficients using a surrogate constructed from the $T(T+1)/2$ evaluations needed to identify a general quadratic on the simplex.

2 Related Work

Model Merging. A *merging operator* specifies how expert parameters or task vectors are transformed and combined. Both the operator and the coefficients controlling each expert's contribution affect merge performance. Task Arithmetic (Ilharco et al., 2023) combines task vectors linearly, while TIES-Merging (Yadav et al., 2023) and DARE (Yu et al., 2024) transform parameter updates to address interference or redundancy. Existing methods select coefficients using data-free rules (Zhou et al., 2024), prediction entropy minimization (Yang et al., 2024), or search guided by measured downstream performance (Akiba et al., 2025). Evaluating candidate LLM merges can be computationally expensive, especially on generation-based benchmarks that require generating and scoring model outputs (Su & Geiping, 2025). To reduce evaluation costs, MERGE^3 (Mencattini et al., 2025) combines evolutionary merging with dataset reduction and performance estimation

based on item response theory. We focus on the widely studied setting of linear task-vector merging of LLM experts fine-tuned from a common base (Yang et al., 2026; Song & Zheng, 2026). Within this setting, we investigate how few direct target-metric evaluations can support coefficient selection that yields competitive merge performance.

Surrogate-Based Coefficient Optimization. Surrogate-based methods fit predictive models to measured merge performance and use their predictions to guide coefficient or configuration search. MM-MO (Li et al., 2026) sequentially selects DARE–TIES coefficients and sparsity levels using Gaussian process surrogates and multi-objective Bayesian optimization. Multi-fidelity merging (Su & Geiping, 2025) uses a random forest surrogate to select configurations and varies the size of the validation set to allocate evaluation resources. MAP (Li et al., 2025) is closely related to our work: it uses surrogates based on quadratic functions to approximate task performance and estimate Pareto fronts. In its basic form, MAP fits these surrogates to scores measured at randomly sampled coefficient vectors, without requiring the coefficients to sum to one. Constructing these surrogates incurs an upfront evaluation cost because the target metrics must be measured on merged models. Bayesian MAP and Nested MAP reduce evaluation costs through adaptive sampling and hierarchical merging, respectively, but both rely on measured scores for surrogate construction. For T experts, SLIM uses a fixed experimental design to interpolate AVG on the coefficient simplex from $T(T+1)/2$ evaluations, the minimum needed to identify a general quadratic on this domain. We test whether this surrogate can predict AVG at unseen multi-expert mixtures and support coefficient selection that yields competitive merge performance.

Mixture Experiments. Mixture experiments are a classical statistical framework for studying how a response depends on the relative proportions of components. Applications include studying how ingredient proportions affect food flavor and how solvent proportions affect solubility in mixed-solvent formulations (Cornell, 2002). Since component proportions are nonnegative and sum to one, the experimental domain is a simplex. We view linear task-vector merging under this simplex constraint as a mixture experiment. The components are LLM experts, their merging coefficients specify the proportions, and the response is the merged model's target metric, AVG. For T components, a general quadratic on the simplex has $T(T+1)/2$ independent coefficients. Scheffé's $\{T, 2\}$ simplex-lattice design uniquely determines this quadratic from evaluations at the T vertices and $\binom{T}{2}$ edge midpoints (Scheffé, 1958). Under a quadratic regression model with uncorrelated, equal-variance errors, a design that allocates observations equally across these points is D-optimal (Kiefer, 1961). When a design is D-optimal, it maximizes the joint precision of the regression coefficient estimates for a fixed number of observations. Recent work applies sparse Scheffé models and optimal design to LLM pretraining data mixtures (Mao & Du, 2026). In that setting, mixture proportions are the fractions of training tokens drawn from each data domain, and the response is validation loss. SLIM adopts the classical simplex-lattice design and its quadratic interpolant. We empirically test whether evaluations of individual experts and equal-weight pairs suffice to predict AVG at unseen mixtures, and whether restricting coefficients to the simplex permits competitive merge performance.

3 METHOD

3.1 PROBLEM SETTING

Consider T expert models $\theta_1, \ldots, \theta_T$ fine-tuned from a common base model θ_{pre}. With task vectors $\tau_k = \theta_k - \theta_{\text{pre}}$, linear merging gives

$$\theta(\lambda) = \theta_{\text{pre}} + \sum_{k=1}^{T} \lambda_k \tau_k. \tag{1}$$

Let $M(\lambda)$ denote the target metric of $\theta(\lambda)$, with higher values indicating better performance. For a feasible coefficient set $\Lambda \subseteq \mathbb{R}^T$, the coefficient-selection problem is

$$\lambda^\star \in \arg\max_{\lambda \in \Lambda} M(\lambda). \tag{2}$$

In our experiments, M is AVG, the unweighted mean of domain scores expressed on a common scale.

3.2 Quadratic Surrogate and Evaluation Design

To select coefficients without repeatedly measuring the expensive target metric during optimization, SLIM approximates M with a quadratic surrogate q (Algorithm 1). Quadratic models capture curvature and pairwise interactions, a choice also used by MAP (Li et al., 2025). Our goal is to construct this surrogate from few measurements and then optimize its predictions.

To reduce the number of surrogate parameters to identify, we restrict the feasible set to the simplex:

$$\Lambda = \Delta^{T-1} = \left\{ \lambda \in \mathbb{R}^T \;\middle|\; \lambda_k \geq 0 \text{ for all } k, \quad \sum_{k=1}^{T} \lambda_k = 1 \right\}. \tag{3}$$

Any quadratic on this domain has the canonical mixture form (Scheffé, 1958)

$$q(\lambda) = \sum_{k=1}^{T} \beta_k \lambda_k + \sum_{i<j} \beta_{ij} \lambda_i \lambda_j. \tag{4}$$

It has $T(T+1)/2$ independent coefficients. Identifying them with the minimum number of measurements requires an evaluation design that supplies independent constraints.

Let e_k be the k-th standard basis vector in $\mathbb{R}^T$. SLIM adopts Scheffé's quadratic $\{T, 2\}$ simplex-lattice design (Scheffé, 1958), evaluating M at the vertices and edge midpoints:

$$\mathcal{D} = \{e_k\}_{k=1}^{T} \cup \left\{ \frac{e_i + e_j}{2} \;\middle|\; 1 \leq i < j \leq T \right\}. \tag{5}$$

These points correspond to individual experts, $\theta(e_k) = \theta_k$, and equal-weight pairs, $\theta((e_i+e_j)/2) = (\theta_i + \theta_j)/2$. The vertex measurements determine the linear coefficients, and the midpoint measurements determine the pairwise coefficients. The total evaluation count is

$$|\mathcal{D}| = T + \binom{T}{2} = \frac{T(T+1)}{2}. \tag{6}$$

Figure 2 illustrates the quadratic design for $T = 3$ and its cubic counterpart used in Section 5.3.

3.3 Quadratic Interpolation

Write $M_k = M(e_k)$ and $M_{ij} = M((e_i + e_j)/2)$. The unique quadratic satisfying these measurements is

$$\widehat{M}(\lambda) = \sum_{k=1}^{T} \lambda_k (2\lambda_k - 1) M_k + \sum_{i<j} 4\lambda_i \lambda_j M_{ij}. \tag{7}$$

Each basis function equals one at its associated design point and zero at the others, so $\widehat{M}$ reproduces all measurements on $\mathcal{D}$. Appendix B.1 derives the coefficients and proves uniqueness and the identification minimum.

By linearity, interpolating AVG directly is equivalent to averaging the domain-wise interpolants (Appendix B.2). Agreement with M is guaranteed at the design points; prediction accuracy at unseen mixtures is assessed in Section 4.2.

3.4 Coefficient Selection

A constrained numerical optimizer returns $\widehat{\lambda}$ by approximately solving

$$\underset{\lambda \in \Delta^{T-1}}{\text{maximize}} \quad \widehat{M}(\lambda). \tag{8}$$

Because $\widehat{M}$ need not be concave, the returned solution is not guaranteed to be a global maximizer.

Surrogate optimization requires no model evaluations beyond the $T(T+1)/2$ design-point measurements. The final merged model is $\theta(\widehat{\lambda})$ from Eq. 1.

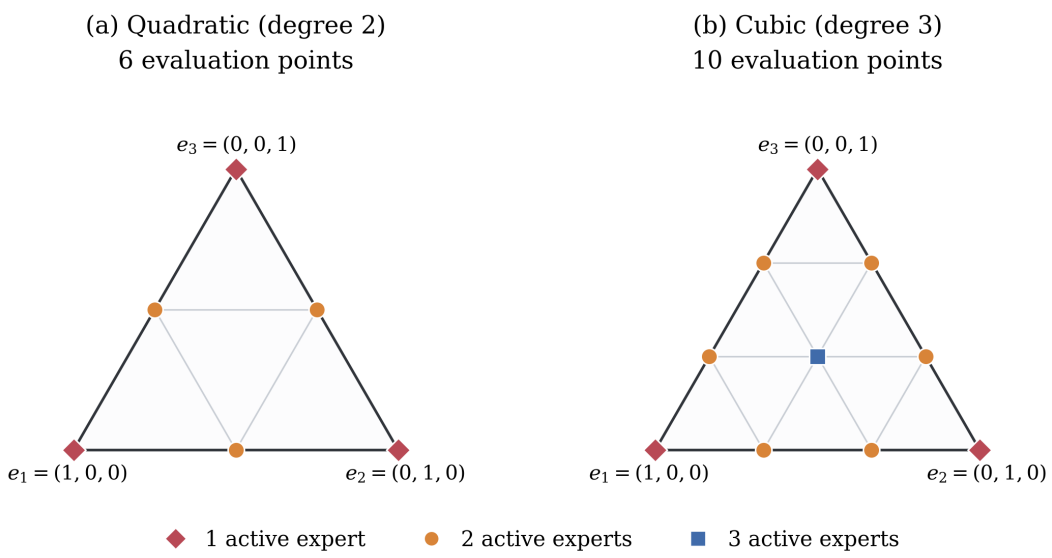

Figure 2: Simplex-lattice designs for $T = 3$. Each point is a coefficient vector with nonnegative entries summing to one. The quadratic design contains three vertices and three edge midpoints; the cubic design contains the vertices, two points on each edge with weights $(2/3, 1/3)$ and $(1/3, 2/3)$, and the barycenter $(1/3, 1/3, 1/3)$. Colors and marker shapes indicate the number of experts with nonzero coefficients. For $T = 8$, the corresponding designs contain 36 and 120 points.

Domain	ID (used for selection)	OOD (held out)
Instruction	IFEval	AlpacaEval 2.0
Math	GSM8K	MATH-500
Coding	MBPP	HumanEval
Multilingual	FLORES+ dev	FLORES+ devtest
Safety	HarmBench	WildGuardTest

Table 1: ID benchmarks for coefficient selection and held-out OOD benchmarks across five domains. Reported ID scores use the selection benchmarks. Metrics, sample counts, and prompting settings appear in Table 6.

4 Experiments

4.1 Experimental Setup

We use Gemma-2-9B (Gemma Team et al., 2024) and Llama-3.1-8B (Grattafiori et al., 2024), each with eight experts fine-tuned from a common base: two each for instruction following, mathematics, and coding, and one each for multilingual translation and safety. All methods use the same experts and linear task-vector merging. The target metric is AVG, the unweighted mean of five domain scores on a 0–100 scale.

Table 1 lists the benchmarks; metrics and evaluation settings are given in Appendix C.2 (Table 6).

Search methods use ID AVG for coefficient selection, and the reported ID scores use these same benchmarks; OOD benchmarks are held out from selection. All merged candidates, including simplex vertices, follow a common evaluation protocol within each architecture and benchmark. Model identifiers, benchmark sources, and implementation details are provided in Appendix C.

Comparisons. Weight averaging (Wortsman et al., 2022) sets $\lambda_k = 1/T$, while MetaGPT (Zhou et al., 2024) uses $\lambda_k = \|\tau_k\|_2^2 / \sum_{j=1}^{T} \|\tau_j\|_2^2$. Both determine $\lambda \in \Delta^{T-1}$ without target-metric

evaluations. Task Arithmetic (Ilharco et al., 2023) evaluates 21 shared coefficients, $\lambda_k = c$ with $c \in \{0, 0.05, \ldots, 1\}$, and selects the candidate with the highest measured ID AVG.

MAP (Li et al., 2025) samples and optimizes coefficients in $[0, 1]^8$. For each seed, one sequence starts with the eight vertices e_k, followed by 127 independent vectors sampled uniformly from $[0, 1]^8$. Budgets $n \in \{45, 90, 135\}$ use the first n points of this sequence. MAP fits domain-wise sigmoid-transformed quadratics, uses NSGA-III to approximate a Pareto front, and selects the vector with the highest predicted AVG.

Evolutionary merging (Akiba et al., 2025) uses CMA-ES to optimize measured ID AVG over $[0, 1]^8$, evaluating 10 candidates per generation and retaining the best observed candidate at each budget. Task Arithmetic, MAP, and evolutionary merging do not impose a sum-to-one constraint.

SLIM constructs a quadratic surrogate of ID AVG from 36 evaluations and optimizes it over Δ^{T-1} using SLSQP. A cubic extension uses 120 evaluations as a degree ablation (Section 5.3). Neither variant requires further target-metric evaluations during surrogate optimization. Implementation details and optimizer settings are given in Appendix C.3.

Evaluation cost and runs. We report #Evals as the hardware-independent coefficient-selection budget and GPU-hours as the measured cost in our environment. Each evaluation constructs, saves, loads, and scores a candidate on all five ID domains using one H100 GPU; GPU-hours sum the elapsed time of these evaluations. Final-model reporting and surrogate-fidelity measurements are excluded. MAP and evolutionary merging are reported as mean $\pm$ standard deviation over seeds 42, 43, and 44; other methods use a single run. SLIM uses a fixed evaluation design and a fixed set of optimizer initializations.

4.2 Surrogate Fidelity

For each architecture, we test the quadratic surrogate on 150 coefficient vectors sampled from $\mathrm{Dirichlet}(1, \ldots, 1)$ with seed 42. Each vector assigns nonzero coefficients to all eight experts, whereas every construction point uses at most two. These points are held out from surrogate construction and coefficient selection; their measured scores use the same ID benchmarks. Since coefficient selection depends on the relative performance of mixtures, we assess agreement between predicted and measured AVG rankings using Spearman's ρ. We also report mean absolute error (MAE) to quantify score prediction errors.

Figure 1 compares predicted and measured scores. For AVG, Spearman's ρ is 0.936 on Gemma-2-9B and 0.959 on Llama-3.1-8B, with MAEs of 1.93 and 1.07 points, respectively. Fidelity varies across domains: for example, Gemma's math scores have $\rho = -0.24$. Domain-wise results are provided in Appendix C.6. The evidence therefore supports approximation of the aggregate objective rather than uniformly accurate prediction in every domain.

We also evaluate the quadratic surrogate at two held-out coefficient vectors on each of the 28 simplex edges. Across these 56 points, MAE is 1.74 on Gemma and 1.38 on Llama; the full edge curves are shown in Appendix C.7.

4.3 Merge Performance and Evaluation Cost

Figure 3 shows the cost–performance trade-off; numerical results at selected budgets are provided in Table 7. Quadratic SLIM uses 36 evaluations, costing 5.2 GPU-hours on Gemma-2-9B and 3.9 on Llama-3.1-8B. It achieves higher AVG than weight averaging in all four architecture-by-distribution settings and exceeds MetaGPT in three; on Gemma OOD, MetaGPT scores slightly higher (50.6 versus 50.3).

SLIM with 36 evaluations outperforms MAP with 45 evaluations in all four settings. On Gemma ID and Llama OOD, SLIM retains higher AVG at every tested MAP budget. On Gemma OOD, SLIM also exceeds MAP at 90 evaluations, while MAP at 135 evaluations reaches 50.6 ± 0.4, close to SLIM's 50.3. On Llama ID, MAP attains higher mean AVG at 90 and 135 evaluations (57.6 ± 5.9 and 60.4 ± 1.2, respectively) than SLIM's 56.6.

At comparable budgets of 30 and 40 evaluations, evolutionary merging scores below SLIM on Gemma ID, Gemma OOD, and Llama OOD. On Llama ID, however, it achieves 58.4 and 58.9,

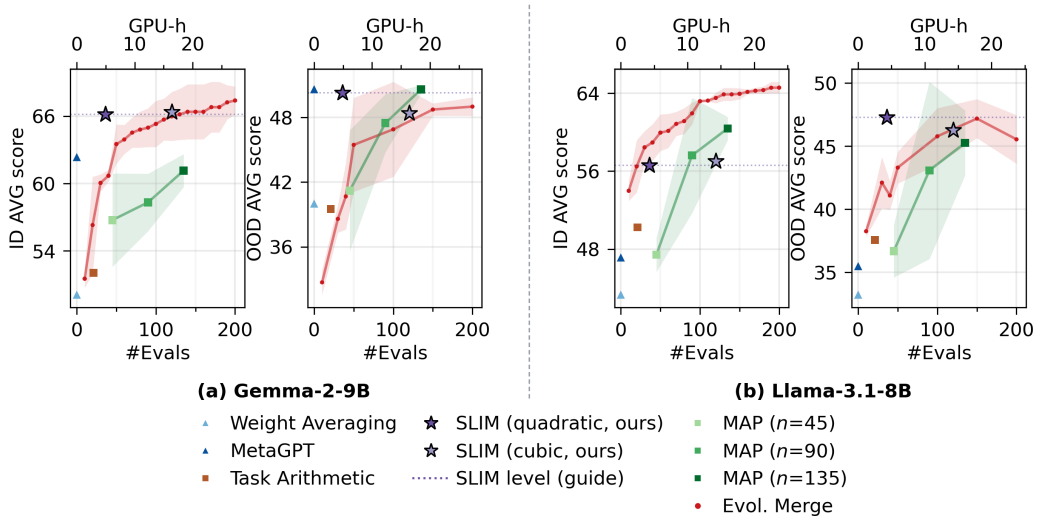

Figure 3: ID and held-out OOD AVG versus the evaluation budget for coefficient selection (ID left, OOD right within each architecture). #Evals counts cumulative ID evaluations per run; upper axes estimate GPU-hours from measured per-evaluation costs. Evolutionary merging evaluates 10 candidates per generation. ID scores are shown after every generation and OOD scores after 10, 30, 40, 50, 100, 150, and 200 evaluations, always for the candidate with the highest ID AVG observed so far. Quadratic and cubic SLIM use 36 and 120 evaluations; dotted lines mark quadratic SLIM's AVG. MAP and evolutionary merging show means over three seeds; shaded bands indicate ± 1 standard deviation.

respectively, exceeding SLIM's 56.6. With evolutionary merging extended to 150–200 evaluations, SLIM remains comparable on Gemma ID, higher on Gemma OOD, and comparable on Llama OOD. These results demonstrate competitive merge performance with a limited evaluation budget in three of the four settings, while also reflecting the methods' different feasible coefficient domains.

4.4 Selected Merging Coefficients

Figure 4 compares the selected coefficients and their sums. Quadratic SLIM concentrates weight on instruction and safety experts on both architectures. Thus, although we evaluate prediction at mixtures involving all eight experts, coefficient optimization can select mixtures concentrated on fewer experts. On Llama, coefficient sums range from 3.31 to 3.75 for MAP and from 2.50 to 2.83 for evolutionary merging, compared with one for SLIM. Both methods achieve higher ID AVG than SLIM at these budgets (Table 7), providing examples of higher-scoring candidates outside the simplex. The selected solutions differ in both relative allocation and overall task-vector scale.

5 Ablation Studies

5.1 Effect of Evaluation-Point Placement

We compare lattice and random evaluation points using the same quadratic function class on the simplex. For each budget n, we draw n of the 150 interior points from Section 4.2 without replacement and fit ID AVG by unregularized least squares (LS) or ridge regression. We repeat this procedure 1,000 times, evaluating all surrogates on the remaining $150 - n$ points in each resample and keeping the 36-point lattice surrogate fixed. Ridge strength is selected by cross-validation using only the fitting points; scaling and fitting details appear in Appendix C.4.

At the matched budget of 36 evaluations, the lattice design achieves higher mean rank correlation and lower mean MAE than either random-design fit on both architectures (Table 2). Ridge reduces the random design's mean MAE from 33.64 to 2.90 on Gemma and from 30.15 to 2.24 on Llama,

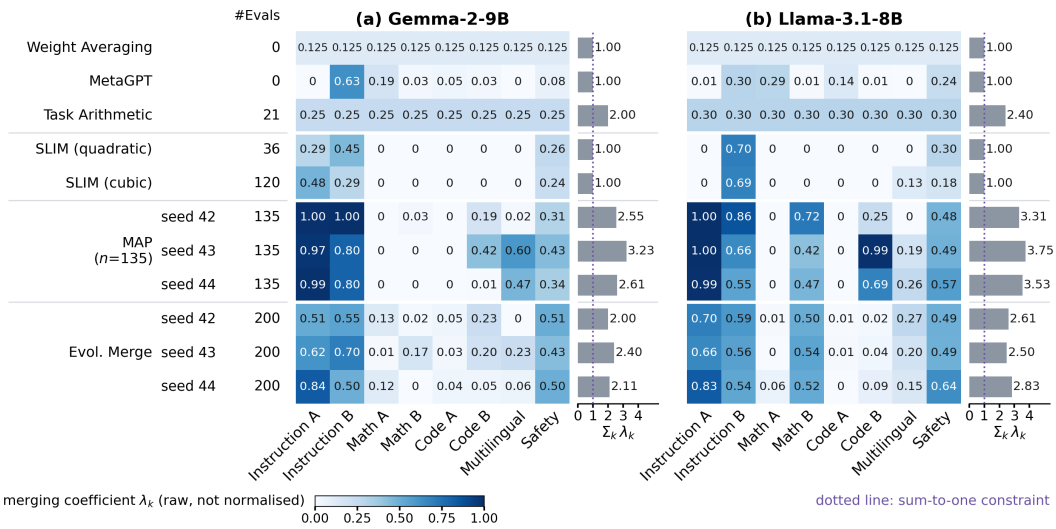

Figure 4: Selected merging coefficients reveal differences in expert allocation and overall scale. Cells show raw coefficients; bars show their sums, with dotted lines marking the sum-to-one constraint. MAP uses 135 evaluations per seed. Evolutionary merging shows the candidate with the highest ID AVG observed within 200 evaluations. Cell labels are rounded, so displayed zeros need not be exact zeros; sums use unrounded values. Expert identities are listed in Table 5.

	Gemma-2-9B				**Llama-3.1-8B**			
n	ρ_{lattice}	ρ_{random}	$\text{MAE}_{\text{random}}$	Lattice win	ρ_{lattice}	ρ_{random}	$\text{MAE}_{\text{random}}$	Lattice win
36	0.935	0.465 / 0.847	33.64 / 2.90	99.5 / 99.5	0.958	0.330 / 0.758	30.15 / 2.24	100.0 / 100.0
45	0.935	0.902 / 0.884	2.46 / 2.50	68.9 / 86.2	0.958	0.819 / 0.800	1.81 / 1.99	99.9 / 99.9
60	0.934	0.954 / 0.948	1.63 / 1.70	12.1 / 18.2	0.957	0.918 / 0.915	1.11 / 1.16	95.6 / 95.9
75	0.933	0.966 / 0.965	1.38 / 1.39	1.7 / 2.4	0.956	0.941 / 0.941	0.93 / 0.94	79.1 / 79.3
90	0.931	0.970 / 0.970	1.27 / 1.27	0.5 / 0.5	0.954	0.950 / 0.950	0.83 / 0.84	57.9 / 57.1
105	0.930	0.972 / 0.972	1.19 / 1.19	0.5 / 0.6	0.953	0.953 / 0.954	0.80 / 0.80	44.2 / 43.8
120	0.924	0.970 / 0.970	1.14 / 1.14	1.1 / 1.0	0.949	0.953 / 0.953	0.75 / 0.76	39.5 / 39.0

Table 2: The lattice improves AVG prediction at the matched budget of 36 evaluations. Random budgets are n; the lattice always uses 36. Values average 1,000 resamples with shared held-out points; pairs show LS / ridge. Lattice win is the percentage with $\rho_{\text{lattice}} > \rho_{\text{random}}$. Lattice MAEs average approximately 1.93 (Gemma) and 1.07 (Llama).

compared with approximately 1.93 and 1.07 for the lattice. Among the tested budgets, the lattice's win rate first falls below 50% at $n = 60$ for Gemma and $n = 105$ for Llama, against random designs fitted with either LS or ridge.

5.2 Effect of Surrogate Optimization

We compare quadratic SLIM with directly selecting the point with the highest measured ID AVG among the same 36 lattice points (Table 3). Most of the ID gain over vertex-only selection comes from adding equal-weight expert pairs to the candidates. Surrogate optimization provides a smaller additional ID gain; OOD AVG decreases on Gemma and increases on Llama.

5.3 Effect of Polynomial Degree

We compare quadratic SLIM with a cubic extension using 120 lattice points and the same simplex-constrained optimization procedure (Appendix B.4). The cubic extension yields small ID gains but

		Gemma-2-9B		Llama-3.1-8B	
Variant	#Evals	ID	OOD	ID	OOD
Best vertex	8	58.51	49.73	54.80	46.41
Best lattice point	36	65.37	**51.85**	56.56	42.93
SLIM (quadratic)	36	66.15	50.30	56.61	**47.27**
SLIM (cubic)	120	**66.35**	48.41	**57.04**	46.27

Table 3: Effects of evaluating expert pairs, optimizing the surrogate, and increasing its degree on merge performance. The first two rows select by measured ID AVG among eight vertices or 36 lattice points; SLIM optimizes its surrogate. #Evals counts ID evaluations for selection; OOD is held out. Bold marks column maxima.

lower OOD AVG on both architectures (Table 3; Figure 3). Increasing the degree and evaluation budget therefore does not consistently improve merge performance in these experiments.

6 Discussion and Limitations

Quadratic Approximation and Model Degree. Evaluations of individual experts and equal-weight pairs support AVG prediction at mixtures of all eight experts on both architectures (Figure 1), although domain-wise fidelity varies (Table 8). Increasing the degree from quadratic to cubic raises the budget from 36 to 120 evaluations but produces small ID gains and lower OOD AVG on both architectures (Table 3). These results support quadratic approximation in the tested settings and show limited benefit from increasing the degree. When greater surrogate expressiveness improves generalization remains an open question.

Evaluation Design and Sample Efficiency. At 36 evaluations, the lattice yields higher mean rank correlation and lower MAE than random designs, including ridge-regularized fits (Table 2). Evaluation placement therefore matters even with the surrogate class and coefficient domain fixed. The $T(T+1)/2$ count is the minimum for identifying a general quadratic on the simplex, not a universal lower bound for finding a good merge.

The Value of Surrogate Optimization. Most ID improvement over the best individual expert comes from adding equal-weight pairs to the candidates (Table 3). Optimizing beyond the best measured lattice point yields small ID gains but lowers OOD AVG on Gemma and raises it on Llama. The surrogate enables continuous coefficient selection without further target-metric evaluations; its additional performance benefit depends on the setting.

The Role of the Simplex Constraint. The OOD results show that the simplex contains competitive merges that generalize beyond the selection benchmarks (Figure 3). On Llama ID, MAP and evolutionary merging select higher-scoring candidates with coefficient sums above one (Figure 4). Although these comparisons do not isolate the effect of coefficient scale, they motivate varying both mixture proportions and overall task-vector scale while controlling the cost of surrogate construction.

7 Conclusion

SLIM selects LLM merging coefficients using a quadratic surrogate built from individual experts and equal-weight pairs, with the minimum evaluations required for quadratic identification on the simplex. Experiments on two architectures show accurate prediction of aggregate performance at unseen mixtures and competitive merging under limited evaluation budgets. Comparisons at a matched budget highlight evaluation-point placement as an important design choice for surrogate-based coefficient selection.

AI USE STATEMENT

We used generative AI tools (ChatGPT, Claude, and Claude Code) for research ideation and discussion, literature search, writing and debugging code for experiments and figures, running experiments and analyzing results, drafting the manuscript, English editing and paraphrasing, creating figures and tables, and checking the mathematical derivations in Appendix B. We have reviewed all AI-assisted work. LLM-generated code was tested by the authors, and the resulting numerical values were verified by manual calculation or against an independent implementation. All references identified with AI assistance were checked against the original sources. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, personally identifiable information, or newly collected data. We are not aware of ethical concerns raised by this work beyond those associated with the publicly available models and benchmarks it uses.

REPRODUCIBILITY STATEMENT

Section 3 and Algorithm 1 describe SLIM, and Appendices B.1– B.4 derive the quadratic interpolant, its uniqueness, its equivalence under domain aggregation, and the cubic extension. Appendix C.1 (Table 5) lists all base and expert checkpoints; Appendix C.2 (Table 6) specifies the benchmarks, metrics, sample counts, and prompting settings; and Appendix C.3 details the implementations of SLIM and all baselines, including random seeds, optimizer settings, and initializations. Upon acceptance, we will release our code together with the measured scores used in our experiments, including those at the lattice design points and the 150 held-out mixtures. These measurements allow the surrogate-fidelity results and SLIM's coefficient selection to be reproduced without re-evaluating merged models.

REFERENCES

Takuya Akiba, Makoto Shing, Yujin Tang, Qi Sun, and David Ha. Evolutionary optimization of model merging recipes. *Nature Machine Intelligence*, 7(2):195–204, 2025.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Laurie Burchell, Jean Maillard, Antonios Anastasopoulos, Christian Federmann, Philipp Koehn, and Skyler Wang. Findings of the WMT 2024 shared task of the open language data initiative. In *Proceedings of the Ninth Conference on Machine Translation*, pp. 110–117, 2024.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

John A. Cornell. *Experiments with Mixtures: Designs, Models, and the Analysis of Mixture Data*. John Wiley & Sons, 3rd edition, 2002.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*, 2024.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. Wildguard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of llms. *Advances in neural information processing systems*, 37:8093–8131, 2024.

Yifei He, Siqi Zeng, Yuzheng Hu, Rui Yang, Tong Zhang, and Han Zhao. Mergebench: A benchmark for merging domain-specialized LLMs. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025. URL https://openreview.net/forum?id=rw50iUoyLu.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

J Kiefer. Optimum designs in regression problems, ii. *The Annals of Mathematical Statistics*, 32(1): 298–325, 1961.

Bingdong Li, Zixiang Di, Yanting Yang, Hong Qian, Peng Yang, Hao Hao, Ke Tang, and Aimin Zhou. It's morphing time: Unleashing the potential of multiple llms via multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 30(4):1699–1713, 2026. doi: 10.1109/TEVC.2025.3613937.

Lu Li, Tianyu Zhang, Zhiqi Bu, Suyuchen Wang, Huan He, Jie Fu, Yonghui Wu, Jiang Bian, Yong Chen, and Yoshua Bengio. Map: Low-compute model merging with amortized pareto fronts via quadratic approximation. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 65032–65064, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a3724221b623bf1966e92cd266ac6177-Paper-Conference.pdf.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Yicheng Mao and Hongru Du. Data mixing as mixture experiment: Response surface methodology and optimal design for large language model pretraining. *arXiv preprint arXiv:2608.23922*, 2026.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35181–35224. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/mazeika24a.html.

Tommaso Mencattini, Adrian Robert Minut, Donato Crisostomi, Andrea Santilli, and Emanuele Rodolà. Merge^3: Efficient evolutionary merging on consumer-grade gpus. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025. URL https://openreview.net/pdf?id=qFXDv0X4yc.

NLLB Team. Scaling neural machine translation to 200 languages. *Nature*, 630:841–846, 2024. doi: https://doi.org/10.1038/s41586-024-07335-x.

Henry Scheffé. Experiments with mixtures. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):344–360, 1958.

Mingyang Song and Mao Zheng. Model merging in the era of large language models: Methods, applications, and future directions. *arXiv preprint arXiv:2603.09938*, 2026.

Guinan Su and Jonas Geiping. Fine, i'll merge it myself: A multi-fidelity framework for automated model merging. *arXiv preprint arXiv:2502.04030*, 2025.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pp. 23965–23998. PMLR, 2022.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in neural information processing systems*, 36:7093–7115, 2023.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. *The Twelfth International Conference on Learning Representations*, 2024.

Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications, and opportunities. *ACM Comput. Surv.*, 58(8):216–257, 2026.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Yuyan Zhou, Liang Song, Bingning Wang, and Weipeng Chen. Metagpt: Merging large language models using model exclusive task arithmetic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1711–1724, 2024.

APPENDIX CONTENTS

A Notation **13**

B Additional Details on the Methods **14**

B.1 Derivation and Uniqueness 14

B.2 Linearity under Domain Aggregation 14

B.3 SLIM Algorithm 15

B.4 Cubic Extension for the Degree Ablation 15

C Additional Details on Experiments **16**

C.1 Models and Checkpoints 16

C.2 Benchmarks and Evaluation Protocol 16

C.3 Coefficient-Selection Implementations 17

C.4 Random-Design Fits 18

C.5 Numerical Results for Figure 3 18

C.6 Domain-wise Surrogate Fidelity 18

C.7 Quadratic Interpolation along Simplex Edges 18

A NOTATION

Symbol	Meaning
T	Number of expert models.
$\theta_{\mathrm{pre}}, \theta_k$	Parameters of the shared base model and expert k.
τ_k	Task vector $\theta_k - \theta_{\mathrm{pre}}$.
λ, λ_k	Merging coefficient vector and its k-th component.
$\theta(\lambda)$	Merged parameters $\theta_{\mathrm{pre}} + \sum_k \lambda_k \tau_k$.
Λ	General feasible set of merging coefficients.
Δ^{T-1}	Coefficient simplex: $\lambda_k \geq 0$ and $\sum_k \lambda_k = 1$.
e_k	k-th standard basis vector; the vertex corresponding to expert k.
$\mathcal{D}$	Evaluation design consisting of the vertices and edge midpoints.
$M(\lambda)$	Measured target metric of $\theta(\lambda)$; AVG in the experiments.
M_k, M_{ij}	Measurements $M(e_k)$ and $M((e_i + e_j)/2)$.
$\widehat{M}(\lambda)$	Quadratic interpolant constructed from measurements on $\mathcal{D}$.
q, β_k, β_{ij}	A quadratic in canonical mixture form and its linear and pairwise coefficients.
$\varphi_a(\lambda)$	Interpolation basis function associated with design point $a \in \mathcal{D}$.
$\lambda^\star$	A maximizer of M over Λ.
$\widehat{\lambda}$	Coefficients returned by numerical optimization of $\widehat{M}$ over Δ^{T-1}.
D, M_d, w_d	Number of domains, domain-d metric, and its fixed aggregation weight; $w_d = 1/D$ for AVG.
$\widehat{M}_d$	Interpolant of domain-d measurements on $\mathcal{D}$.
n	Number of evaluation points used by MAP or a random-design surrogate.

Table 4: Symbols defining the merging coefficients, target metric, and surrogate construction.

B Additional Details on the Methods

B.1 Derivation and Uniqueness

On the simplex, $1 = \sum_k \lambda_k$ and $\lambda_k^2 = \lambda_k - \sum_{j \neq k} \lambda_k \lambda_j$. Thus, constant and squared terms can be absorbed into linear and pairwise terms, giving the canonical mixture form in Eq. 4 (Scheffé, 1958).

At each vertex, interpolation requires $\beta_k = q(e_k) = M_k$. At the midpoint of experts i and j,

$$q\left(\frac{e_i + e_j}{2}\right) = \frac{\beta_i + \beta_j}{2} + \frac{\beta_{ij}}{4} = M_{ij}, \tag{9}$$

so

$$\beta_{ij} = 4M_{ij} - 2M_i - 2M_j. \tag{10}$$

The vertex and midpoint values uniquely determine all $T + \binom{T}{2}$ coefficients. Conversely, each evaluation supplies one linear constraint on these coefficients, so fewer evaluations cannot identify an unrestricted member of this quadratic space.

Substituting $\beta_k = M_k$ and Eq. 10 into Eq. 4 gives

$$\begin{aligned} q(\lambda) &= \sum_k M_k \lambda_k + \sum_{i<j} (4M_{ij} - 2M_i - 2M_j)\lambda_i \lambda_j \\ &= \sum_k M_k \lambda_k + \sum_{i<j} 4M_{ij}\lambda_i\lambda_j - 2\sum_{i<j}(M_i + M_j)\lambda_i\lambda_j \\ &= \sum_k M_k \lambda_k + \sum_{i<j} 4M_{ij}\lambda_i\lambda_j - 2\left(\sum_{i<j} M_i\lambda_i\lambda_j + \sum_{i<j} M_j\lambda_i\lambda_j\right). \end{aligned}$$

Regrouping the pairwise terms by expert index k gives

$$\begin{aligned} q(\lambda) &= \sum_k M_k \lambda_k + \sum_{i<j} 4M_{ij}\lambda_i\lambda_j - 2\sum_k M_k\lambda_k\left(\sum_{\ell>k}\lambda_\ell + \sum_{\ell<k}\lambda_\ell\right) \\ &= \sum_k M_k \lambda_k + \sum_{i<j} 4M_{ij}\lambda_i\lambda_j - 2\sum_k M_k\lambda_k \sum_{\ell\neq k}\lambda_\ell. \end{aligned}$$

The simplex constraint implies $\sum_{\ell\neq k}\lambda_\ell = \sum_\ell \lambda_\ell - \lambda_k = 1 - \lambda_k$. Therefore,

$$\begin{aligned} q(\lambda) &= \sum_k M_k \lambda_k + \sum_{i<j} 4M_{ij}\lambda_i\lambda_j - 2\sum_k M_k\lambda_k(1-\lambda_k) \\ &= \sum_k \lambda_k(2\lambda_k - 1)M_k + \sum_{i<j} 4\lambda_i\lambda_j M_{ij}, \end{aligned}$$

which is Eq. 7.

B.2 Linearity under Domain Aggregation

For domain metrics M_d and fixed weights w_d, define

$$M(\lambda) = \sum_{d=1}^{D} w_d M_d(\lambda). \tag{11}$$

AVG corresponds to $w_d = 1/D$. Let φ_a denote the basis function in Eq. 7 associated with design point $a \in \mathcal{D}$. Since these functions do not depend on the measured values,

$$\begin{aligned} \widehat{M}(\lambda) &= \sum_{a\in\mathcal{D}} \varphi_a(\lambda) \sum_{d=1}^{D} w_d M_d(a) \\ &= \sum_{d=1}^{D} w_d \sum_{a\in\mathcal{D}} \varphi_a(\lambda) M_d(a) = \sum_{d=1}^{D} w_d \widehat{M}_d(\lambda). \end{aligned}$$

Thus, interpolating aggregated measurements and aggregating domain-wise interpolants give the same surrogate. No additional model evaluations are needed when the domain scores are recorded at the same design points.

B.3 SLIM ALGORITHM

Algorithm 1 SLIM: Simplex-Lattice Interpolation Merging

Require: Base parameters θ_{pre}; task vectors $\{\tau_k\}_{k=1}^{T}$; target metric M
Ensure: Merge coefficients $\widehat{\lambda}$
1: Construct $\mathcal{D}$ using Eq. 5
2: **for all** $\lambda \in \mathcal{D}$ **do**
3: Form $\theta(\lambda)$ using Eq. 1
4: Evaluate the model and store $M(\lambda)$
5: **end for**
6: Construct $\widehat{M}$ using Eq. 7
7: Numerically maximize $\widehat{M}$ over Δ^{T-1} to obtain $\widehat{\lambda}$
8: **return** $\widehat{\lambda}$

B.4 CUBIC EXTENSION FOR THE DEGREE ABLATION

For the polynomial-degree ablation, we construct a cubic interpolant on the same coefficient simplex using the $\{T, 3\}$ simplex-lattice design (Scheffé, 1958):

$$\mathcal{A}_3 = \left\{ \alpha \in \mathbb{Z}_{\geq 0}^{T} \,\middle|\, \sum_{k=1}^{T} \alpha_k = 3 \right\}, \qquad \mathcal{D}_3 = \left\{ \frac{\alpha}{3} \,\middle|\, \alpha \in \mathcal{A}_3 \right\}. \tag{12}$$

Figure 2 illustrates the quadratic and cubic designs for $T = 3$.

The design contains the vertices, two points per expert pair with weights $(2/3, 1/3)$ and $(1/3, 2/3)$, and the equal-weight mixture of each expert triple. Its size is

$$|\mathcal{D}_3| = T + 2\binom{T}{2} + \binom{T}{3} = \binom{T+2}{3}, \tag{13}$$

giving $8 + 56 + 56 = 120$ evaluations for $T = 8$.

Using the measured ID AVG at each design point, we construct

$$\begin{aligned} \widehat{M}_3(\lambda) &= \sum_{\alpha \in \mathcal{A}_3} \varphi_{\alpha}^{(3)}(\lambda)\, M(\alpha/3), \\ \varphi_{\alpha}^{(3)}(\lambda) &= \prod_{k=1}^{T} \prod_{\ell=0}^{\alpha_k - 1} \frac{3\lambda_k - \ell}{\alpha_k - \ell}, \end{aligned} \tag{14}$$

where an empty product equals one.

At a design point $\gamma/3$, the basis function equals one if $\gamma = \alpha$. Otherwise, some $\gamma_k < \alpha_k$, making a factor in the product zero. Thus, $\widehat{M}_3$ reproduces all measured values. The number of cardinal basis functions equals the dimension $\binom{T+2}{3}$ of polynomials of degree at most three on the simplex, so the interpolant is unique in this space.

We numerically maximize $\widehat{M}_3$ over Δ^{T-1} using the same SLSQP settings and initializations as quadratic SLIM (Appendix C.3). This optimization requires no further model evaluations, leaving the budget at $|\mathcal{D}_3| = 120$ for $T = 8$.

Role	Hugging Face identifier
Gemma-2-9B	
Base	`google/gemma-2-9b`
Instruction A	`MergeBench/gemma-2-9b_instruction`
Instruction B	`google/gemma-2-9b-it`
Math A	`MergeBench/gemma-2-9b_math`
Math B	`POME-OPTIM/Gemma2-9B-MetaMathQA`
Code A	`MergeBench/gemma-2-9b_coding`
Code B	`testmoto/gemma-2-9b-synthetic_coding`
Multilingual	`MergeBench/gemma-2-9b_multilingual`
Safety	`MergeBench/gemma-2-9b_safety`
Llama-3.1-8B	
Base	`meta-llama/Llama-3.1-8B`
Instruction A	`MergeBench/Llama-3.1-8B_instruction`
Instruction B	`meta-llama/Llama-3.1-8B-Instruct`
Math A	`MergeBench/Llama-3.1-8B_math`
Math B	`pmahdavi/Llama-3.1-8B-math-reasoning`
Code A	`MergeBench/Llama-3.1-8B_coding`
Code B	`pmahdavi/Llama-3.1-8B-coding`
Multilingual	`MergeBench/Llama-3.1-8B_multilingual`
Safety	`MergeBench/Llama-3.1-8B_safety`
External evaluator	
Refusal classifier	`allenai/wildguard`

Table 5: Shared bases and eight experts per architecture, with identifiers linked to their releases. A and B distinguish experts in the same domain. WildGuard is the external evaluator and is not merged.

C Additional Details on Experiments

C.1 Models and Checkpoints

Table 5 lists the checkpoints used for Gemma-2-9B (Gemma Team et al., 2024) and Llama-3.1-8B (Grattafiori et al., 2024). Each expert set contains five domain-specific checkpoints from MergeBench (He et al., 2025) and three additional checkpoints: Instruction B, Math B, and Code B. Instruction B is the official instruction-tuned model.

C.2 Benchmarks and Evaluation Protocol

Table 6 provides benchmark sources and evaluation settings. ID benchmarks are used for coefficient selection and the reported ID results; OOD benchmarks are held out from selection.

We use `openlanguagedata/flores_plus` for translation from English into German, French, Spanish, Chinese, Russian, Arabic, Hindi, and Japanese. ID evaluation reuses a fixed subset of 100 dev sentences per target language (seed 42); OOD evaluation uses all 1,012 devtest sentences per language. The multilingual score is the mean chrF2 across these eight languages. WildGuardTest has 749 evaluated examples after five entries with missing values are excluded. Accuracy and pass@1 scores are multiplied by 100; chrF2, refusal rates, and the AlpacaEval win rate are expressed on the same 0–100 scale.

Merged candidates inherit the base model's configuration and tokenizer, with benchmark-specific prompt formatting applied consistently. The lm-eval tasks and FLORES+ use raw prompts; AlpacaEval uses the architecture's Instruct chat template, and safety evaluation uses a shared chat format. Simplex vertices follow this pipeline: their weights equal those of the corresponding experts, but their scores can differ from evaluations using each expert's native tokenizer, prompts, and generation settings.

Benchmark	Role	Examples	Shots	Scoring
IFEval (Zhou et al., 2023)	ID	541	0	Prompt-level strict accuracy
AlpacaEval 2.0 (Dubois et al., 2024)	OOD	805	0	Length-controlled win rate
GSM8K-CoT (Cobbe et al., 2021)	ID	1,319	8	Strict-match exact match
MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024)	OOD	500	4	Verified accuracy (`math_verify`)
MBPP (Austin et al., 2021)	ID	500	3	pass@1
HumanEval (Chen et al., 2021)	OOD	164	0	pass@1
FLORES+ dev (NLLB Team, 2024; Burchell et al., 2024)	ID	100×8	0	chrF2
FLORES+ devtest	OOD	$1{,}012 \times 8$	0	chrF2
HarmBench (Mazeika et al., 2024)	ID	320	0	Refuse-to-answer rate
WildGuardTest (Han et al., 2024)	OOD	749	0	Refuse-to-answer rate

Table 6: Sample counts, prompting, and scoring for ID selection and held-out OOD evaluation. Shots denotes in-context examples. FLORES+ counts are sentences per target language times eight languages. Refusal rates use WildGuard.

C.3 Coefficient-Selection Implementations

MetaGPT. MetaGPT (Zhou et al., 2024) computes $\lambda_k = \|\tau_k\|_2^2 / \sum_j \|\tau_j\|_2^2$ from the full task vectors. This rule gives nonnegative coefficients summing to one and requires no target-metric evaluations.

MAP. For each seed, we construct one sequence of 135 coefficient vectors. With zero-based indexing, indices 0–7 are $e_1, \ldots, e_8$; indices 8–134 are 127 independent draws from $\mathcal{U}([0,1]^8)$, with all coordinates sampled independently. We initialize `np.random.default_rng(seed)` and generate these coordinates with `uniform(0, 1)`. The fit at budget $n \in \{45, 90, 135\}$ uses the first n points, so all three budgets contain the eight vertices and form nested samples.

For $T = 8$, each surrogate's quadratic component has $(T+1)(T+2)/2 = 45$ coefficients. The budgets of 45, 90, and 135 evaluations are one, two, and three times this count; 90 is the released implementation's default. We adapt the released implementation in three ways. First, we divide scores by 100 to match the sigmoid output range, corresponding to a bounded-response model on the original 0–100 score scale. Second, we replace all-ones surrogate parameter initialization with zeros to avoid initial sigmoid saturation. Third, we replace the released code's search bounds $[-3, 3]^8$ with $[0, 1]^8$ so that optimization remains within the domain from which the surrogate-fitting points are sampled. All reported MAP results use these adjustments to the released implementation. NSGA-III in DEAP uses a population of 100 for 150 generations over $x \in [0,1]^8$. We evaluate the surrogates at x, select the vector with the highest predicted AVG from the approximate Pareto front, and use that same vector directly as the merging coefficients, $\lambda = x$. The sigmoid is applied only to the surrogate outputs, not to the merging coefficients. This selection requires no additional target-metric evaluations.

Evolutionary merging. For $T = 8$, CMA-ES uses a population size of $4 + \lfloor 3 \ln T \rfloor = 10$, an initial mean of 0.5 in every coordinate, an initial step size of $1/6$, and bounds $[0,1]^8$. For each seed, we use a single run of 20 generations and report ID scores after every generation. The OOD evaluation checkpoints of 10, 30, 40, 50, 100, 150, and 200 cumulative evaluations correspond to generations 1, 3, 4, 5, 10, 15, and 20, respectively. At each checkpoint, we report the ID and OOD scores of the candidate with the highest measured ID AVG among all candidates evaluated so far.

SLIM. We interpolate the measured ID AVG directly; by linearity, this is equivalent to averaging the domain-wise interpolants (Appendix B.2). For both quadratic and cubic variants, SLSQP uses analytic gradients, $\sum_k \lambda_k = 1$, bounds $0 \le \lambda_k \le 1$, `ftol`$= 10^{-12}$, and at most 1,000 iterations per initialization. The 20 initializations comprise the barycenter, eight points near the vertices, and 11 random simplex points generated with seed 42. We retain the converged solution with the highest surrogate value. Given the measurements and these fixed initializations, the procedure is

		Gemma-2-9B			Llama-3.1-8B		
Method	#Evals	GPU-h	ID AVG	OOD AVG	GPU-h	ID AVG	OOD AVG
Weight Averaging	0	0	50.1	40.0	0	43.3	33.2
MetaGPT	0	0	62.4	**50.6**	0	47.1	35.5
Task Arithmetic	21	3.0	52.0	39.5	2.7	50.2	37.6
Evol. Merge (gen=1)	10	1.4	$51.5_{\pm 0.8}$	$32.7_{\pm 1.1}$	1.3	$54.0_{\pm 1.0}$	$38.3_{\pm 0.4}$
Evol. Merge (gen=3)	30	4.1	$60.1_{\pm 0.8}$	$38.6_{\pm 1.3}$	3.9	$58.4_{\pm 1.7}$	$42.1_{\pm 2.0}$
Evol. Merge (gen=4)	40	5.4	$60.7_{\pm 0.5}$	$40.7_{\pm 3.1}$	5.0	$58.9_{\pm 1.9}$	$41.1_{\pm 1.1}$
Evol. Merge (gen=5)	50	6.7	$63.5_{\pm 1.8}$	$45.5_{\pm 4.4}$	6.2	$60.0_{\pm 1.9}$	$43.3_{\pm 1.3}$
Evol. Merge (gen=10)	100	13.8	$65.3_{\pm 2.1}$	$46.9_{\pm 4.4}$	11.9	$63.2_{\pm 0.1}$	$45.8_{\pm 2.2}$
Evol. Merge (gen=15)	150	20.4	$66.4_{\pm 2.4}$	$48.7_{\pm 0.6}$	17.6	$63.9_{\pm 0.6}$	$47.2_{\pm 1.5}$
Evol. Merge (gen=20)	200	26.8	$\mathbf{67.4}_{\pm 1.2}$	$49.0_{\pm 0.9}$	23.2	$\mathbf{64.6}_{\pm 0.6}$	$45.5_{\pm 1.9}$
MAP ($n = 45$)	45	6.3	$56.7_{\pm 4.2}$	$41.2_{\pm 5.6}$	5.5	$47.4_{\pm 1.8}$	$36.7_{\pm 2.1}$
MAP ($n = 90$)	90	12.4	$58.3_{\pm 2.6}$	$47.5_{\pm 2.7}$	11.1	$57.6_{\pm 5.9}$	$43.1_{\pm 7.0}$
MAP ($n = 135$)	135	18.5	$61.2_{\pm 1.5}$	$50.6_{\pm 0.4}$	16.7	$60.4_{\pm 1.2}$	$45.3_{\pm 2.6}$
SLIM (quadratic)	36	5.2	66.2	50.3	3.9	56.6	**47.3**
SLIM (cubic)	120	16.4	66.4	48.4	12.9	57.0	46.3

Table 7: ID and held-out OOD performance across coefficient-selection budgets ($T = 8$; Figure 3). AVG is the unweighted mean of five domain scores on a 0–100 scale. MAP and evolutionary merging report mean $\pm$ standard deviation over three seeds; their GPU-hours are means. Costs cover coefficient selection per run, excluding final reporting and fidelity measurements. Zero cost denotes no target-metric evaluations. Bold marks the highest AVG in each column before rounding. On Gemma OOD, MetaGPT (50.62) and MAP with $n = 135$ (50.60) both round to 50.6. The cubic variant is a degree ablation.

deterministic and uses no additional target-metric evaluations. The cubic extension is defined in Appendix B.4.

C.4 Random-Design Fits

The evaluation-placement ablation uses the same Scheffé basis for LS and ridge: eight linear terms and 28 pairwise terms, with no explicit intercept. For ridge, each feature is divided by its root mean square without centering. We select $\alpha \in \{10^{-4}, 10^{-3}, \ldots, 10^{2}\}$ by five-fold cross-validation on the n fitting points, minimizing mean squared error and recomputing feature scales within each training fold. The remaining $150 - n$ points are used only to assess prediction fidelity.

C.5 Numerical Results for Figure 3

Table 7 gives the numerical results underlying Figure 3. All coefficients are selected using ID evaluations only.

C.6 Domain-wise Surrogate Fidelity

Table 8 reports fidelity on the same 150 held-out mixtures as Figure 1. The interquartile range (IQR) of measured scores provides context for the MAE. The AVG row evaluates predictions of the aggregate objective; it is not an average of the domain-wise fidelity statistics.

C.7 Quadratic Interpolation along Simplex Edges

For each expert pair (i, j), we consider $\lambda(t) = (1 - t)e_i + te_j$ for $t \in [0, 1]$. The quadratic interpolant uses measurements at $t = 0, 1/2, 1$. We assess its predictions at $t = 1/3, 2/3$, giving 56 held-out points across 28 edges per architecture. These points are excluded from quadratic surrogate construction and coefficient selection, although they belong to the cubic design. All scores are measured on the ID benchmarks.

	Gemma-2-9B			Llama-3.1-8B		
Domain	ρ	MAE	IQR	ρ	MAE	IQR
Instruction	0.97	2.60	15.9	0.96	2.46	10.9
Math	-0.24	4.88	5.0	0.85	2.33	6.5
Coding	0.86	1.89	4.2	0.70	1.93	3.9
Multilingual	0.63	1.40	1.1	0.56	1.22	0.6
Safety	0.94	9.20	47.2	0.95	4.39	19.9
AVG	0.936	1.93	10.1	0.959	1.07	4.6

Table 8: AVG is accurately predicted despite variation in domain-wise fidelity. On 150 held-out mixtures, Spearman's ρ compares predicted and measured rankings; MAE and the IQR of measured scores are in score points. All interpolants use the same 36 construction evaluations.

MAE is 1.74 on Gemma and 1.38 on Llama, with maximum absolute errors of 8.08 and 4.28, respectively. The curves show useful interpolation overall, with localized deviations. This analysis complements the interior-mixture evaluation in Section 4.2.

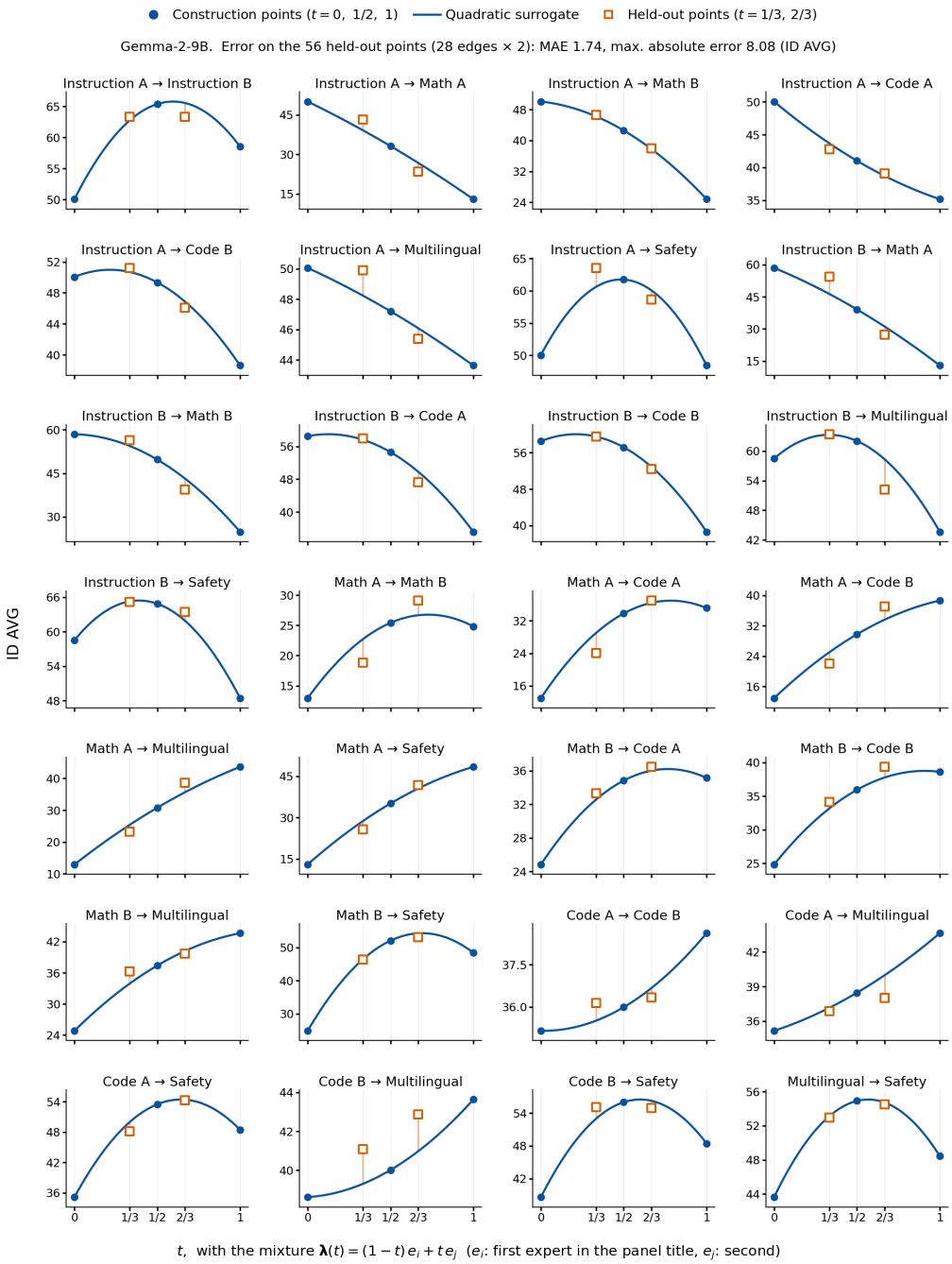

Figure 5: Quadratic predictions and measured ID AVG along all 28 simplex edges for Gemma-2-9B. Blue circles mark construction points and blue curves show the quadratic surrogate. Orange squares mark points held out from quadratic construction; orange segments show prediction errors. Panel titles list the experts at $t = 0$ and $t = 1$. Y-axis ranges vary across panels.

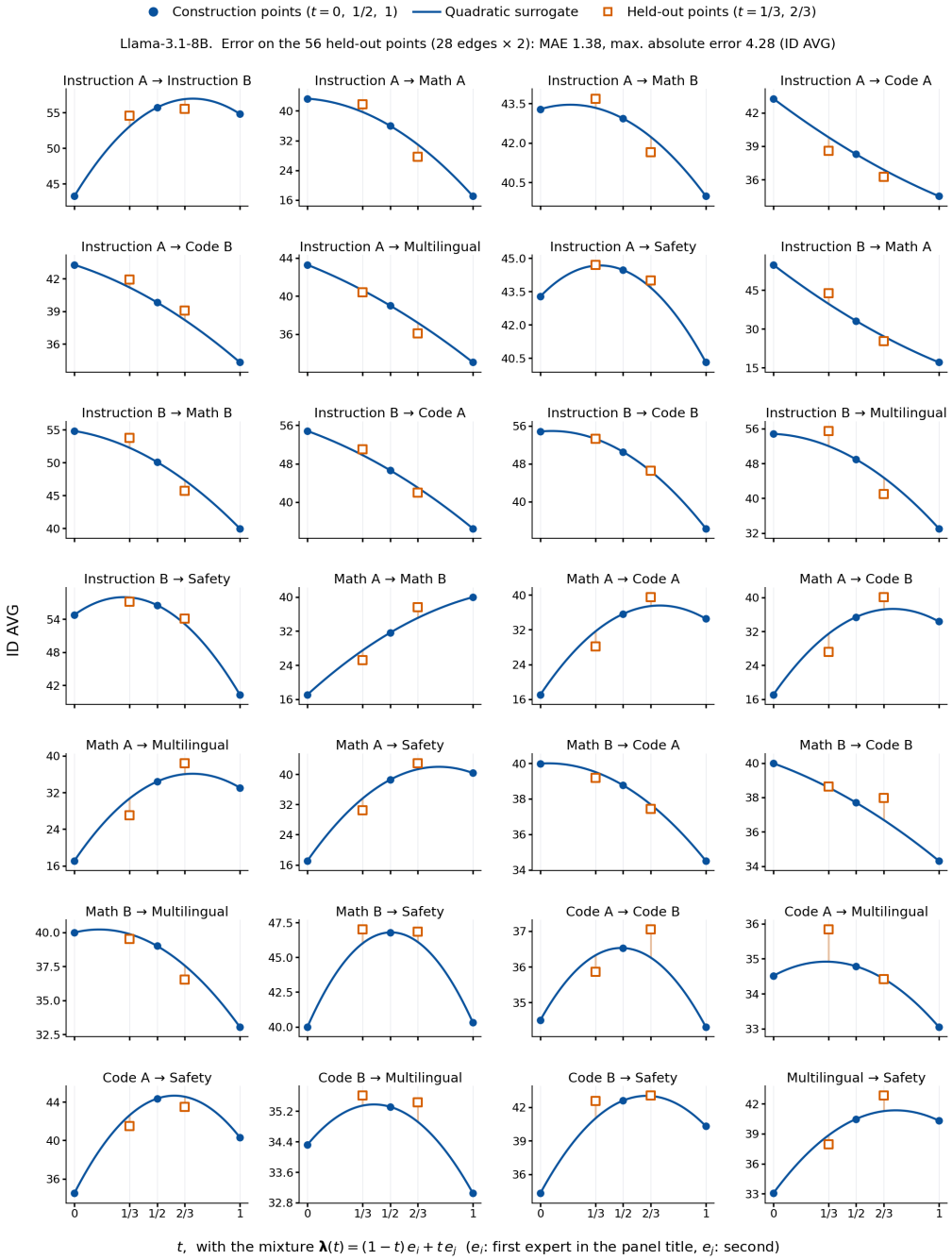

Figure 6: Quadratic predictions and measured ID AVG along all 28 simplex edges for Llama-3.1-8B. Blue circles mark construction points and blue curves show the quadratic surrogate. Orange squares mark points held out from quadratic construction; orange segments show prediction errors. Panel titles list the experts at $t = 0$ and $t = 1$. Y-axis ranges vary across panels.