PRINCIPIA KAN: DECOUPLING SYMBOLIC ROUTING FROM COEFFICIENT ESTIMATION FOR STABLE EQUATION RECOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Symbolic regression seeks both low prediction error and exact recovery of the generating structure. In differentiable recovery from a constrained dictionary, these objectives can diverge when candidate selection shares an amplitude parameter with coefficient allocation. The selector must then determine which basis is active while also distributing coefficient mass, a failure we formalize as the *Shared-Amplitude-Driven Coupling of Symbolic Routing and Coefficient*. We propose Principia KAN (PKAN), which retains the KAN principle of learnable univariate edge functions and constrains each edge in the recovery layer to a physics-informed dictionary. Its physical expectation layer uses a decoupled selector and gives every candidate a signed coefficient; their product is the effective contribution in physical units. The complete system combines temperature annealing in two stages and sparsity on effective contributions with constrained support projection and coefficient refitting. Across 100 tasks and six levels of observation noise, it reaches a 96.83% exact formula recovery rate (EFRR), defined here as exact recovery of the fixed dictionary term set. Under the same training and decoding protocol, strict direct readout before projection and refitting reaches 74.00% for PKAN and 59.17% for the shared amplitude control. The higher direct-readout recovery rate indicates that the decoupled parameterization makes the correct dictionary terms more identifiable inside the trained layer, a difference that the projection and refit decoder can conceal. We evaluate additive equations over three variables, using 36 unary bases per variable and benchmark tasks compatible with this candidate dictionary.

1 INTRODUCTION

Symbolic regression has two distinct objectives: fitting observed data and recovering the exact analytic structure that generated them (Schmidt & Lipson, 2009; Udrescu & Tegmark, 2020). Modern evaluations therefore consider predictive accuracy together with expression structure and complexity (La Cava et al., 2021). The distinction is scientific rather than cosmetic. Correlated expressions can attain similar errors on a finite sample while assigning the response to different mechanisms, so low prediction error alone is not evidence of correct structural recovery.

Expression search treats structure as a discrete program (Koza, 1992; Petersen et al., 2021), while sparse library regression estimates coefficients over a fixed dictionary (Brunton et al., 2016). Neural symbolic architectures and KAN models instead place analytic functions inside trainable networks (Martius & Lampert, 2017; Sahoo et al., 2018; Liu et al., 2025). When a differentiable selector weights a candidate bank by probabilities, its parameterization determines whether function choice can be separated from coefficient estimation. Consider three candidates for one variable, $\sin(x)$, $\cos(x)$, and $\exp(x)$, combined as $a[p_{\sin}\sin(x) + p_{\cos}\cos(x) + p_{\exp}\exp(x)]$. The shared signed amplitude a fixes the total coefficient mass, so the probabilities must encode both which function is preferred and how its coefficient should be scaled. Differences in candidate scale, dictionary correlation, and observation noise can then favor a numerically convenient but structurally incorrect term. We call this failure the *Shared-Amplitude-Driven Coupling of Symbolic Routing and Coefficient*.

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.

This observation leads to the question studied here: how can differentiable training make the correct dictionary terms identifiable inside the learned representation? We propose PKAN, which parameterizes each learnable KAN edge function as an expansion over a physics-informed dictionary. Within that symbolic edge, every named candidate receives a routing logit and a signed coefficient. Their product $p_{ik}w_{ik}$ is converted from normalized training coordinates to an effective contribution in physical units. Routing records competition among named hypotheses; the effective contribution in physical units records direction and magnitude. Annealing in two stages first explores the candidate set and then sharpens the routes with entropy regularization and sparsity on these effective contributions. Direct readout obtains the support represented by the trained layer, and the projection and refit decoder produces the final explicit equation.

Our contributions center on separating symbolic routing from coefficient estimation in dictionary-constrained symbolic equation recovery.

1. We identify and formalize the *Shared-Amplitude-Driven Coupling of Symbolic Routing and Coefficient*. Under a shared-amplitude parameterization, candidate contributions associated with the same input variable must share a sign and have a fixed total absolute mass. PKAN removes these restrictions through a dictionary-constrained symbolic KAN edge that assigns candidate-specific routing logits and signed coefficient parameters. Routing probabilities represent competition among candidate structures, while their products with the signed coefficients define effective contributions that are mapped back to physical units.
2. We train this parameterization using two-stage temperature annealing, entropy regularization, and sparsity on effective contributions. The first stage maintains broad competition among candidate functions. The second sharpens the routing distribution and suppresses negligible contributions, producing a sparse internal representation of candidate support.
3. We distinguish two levels of equation recovery. Direct readout extracts a support set from the effective contributions within the trained layer. The projection and refit decoder selects the final support under structural constraints, then re-estimates the intercept and coefficients in physical units.
4. We evaluate the resulting system on 100 tasks at six levels of observation noise. The benchmark comparison measures final formula recovery, while a strict comparison under matched training and decoding conditions isolates the parameterization change and shows higher direct-readout exact support recovery across the tested noise range.

2 Related work

Classical symbolic regression uses evolutionary search over symbolic expressions (Koza, 1992; Cranmer, 2023). AI Feynman narrows this search through physics-motivated decomposition (Udrescu & Tegmark, 2020). DSR learns a distribution over expressions with risk-seeking policy gradients (Petersen et al., 2021). SPL, TPSR, and MCTS-4-SR use Monte Carlo tree search, with MCTS-4-SR adding an extreme-bandit allocation rule and evolution-inspired state jumps (Sun et al., 2023; Shojaee et al., 2023; Huang et al., 2025). NeSymReS pretrains a transformer to generate equations, whereas SNIP jointly pretrains numeric and symbolic representations (Biggio et al., 2021; Meidani et al., 2024).

Other systems organize symbolic search through parametric families, retrieval, description length, or an equation generative space (Scholl et al., 2025; Zhang et al., 2025; Yu et al., 2025; Li et al., 2026). SyMANTIC combines mutual information screening, feature expansion, and sparse regression to construct parsimonious equations (Muthyala et al., 2025). These methods address broad symbolic spaces. PKAN studies the complementary regime in which scientific knowledge supplies an admissible candidate dictionary and the unresolved problem is stable recovery of its sparse support and coefficients.

Sparse dictionary methods assume that the target has a small support in a supplied library. SINDy estimates the sparse coefficient vector directly (Brunton et al., 2016). Equation Learner networks place fixed analytic operations in a differentiable sparse architecture (Martius & Lampert, 2017; Sahoo et al., 2018). Evaluation must distinguish predictive accuracy from symbolic solution recovery (La Cava et al., 2021). Syntactic agreement, numerical behavior, and algebraic equivalence can also diverge (Li et al., 2026); exact support recovery in this paper is a separate, explicitly defined criterion.

KANs replace scalar edge weights with learnable univariate edge functions (Liu et al., 2025). SR-KAN uses KAN-guided decomposition for symbolic regression, while other recent variants explore in-context operator replacement (Bühler & Guillén-Gosálbez, 2026; Sovrano et al., 2026). Symbolic-KAN constructs an interpretable hierarchical symbolic network by selecting analytic primitives on learned scalar projections and progressively hardening primitive, edge, and unit gates (Faroughi et al., 2026). PKAN retains the KAN edge-function view but constructs each edge from named candidates in a physics-informed dictionary. Every candidate is therefore a testable support hypothesis whose effective contribution can be mapped to physical units. The resulting path from candidate competition to effective contribution, direct readout, constrained projection, and coefficient refitting makes the model's internal structure recognition directly inspectable during exact equation recovery.

3 Background and problem setup

Let $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^{N}$, where $\mathbf{x}^{(n)} \in \mathbb{R}^d$ and $y^{(n)} \in \mathbb{R}$. Symbolic regression seeks an analytic predictor with low error and a compact set of active terms. In the dictionary-constrained setting, a predefined, physics-informed library $\Psi = \{\psi_j\}_{j=1}^{M}$ gives the additive model

$$\hat{y}(\mathbf{x}) = b + \sum_{j=1}^{M} \gamma_j \psi_j(\mathbf{x}), \qquad \|\boldsymbol{\gamma}\|_0 \ll M. \tag{1}$$

The support $S^\star = \{j : \gamma_j \neq 0\}$ identifies the active scientific hypotheses, while b and γ_j determine the recovered expression in physical units. Prediction error and support recovery are evaluated separately because a low-error fit need not have the correct term set.

The benchmark uses an additive unary dictionary with $d = 3$ variables and $K = 36$ bases per variable. Thus $M = dK = 108$, $j = (i, k)$, and $\psi_{ik}(\mathbf{x}) = \phi_k(x_i)$. The admissible support family is

$$\mathcal{S}_{\text{eval}} = \{S \subseteq [d] \times [K] : |S \cap (\{i\} \times [K])| \leq 1,\ |S| \leq d\}\,. \tag{2}$$

It allows at most one unary term for each variable and is used by both direct readout and support projection. The full list of bases and data domains is given in Appendix **??**.

For R evaluation conditions, let $\widehat{S}_r$ denote the recovered support and let $S_r^\star$ denote the ground truth support specified by the benchmark manifest. We define

$$\text{EFRR} = \frac{100}{R} \sum_{r=1}^{R} \mathbf{1}\Big\{\widehat{S}_r = S_r^\star\Big\}\,\%.$$

EFRR is exact term set recovery within the fixed dictionary: the predicted support must equal the ground truth support. It does not compare serialized formula strings, and coefficient error is outside this metric. Algebraically equivalent expressions built from a different term set do not receive credit. The refitted coefficients and intercept are retained as part of the final recovered expression.

We distinguish three evaluation objects. Prediction accuracy is measured by normalized mean squared error. Strict direct readout obtains $\widehat{S}_{\text{dir}}$ from the inverse-normalized effective contributions before candidate screening, support projection, or coefficient refitting, and therefore probes the structure represented inside the trained layer. The projection and refit decoder returns $\widehat{S}_{\text{proj}}$ and refits b and $\boldsymbol{\gamma}$ on the selected support. The strict ablation compares $\widehat{S}_{\text{dir}}$, whereas the complete-system result evaluates the projected and refitted expression.

4 Shared-Amplitude-Driven Coupling of Symbolic Routing and Coefficient

For variable i, consider routing logits s_{ik} and probabilities

$$p_{ik}(\tau) = \frac{\exp(s_{ik}/\tau)}{\sum_{\ell=1}^{K} \exp(s_{i\ell}/\tau)}, \qquad p_{ik} \geq 0, \quad \sum_k p_{ik} = 1. \tag{3}$$

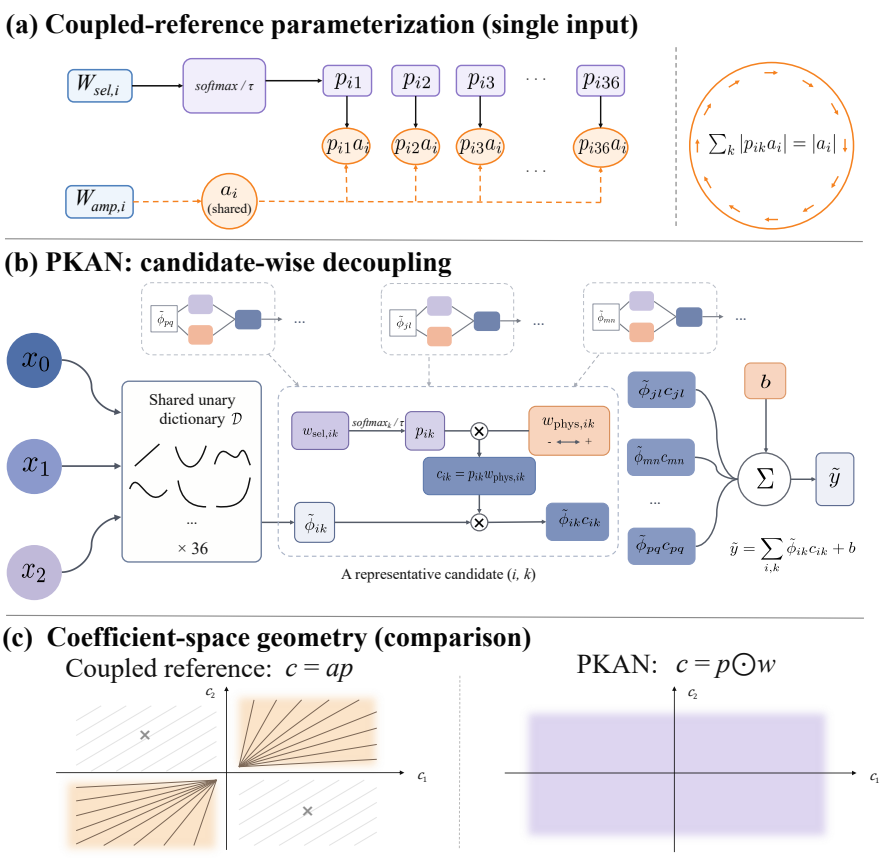


Figure 1: Shared amplitude coupling and PKAN coefficients. (a) Shared amplitude enforces a common sign and fixed absolute mass. (b) PKAN uses candidate-specific logits and signed coefficients, $c_{ik} = p_{ik} w_{\mathrm{phys},ik}$. (c) The same-sign constraint and its removal in two dimensions.

In the normalized coordinates used by the implementation, with standardized bases defined in Eq. (8), a shared-amplitude selector uses one signed value a_i for all bases of that variable:

$$\widetilde{f}_i(x_i) = a_i \sum_{k=1}^{K} p_{ik} \widetilde{\phi}_{ik}(x_i) = \sum_{k=1}^{K} c_{ik}^{\mathrm{cpl}} \widetilde{\phi}_{ik}(x_i), \qquad c_{ik}^{\mathrm{cpl}} = a_i p_{ik}. \tag{4}$$

Proposition 1 (Shared-amplitude geometry) *For fixed i, every nonzero entry of $\mathbf{c}_i^{\mathrm{cpl}} = a_i \mathbf{p}_i$ has the sign of a_i, and*

$$\|\mathbf{c}_i^{\mathrm{cpl}}\|_1 = |a_i|, \qquad \frac{c_{ik}^{\mathrm{cpl}}}{c_{i\ell}^{\mathrm{cpl}}} = \frac{p_{ik}}{p_{i\ell}} \quad \textit{when } p_{i\ell} > 0. \tag{5}$$

The result follows from the simplex constraints. Its implication is about optimization. During soft selection, $\mathbf{p}_i$ cannot encode only symbolic preference. It must allocate the shared coefficient mass and cannot give competing candidates opposite signs. A gradient that compensates for numerical scale therefore changes the apparent symbolic route. Once the route is exactly one-hot, either parameterization can express an arbitrary signed coefficient for its selected basis. The distinction lies in the path to that discrete choice, not in the final hypothesis class.

Decoupling does not resolve every identifiability problem. Collinear bases, algebraically equivalent functions, a missing true basis, or insufficient observations can still make the support ambiguous. Figure 1 identifies one removable constraint rather than claiming unique recovery in general.

5 PKAN

5.1 From KAN edges to dictionary-constrained symbolic edges

A KAN layer represents every connection by a learnable univariate function and sums the incoming edge functions (Liu et al., 2025):

$$h_j^{(\ell+1)} = \sum_i \Phi_{ji}^{(\ell)}\left(h_i^{(\ell)}\right). \tag{6}$$

PKAN preserves this edge-function construction but restricts an edge in the recovery layer to named candidates from the scientific dictionary,

$$\Phi_{ji}(x_i) = \sum_{k=1}^{K} c_{jik}\widetilde{\phi}_{ik}(x_i), \qquad c_{jik} = p_{jik}w_{jik}. \tag{7}$$

Here p_{jik} represents competition among candidate functions and w_{jik} is the corresponding signed coefficient in normalized coordinates. The evaluated model has one scalar output, so we suppress j below. Summing its symbolic edge functions over the inputs yields the additive equation in Eq. (1). Thus a KAN edge becomes a dictionary-constrained symbolic edge, its route and coefficient are learned separately, and their effective contribution is mapped back to physical units for direct readout and final decoding. This changes the continuous parameterization while leaving the discrete hypothesis class unchanged.

5.2 Normalization and initialization

The model evaluates all K bases on every coordinate. For training-set statistics μ_{ik}, σ_{ik}, μ_y, and σ_y, it forms

$$\widetilde{\phi}_{ik}(x_i) = \frac{\phi_k(x_i) - \mu_{ik}}{\sigma_{ik}}, \qquad \widetilde{y} = \frac{y - \mu_y}{\sigma_y}, \tag{8}$$

with numerical floors for degenerate standard deviations. Test observations never enter these statistics. The normalized correlation

$$\rho_{ik} = \frac{1}{N}\sum_{n=1}^{N} \widetilde{\phi}_{ik}(x_i^{(n)})\widetilde{y}^{(n)} \tag{9}$$

initializes the routing score as $s_{ik} = 4|\rho_{ik}|$ and the signed coefficient as $w_{ik} = \rho_{ik}$.

5.3 Dictionary-aligned physical-expectation layer

Each candidate has an independent logit s_{ik} and signed coefficient w_{ik}. During training, independent Gumbel noise g_{ik} gives

$$p_{ik}(\tau) = \mathrm{softmax}_k(s_{ik}/\tau + g_{ik}), \qquad c_{ik} = p_{ik}(\tau)w_{ik}. \tag{10}$$

The normalized prediction is

$$\widehat{\widetilde{y}} = b_{\mathrm{norm}} + \sum_{i=1}^{d}\sum_{k=1}^{K} c_{ik}\widetilde{\phi}_{ik}(x_i). \tag{11}$$

We omit Gumbel noise at evaluation. During training, Gumbel–Softmax provides a differentiable relaxation of categorical sampling (Jang et al., 2017). Unlike Eq. (4), candidate coefficients may adapt in sign and magnitude without forcing the route to perform the same numerical correction. In PKAN, p_{ik} represents competition among fixed dictionary hypotheses, while w_{ik} carries the signed candidate magnitude before conversion to physical scale. Their product determines the effective contribution used for direct readout. The prediction constrains $p_{ik}w_{ik}$ directly, while the training objective and decoding procedure assign the two factors their operational roles.

The effective contribution in physical units and the physical intercept are

$$\gamma_{ik} = \frac{\sigma_y}{\sigma_{ik}}p_{ik}w_{ik}, \qquad b_{\mathrm{phys}} = \mu_y + \sigma_y b_{\mathrm{norm}} - \sum_{i,k}\gamma_{ik}\mu_{ik}. \tag{12}$$

Reporting w_{ik} alone would confuse the learned normalized coefficient with the scale of a dictionary column.

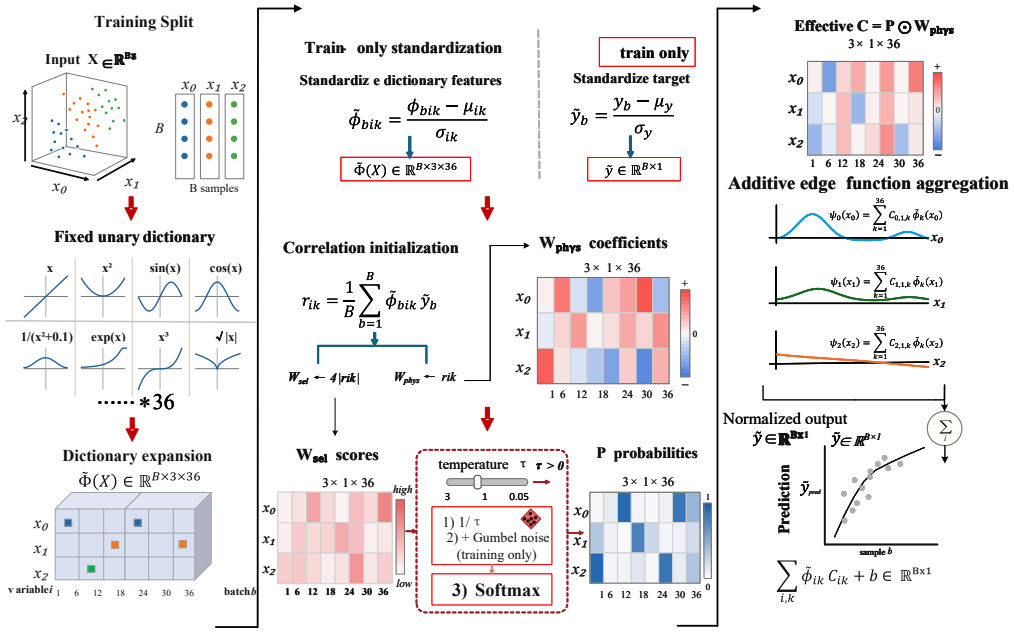


Figure 2: PKAN computation. Dictionary correlations initialize W_{sel} and W_{phys}; Gumbel routing forms P and $C = P \odot W_{\text{phys}}$, which weight 36 unary bases on each of three input edges before summation with b.

5.4 Exploration and Crystallization

Training lasts 800 epochs. For the first 300, the temperature decreases linearly from 3 to 1 and the objective is normalized mean squared error:

$$\mathcal{L}_{\text{exp}} = \frac{1}{N} \sum_{n} (\widehat{\tilde{y}}^{(n)} - \tilde{y}^{(n)})^2. \tag{13}$$

During the remaining epochs, the temperature falls from 1 to 0.05 and

$$\mathcal{L}_{\text{crys}} = \mathcal{L}_{\text{exp}} + 0.2 \frac{1}{d} \sum_{i} H(\mathbf{p}_i) + 0.005 \sum_{i,k} |p_{ik} w_{ik}|. \tag{14}$$

The entropy term sharpens routes. The sparsity term acts on effective contributions, not latent coefficients. AdamW uses learning rate 0.03, weight decay 10^{-4}, and gradient norm clipping at 1.

5.5 Decoding and Recovery Trace

Eight restarts are trained, with restart seeds separated by 100003. Direct readout ranks the inverse-normalized effective contributions $|\gamma_{ik}|$, applies the admissible support rule in Eq. (2), and enforces the configured coefficient threshold. The mechanism experiment retains the restart with the lowest direct training NMSE. This readout exposes what the learned layer itself recognizes.

The projection and refit decoder adds a combinatorial support search. For each variable, its pool is the union of the six largest learned effective contributions in physical units and the six strongest univariate least-squares candidates. It enumerates supports in $\mathcal{S}_{\text{eval}}$ and fits an intercept and coefficients on the training split. A support S is scored by

$$J(S) = \text{NMSE}_{\text{train}}(S) + 10^{-4}|S| + 10^{-6} \sum_{(i,k)\in S} (1 + 0.02k), \tag{15}$$

subject to a minimum fitted coefficient magnitude. The active physical threshold is 0.03. The selected support is refitted once more on the training split before test evaluation. The restart with the lowest refitted training NMSE is retained.

These quantities record each stage of recovery. Routing probabilities show candidate competition. Effective contributions in physical units give the signed contribution learned before projection. The candidate pool records which possibilities reach the projection and refit decoder. The projected support and refitted coefficients define the explicit equation. Comparing these stages reveals whether a term was lost in training or repaired downstream. This comparison does not establish causal truth or posterior confidence in the recovered law.

6 EXPERIMENTS

6.1 BENCHMARK AND EXPERIMENTAL SETUP

The benchmark instantiates the target setting with 100 three-input sparse equations. Twenty are internal synthetic equations. Eighty are dictionary-compatible slices or variants associated with AI Feynman, Korns, Nguyen, Keijzer, Vladislavleva, and Pagie families. They are not the original public tasks used unchanged. The target supports contain 25 one-term, 60 two-term, and 15 three-term equations.

Each condition has 3,000 training and 3,000 independently sampled test observations. The 20 internal equations use $[-1.5, 2.5]^3$; the 80 public-benchmark-compatible equations use that domain by default and apply equation-specific all-coordinate or variable-wise ranges where required by the functional form. Gaussian noise with standard deviation $\eta \,\mathrm{std}(y_{\mathrm{clean}})$ is added for $\eta \in \{0, 0.01, \ldots, 0.05\}$, producing 600 dataset and noise conditions. The dictionary contains 36 unary bases per variable, or 108 candidates. PKAN has 108 routing logits, 108 signed coefficients, and one bias, for 217 trainable parameters. Runs used PyTorch 2.11.0 with CUDA 12.8 on an NVIDIA A100-SXM4-40GB.

Exact formula recovery is strict equality between the predicted and manifest support sets. It does not grant credit for algebraically equivalent expressions. NMSE is mean squared error divided by target variance. For the matched decoupling comparison, we report a 95% percentile bootstrap interval from 100,000 resamples of the 100 datasets, retaining all six noise conditions within each sampled cluster. An exact paired sign-flip test uses the per-dataset difference summed over noise levels. The raw result tables do not contain coefficient-error measurements or repeated external-seed trials, so we do not report physical-coefficient accuracy or cross-seed stability as empirical findings.

6.2 COMPLETE-SYSTEM COMPARISON

Figure 3 compares PKAN with eleven baselines across all 600 dataset and noise conditions. The complete PKAN system recovered 581/600 supports (96.83%), compared with 551/600 (91.83%) for GP symbolic search and 486/600 (81.00%) for RL symbolic search. Recovery stayed between 96% and 97% at every noise level and was highest in six of the seven equation families. GP was slightly higher on Keijzer (93.06% versus 91.67%). Panel (c) places recovery beside the median recorded runtime.

All methods use the same data manifest and exact support criterion but retain their native fitting procedures. GP and RL search the dictionary of 108 terms, SINDy fits it, and predictions from the other baselines are mapped to it for scoring. MCTS-4-SR uses the compiled implementation of Huang et al. (2025); Symbolic-KAN is reimplemented locally. Search budgets and postprocessing differ, so Figure 3 reports performance under this protocol rather than a universal ranking.

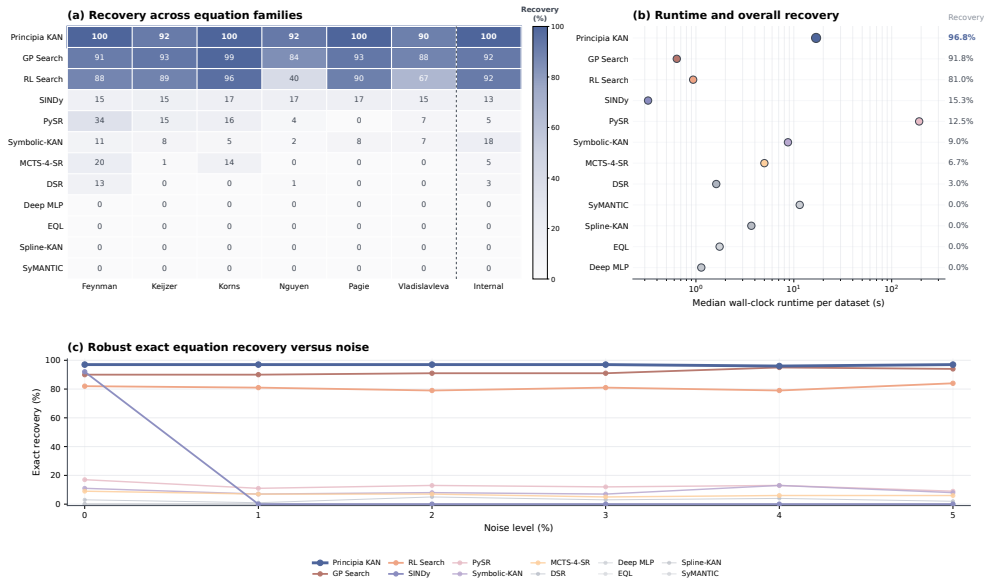


Figure 3: Comparison across 100 equations, six noise levels, and 600 conditions: (a) recovery by noise, (b) recovery by source family, and (c) overall recovery versus median runtime. Symbolic-KAN is our local reimplementation.

6.3 Decoupling and internal structural recognition

The mechanism experiment keeps the data, dictionary, normalization, correlation-derived initialization, optimizer, temperature schedule, regularizers, restart count, and direct readout fixed. The control replaces w_{ik} with one amplitude a_i per variable, initialized from the strongest signed correlation for that variable. Tying the candidate coefficients in this way produces a matched shared-amplitude control that isolates the geometry in Eq. (4) while preserving the remaining training and decoding protocol. Model-specific random streams are independent under the experiment runner. Pairing is therefore defined at the dataset and noise-condition level, with the best of eight restarts selected for each model. Candidate-specific coefficients increase the parameter count, so this estimate applies to the implemented parameterizations without enforcing equal capacity.

Direct readout yields 444/600 exact supports for PKAN and 355/600 for the shared-amplitude control (Table 1). The 14.83 point gain has a dataset-cluster bootstrap interval of $[8.83, 21.17]$ points. The exact cluster-level paired sign-flip test gives $p = 4.63 \times 10^{-6}$. Among the 600 paired conditions, both succeed in 319, only PKAN succeeds in 125, only the control succeeds in 36, and both fail in 120. Median direct readout test NMSE falls from 1.318×10^{-3} to 9.176×10^{-4}.

The direct readout difference is positive at every tested noise level (Table 2). It is 21 points at zero noise, remains between 10 and 14 points from one to four percent noise, and returns to 21 points at five percent. The control alone succeeds in 36 pairs, and the polynomial category is 3.57 points lower for PKAN. Ten of the eleven structure categories improve, with the largest gains in mixed-function targets. The result shows that the internal-recovery advantage persists across the tested observation-noise range. It does not imply monotone degradation, universal dominance, or stability across random seeds.

The two parameterizations differ under direct readout before projection and refitting. When the projection and refit decoder is applied, both parameterizations recover 581/600 supports, and their predicted support rows are identical. The downstream search can repair many internal routing mistakes. The complete system result establishes end-to-end effectiveness under this protocol, while direct readout shows that the difference is already present in the learned effective contributions before projection and refitting.

Table 1: Direct-readout comparison under matched data, training, and decoding. The readout uses learned effective contributions in physical units without projection or coefficient refitting. Values in parentheses are counts out of 600.

Parameterization	Parameters	EFRR (%)	Mean NMSE	Median NMSE
PKAN candidate coefficients	217	**74.00** (444)	1.486×10^{-3}	9.176×10^{-4}
Shared amplitude per variable	112	59.17 (355)	2.857×10^{-3}	1.318×10^{-3}

Table 2: Exact recovery under direct readout by observation-noise level. Each cell is a percentage over 100 datasets.

Noise (%)	PKAN	Shared amplitude	Difference
0	75	54	+21
1	74	64	+10
2	74	62	+12
3	71	57	+14
4	70	59	+11
5	80	59	+21

7 Conclusion

We formalized the *Shared-Amplitude-Driven Coupling of Symbolic Routing and Coefficient* and developed PKAN as a dictionary-constrained symbolic parameterization of KAN edge functions for equation recovery in physical units. The recorded stages connect each named hypothesis to its route, effective contribution, projected support, and refitted coefficient. With the matched direct readout procedure, decoupling improves exact support recovery at every tested noise level. Identical supports after projection and refitting locate this gain within the learned representation rather than the projection and refit decoder.

AI use statement

AI tools were used only for language polishing and assistance with LaTeX formatting. The research content, experimental design, data analysis, and conclusions were developed independently by the authors.

References

Luca Biggio, Tommaso Bendinelli, Alexander Neitz, Aurélien Lucchi, and Giambattista Parascandolo. Neural symbolic regression that scales. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 936–945, 2021. URL https://proceedings.mlr.press/v139/biggio21a.html.

Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016. doi: 10.1073/pnas.1517384113.

Marco Andrea Bühler and Gonzalo Guillén-Gosálbez. SR-KAN: A kolmogorov–arnold network guided symbolic regression framework. *Computers & Chemical Engineering*, 213:109721, 2026. doi: 10.1016/j.compchemeng.2026.109721. URL https://doi.org/10.1016/j.compchemeng.2026.109721.

Miles Cranmer. Interpretable machine learning for science with PySR and SymbolicRegression.jl. *arXiv preprint arXiv:2305.01582*, 2023. URL https://arxiv.org/abs/2305.01582.

Salah A. Faroughi, Farinaz Mostajeran, Amirhossein Arzani, and Shirko Faroughi. Symbolic–KAN: Kolmogorov-arnold networks with discrete symbolic structure for interpretable learning. *Journal of Computational Physics*, 566:115223, 2026. doi: 10.1016/j.jcp.2026.115223. URL https://doi.org/10.1016/j.jcp.2026.115223.

Zhengyao Huang, Daniel Zhengyu Huang, Tiannan Xiao, Dina Ma, Zhenyu Ming, Hao Shi, and Yuanhui Wen. Improving monte carlo tree search for symbolic regression. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/3a28dd36daed2e3d66a3a143ab8b53cc-Abstract-Conference.html.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rkE3y85ee.

John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

William La Cava, Patryk Orzechowski, Bogdan Burlacu, Fabrício Olivetti de França, Marco Virgolin, Ying Jin, Michael Kommenda, and Jason H. Moore. Contemporary symbolic regression methods and their relative performance. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/c0c7c76d30bd3dcaefc96f40275bdc0a-Abstract-round1.html.

Qian Li, Yuxiao Hu, Juncheng Liu, and Yuntian Chen. GenSR: Symbolic regression based on equation generative space. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=8emIjwUQZg.

Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y. Hou, and Max Tegmark. KAN: Kolmogorov-arnold networks. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/afaed89642ea100935e39d39a4da602c-Abstract-Conference.html.

Georg Martius and Christoph H. Lampert. Extrapolation and learning equations. In *International Conference on Learning Representations Workshop Track*, 2017. URL https://openreview.net/forum?id=H1gNOeHKPS.

Kazem Meidani, Parshin Shojaee, Chandan K. Reddy, and Amir Barati Farimani. SNIP: Bridging mathematical symbolic and numeric realms with unified pre-training. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=KZSEgJGPxu.

Madhav R. Muthyala, Farshud Sorourifar, You Peng, and Joel A. Paulson. SyMANTIC: An efficient symbolic regression method for interpretable and parsimonious model discovery in science and beyond. *Industrial & Engineering Chemistry Research*, 64(6):3354–3369, 2025. doi: 10.1021/acs.iecr.4c03503. URL https://doi.org/10.1021/acs.iecr.4c03503.

Brenden K. Petersen, Mikel Landajuela, T. Nathan Mundhenk, Claudio P. Santiago, Soo K. Kim, and Joanne T. Kim. Deep symbolic regression: Recovering mathematical expressions from data via risk-seeking policy gradients. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=m5Qsh0kBQG.

Subham S. Sahoo, Christoph H. Lampert, and Georg Martius. Learning equations for extrapolation and control. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4442–4450, 2018. URL https://proceedings.mlr.press/v80/sahoo18a.html.

Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85, 2009. doi: 10.1126/science.1165893.

Philipp Scholl, Katharina Bieker, Hillary Hauger, and Gitta Kutyniok. ParFam – (neural guided) symbolic regression via continuous global optimization. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8y5Uf6oEiB.

Parshin Shojaee, Kazem Meidani, Amir Barati Farimani, and Chandan K. Reddy. Transformer-based planning for symbolic regression. In *Advances in Neural Information Processing Systems*, volume 36, pp. 45907–45919, 2023. doi: 10.52202/075280-1990. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8ffb4e3118280a66b192b6f06e0e2596-Abstract-Conference.html.

Francesco Sovrano, Lidia Losavio, Giulia Vilone, and Marc Langheinrich. In-context symbolic regression for robustness-improved kolmogorov-arnold networks. *arXiv preprint arXiv:2603.15250*, 2026. URL https://arxiv.org/abs/2603.15250.

Fangzheng Sun, Yang Liu, Jian-Xun Wang, and Hao Sun. Symbolic physics learner: Discovering governing equations via monte carlo tree search. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ZTK3SefE8_Z.

Silviu-Marian Udrescu and Max Tegmark. AI Feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, 2020. doi: 10.1126/sciadv.aay2631.

Zihan Yu, Jingtao Ding, Yong Li, and Depeng Jin. Symbolic regression via MDLformer-guided search: From minimizing prediction error to minimizing description length. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a402493de088886740b5939f666a6e56-Abstract-Conference.html.

Hengzhe Zhang, Qi Chen, Bing Xue, Wolfgang Banzhaf, and Mengjie Zhang. RAG-SR: Retrieval-augmented generation for neural symbolic regression. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/19a8e70828c01059631f913442ae31e6-Abstract-Conference.html.

A Formal properties of the parameterization

For one input coordinate and one output, write the route probabilities as

$$\mathbf{p}(\tau) = \mathrm{softmax}(\mathbf{s}/\tau), \qquad \mathbf{p} \in \Delta^{K-1}.$$

The normalized dictionary columns are denoted by $z_k = (\phi_k - \mu_k)/\sigma_k$. The coupled parameterization uses $c_k = a p_k$, where one signed amplitude a is shared by all K candidates. PKAN uses $c_k = p_k w_k$, with one signed coefficient w_k per candidate.

A.1 Geometry of shared amplitude coupling

Proposition 1. For a fixed input coordinate, every coupled contribution vector $\mathbf{c} = a\mathbf{p}$ has a common sign and satisfies

$$\sum_{k=1}^{K} |c_k| = |a|. \tag{16}$$

Conversely, every nonzero vector whose entries are all nonnegative or all nonpositive lies in the closure of this parameterization.

Proof. Since $p_k \geq 0$ and $\sum_k p_k = 1$, the sign of each nonzero c_k is the sign of a, and

$$\sum_k |c_k| = \sum_k |a| p_k = |a|.$$

For a vector $\mathbf{c}$ with a common sign, let $s \in \{-1, 1\}$ be the sign of its nonzero entries and choose

$$a = s \sum_k |c_k|, \qquad p_k = \frac{|c_k|}{\sum_j |c_j|}.$$

Softmax probabilities are strictly positive at finite logits; zero entries are obtained as limits of logits. This proves the claim.

The result formalizes the Shared-Amplitude-Driven Coupling of Symbolic Routing and Coefficient. Routing probabilities must allocate a fixed absolute contribution mass while also choosing a basis. Candidates under the same variable cannot carry opposite signed contributions in this form.