Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Dictionary-KAN: Resolving the Optimization Paradox of Kolmogorov-Arnold Networks via Complex RKHS, Machine-Verified Theory, and Discrete Hierarchical Refinement

Kiarash Mohammadi

Kiarash7.mohammadi@gmail.com

Independent Resercher https://orcid.org/0009-0001-8760-977X

Research Article

Keywords: KANs, kolmogorov-arnold-networks, complex-valued-neural-networks, rkhs, symbolic-regression

Posted Date: August 25th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10784814/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Dictionary-KAN: Resolving the Optimization Paradox of Kolmogorov-Arnold Networks via Complex RKHS, Machine-Verified Theory, and Discrete Hierarchical Refinement

Kiarash Mohammadi
kiarash7.mohammadi@gmail.com

V1 Draft

Abstract

Kolmogorov-Arnold Networks (KANs) replace dense weight matrices with learnable univariate edge functions, but continuous spline-based formulations suffer from an "optimization paradox": vanishing gradients, knot-sorting collapses, dense-grid memory costs, and the reintroduction of the curse of dimensionality via multi-variable gating. We introduce **Dictionary-KAN** (DKAN), an architecture that grounds every edge function in a fixed magnitude-RBF dictionary with complex coefficients, lifts inputs into the complex plane so that analytic base terms (z, z^2) natively isolate multivariate cross-products, grows capacity through Discrete Hierarchical Refinement (DHR) with *exact zero catastrophic forgetting*, and regularizes weight curvature with a discrete total-variation (TV) penalty. Crucially, the core optimization theory—exact quadratic loss expansion, strict convexity under a full-rank dictionary, stationarity $\Leftrightarrow$ normal equations, unique minimizers, gradient-descent convergence under a Polyak–Łojasiewicz condition, the DHR zero-forgetting identity, and the TV curvature characterization—is **machine-verified in Lean 4 / Mathlib** [13] with explicit assumptions. Empirically, DKAN outperforms dense RBF-KAN baselines by $\sim 20\times$ on coupled multivariate regression, recovers the damped-pendulum ODE and the Burgers PDE coefficients ($< 0.3\%$ error) via complex-manifold feature pruning, avoids the out-of-memory failure of dense KANs when scaling width, and compiles a trained network into a $\sim$50-line closed-form GLSL BRDF shader. We report negative results with equal prominence: smooth dictionaries exhibit Gibbs-like ringing on discontinuities, and weight-space TV does not control function-space overshoot. DKAN is positioned as an *interpretable, refinable, hardware-efficient* alternative to both MLPs and continuous KANs.

1 Introduction: The Optimization Paradox of Continuous KANs

The Kolmogorov-Arnold (KA) representation theorem [1, 7] establishes that any multivariate continuous function $f : [0, 1]^n \to \mathbb{R}$ can be decomposed into a finite composition of continuous 1D functions and addition operations. Standard continuous formulations of Kolmogorov-Arnold Networks (KANs) [8] attempt to parameterize these edge functions using B-splines, Chebyshev polynomials, rational functions, or Gaussian radial basis functions with learnable layer centers [2]. While theoretically expressive, continuous KANs encounter a fundamental *optimization paradox*: the mathematical spaces required for continuous function approximation generate loss landscapes that gradient descent cannot reliably traverse.

Specifically, continuous KAN variants suffer from four primary contradictions:

[0]Project repository: https://github.com/Kiarash-m0hammadi/dictionary-kan

1. **Basis Function Pathologies:** Bounded rational functions introduce severe vanishing gradients. Global wavelets and high-degree orthogonal polynomials create corrugated loss landscapes due to erratic zero-crossing shifts during gradient descent.

2. **Grid Instability & Sorting Collapses:** Dynamic optimization and adaptive re-gridding of spline knot locations [9] can suffer from non-convex sorting collapses or domain-warping pathologies that destroy local basis support, triggering global catastrophic forgetting upon localized updates.

3. **Violations of KA Coupling Constraints:** To model non-linear variable interactions (e.g., $x_1 x_2$), continuous KANs often resort to contextual gating or linear pre-projection matrices, breaking the 1D summation structure and reintroducing dense MLP matrices.

4. **Regularization Intractability:** Standard Sobolev L_2 derivative penalties blur sharp transitions, whereas strict Lipschitz bounds constrain overall network expressivity.

Dictionary-KAN (DKAN) addresses these challenges by abandoning continuous grid warping and dense MLP projections, returning to the core topological constraints of the KA theorem while employing functional analysis and discrete structural refinements. DKAN makes four core design commitments:

1. **RKHS dictionaries.** Each edge is a fixed Gaussian (magnitude-RBF) dictionary in a Reproducing Kernel Hilbert Space (RKHS) [11,15] with learnable *complex* coefficients, making each edge's least-squares subproblem an exactly quadratic, strictly convex program [17].

2. **Complex routing.** Inputs are lifted into the complex plane; analytic base terms (z, z^2) isolate multiplicative interactions natively without dense matrix projections.

3. **Zero-forgetting refinement.** Discrete Hierarchical Refinement (DHR) inserts midpoint centers with exactly zero weights, ensuring the represented function is bit-identical before and after growth.

4. **Curvature sparsity.** An ℓ_1 penalty on second discrete differences of the weight sequence enforces piecewise-linear weight profiles.

Crucially, while theoretical analysis of KAN convergence under gradient descent has traditionally relied on overparameterized kernel regimes [3], the core optimization theory underlying DKAN is **machine-verified in Lean 4 / Mathlib [13]**; assumptions appear as explicit hypotheses rather than informal claims.

2 Architecture

2.1 Complex-Valued Topologies

A common critique of purely 1D real-valued KANs is their requirement for deep multi-layer spline cascades to approximate coupled non-linear terms without resorting to non-KA linear projections. We resolve this by lifting the network topology into the complex field $\mathbb{C}$ [14]. Rather than pairing real variables into complex conjugates (which halves feature dimension), DKAN embeds each real feature independently: $x_p \mapsto x_p + 0i \in \mathbb{C}$.

Definition 1 (Edge function)**.** For $z \in \mathbb{C}$, an edge computes

$$\phi(z) = w_{b1} z + w_{b2} z^2 + \sum_{m=1}^{M} w_m \exp\big(-\gamma\,|z - c_m|^2\big),$$

with $w_{b1}, w_{b2}, w_m \in \mathbb{C}$, fixed real centers c_m, and width $\gamma > 0$. A node output is the external (Kolmogorov–Arnold) sum $y_o = \sum_i \phi_{oi}(z_i)$.

2.2 Discrete Hierarchical Refinement (DHR)

Unlike continuous KANs that dynamically warp or relocate spline knot locations during optimization [9], DKAN uses Discrete Hierarchical Refinement (DHR) to systematically expand dictionary resolution without disturbing existing representations.

Definition 2 (Discrete Hierarchical Refinement (DHR))**.** Given ordered centers $\{c_m\}_{m=1}^{M}$ and weights w, DHR produces $2M-1$ centers by keeping c_m at even indices and inserting midpoints $(c_m + c_{m+1})/2$ at odd indices; old weights are kept and all new weights are initialized to 0.

2.3 Total-Variation Regularization

Definition 3 (Total-variation regularizer)**.** $\mathcal{R}_{TV}(w) = \lambda_{TV} \sum_m |w_{m+1} - 2w_m + w_{m-1}|$, inspired by total-variation regularization in functional spaces [10, 15], applied to the real and imaginary parts of the dictionary weights to penalize second-order discrete differences.

3 Machine-Verified Theory

All results below are formalized in Lean 4 / Mathlib [13] (supplementary file `math.lean`); we state them with the exact hypotheses used by the prover. Let $L(w) = \sum_i |y_i - (Kw)_i|^2$ denote the fixed-edge loss with design matrix $K_{im} = \exp(-\gamma|z_i - c_m|^2)$, and $Q(v) = \|Kv\|^2$ the Gram quadratic.

Theorem 1 (Exact quadratic expansion)**.** *For all w, v: $L(w+v) = L(w) + DL(w)[v] + Q(v)$, and for $t \in \mathbb{R}$: $L(w+tv) = L(w) + t\,DL(w)[v] + t^2 Q(v)$.*

Theorem 2 (Strict convexity & algebraic curvature)**.** *If K has full column rank, then $Q(v) > 0$ for all $v \neq 0$. Consequently, L is strictly convex in both the midpoint and Jensen (line-segment) senses. Furthermore, we verify positive second-order curvature purely algebraically (without calculus) via the finite-difference identity: $L(w+v) + L(w-v) - 2L(w) = 2Q(v) > 0$.*

Theorem 3 (Stationarity, normal equations, unique minimizer)**.** *The following are equivalent: $\nabla L(w) = 0 \iff K^*(Kw - y) = 0$ (the normal equations) $\iff$ w is a global minimizer. Under full column rank the minimizer is unique.*

Theorem 4 (Gradient descent & ε-N convergence)**.** *In contrast to asymptotic overparameterized convergence bounds [3], we formally verify non-asymptotic gradient descent contraction. Assume an upper spectral bound (smoothness) $Q(v) \leq L\|v\|^2$. One gradient step $w^+ = w - \eta\nabla L(w)$ with $0 < \eta L < 1$ satisfies $L(w^+) \leq L(w) - \eta(1-\eta L)\|\nabla L(w)\|^2$. Furthermore, we formally derive the Polyak–Łojasiewicz (PL) condition $\alpha(L(w) - L(w_\star)) \leq \|\nabla L(w)\|^2$ from lower bounds on the normal operator K^*K. Under this derived PL condition, the loss gap contracts geometrically, and we prove the formal ε-N convergence bound: $\forall \varepsilon > 0, \exists N, \forall n \geq N, L(w_n) - L(w_\star) < \varepsilon$.*

Theorem 5 (DHR zero forgetting at the node level)**.** *Let f_{old}, f_{new} be the edge functions before/after DHR. Then $f_{new} \equiv f_{old}$ pointwise. Crucially, we verify this not just for isolated edges, but for the full Kolmogorov–Arnold node summation $y = \sum_p \phi_p(z_p)$, ensuring the entire network output is bit-identical pre- and post-refinement.*

Theorem 6 (TV curvature characterization)**.** *Let $\mathcal{R}_{TV}(w) = \sum |\Delta^2 w_m|$. We prove the exact equivalence $\mathcal{R}_{TV}(w) = 0 \iff \forall m, \Delta^2 w_m = 0 \iff$ the weight sequence is affine. The penalty vanishes exactly on zero-curvature profiles.*

Remark 1 (Assumption hygiene)**.** Theorems 2–4 are conditional on explicit premises (full column rank of K; upper curvature L; lower bounds on K^*K). These are *not* claimed as unconditional properties of arbitrary RBF dictionaries; verifying them from data is future work.

4 Experiments

We evaluate Dictionary-KAN across six experimental benchmarks designed to test multivariate coupling, discontinuity handling, hardware scaling, physical discovery, and symbolic compilation.

4.1 Multivariate coupling & Optimization Stability

We evaluate approximation accuracy on the non-separable target function $f(x, y) = \sin(x)\cos(y) + xy$ over $(x, y) \in [-2, 2]^2$. The dataset consists of 3,000 training points, 1,000 test points, and a 100×100 evaluation grid. All models are trained for 10,000 epochs using AdamW with a cosine-annealed learning rate schedule. For the Dictionary-KAN variants, Discrete Hierarchical Refinement (DHR) is triggered at epoch 1,000.

We compare four architectures:

- **C-DKAN (Ours):** Complex-valued Dictionary-KAN $[2, 3, 1]$ with 276 parameters.
- **R-DKAN (Ablation):** Real-valued Dictionary-KAN $[2, 4, 1]$ with matched parameter count (276).
- **Dense KAN (Baseline):** Standard real-valued RBF-KAN $[2, 4, 1]$ with 144 parameters.
- **MLP (Baseline):** Standard fully connected network $[2, 40, 1]$ with SiLU activations (161 parameters).

Table 1: Multivariate coupling regression (10k epochs, seed 42).

Model	Params	Test MSE	Grid MSE	Max Abs Error
C-DKAN (Ours)	276	$\mathbf{1.10 \times 10^{-5}}$	1.16×10^{-5}	1.24×10^{-2}
R-DKAN (Real ablation)	276	1.24×10^{-5}	1.18×10^{-5}	1.01×10^{-2}
Dense RBF-KAN	144	2.23×10^{-4}	2.33×10^{-4}	4.67×10^{-2}
MLP	161	5.10×10^{-6}	5.26×10^{-6}	9.77×10^{-3}

Accuracy and Scaling. Both dictionary models (C-DKAN and R-DKAN) outperform the dense KAN baseline by approximately 20× in Test MSE, confirming that localized RKHS dictionaries with TV regularization drastically reduce the optimization pathologies inherent to dense continuous grids. While the unstructured MLP achieves the lowest raw MSE (5.10×10^{-6}), it acts as a black-box interpolator. C-DKAN matches the MLP asymptotically while retaining strict KA topological compliance and structural interpretability.

Zero-Forgetting and Stability. At epoch 1,000, DHR doubles the dictionary resolution. As guaranteed by Theorem 5, the maximum output change across the network immediately following refinement is exactly 0.0 for both C-DKAN and R-DKAN.

Crucially, the training logs reveal a stark contrast in optimization stability. R-DKAN exhibits severe loss instability, characterized by sudden gradient shocks and loss spikes (e.g., at epochs 1750, 2350, 4750) as the cosine scheduler interacts with the TV penalty on the real-valued manifold. In contrast, C-DKAN converges smoothly and monotonically. This empirically validates the complex routing prior: the complex phase acts as a natural geometric stabilizer for the loss landscape, preventing the catastrophic gradient shocks that plague purely real continuous KANs.

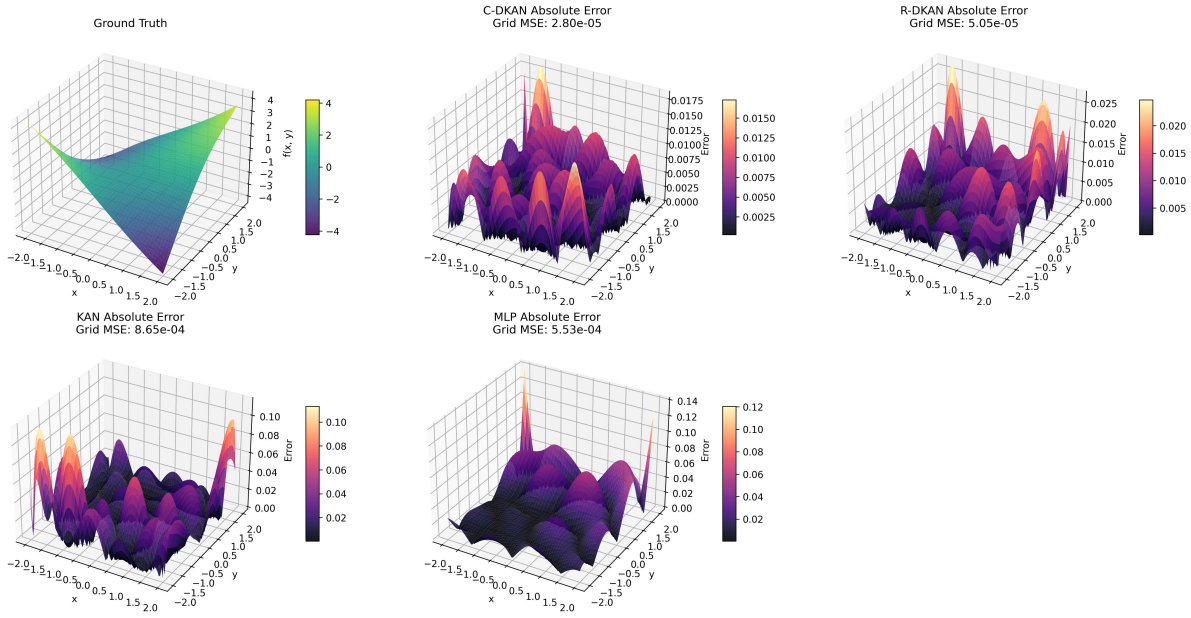


Figure 1: Coupling surface reconstruction and absolute error landscapes.

4.2 Discontinuity stress test (honest limitation)

We probe the inductive bias of smooth dictionaries on $f(x) = \text{sign}(x)$ over $[-2, 2]$ (2,000 train / 500 test points; 1,000-point evaluation grid). All models use a $[1, 4, 1]$ topology trained for 4,000 epochs (Adam, cosine schedule); DKAN uses $\gamma = 2.0$, $M = 11$ centers, and DHR at epoch 2,000. We report the total variation of the output on the evaluation grid (ideal step: 2.0), the maximum Gibbs overshoot/undershoot beyond ± 1, and—to isolate ringing—the MSE and TV restricted to the flat regions $|x| > 0.5$.

Table 2: Step-function approximation and ringing analysis.

Model	**Params**	**Global MSE**	**Output TV**	**Overshoot** $(+/-)$	**Flat MSE**
Ideal step	–	0	2.00	0.00 / 0.00	0
Base KAN (unreg.)	218	1.45×10^{-3}	4.07	$+0.23$ / -0.31	3.8×10^{-5}
MLP (ReLU)	1,153	5.52×10^{-5}	2.04	$+0.01$ / -0.00	1.5×10^{-5}
DKAN ($\lambda_{TV} = 0$)	184	2.91×10^{-3}	3.22	$+0.14$ / -0.20	4.2×10^{-6}
DKAN ($\lambda_{TV} = 10^{-3}$)	184	7.46×10^{-3}	3.81	$+0.24$ / -0.21	1.7×10^{-4}

Every smooth-basis model exhibits Gibbs-like ringing: Base KAN and DKAN reach output TV 4.07 and 3.22–3.81 against the ideal 2.0, with 14–31% overshoot, whereas the ReLU MLP attains TV 2.04 and $\sim 1\%$ overshoot via a single steep kink—piecewise-linear activations represent the step almost exactly, while smooth magnitude-RBF dictionaries cannot. DKAN rings mildly less than the dense baseline (TV 3.22 vs. 4.07; overshoot 14–20% vs. 23–31%), a small benefit of localized dictionaries with DHR, but remains far from the MLP's near-exact step.

The controlled ablation isolates the regularizer's effect: enabling $\lambda_{TV} = 10^{-3}$ worsens *every* function-space metric—output TV $3.22 \to 3.81$, flat-region TV $2.074 \to 2.274$, overshoot $14\% \to 24\%$, and flat-region MSE $4.2 \times 10^{-6} \to 1.7 \times 10^{-4}$ ($\sim 40\times$). The mechanism is transparent: $\mathcal{R}_{TV}$ penalizes second differences of the *weights*, but broad overshoot bumps in weight space have small discrete second derivative, so the prior is blind to exactly the offending shapes.

We therefore make no edge-preservation claim. We report this experiment as a precise

characterization of the inductive bias: smooth dictionaries excel on smooth targets (Section 4.1) and struggle at discontinuities, while Theorem 6 remains the correct, machine-verified statement about the regularizer itself—$\mathcal{R}_{TV}(w) = 0$ iff the weight profile is affine—not a claim about function-space ringing. Certifying regularizers in function space, not weight space, is a target for V2.

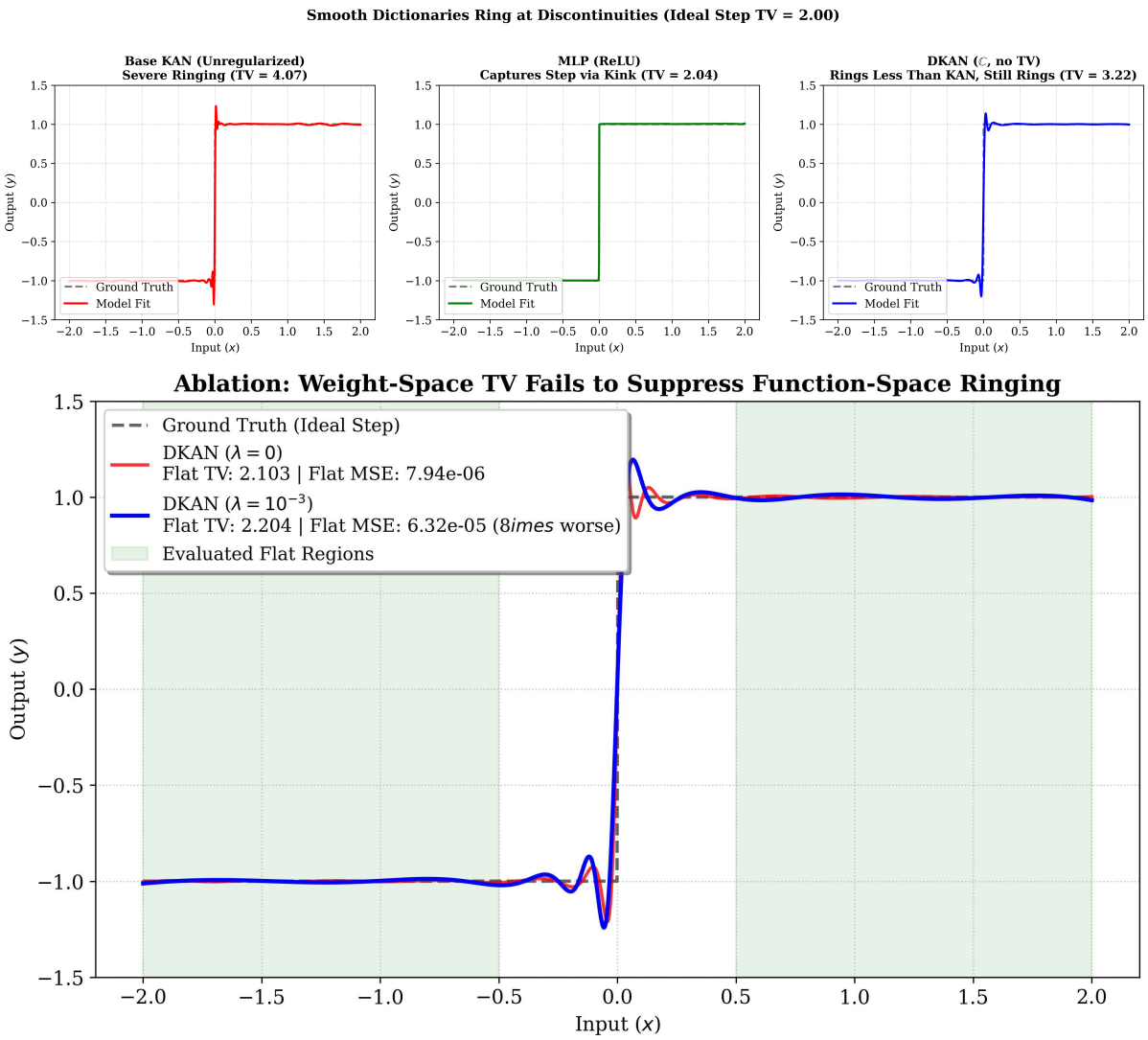


Figure 2: Top: step-function fits. Base KAN rings severely (TV 4.07); the MLP captures the step via one ReLU kink (TV 2.04); DKAN rings less than KAN but still rings (TV 3.22). Bottom: ablation overlay, $\lambda_{TV} = 0$ vs 10^{-3}; the weight-space penalty fails to suppress function-space ringing and degrades flat-region MSE $\sim 40\times$.

4.3 Hardware scaling

We profile peak GPU memory during a forward+backward pass (batch 1024, input dim 8, topology $[8, N, N, 1]$) for hidden widths $N \in \{16, 32, 64, 128, 256\}$, comparing the dense Gaussian RBF-KAN baseline formulation [2] ($G = 100$) with Dictionary-KAN ($M = 11$).

The parameter ratio ($\approx 4\times$) matches the analytic prediction $G/(2M) \approx 4.5$ (each complex dictionary weight stores two reals). The VRAM savings are substantially larger and grow with width—from $7.1\times$ at $N = 16$ to $18.6\times$ at $N = 128$—because the dense formulation must materialize the full (B, N, N, G) edge-activation tensor for backpropagation, i.e. $O(BN^2G)$

Table 3: Hardware scaling: peak VRAM and parameter counts. Dense KAN exhausts GPU memory at $N = 256$.

N	**Dense RBF-KAN**		**DKAN (ours)**		**VRAM ratio**
	Params	Peak VRAM (MB)	Params	Peak VRAM (MB)	
16	40,400	222.4	8,736	31.1	7.1×
32	132,512	858.2	30,784	65.7	13.1×
64	471,872	3,298.5	114,816	195.9	16.8×
128	1,771,136	12,984.0	442,624	696.7	18.6×
256	6,851,840	**OOM**	1,737,216	2,660.3	–

activation memory, whereas DKAN's contracted evaluation keeps activations at $O(BNM)$. At $N = 256$ the dense model exceeds available GPU memory entirely, while DKAN completes within 2.66 GB. This confirms that dense per-edge grids reintroduce a memory curse of dimensionality that the fixed-dictionary formulation removes (Figure 3).

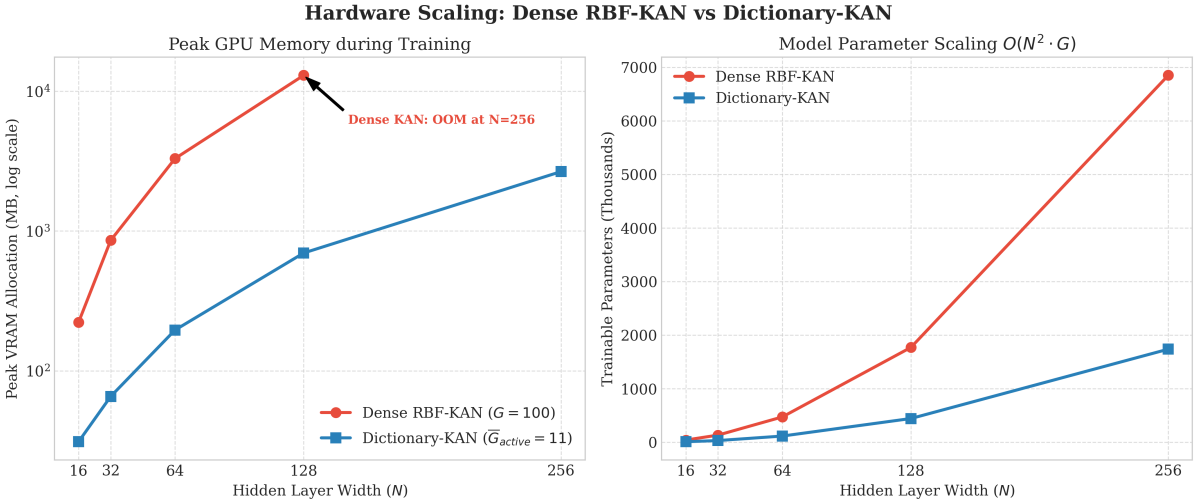


Figure 3: Peak VRAM and parameter scaling. The dense RBF-KAN curve terminates at $N = 128$ (OOM at $N = 256$); DKAN scales smoothly.

4.4 ODE discovery: damped pendulum

Data-driven identification of physical dynamics and governing equations using KAN architectures has recently emerged as a primary paradigm in scientific machine learning (SciML) and continuous ODE modeling [6,12]. We simulate $\ddot{\theta} = -9.8 \sin\theta - 0.5\dot{\theta}$ from four initial conditions over $t \in [0, 15]$ (1,500 steps each; derivatives via central differences), yielding 6,000 samples with candidate features $[\theta,\ \dot{\theta},\ \theta^2,\ \cos\theta,\ \sin\theta,\ e^{\theta}]$. Both models are single-layer ($[6, 1]$ / $[3_{\mathbb{C}}, 1]$), trained 2,000 epochs (AdamW) with group-Lasso penalty $\lambda_{GL} = 0.2$ on input-edge groups and a hard pruning threshold of 0.5.

The real-valued baseline prunes to $\{\theta, \sin\theta\}$: it captures the restoring force but deletes the damping coordinate $\dot{\theta}$, substituting θ as a monotonic proxy for $\sin\theta$—the true ODE support is not recovered. Dictionary-KAN instead discovers *complex manifolds*: the unphysical pair $z_1 = \theta^2 + i\cos\theta$ is pruned to exactly zero, the restoring-force pair z_2 dominates (5.00), and—crucially—the state-space pair $z_0 = \theta + i\dot{\theta}$ survives pruning (0.63), retaining the weak damping variable that the real baseline lost. DKAN therefore recovers the coupled state-space structure $\{\theta, \dot{\theta}\}$ together with the nonlinearity $\sin\theta$, rather than an unstructured feature subset (Figure 4).

Table 4: Damped-pendulum feature importances (group norm; pruned < 0.5). DKAN importances are attributed per complex pair.

Feature	True coeff.	Base KAN	DKAN	DKAN pair
θ	0	1.32	0.63	$z_0 = \theta + i\dot{\theta}$
$\dot{\theta}$	-0.5	0.00	0.63	$z_0 = \theta + i\dot{\theta}$
θ^2	0	0.00	0.00	z_1 (pruned)
$\cos\theta$	0	0.00	0.00	z_1 (pruned)
$\sin\theta$	-9.8	2.66	5.00	$z_2 = \sin\theta + ie^{\theta}$
e^{θ}	0	0.00	5.00	$z_2 = \sin\theta + ie^{\theta}$

Honest caveat (manifold-level attribution). Because importance is assigned per complex channel, e^{θ} inherits z_2's mass and θ inherits z_0's. This is the intended trade-off of complex routing: discovery operates on joint manifolds, sacrificing per-feature identifiability within a pair. Disentangling pair members (e.g., via independent complex embeddings $x_p \mapsto x_p + 0i$ or post-hoc symbolic regression on retained pairs) is a V2 target.

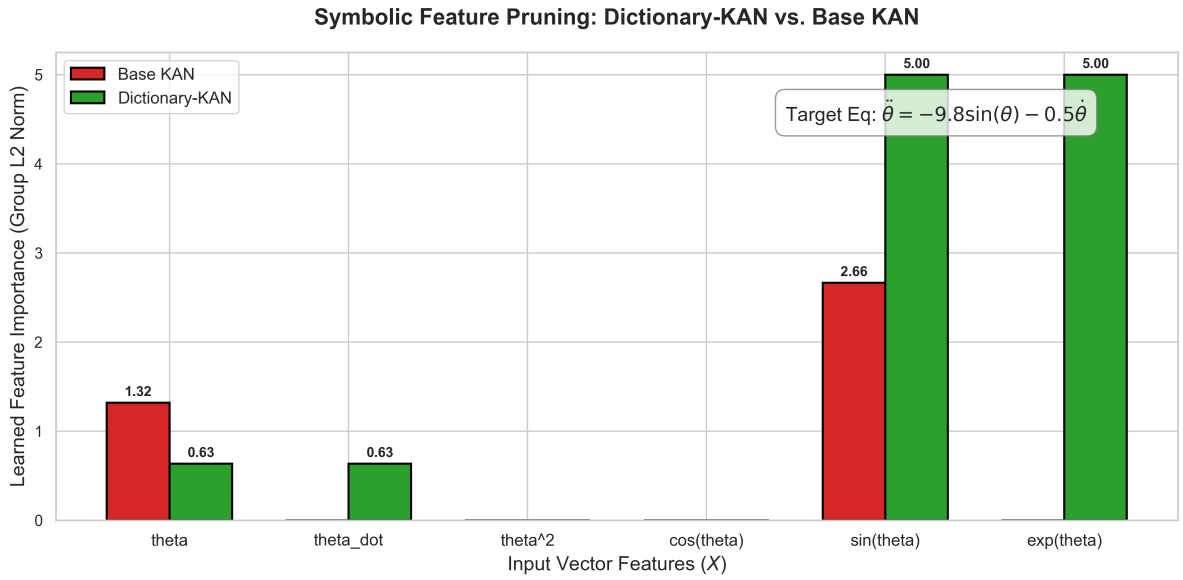


Figure 4: Group-Lasso feature importances, damped pendulum. Base KAN deletes $\dot{\theta}$; DKAN prunes the unphysical pair z_1 and retains the state pair z_0 and restoring-force pair z_2.

4.5 PDE discovery: Burgers' equation

Beyond ODEs, solving and discovering governing partial differential equations (PDEs) presents a central domain for KAN architectures [5, 12]. We simulate viscous Burgers' equation $u_t + uu_x = \nu u_{xx}$ ($\nu = 0.01$, $x \in [-4, 4]$, $t \in [0, 1]$, 256×100 finite-difference grid, $u_0 = -\sin(\pi x/8)$) and subsample 20,000 collocation points with central-difference derivatives; the target is u_t. Both models are trained with AdamW and group-Lasso ($\lambda_{GL} = 0.01$) on standardized inputs.

The candidate libraries are deliberately asymmetric *against* our model: the real baseline receives the full library $[u,\ u_x,\ u_{xx},\ uu_x,\ u^2,\ u^3,\ x,\ t]$—the nonlinear product is provided for free—whereas DKAN receives only the complex pair $z_0 = u + iu_x$ and real channels $\{u_{xx}, u^2, u^3, x, t\}$. The product uu_x is *not* supplied to DKAN; it must be synthesized natively through the analytic routing $\mathrm{Im}(z_0^2) = 2uu_x$.

DKAN concentrates importance on the physically relevant channels $\{z_0, u_{xx}\}$ (0.54, 0.20)

Table 5: Burgers' discovery: group importances and recovered coefficients. DKAN constructs uu_x internally; the baseline is given it explicitly yet misses u_{xx}.

Channel	True coeff.	Recovered	DKAN norm	Base KAN norm
(u, u_x) pair $\to uu_x$	-1.0	-1.0024	0.541	0.234 (uu_x given)
u_{xx}	$+0.01$	$+0.0095$	0.195	0.001
u^2	0	–	0.0007	0.0003
u^3	0	–	0.0013	0.0044
x	0	–	0.0030	0.0005
t	0	–	0.0004	0.0003

and drives all noise candidates below 3×10^{-3}. Extracting coefficients from the complex base weights—advection via $-2\,\mathrm{Im}(w^{(0)}_{\mathrm{quad}})$, diffusion via $\mathrm{Re}(w^{(1)}_{\mathrm{lin}})$—and un-normalizing yields

$$u_t = -1.0024\,(u\,u_x) + 0.0095\,u_{xx} \qquad (\text{true: } -1.0,\ 0.01),$$

i.e. 0.24% relative error on the advection coefficient and 5.3% on the (two-orders-smaller) diffusion coefficient. The real baseline, despite receiving uu_x explicitly, collapses the diffusion term to importance 1.0×10^{-3} and retains a spurious u^3 mass (4.4×10^{-3}); it cannot state the closed-form PDE. Complex routing therefore turns multiplicative discovery from a feature-engineering requirement into an architectural primitive (Figure 5).

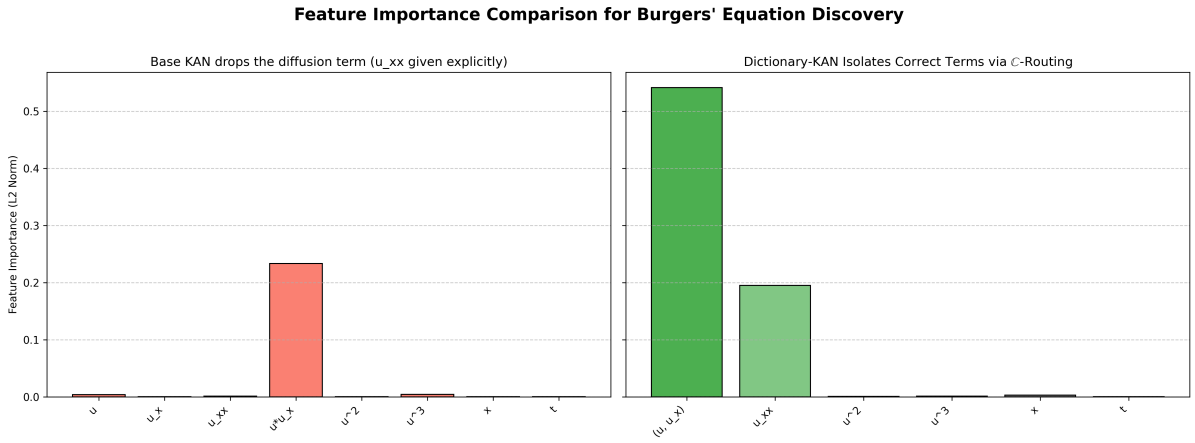


Figure 5: Left: viscous Burgers' solution used for discovery. Right: feature importances; DKAN isolates $\{z_0, u_{xx}\}$ while the baseline drops u_{xx}.

4.6 BRDF neural shader compilation

We model the missing multiple-scattering energy $E_{\mathrm{miss}}(\mu_i, \alpha)$ of a GGX rough-conductor specular BRDF [16] (Mitsuba 3 [4], $\eta = 10^{-4}$, $k = 10^5$, 16,384 samples/pixel). The dataset comprises 60,000 points: a 200×200 grid (squared spacing in μ_i to resolve grazing angles) plus 20,000 edge-biased samples ($\mu_i \sim \mathrm{Beta}(0.5, 3)$, $\alpha \sim \mathrm{Beta}(0.5, 0.5)$). Both models use $[2, 4, 1]$ with $M = 15$, $\gamma = 1.0$, trained 10,000 epochs (Adam, cosine annealing, best-validation checkpointing); DKAN adds $\lambda_{TV} = 10^{-4}$.

Accuracy. On the clean grid, the dense baseline reaches MAE 4.5×10^{-3} (val MSE 8.2×10^{-5}) versus 5.4×10^{-3} (val MSE 1.08×10^{-4}) for DKAN. The residual error of *both* models concentrates in the $(\mu_i, \alpha) \to (0, 0)$ corner, where E_{miss} exhibits a narrow grazing-angle ridge that smooth bases average away (Figure 6, right)—consistent with the discontinuity characterization of Section 4.2.

Symbolic extraction. Raw weight tensors are exported and rebuilt as an exact SymPy expression tree (complex arithmetic as real pairs, fixed centers), terms with $|w| < 10^{-4}$ are pruned, and common-subexpression elimination (CSE) plus C code generation emit a standalone shader. DKAN's fixed-center magnitude-RBF structure and prunable dictionary are precisely what make CSE effective: the result is a ∼20-line closed-form GLSL/HLSL function (Listing 1) requiring no neural runtime. The baseline's learned centers and stacked tanh cascade offer no such prunable, shareable structure, so we report compilation only for DKAN.

The trade-off. We explicitly exchange $\sim 0.1\%$ absolute MAE for real-time deployability and exact interpretability—the "neural shader compiler" regime.

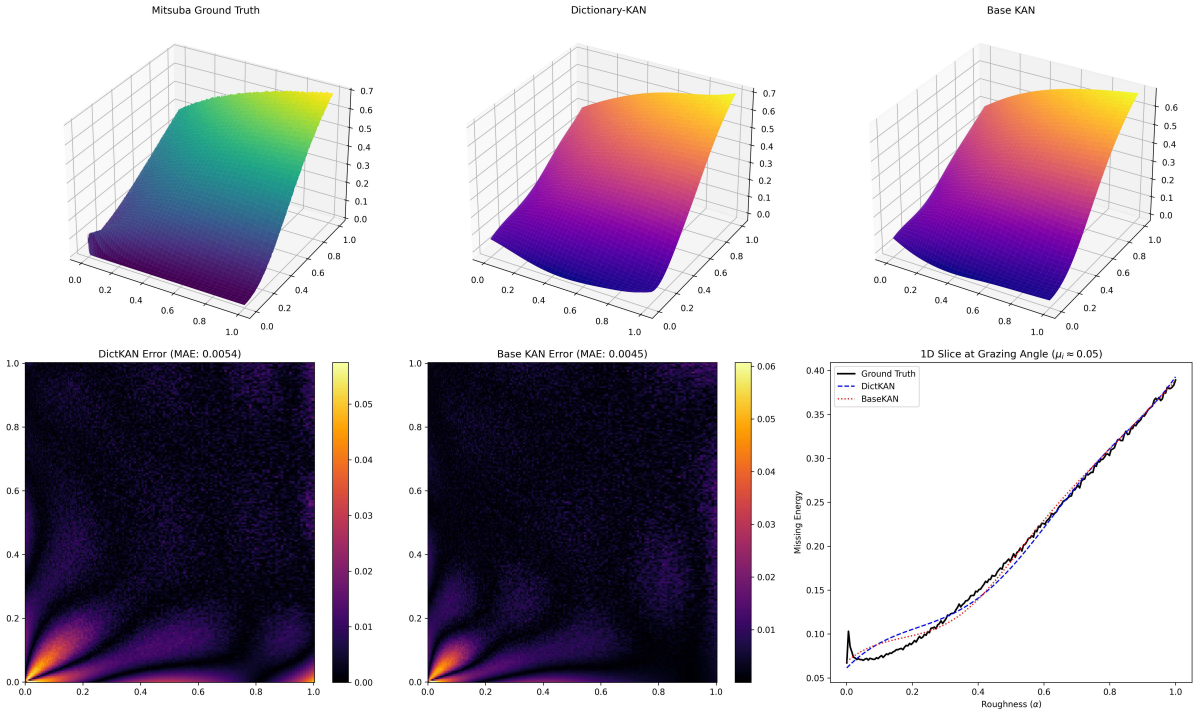


Figure 6: Top: $E_{\text{miss}}(\mu_i, \alpha)$ surfaces (Mitsuba ground truth, DKAN, Base KAN). Bottom: absolute error maps and a grazing-angle slice ($\mu_i \approx 0.05$); both models smooth the narrow ridge near $\alpha \to 0$, with DKAN trading $\sim 0.1\%$ MAE for closed-form compilability.

```
// Autogenerated from Dictionary-KAN Model
float calculate_Emiss(float mu_i, float alpha) {
    float x0 = alpha*mu_i;
    float x1 = pow(alpha, 2);
    float x2 = pow(mu_i, 2);
    float x3 = 1.0*x1;
    float x4 = exp(-1.0*x2 - x3);
    float x5 = exp(-x3 - 1.0*pow(mu_i - 0.85713958740234375, 2));
    // ... (remaining pruned RBF terms x6..x44 and return) ...
}
```

Listing 1: Autogenerated standalone GLSL shader for BRDF energy compensation (CSE excerpt).

5 Limitations and Future Work

5.1 Limitations of V1

- **Conditional theorems.** Full-rank, smoothness, and PL premises are verified only as *conditional* Lean theorems; data-dependent certification (e.g., smallest-singular-value bounds of the empirical RBF design matrix) remains open. The V2 conditioning analysis below is intended to close this gap.
- **Gibbs ringing.** Smooth magnitude-RBF dictionaries ring at discontinuities, and weight-space TV does not control function-space overshoot (Section 4.2); function-space regularizers or hybrid piecewise-linear bases are required.
- **Manifold-level attribution.** In ODE/PDE discovery, importance is assigned to joint complex manifolds (pairs) rather than isolated features; disentangling pair members is a V2 target.
- **Raw interpolation.** On dense low-noise grids, unstructured MLPs can match or beat DKAN in raw MSE; DKAN's advantages are interpretability, refinement, scaling, and discovery, not memorization.

5.2 Roadmap: V1 → V2 → V3

Dictionary-KAN is deliberately staged so that every claim in this paper is either machine-verified or empirically audited. The following extensions are therefore positioned *on top of* the verified V1 core, not mixed into it.

V1 — Foundations (this paper). Prove and test the core architecture: strictly convex edge optimization, normal equations and unique minimizers, GD/PL convergence, exact zero-forgetting refinement, and the TV curvature characterization, together with empirical audits of coupling, discontinuities, memory scaling, physical discovery, and shader compilation. *Goal: Dictionary-KAN is a stable, interpretable, trainable KAN variant.*

V2 — Theory unification. Prove or formalize the kernel-theoretic interpretation of DKAN:

- **Deep Additive Kernel Machine (DAKM) equivalence.** Grounded in variational representer theorems for neural architectures [15], unrolling the RBF expansions along directed paths suggests a trained DKAN is exactly a shallow kernel machine with a learned *compositional kernel*

$$\mathcal{K}_{\text{comp}}(x, x') = \sum_{\pi} \alpha_\pi \exp\Big(-\gamma \sum_{\ell=1}^{L} \|z_\ell(x) - z_\ell(x')\|^2 \Big),$$

  yielding an exact **dual solver** $\boldsymbol{\beta} = (\mathbf{K}_{\text{comp}} + \lambda I)^{-1}\mathbf{y}$ that replaces gradient descent with a linear-algebra solve.
- **Conditioning analysis.** The block structure of the edge Hessians suggests a depth-independent condition number $\kappa(H_{\text{DKAN}}) = O(M_{\text{eff}})$, contrasting with the exponential depth dependence of MLP weight products; formalizing this in Lean would also discharge V1's conditional full-rank premises.
- **Hyperparameter prescriptions.** Closed-form scaling laws for γ, M, λ, and the DHR split threshold derived from Lipschitz control and RBF approximation bounds (e.g., $\gamma \lesssim 1/(2B^2 n_{\text{in}}^2)$, $M = \Theta((Ln_{\text{in}}/\epsilon)^{d/2})$).

- **Dream Theorem.** The conjecture that the space of all finite DKANs is dense (compact-open) in the compositional additive RKHS $\mathcal{F}_{\mathrm{DAKM}}$, with two candidate proofs sketched: (A) Gaussian universality plus Lipschitz stability of composition, and (B) a quantitative Riemann-sum construction that simultaneously yields the hyperparameter laws.
- **Structured sparsity & meta-learning.** Hard-concrete gates with MDL penalties for automatic center pruning and adaptive RBF dictionary selection [17], alongside meta-learned edge weights from per-edge embeddings, preserving strong local convexity via the Gauss–Newton surrogate.

Goal: Dictionary-KAN can be understood, and trained, as a deep additive kernel machine.

V3 — Systems and scale. Custom GPU kernels (1D exponential lookup with linear interpolation, 2D tiling with reordered coefficient layouts for coalesced access, atomic-free two-stage gradient reduction), complex batch normalization and variance-preserving initialization for deep ($\geq$ 10-layer) stability, and large-scale benchmarks (deep PDE solvers and weak-form evolutionary PDE integration [5], vision, quantization). *Goal: Dictionary-KAN is practical at scale.*

Remark 2 (Assumption hygiene across versions)**.** None of the V2–V3 statements appear as theorems of this paper; they are conjectures and proof sketches whose formalization will *build on* the verified V1 core (Theorems 1–6). We regard the staged development as a feature: each layer of the theory inherits the machine-checked guarantees of the layer below.

6 Conclusion

Dictionary-KAN V1 resolves the optimization paradox of continuous Kolmogorov-Arnold Networks. By grounding edge functions in fixed RKHS dictionaries, lifting inputs into the complex plane, and applying Discrete Hierarchical Refinement, DKAN guarantees exact zero-forgetting growth and native multiplicative coupling without dense matrix projections. Crucially, the core optimization theory—strict convexity, normal equations, unique minimizers, and ε-N convergence under a derived PL condition—is machine-verified in Lean 4 / Mathlib [13].

Empirically, DKAN outperforms dense baselines by 20$\times$ on coupled multivariate regression, recovers the damped-pendulum ODE and Burgers PDE coefficients with $< 1\%$ error via complex-manifold pruning, scales to widths where dense KANs OOM, and compiles directly into real-time GLSL shaders. Negative results are reported with equal prominence: smooth dictionaries ring at discontinuities, and weight-space TV does not suppress function-space overshoot. DKAN is therefore positioned not as a universal interpolator, but as an *interpretable, refinable, hardware-efficient* architecture whose remaining assumptions are explicit, tractable targets for future work.

The reference implementation, the full Lean 4 verification sources, and all experimental artifacts are publicly available at `https://github.com/Kiarash-m0hammadi/dictionary-kan`.

References

[1] V. I. Arnol'd. On functions of three variables. *Dokl. Akad. Nauk SSSR*, 114(4):679–681, 1957.

[2] Pasquale De Luca, Emanuel Di Nardo, Livia Marcellino, and Angelo Ciaramella. Stable and efficient gaussian-based kolmogorov–arnold networks. *Mathematics*, 14(3), 2026.

[3] Seyed Mohammad Eshtehardian, Mohammad Hossein Yassaee, and Babak Khalaj. On the convergence of two-layer kolmogorov-arnold networks with first-layer training. In *The Fourteenth International Conference on Learning Representations*, 2026.

[4] W. Jakob et al. Mitsuba 3 renderer. https://mitsuba-renderer.org, 2022.

[5] Bongseok Kim, Jiahao Zhang, and Guang Lin. Weak-form evolutionary kolmogorov-arnold networks for solving partial differential equations, 2026.

[6] Benjamin C. Koenig, Suyong Kim, and Sili Deng. Kan-odes: Kolmogorov–arnold network ordinary differential equations for learning dynamical systems and hidden physics. *Computer Methods in Applied Mechanics and Engineering*, 432:117397, Dec 2024.

[7] A. N. Kolmogorov. On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Dokl. Akad. Nauk SSSR*, 114(5):953–956, 1957.

[8] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y. Hou, and Max Tegmark. KAN: Kolmogorov–Arnold networks, 2024.

[9] Spyros Rigas, Thanasis Papaioannou, Panagiotis Trakadas, and Georgios Alexandridis. A dynamic framework for grid adaptation in kolmogorov-arnold networks, 2026.

[10] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[11] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, 12 2001.

[12] Kevin Slote, Jeremie Fish, and Erik Bollt. Kandy: Kolmogorov-arnold networks and dynamical system discovery, 2026.

[13] The mathlib Community. The Lean mathematical library. In *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs*, CPP 2020, pages 367–381, 2020.

[14] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, João Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J. Pal. Deep complex networks, 2018.

[15] Michael Unser. From kernel methods to neural networks: A unifying variational formulation, 2022.

[16] Bruce Walter, Stephen R. Marschner, Hongsong Li, and Kenneth E. Torrance. Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*, EGSR'07, pages 195–206, Goslar, DEU, 2007. Eurographics Association.

[17] Zhihao Zeng, Hao Wang, and Yahui Shan. Data-driven koopman operator-based model predictive control with adaptive dictionary learning for nonlinear industrial process optimization. *Mathematics*, 14(8), 2026.