Beyond the Grid: Continuous Dictionary Pursuit for Interpretable Signal Decomposition

Nick Janssen
Institute of Information Processing
Leibniz University Hannover
janssen@tnt.uni-hannover.de

Melanie Schaller
Institute of Information Processing
Leibniz University Hannover
schaller@tnt.uni-hannover.de

Bodo Rosenhahn
Institute of Information Processing
Leibniz University Hannover
rosenhahn@tnt.uni-hannover.de

Abstract

Deep learning models achieve high accuracy in signal processing but remain difficult to interpret. Classical matching pursuit and sparse coding offer better transparency but suffer from spectral leakage and grid mismatch when signal frequencies do not align with discrete dictionary atoms. We introduce Continuous Dictionary Pursuit (CDP), a framework for interpretable signal decomposition that learns the parameters of closed-form functions directly from data. CDP optimizes over a continuous manifold of differentiable atoms and, unlike prior continuous pursuit methods that operate on a single analytic family, searches a heterogeneous library spanning periodic, trend, transient, and discontinuous structures. The algorithm employs a greedy iterative strategy using data-driven priors and gradient-based optimization to isolate individual signal components, with discontinuous waveforms made differentiable through spectral sharpness annealing. To ensure a sparse atom set for reconstruction, the framework utilizes a stopping criterion based on a predefined atom budget and residual error convergence. Evaluations on synthetic and real-world datasets show that CDP removes the grid discretization error for atom families contained in its library, recovers generative parameters to within 1% on stationary synthetic signals expressible in that library, and achieves the best sparsity-accuracy trade-off among classical, continuous-pursuit, and neural baselines in the sparse regime. Beyond decomposition, we demonstrate that the extracted atoms can serve as structured curriculum targets for training sequence models, improving forecasting accuracy over standard end-to-end training on most evaluated datasets. The method offers a glass-box alternative for signal decomposition by providing high reconstruction fidelity alongside direct physical interpretability.

1 Introduction

The decomposition of univariate time series into interpretable components is a fundamental problem across a wide range of scientific and engineering disciplines, from energy consumption planning [22, 34, 65] and financial market analysis [12, 57] to medical signal processing [10, 44, 58] and industrial sensor networks [21, 47]. Across these domains, the ability to express a signal as a sum of physically meaningful components is often as valuable as predictive accuracy, since the decomposition itself enables diagnosis, anomaly attribution, and scientific understanding.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

With increasing signal complexity, deep learning approaches achieve strong results on reconstruction benchmarks yet function as black boxes, offering little insight into the underlying signal structure [46]. Interpretable-by-design models address this by construction, for example through component-based image classifiers [37], and CDP pursues the same principle for signal decomposition. This matters in safety-critical settings such as healthcare [14, 49] and autonomous systems [27, 39, 61], where understanding a model's output is a practical and increasingly regulatory requirement [18, 23].

Classical approaches face a different set of limitations. Methods such as STL [13] produce components with clear semantic labels (trend, seasonality, residual) but no explicit functional form, while sparse coding methods such as Orthogonal Matching Pursuit (OMP) [40] represent signals as linear combinations of dictionary atoms and offer explicit, term-by-term reconstructions. However, operating on a fixed discrete dictionary exposes them to the grid mismatch problem [19, 41]: when a signal component's true parameters fall between grid points, the reconstruction requires multiple atoms to approximate a single physical feature, inflating model complexity and destroying interpretability.

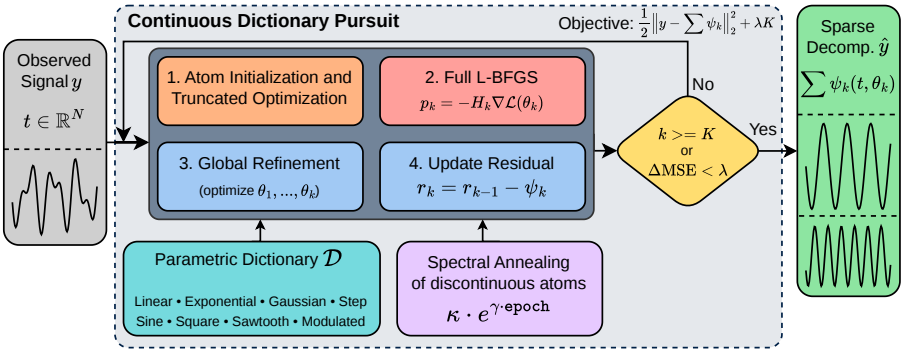

Figure 1: **Continuous Dictionary Pursuit:** The iterative greedy procedure initializes atoms from data-driven priors, identifies a candidate via truncated optimization, fully optimizes it with L-BFGS [30], and adds it to the additive model until the atom budget is exhausted or the residual error stagnates.

We propose Continuous Dictionary Pursuit (CDP), illustrated in Figure 1, a greedy iterative framework that optimizes directly over a continuous manifold of parameterized closed-form functions, removing the grid discretization error for atom families contained in its library through FFT-based spectral priors, truncated parallel optimization, and a global joint refinement step. We evaluate CDP against nine baselines ranging from classical decomposition to deep learning, including the closest continuous pursuit methods NOMP and SFW [33, 17], demonstrating its utility as a glass-box model in which every component has a precise mathematical definition and a direct physical interpretation.

The contributions of this work are as follows:

- **Continuous Dictionary Pursuit (CDP):** A greedy decomposition framework that optimizes over a continuously parameterized manifold of closed-form functions spanning periodic, trend, transient, and discontinuous waveform families, removing the grid discretization error of fixed-dictionary methods for atom families contained in its library.
- **Greedy Forward Selection:** A hybrid selection strategy combining FFT-based spectral priors for candidate initialization with truncated parallel optimization and full L-BFGS refinement under soft barrier penalties.
- **Pareto-Efficient Reconstruction:** On real-world LTSF benchmarks, CDP achieves the best sparsity-accuracy trade-off among all evaluated baselines in the sparse regime.
- **Atom-Based Curriculum Learning:** CDP atoms serve as structured curriculum targets for sequence models, reducing forecasting error over standard end-to-end training.
- **Open Source Release:** The full implementation of CDP is publicly available on GitHub.[1]

[1] `https://github.com/nick-janssen/ContinuousDictionaryPursuit`

2 Related Work

Classical and Spectral Decomposition: The ARIMA family [8] and exponential smoothing produce closed-form representations with statistically interpretable coefficients, but their linearity limits their ability to capture non-stationary dynamics [63]. STL [13] and MSTL [4] separate signals into seasonal, trend, and remainder components via iterative LoESS smoothing, yielding semantically labeled but non-closed-form expressions, and Prophet [54] fits piecewise linear trends with Fourier seasonalities yet cannot adapt to non-sinusoidal or aperiodic behavior. Subspace methods such as SSA [59] and DMD [48] decompose the trajectory matrix via Singular Value Decomposition, but their components are linear combinations of lagged observations that require additional analysis to link to physical phenomena. CDP instead learns an additive functional form whose components are members of named parametric families with directly readable parameters.

Sparse Coding and Dictionary Learning: Matching Pursuit (MP) [32] and Orthogonal Matching Pursuit (OMP) [40] greedily represent signals as sparse linear combinations of dictionary atoms, and K-SVD [2] further adapts the atoms to data, though they remain unstructured and physically uninterpretable. All fixed-dictionary methods share the grid mismatch problem [19, 41]: when a component's true frequency falls between grid points, several neighboring atoms are required to approximate a single physical feature, inflating model complexity and degrading interpretability. Atomic norm minimization [6] addresses this for the sinusoidal case with exact recovery guarantees [52, 11], but the underlying Toeplitz structure has no counterpart for mixed atom families containing Gaussians, polynomials, or non-sinusoidal waveforms.

Structured, Analytic, and Continuous Dictionaries: A rich body of work imposes structure on learned dictionaries [20], learns analysis-model operators instead of synthesis atoms [45], adapts dictionaries across domains [42], or constrains representations geometrically [53]. Morphological component analysis (MCA) [50, 7] separates signals over unions of analytic dictionaries, such as Fourier and wavelet bases, exploiting the morphological distinctness of the sub-dictionaries. In all of these approaches the learned or selected atom is ultimately a numerical vector, either learned under structural constraints or drawn from a parametric family discretized onto a grid, so interpretation requires reading structure out of the vector after the fact. Closest to our work are continuous pursuit methods that refine atom parameters off the grid: Newtonized OMP [33] augments greedy frequency estimation with Newton refinement over the continuum, and the Sliding Frank-Wolfe algorithm [17] solves the continuous sparse approximation problem (BLASSO) with exact local parameter updates. Both operate on a single smooth analytic family, which is what makes these exact update steps available. CDP belongs to this class of continuous pursuit methods and extends it to a heterogeneous library of periodic, trend, transient, and discontinuous waveforms with the atom type selected per iteration (section 3), and both methods serve as baselines in our evaluation (subsection 4.3).

Deep Sequence Models: N-BEATS [38] decomposes signals into basis expansions via stacked residual blocks with some structural interpretability through Fourier and polynomial constraints. Transformer and state-space architectures [60, 65, 29, 24] and pre-trained foundation models [15, 3] achieve strong forecasting accuracy, yet the underlying signal structure remains distributed across millions of parameters with no compact mathematical description. We target this gap, by decomposing the signal into a small set of closed-form components whose parameters are directly readable.

3 Methodology

We formulate the interpretable signal decomposition problem as finding a sparse set of parametric functions that best approximate the observed time series $\mathbf{y} \in \mathbb{R}^N$ over time $\mathbf{t} = [t_1, \ldots, t_N]^\top \in \mathbb{R}^N$, where $t \in \mathbb{R}$ denotes the continuous time variable and $\psi_k(\mathbf{t};\, \theta_k) \in \mathbb{R}^N$ denotes component-wise evaluation over the grid.

Problem Formulation: Let $\mathcal{A}$ be a set of atom families (e.g., Sine, Gaussian, Linear). Each family $a \in \mathcal{A}$ defines a parametric manifold $\mathcal{M}_a = \{\psi(t;\, \theta) : \theta \in \Theta_a\}$, where θ collects the continuous parameters (frequency, amplitude, width, etc.). The union of these manifolds forms our continuous dictionary $\mathcal{D} = \bigcup_{a \in \mathcal{A}} \mathcal{M}_a$.

Our goal is to find a set of K atoms $\{\boldsymbol{\psi}_k\}_{k=1}^K$ and their parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K$ to minimize the reconstruction error subject to a sparsity constraint. This can be formalized as an ℓ_0 regularized

nonlinear least squares problem:

$$\min_{K,\{\boldsymbol{\theta}_k\}_{k=1}^{K}} \underbrace{\frac{1}{2}\left\|\mathbf{y} - \sum_{k=1}^{K} \boldsymbol{\psi}_k(\mathbf{t};\boldsymbol{\theta}_k)\right\|_2^2}_{\text{Data Fidelity Term}} + \underbrace{\lambda K}_{\text{Sparsity Penalty}} \tag{1}$$

Here, the objective function balances two competing goals: accuracy and simplicity. The first term measures reconstruction error, driving parameters θ_k to capture signal morphology precisely. The second term penalizes model complexity, admitting a new atom only when it reduces the error by more than λ, which determines the model order automatically and prevents overfitting. The discrete analogue of this problem with a fixed dictionary is NP-hard [35, 16], and the continuous formulation considered here remains non-convex with no known efficient global solver. CDP therefore approximates the solution via greedy forward selection with gradient-based refinement. Instead of tuning λ directly, we treat the model order K as the sparsity control. By sweeping K over a set budget during our evaluations, we trace the Pareto frontier of accuracy vs. sparsity, where each fixed-K solution implicitly optimizes the objective for some λ.

3.1 The Parametric Dictionary Library

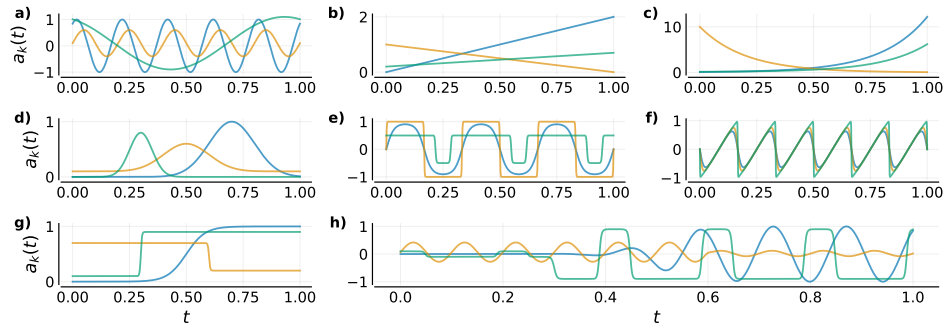

Figure 2: **Parametric Dictionary Library:** Example parameterizations of atom types: (a) SineAtom, (b) LinearAtom, (c) ExponentialAtom, (d) GaussianAtom, (e) SquareAtom, (f) SawtoothAtom, (g) StepAtom, (h) ModulatedAtom (for example `SineAtom` · `StepAtom`).

The efficacy of the CDP framework relies on a diverse library of parametric continuous and discontinuous atoms categorized into periodic, trend, shape, and modulated families. Every atom is built on a shared structure decomposing $\theta_k = (\phi_k, b_k)$ into atom specific parameters ϕ_k and a scalar bias $b_k \in \mathbb{R}$:

$$\psi_k(t;\theta_k) = a_k(t;\phi_k) + b_k \tag{2}$$

The bias b_k enables individual components to adapt to localized mean offsets without requiring a separate trend atom. For the discontinuous atoms, CDP offers differentiable approximations controlled by a sharpness scalar $\kappa > 0$, which is exponentially annealed during optimization, providing smooth gradients early and sharp waveform edges in the final epochs:

$$\kappa(\text{epoch}) = \min\left(\kappa_{\text{end}},\ \kappa_{\text{start}} \cdot e^{\gamma \cdot \text{epoch}}\right) \tag{3}$$

Trend and Shape Atoms Global and transient structure is covered by four families (Figure 2b,c,d,g). The LinearAtom and ExponentialAtom capture monotonic behavior, the GaussianAtom isolates localized transient events, and the StepAtom models abrupt level shifts. The ExponentialAtom is conditionally time-shifted based on the sign of the growth rate r, bounding the unscaled exponential

to a maximum of 1 and decoupling amplitude from rate to prevent gradient explosion [9]:

$$a_{\text{linear}}(t;\,\phi) = m \cdot t \tag{4}$$

$$a_{\text{exp}}(t;\,\phi) = A \cdot \exp\left(\begin{cases} r(t-T) & \text{if } r \geq 0 \\ r \cdot t & \text{if } r < 0 \end{cases}\right) \tag{5}$$

$$a_{\text{gauss}}(t;\,\phi) = A \cdot \exp\left(-\tfrac{1}{2}\left(\tfrac{t-\mu}{\sigma}\right)^2\right) \tag{6}$$

$$a_{\text{step}}(t;\,\phi) = A \cdot \sigma\left(\tfrac{8\kappa}{T}(t - t_{\text{step}})\right) \tag{7}$$

where m, A, r, T, μ, σ, t_{step} are the scalar entries of ϕ. The normalization factor $\frac{8}{T}$ in the StepAtom makes κ scale-invariant across series of different lengths.
The ModulatedAtom (Figure 2h) multiplies any carrier and modulator pair from the library, enabling composite structures such as damped oscillations or localized burst periodicities:

$$a_{\text{mod}}(t;\,\phi_{\text{c}}, \phi_{\text{m}}) = a_{\text{carrier}}(t;\,\phi_{\text{c}}) \cdot a_{\text{modulator}}(t;\,\phi_{\text{m}}) \tag{8}$$

Periodic Atoms Oscillatory behavior is anchored by the SineAtom. For non-sinusoidal periodicity, CDP provides differentiable approximations of two further waveform families (Figure 2a,e,f). The SquareAtom supports an adjustable duty cycle $\text{D} \in (0, 1)$ that widens or narrows the rectangular pulses [62], and the SawtoothAtom captures asymmetric rise and fall patterns [36]:

$$a_{\text{sine}}(t;\,\phi) = A \cdot \sin(2\pi f t + \varphi) \tag{9}$$

$$a_{\text{square}}(t;\,\phi) = A \cdot \tanh\left(\tfrac{\kappa}{2} \cdot (\sin(2\pi f t + \varphi) - \cos(\pi D))\right) \tag{10}$$

$$a_{\text{saw}}(t;\,\phi) = A \cdot (\tanh(2\kappa \cos\varphi_1)\sin\varphi_1 + \tanh(2\kappa\cos\varphi_2)\sin\varphi_2) \tag{11}$$

where $\varphi_{1,2} = \frac{1}{4}(2\pi f t + \varphi \pm \pi)$ and A, f, φ, D are the scalar entries of ϕ.

Theoretical Properties. The SineAtom manifold contains the $N \times N$ DCT-II basis as a special case and therefore spans $\mathbb{R}^N$, so the dictionary is complete (proof in subsubsection A.1.2). The remaining atom families enrich the dictionary with morphological structures that a pure sinusoidal basis cannot represent compactly. Completeness alone establishes representability, not convergence or recovery. In subsubsection A.1.3 we therefore show that CDP is an instance of the weak orthogonal greedy algorithm [55]. In this framework, completeness guarantees that the atomic norm $|\mathbf{y}|_{\mathcal{A}_1(\mathcal{D})}$ is finite, which is exactly the hypothesis of an explicit residual decay bound. The single condition of this bound is a per-step approximation ratio that can be computed exactly from every fitted model. Verifying it on all 13 real-world datasets, the certified residual bound tracks the actual training error with a median ratio of 1.12, and against a randomized reference search over the continuous dictionary the realized weakness parameter exceeds 0.8 in 85% of the greedy steps (subsubsection A.1.3). In subsubsection A.1.4 we further prove that the annealed surrogate objectives for discontinuous atoms converge uniformly to their ideal sharp-limit counterparts at an explicit $O(\kappa^{-1/2})$ rate, which implies Γ-convergence, so the smoothing is a consistent approximation rather than a heuristic. Formal global recovery and identifiability guarantees for mixed parametric families remain open for all methods in this class. Existing guarantees [6, 52, 11] cover only the sinusoidal case, and our synthetic parameter-recovery experiments (subsection 4.2) serve as their empirical counterpart.

3.2 Greedy Forward Selection via Data-Driven Priors and Truncated Optimization

We iteratively solve the sub-problem of finding the optimal atomic component to approximate the current residual $\mathbf{r}_{k-1} = \mathbf{y} - \sum_{i=1}^{k-1} \psi_i(\mathbf{t};\,\theta_i) \in \mathbb{R}^N$ as shown in Figure 1. At iteration k, we seek parameters θ^* such that:

$$\theta_k^* = \arg\min_{a \in \mathcal{A},\ \theta \in \Theta_a} \|\mathbf{r}_{k-1} - \psi(\mathbf{t};\,\theta)\|_2^2 \tag{12}$$

The highly non-convex nature of this parametric landscape makes random initialization susceptible to poor local minima. CDP therefore initializes candidates from a data-driven proposal distribution: periodic atoms start at dominant frequencies extracted via the Fast Fourier Transform (FFT) of the residual, while shape-based atoms are sampled from the residual's empirical variance and temporal support, with localization parameters bounded within the active time window and amplitudes scaled to the residual energy (subsubsection A.1.5).

To isolate the best candidate efficiently, a diverse population of instantiated atoms is optimized in parallel for a fixed truncated budget, which shifts candidates toward promising basins of attraction

without the cost of full convergence. The candidate with the minimal reconstruction error is then added to the active dictionary and fully optimized before the next greedy iteration. A global refinement step additionally performs a joint optimization over all active parameters $\Theta_k = (\theta_1, \ldots, \theta_k)$, correcting suboptimal assignments from earlier iterations by accounting for correlations between selected components, with a configurable frequency from after every atom to periodically after a predefined batch.

All optimization stages use L-BFGS [30], whose second-order curvature information decouples dimensions with highly disparate sensitivities such as frequency versus amplitude. To prevent physically invalid parameters, for example negative frequencies or vanishing bandwidths, the MSE objective is constrained with soft barrier penalties that smoothly force each parameter to its physically meaningful range (details in subsubsection A.1.6).

4 Experiments

4.1 Experimental Setup

We evaluate CDP on both synthetic and real-world datasets. CDP's gradient-based optimization trades a higher one-time training cost for a closed-form model with efficient inference, and hardware details, runtimes, and hyperparameters are reported in Appendices A.7 and A.9. All aggregated experiments and figures report the Interquartile Mean (IQM) across random seeds with 95% confidence intervals from percentile bootstrap, which is robust to outliers in small-sample evaluations [1]. All evaluations use 10 seeds, covering the stochastic methods of the main benchmark (CDP, CDP (Modulated), N-BEATS, NOMP, SFW, MCA, Analysis K-SVD) and the curriculum experiments. The deterministic baselines (MSTL, SSA, OMP, Prophet) are run once.

Evaluation Datasets: As the true decomposition of real-world signals is typically unknown, we first constructed a custom synthesizer to generate signals with varying complexities, including harmonics, trends, damped oscillations, and non-stationarities, which allows exact ground-truth validation of parameter recovery. For real-world validation, we evaluate on 13 series drawn from the standard LTSF benchmarks ETT [64], Electricity [64, 28], Exchange [28], PEMS [26], Solar [28], Traffic [28], and Weather [60], spanning diverse domains, sampling resolutions, and signal characteristics. Although these benchmarks are multivariate, CDP is designed for univariate decomposition, and we follow established practice by selecting the first variate per dataset.

Competitors: We benchmarked CDP against methods spanning classical decomposition, sparse approximation, additive modeling, continuous pursuit, structured dictionary learning, and deep learning. MSTL [4] and SSA [59] represent classical structural decomposition, OMP [40] with a discrete Fourier dictionary serves as a direct sparse spectral baseline, Prophet [54] as a widely used parametric additive model, and N-BEATS [38] as a neural baseline whose residual block architecture is explicitly motivated by signal decomposition. We further include the two closest continuous pursuit methods, Newtonized OMP (NOMP) [33] and Sliding Frank-Wolfe (SFW) [17], which refine atom parameters off the grid but operate on a single analytic family. Finally, we add two structured-dictionary baselines. Morphological component analysis (MCA) [50, 7] is instantiated with the same primitive families as CDP but with parameters frozen on discrete grids, which makes it a discretized ablation of our method, and Analysis K-SVD [45] represents analysis-model dictionary learning. Implementation details for all baselines are provided in subsection A.8.

4.2 Synthetic Evaluation and Parameter Extraction

As a basic sanity check, we test whether CDP identifies the true underlying components, all of which can be expressed by atoms in the library, rather than simply fitting their noisy sum. Both input and target signals are corrupted with additive Gaussian noise ($\sigma = 0.1$), while reconstruction quality is assessed against the clean ground truth.

We evaluate six signal structures of increasing complexity: a linear trend, a sine wave, a square wave, an off-grid frequency superposition, a damped sine wave, and a non-stationary amplitude shift. Across all six, CDP recovers the generative parameters to high precision, as shown in Figure 3. The recovered parameters deviate from the ground truth by less than 1% in all stationary cases. For off-grid frequencies, where standard OMP produces a reconstruction MSE of 0.36 due to spectral leakage, CDP achieves $7.00\mathrm{e}{-5}$. For the non-stationary amplitude shift, CDP places the transition

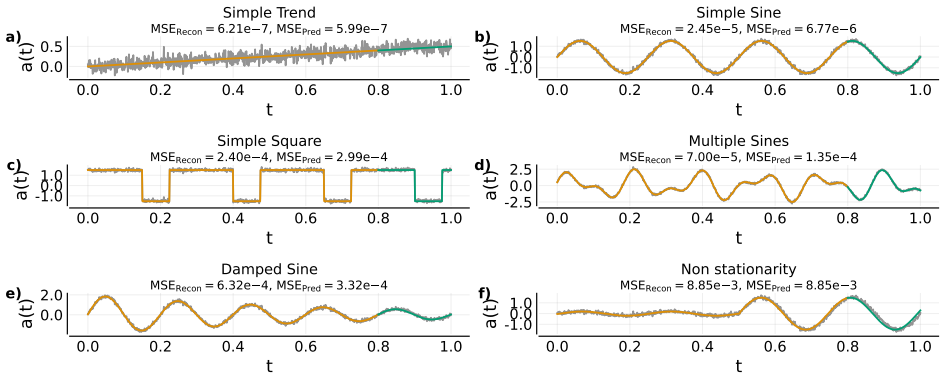

Figure 3: **Parameter recovery on synthetic datasets:** CDP reconstructs the true mathematical components from noisy targets ($\sigma = 0.1$): (a) linear trend, (b) sine, (c) square wave, (d) off-grid frequency superposition, (e) damped sine via a modulated atom, (f) non-stationary amplitude shift. Reconstruction and prediction MSE are given per panel.

point within 1% of the true location using a single modulated atom, while OMP yields an MSE of 0.31 and fails entirely on forecasting. The full parameter recovery equations for all six experiments are provided in subsection A.2.

4.3 Sparsity vs. Accuracy (Pareto Analysis)

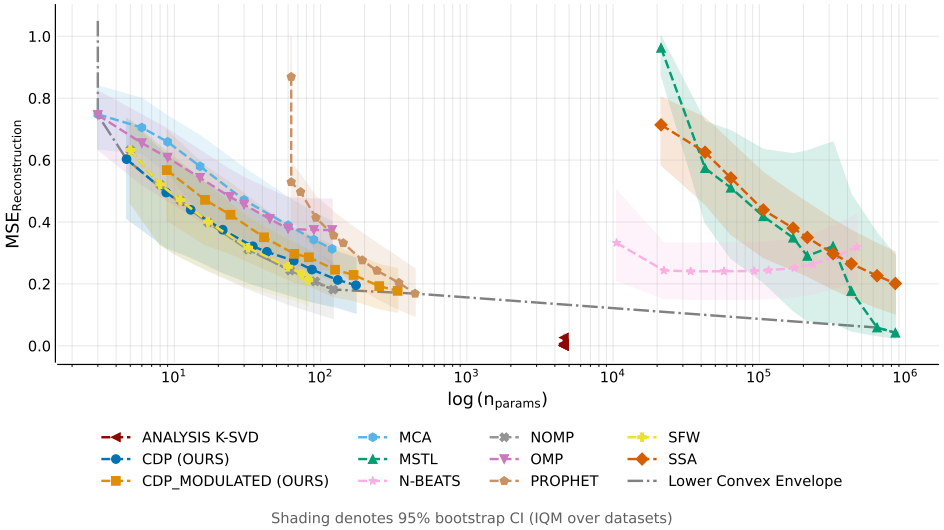

Figure 4: **Pareto frontier of parameter count versus training reconstruction MSE:** CDP and CDP (Modulated) achieve the optimal trade-off in the sparse regime across all baselines. The lower convex envelope excludes Analysis K-SVD, whose near-zero training residual reflects in-sample denoising rather than a parametric decomposition (subsection A.8).

To evaluate the trade-off between the number of parameters and the reconstruction Mean Squared Error (MSE), we constructed a Pareto frontier across the real-world LTSF datasets (Figure 4). The individual evaluations of each dataset can be found in the appendix (A.10).

A core objective of CDP is achieving high reconstruction accuracy with minimal parameters. Both CDP and CDP (Modulated) define the lower convex envelope at low parameter counts (10^1–10^2), meaning no baseline achieves a better accuracy-sparsity trade-off in this regime. Because a single continuous atom can resolve a physical component exactly, CDP reduces error faster per added

parameter than OMP, which requires multiple neighboring atoms to approximate off-grid frequencies. High-parameter baselines such as MSTL and SSA only enter the plot beyond 10^4 parameters and do not reach competitive error until 10^5–10^6, while Prophet plateaus at a high MSE regardless of parameter count. CDP thus occupies the efficient frontier in the sparse regime it targets while remaining fully parametrically transparent. Out-of-sample forecasting results follow a similar pattern, with performance degrading only when non-stationary domain shifts occur between train and test sets (See subsection A.5 for full analysis).

Comparison with continuous pursuit methods: The comparison with NOMP and SFW, which share CDP's off-grid refinement but not its mixed-morphology library, reveals two regimes. In the sparse regime, CDP achieves the best median test error across the 13 datasets, for example 0.62 against 1.05 for NOMP and 0.84 for SFW at K=3, while at larger budgets ($K \geq 20$) the single-family methods win through their exact update steps (subsection A.8). The morphology breakdown in Table 1 (analysis in subsection A.8) separates the two effects: with a matched sinusoid-only library, CDP's optimizer is equivalent to the single-family baselines on strongly periodic data, and the full library adds coverage gains of 13–28% on datasets with switching or spiky structure. The

Table 1: **Per-dataset training reconstruction loss at** $K = 5$ for the continuous and structured baselines, the CDP-sine ablation, and full CDP. The best value per dataset is bold, excluding Analysis K-SVD, whose near-zero training residual reflects in-sample denoising rather than a parametric decomposition (subsection A.8).

Dataset	NOMP	SFW	MCA	AK-SVD	CDP-sine	CDP
Electricity	0.562	0.567	0.622	0.007	0.593	**0.514**
ETTh1	**0.418**	0.421	0.594	0.005	0.486	0.427
ETTh2	**0.539**	0.545	0.760	0.004	0.692	0.545
ETTm1	**0.420**	0.422	0.575	0.002	0.488	0.426
ETTm2	0.537	0.542	0.740	0.002	0.691	**0.491**
Exchange	**0.055**	0.060	0.135	0.000	0.145	0.066
PEMS03	**0.196**	**0.196**	0.384	0.001	**0.196**	0.201
PEMS04	0.250	0.250	0.475	0.002	0.253	**0.238**
PEMS07	0.429	0.429	0.590	0.002	0.436	**0.313**
PEMS08	**0.138**	**0.138**	0.490	0.001	**0.138**	0.157
Solar	**0.161**	0.167	0.187	0.001	0.167	0.182
Traffic	**0.550**	**0.550**	0.712	0.003	**0.550**	0.579
Weather	0.589	0.595	0.791	0.000	0.599	**0.496**

matched-family MCA ablation isolates the value of continuous parameter optimization itself, as reconstruction with identical primitive families but grid-frozen parameters is worse at every budget (median 0.59 against 0.43 at K=5). Analysis K-SVD illustrates the analysis-model alternative, which achieves a near-zero in-sample residual but offers unstable extrapolation and no closed-form atoms.

4.4 Interpretability in Real-World Data

We qualitatively assessed the physical meaning of the extracted components on three example datasets. Because each atom's frequency maps directly to a physical period, the extracted components can be read in real time units. On PEMS07, CDP extracts a square-wave daily pattern whose duty cycle of 0.76 matches the split between high daytime and low nighttime traffic. On ETTm1, the dominant component is the 24-hour temperature cycle, recovered without any domain-specific prior. For PEMS04, shown in Figure 5, the decomposition separates a daily pattern, a weekly pattern capturing weekend drops, and localized rush-hour peaks, each corresponding to a single atom. To quantify how each method distributes explanatory weight across its components, and whether individual atoms capture distinct structures rather than spreading their contribution, we conduct an additional SHAP analysis [31] in subsection A.6.

4.5 Atom-Based Curriculum Learning

A key motivation for obtaining a structured parametric decomposition is that the recovered atoms encode the signal's components in a form that can directly benefit downstream learning. Following Bengio et al. [5], we train a vanilla S4D state space sequence model [25] by introducing atoms

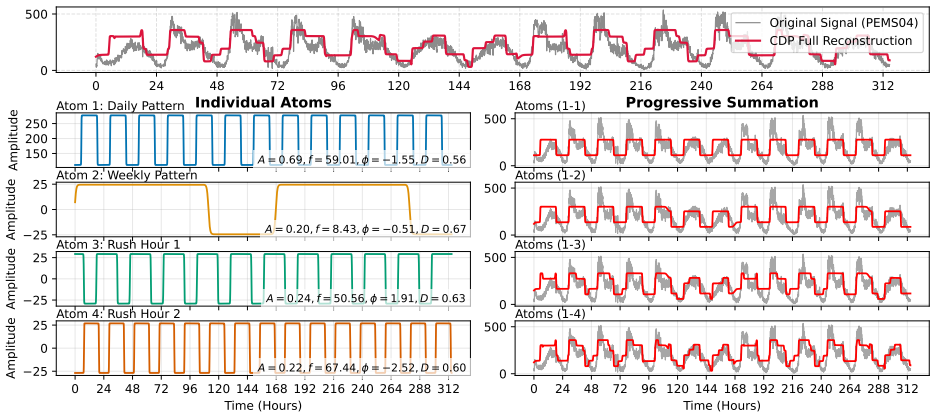

Figure 5: **Signal decomposition of the PEMS04 traffic dataset:** Top: the original signal overlaid with the full CDP reconstruction. Left: the individual extracted atoms, capturing daily cycles, weekly patterns, and localized rush-hour modulations. Right: their progressive summation toward the target signal.

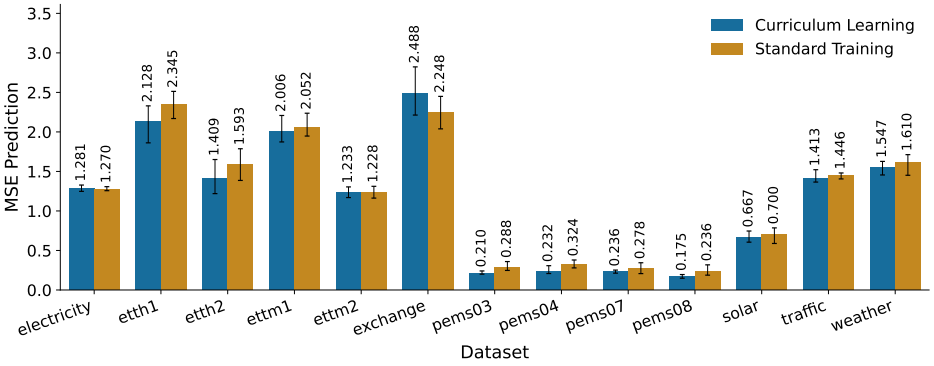

Figure 6: **Forecasting MSE for curriculum learning versus standard end-to-end training** across all 13 real-world LTSF datasets (IQM over 10 seeds, whiskers show 95% bootstrap confidence intervals).

one at a time, blending each into the target via a sigmoid-shaped curriculum weight, before a final fine-tuning stage on the normalized real signal. The full procedure is described in subsection A.4.

All curricula are evaluated under a single protocol on the 13 real-world LTSF datasets. The S4D model, the staging schedule, and the fine-tuning phase are held fixed, and only the curriculum targets vary, covering CDP atoms in greedy coarse-to-fine order, the components of the additive baselines (SSA, MSTL, OMP, Prophet, NOMP, SFW), CDP atoms in reversed fine-to-coarse order, a decomposition-free low-pass smoothing of the raw signal, and standard end-to-end training. Every condition uses the stored seed-0 decompositions at an atom budget of 10, and every reported value is the Interquartile Mean over 10 training seeds with 95% percentile bootstrap confidence intervals.

Across the 13 datasets, staging the CDP atoms in greedy coarse-to-fine order reduces the out-of-sample forecasting error relative to standard end-to-end training on 10 of 13 datasets, with relative reductions of 25 to 29 percent on PEMS03, PEMS04, and PEMS08 (Figure 6). The confidence intervals separate fully on PEMS03, and the curriculum is not significantly worse than standard training on any dataset. The reversed fine-to-coarse ordering performs worse than the greedy coarse-to-fine ordering on 11 of 13 datasets, so the ordering contributes a substantial part of the gain.

Comparing curricula built from all baseline decompositions (Table 2), CDP atoms are the best target on 8 of 13 datasets, and the benefit tracks the forecast fidelity of the underlying decomposition. On

Exchange and Weather, where CDP extrapolates poorly (subsection A.5), the curriculum advantage shrinks. The diverging SFW curriculum on Exchange is the extreme case: it selects nearly identical atoms whose large weights largely offset each other (subsection A.6), so partial sums grow far larger than the signal itself. CDP atoms are thus useful structured training targets precisely where CDP decomposes the signal accurately.

Table 2: **Curriculum evaluation on all datasets.** Out-of-sample forecast MSE of the S4D model for every curriculum target (IQM over 10 seeds). The best value per dataset is bold.

Dataset	CDP	CDP (rev.)	SSA	MSTL	OMP	Prophet	NOMP	SFW	Smooth.	Standard
Electricity	1.281	1.277	1.286	1.295	1.269	1.273	**1.254**	1.269	1.261	1.270
ETTh1	**2.128**	2.244	2.272	2.285	2.216	2.199	2.222	2.241	2.283	2.345
ETTh2	1.409	1.675	1.566	1.449	1.503	1.496	1.497	**1.397**	1.492	1.593
ETTm1	**2.006**	2.105	2.162	2.158	2.100	2.177	2.065	2.136	2.183	2.052
ETTm2	1.233	1.272	1.233	**1.218**	1.227	1.254	1.220	1.285	1.236	1.228
Exchange	2.488	2.340	2.265	2.260	2.062	2.335	2.323	9.8×10^4	**1.824**	2.248
PEMS03	**0.210**	0.251	0.298	0.235	0.310	0.292	0.291	0.296	0.453	0.288
PEMS04	**0.232**	0.264	0.298	0.583	0.247	0.314	0.286	0.293	0.348	0.324
PEMS07	**0.236**	0.248	0.489	0.401	0.379	0.481	0.418	0.446	0.645	0.278
PEMS08	**0.175**	0.304	0.301	0.276	0.269	0.266	0.244	0.261	0.402	0.236
Solar	**0.667**	0.678	0.768	0.750	0.694	0.699	0.724	6.451	0.799	0.700
Traffic	**1.413**	1.574	1.434	1.582	1.495	1.463	1.428	1.443	1.610	1.446
Weather	1.547	1.647	1.566	1.554	**1.322**	1.501	1.456	1.500	1.414	1.610

5 Limitations

CDP rests on a non-convex greedy optimization. The convergence result in subsubsection A.1.3 is conditional on a per-step approximation ratio that we measure rather than prove, formal identifiability guarantees for mixed parametric families remain open, and the high stability of the recovered atoms across seeds, time windows, and hyperparameters (subsection A.3) is an empirical observation rather than a guarantee. The method further depends on its predefined atom library, and for out-of-library structure the fit degrades toward a best-in-library approximation, so atoms extracted from real-world signals, where no ground-truth decomposition exists, should not be over-interpreted as true physical components. Finally, fitting is more expensive than classical grid-based greedy methods (subsection A.7), although the cost is a one-time investment and inference on the closed-form model is the cheapest of all evaluated methods, and CDP is currently limited to univariate signals, with a multivariate extension that shares atoms across channels left to future work.

6 Conclusion

This paper presents Continuous Dictionary Pursuit (CDP), a greedy iterative framework that removes the grid discretization error and the resulting spectral leakage of fixed-dictionary sparse coding for atom families contained in its library, by optimizing directly over a continuous parametric manifold that includes differentiable approximations of discontinuous shapes. On real-world benchmarks, CDP occupies a favorable position on the accuracy-sparsity Pareto frontier with orders of magnitude fewer parameters than deep learning baselines, and the extracted atoms serve as structured curriculum targets that reduce the out-of-sample forecasting error of neural sequence models on most evaluated datasets. The full implementation is publicly available to support reproducibility and further research.

CDP represents a meaningful step toward machine learning models that are not only accurate but also fully transparent, producing compact closed-form decompositions whose components can be interpreted directly.

Acknowledgements

This work has been supported by the European Union under grant agreement no. 101311180 (DEEP-HEALTH), the MWK of Lower Saxony within Hybrint (VWZN4219) and LCIS (VWZN4704), and the DFG under Germany's Excellence Strategy within the Clusters of Excellence PhoenixD (EXC 2122) and Quantum Frontiers (EXC 2123).

References

[1] Agarwal, R., Schwarzer, M., Castro, P.S., Courville, A.C., Bellemare, M.: Deep reinforcement learning at the edge of the statistical precipice. In: Advances in Neural Information Processing Systems. vol. 34, pp. 29304–29320 (2021)

[2] Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on signal processing **54**(11), 4311–4322 (2006)

[3] Ansari, A.F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S.S., Arango, S.P., Kapoor, S., et al.: Chronos: Learning the language of time series. Transactions on Machine Learning Research (2024)

[4] Bandara, K., Hyndman, R.J., Bergmeir, C.: Mstl: A seasonal-trend decomposition algorithm for time series with multiple seasonal patterns. International Journal of Operational Research **52**(1), 79–98 (2025)

[5] Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning. pp. 41–48 (2009)

[6] Bhaskar, B.N., Tang, G., Recht, B.: Atomic norm denoising with applications to line spectral estimation. IEEE Transactions on Signal Processing **61**(23), 5987–5999 (2013)

[7] Bobin, J., Starck, J.L., Fadili, J.M., Moudden, Y., Donoho, D.L.: Morphological component analysis: An adaptive thresholding strategy. IEEE Transactions on Image Processing **16**(11), 2675–2681 (2007)

[8] Box, G.E.P., Jenkins, G.M.: Time Series Analysis: Forecasting and Control. Holden-Day, San Francisco (1976)

[9] Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge university press (2004)

[10] Bui, C., Pham, N., Vo, A., Tran, A., Nguyen, A., Le, T.: Time series forecasting for healthcare diagnosis and prognostics with the focus on cardiovascular diseases. In: 6th International Conference on the Development of Biomedical Engineering in Vietnam (BME6). pp. 809–818. Springer Singapore, Singapore (2018)

[11] Candès, E.J., Fernandez-Granda, C.: Towards a mathematical theory of super-resolution. Communications on Pure and Applied Mathematics **67**(6), 906–956 (2014)

[12] Chakraborti, A., Patriarca, M., Santhanam, M.: Financial time-series analysis: A brief overview. In: Econophysics of Markets and Business Networks: Proceedings of the Econophys-Kolkata III. pp. 51–67. Springer (2007)

[13] Cleveland, R.B., Cleveland, W.S., McRae, J.E., Terpenning, I.: STL: A seasonal-trend decomposition procedure based on loess. Journal of Official Statistics **6**(1), 3–73 (1990)

[14] Da Cruz, H.F., Pfahringer, B., Martensen, T., Schneider, F., Meyer, A., Böttinger, E., Schapranow, M.P.: Using interpretability approaches to update "black-box" clinical prediction models: an external validation study in nephrology. Artificial Intelligence in Medicine **111**, 101982 (2021)

[15] Das, A., Kong, W., Sen, R., Zhou, Y.: A decoder-only foundation model for time-series forecasting. In: Forty-first international conference on machine learning (2024)

[16] Davis, G., Mallat, S., Avellaneda, M.: Adaptive greedy approximations. Constructive approximation **13**(1), 57–98 (1997)

[17] Denoyelle, Q., Duval, V., Peyré, G., Soubies, E.: The sliding Frank-Wolfe algorithm and its application to super-resolution microscopy. Inverse Problems **36**(1), 014001 (2020)

[18] Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608 (2017)

[19] Duarte, M.F., Baraniuk, R.G.: Spectral compressive sensing. Applied and Computational Harmonic Analysis **35**(1), 111–129 (2013)

[20] Dumitrescu, B., Irofti, P.: Dictionary Learning Algorithms and Applications. Springer, Cham (2018)

[21] Farahani, M.A., Kalach, F.E., Harper, A., McCormick, M., Harik, R., Wuest, T.: Time-series forecasting in smart manufacturing systems: An experimental evaluation of the state-of-the-art algorithms. Robotics and Computer-Integrated Manufacturing **95**, 103010 (2025)

[22] Forootani, A., Aliabadi, D.E., Thraen, D.: Climate aware deep neural networks (cadnn) for wind power simulation (2024)

[23] Goodman, B., Flaxman, S.: European union regulations on algorithmic decision-making and a "right to explanation". AI magazine **38**(3), 50–57 (2017)

[24] Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. In: First conference on language modeling (2024)

[25] Gu, A., Goel, K., Gupta, A., Ré, C.: On the parameterization and initialization of diagonal state space models. In: Advances in Neural Information Processing Systems. vol. 35, pp. 35971–35983 (2022)

[26] Guo, S., Lin, Y., Feng, N., Song, C., Wan, H.: Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 922–929 (2019)

[27] Jing, T., Xia, H., Tian, R., Ding, H., Luo, X., Domeyer, J., Sherony, R., Ding, Z.: Inaction: Interpretable action decision making for autonomous driving. In: European Conference on Computer Vision. pp. 370–387. Springer (2022)

[28] Lai, G., Chang, W.C., Yang, Y., Liu, H.: Modeling long-and short-term temporal patterns with deep neural networks. In: The 41st international ACM SIGIR conference on research & development in information retrieval. pp. 95–104 (2018)

[29] Lim, B., Arık, S.Ö., Loeff, N., Pfister, T.: Temporal fusion transformers for interpretable multi-horizon time series forecasting. International journal of forecasting **37**(4), 1748–1764 (2021)

[30] Liu, D.C., Nocedal, J.: On the limited memory bfgs method for large scale optimization. Mathematical programming **45**(1), 503–528 (1989)

[31] Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems. vol. 30 (2017)

[32] Mallat, S.G., Zhang, Z.: Matching pursuits with time-frequency dictionaries. IEEE Transactions on signal processing **41**(12), 3397–3415 (1993)

[33] Mamandipoor, B., Ramasamy, D., Madhow, U.: Newtonized orthogonal matching pursuit: Frequency estimation over the continuum. IEEE Transactions on Signal Processing **64**(19), 5066–5081 (2016)

[34] Mustafa, A.T., Al-Yozbaky, O.S.A.D.: Forecasting energy demand and generation using time series models: A comparative analysis of classical, grey, fuzzy, and intelligent approaches. Franklin Open p. 100350 (2025)

[35] Natarajan, B.K.: Sparse approximate solutions to linear systems. SIAM journal on computing **24**(2), 227–234 (1995)

[36] Noble, D.: A modification of the Hodgkin–Huxley equations applicable to purkinje fibre action and pacemaker potentials. The Journal of physiology **160**(2), 317 (1962)

[37] Norrenbrock, T., Kaiser, T., Biswas, S., Kose, N., Manuvinakurike, R., Rosenhahn, B.: CHiQPM: Calibrated hierarchical interpretable image classification. In: Advances in Neural Information Processing Systems (NeurIPS) (2025)

[38] Oreshkin, B.N., Carpov, D., Chapados, N., Bengio, Y.: N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. In: International Conference on Learning Representations (2020)

[39] Özkurt, C.: Interpretable ai analysis of chaos systems distribution in time series data from industrial robotics. Turkish Journal of Engineering (TUJE) **8**(4) (2024)

[40] Pati, Y.C., Rezaiifar, R., Krishnaprasad, P.S.: Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In: Proceedings of 27th Asilomar conference on signals, systems and computers. pp. 40–44. IEEE (1993)

[41] Pezeshki, A., Chi, Y., Scharf, L., Chong, E.: Compressed Sensing, Sparse Inversion, and Model Mismatch, pp. 75–95. Springer (01 2015)

[42] Qiu, Q., Patel, V.M., Turaga, P., Chellappa, R.: Domain adaptive dictionary learning. In: European Conference on Computer Vision (ECCV). pp. 631–645. Springer (2012)

[43] Rao, K.R., Yip, P.: Discrete cosine transform: algorithms, advantages, applications. Academic press (2014)

[44] Ríos, M., García, J.M., Cubedo, M., Pérez, D.: [time series in the epidemiology of typhoid fever in spain]. Medicina clinica **106 18**, 686–9 (1996)

[45] Rubinstein, R., Peleg, T., Elad, M.: Analysis K-SVD: A dictionary-learning algorithm for the analysis sparse model. IEEE Transactions on Signal Processing **61**(3), 661–677 (2013)

[46] Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature machine intelligence **1**(5), 206–215 (2019)

[47] Schaller, M., Hloch, S., Nag, A., Klichová, D., Pude, F., Zeleňák, M., Janssen, N., Rosenhahn, B.: Audio monitoring of bone cement disintegration in pulsating fluid jet surgery under laboratory conditions. Results in Engineering **29**, 108762 (2026)

[48] Schmid, P.J.: Dynamic mode decomposition of numerical and experimental data. Journal of fluid mechanics **656**, 5–28 (2010)

[49] Singh, A., Sengupta, S., Lakshminarayanan, V.: Explainable deep learning models in medical image analysis. Journal of imaging **6**(6), 52 (2020)

[50] Starck, J.L., Elad, M., Donoho, D.L.: Image decomposition via the combination of sparse representations and a variational approach. IEEE Transactions on Image Processing **14**(10), 1570–1582 (2005)

[51] Štrumbelj, E., Kononenko, I.: Explaining prediction models and individual predictions with feature contributions. Knowledge and information systems **41**(3), 647–665 (2014)

[52] Tang, G., Bhaskar, B.N., Shah, P., Recht, B.: Compressed sensing off the grid. IEEE transactions on information theory **59**(11), 7465–7490 (2013)

[53] Tasissa, A., Tankala, P., Murphy, J.M., Ba, D.: K-deep simplex: Manifold learning via local dictionaries. IEEE Transactions on Signal Processing **71**, 3741–3754 (2023)

[54] Taylor, S.J., Letham, B.: Forecasting at scale. The American Statistician **72**(1), 37–45 (2018)

[55] Temlyakov, V.N.: Weak greedy algorithms. Advances in Computational Mathematics **12**(2–3), 213–227 (2000)

[56] Temlyakov, V.N.: Nonlinear methods of approximation. Foundations of Computational Mathematics **3**(1), 33–107 (2003)

[57] Tsay, R.S.: Analysis of financial time series. John wiley & sons (2005)

[58] Vandith Sreenivas, K., Ganesan, M., Lavanya, R.: Classification of arrhythmia in time series ecg signals using image encoding and convolutional neural networks. In: 2021 Seventh International conference on Bio Signals, Images, and Instrumentation (ICBSII). pp. 1–6 (2021)

[59] Vautard, R., Yiou, P., Ghil, M.: Singular-spectrum analysis: A toolkit for short, noisy chaotic signals. Physica D: Nonlinear Phenomena **58**(1-4), 95–126 (1992)

[60] Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In: Advances in Neural Information Processing Systems. vol. 34, pp. 22419–22430 (2021)

[61] Yan, J., Wang, H.: Self-interpretable time series prediction with counterfactual explanations. In: International Conference on Machine Learning. pp. 39110–39125. PMLR (2023)

[62] Yu, Z., Mohammed, A., Panahi, I.: A review of three pwm techniques. In: Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041). vol. 1, pp. 257–261. IEEE (1997)

[63] Zhang, G.P.: Time series forecasting using a hybrid arima and neural network model. Neurocomputing **50**, 159–175 (2003)

[64] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: Beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 11106–11115 (2021)

[65] Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., Jin, R.: Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In: International conference on machine learning. pp. 27268–27286. PMLR (2022)

A Appendix

A.1 Methodology Details

This section provides expanded derivations and implementation details that supplement the main paper

A.1.1 Parametric Dictionary Library

ExponentialAtom: Numerical Stability via Reparameterization. A naive parameterization Ae^{rt} causes severe numerical problems during L-BFGS optimization: for large $|r|$ and a long horizon T, the output explodes, leading to an ill-conditioned Hessian and diverging gradient steps. The conditional time-shift in the main paper resolves this by ensuring the exponent is always non-positive:

$$\max_{t\in[0,T]} \exp(\cdots) = e^0 = 1 \tag{13}$$

This bounds the unscaled exponential to $[0, 1]$ and completely decouples A from r. The amplitude A then directly represents the peak value, equal to the signal at $t = T$ for growth and at $t = 0$ for decay, which is both physically interpretable and well-conditioned for optimization [9].

StepAtom: Scale-Invariant Sharpness. The normalization factor $\frac{8}{T}$ in the StepAtom makes κ scale-invariant across time series of different lengths. Since the sigmoid completes the majority of its transition between inputs -4 and $+4$, multiplying by $\frac{8}{T}$ ensures that at $\kappa = 1$ the transition spans the full observation window. Increasing κ sharpens the step independently of whether the series spans milliseconds or years, so the optimizer behaves consistently across datasets.

SquareAtom: Duty Cycle Threshold Mechanism. The term $\cos(\pi D)$ in the SquareAtom acts as a learnable threshold on the underlying sinusoid before the $\tanh$ activation. Increasing D toward 1 lowers the threshold, widening the active pulse phase, while decreasing D toward 0 narrows it. As $\kappa \to \infty$ the atom converges exactly to $A \cdot \text{sign}(\sin(2\pi ft + \varphi) - \cos(\pi D))$, recovering an ideal square wave with the specified duty cycle [62].

SawtoothAtom: Phase-Shifted Construction. The two arguments $\varphi_1 = \frac{1}{4}(2\pi ft + \varphi - \pi)$ and $\varphi_2 = \frac{1}{4}(2\pi ft + \varphi + \pi)$ represent the same fundamental frequency sampled at two opposing phase offsets. Each contributes one half of the sawtooth profile through the $\tanh$-weighted sinusoidal product, and their sum produces the asymmetric rise-fall pattern [36]. Like the other discontinuous atoms, the sharpness κ controls the transition steepness during optimization.

A.1.2 Dictionary Completeness

We formalize the claim from subsection 3.1 that any discrete signal in $\mathbb{R}^N$ is exactly representable as a finite sum of SineAtoms, and therefore as a finite sum of CDP atoms.

Setup. Let $\mathbf{t} = (t_0, t_1, \ldots, t_{N-1})^\top$ denote the discrete sampling grid with $t_n = n\Delta t$ for some sampling interval $\Delta t > 0$. The SineAtom family, written componentwise on the grid, is

$$\psi_{\text{sine}}(\mathbf{t}; A, f, \varphi, b) = A\sin(2\pi f\mathbf{t} + \varphi) + b \tag{14}$$

where $A, f, \varphi, b \in \mathbb{R}$. The Type-II Discrete Cosine Transform (DCT-II) basis on $\mathbb{R}^N$ consists of the vectors

$$\mathbf{c}_k[n] = \cos\left(\frac{\pi(2n+1)k}{2N}\right), \qquad n = 0, 1, \ldots, N-1, \qquad k = 0, 1, \ldots, N-1 \tag{15}$$

The set $\{\mathbf{c}_k\}_{k=0}^{N-1}$ is orthogonal under the standard inner product on $\mathbb{R}^N$, has full rank N, and therefore spans $\mathbb{R}^N$ [43].

Theorem (Completeness). For any signal $\mathbf{y} \in \mathbb{R}^N$, there exist coefficients $\{\alpha_k\}_{k=0}^{N-1}$ and parameter tuples $\{(A_k, f_k, \varphi_k, b_k)\}_{k=0}^{N-1}$ such that

$$\mathbf{y} = \sum_{k=0}^{N-1} \alpha_k\, \psi_{\text{sine}}(\mathbf{t}; A_k, f_k, \varphi_k, b_k) \tag{16}$$

exactly on the sampling grid $\mathbf{t}$.

Proof. The proof proceeds in two steps.

Step 1. Each DCT-II basis vector is a SineAtom evaluated on the grid. Using the identity $\cos(x) = \sin(x + \pi/2)$, we can write

$$\mathbf{c}_k[n] = \cos\left(\frac{\pi(2n+1)k}{2N}\right) = \sin\left(\frac{\pi(2n+1)k}{2N} + \frac{\pi}{2}\right) \tag{17}$$

Substituting $t_n = n\Delta t$ and matching to the SineAtom argument $2\pi f t_n + \varphi$ yields the parameter assignment

$$A_k = 1, \qquad f_k = \frac{k}{2N\Delta t}, \qquad \varphi_k = \frac{\pi k}{2N} + \frac{\pi}{2}, \qquad b_k = 0 \tag{18}$$

which gives $\psi_{\text{sine}}(\mathbf{t}; A_k, f_k, \varphi_k, b_k) = \mathbf{c}_k$ exactly on the sampling grid for every $k \in \{0, 1, \ldots, N-1\}$. Each frequency f_k lies in $\left[0, \frac{N-1}{2N\Delta t}\right]$, which is below the Nyquist frequency $\frac{1}{2\Delta t}$ and therefore within the admissible parameter range of the SineAtom, where the constant $k = 0$ basis vector is realized in CDP by the bias term rather than a zero-frequency sinusoid.

Step 2. The DCT-II basis spans $\mathbb{R}^N$. Since $\{\mathbf{c}_k\}_{k=0}^{N-1}$ is an orthogonal full-rank basis of $\mathbb{R}^N$, every signal $\mathbf{y} \in \mathbb{R}^N$ admits an exact expansion

$$\mathbf{y} = \sum_{k=0}^{N-1} \alpha_k \mathbf{c}_k \tag{19}$$

where $\alpha_k = \langle \mathbf{y}, \mathbf{c}_k \rangle / \|\mathbf{c}_k\|_2^2$ are the standard DCT coefficients.

Combining the two steps. Substituting the SineAtom representation of $\mathbf{c}_k$ from Step 1 into the DCT expansion from Step 2 yields the claimed sum of N SineAtoms with continuous parameters.

Remarks. The proof uses only the SineAtom family, so completeness holds even before the trend, transient, and waveform atom families are added. The remaining atoms enrich the dictionary in the sense that signals with non-sinusoidal physical structure can be represented by far fewer than N atoms. The completeness statement is therefore about representability, not about sparsity. Sparsity in the CDP framework is an empirical property that depends on the match between the atom library and the signal generating process, and is evaluated in subsection 4.3. Beyond representability, completeness supplies the hypothesis of the convergence guarantee established next: it ensures that the atomic norm $|\mathbf{y}|_{\mathcal{A}_1(\mathcal{D})}$ is finite for every $\mathbf{y} \in \mathbb{R}^N$.

A.1.3 From Completeness to Conditional Convergence

Completeness states that the dictionary is rich enough to represent any signal. It does not state that the greedy procedure actually approaches such a representation. This section closes that gap. We show that CDP follows the template of the weak orthogonal greedy algorithm (WOGA) [55, 56], a classical scheme for greedy approximation in which each step is allowed to select a good atom rather than the best one. The value of this connection is a residual decay bound whose only unknown is one number per iteration, and this number can be computed exactly from every fitted model. The non-convexity of the atom search therefore does not invalidate the analysis, it only determines how large these measured numbers turn out to be.

Setup. Let $\mathcal{D} = \{\psi(\cdot;\theta)/\|\psi(\cdot;\theta)\|_2 : a \in \mathcal{A}, \theta \in \Theta_a\}$ denote the continuous dictionary with every atom rescaled to unit norm, the standard normalization in greedy approximation theory. Each parameter domain Θ_a can be taken compact: frequencies lie in a closed interval below the Nyquist limit enforced by the barrier penalties, temporal anchors lie in the observation window, and amplitudes and biases can be restricted to a bounded box without loss, since the least-squares optimal coefficient of any unit-norm atom is bounded by $\|\mathbf{y}\|_2$ through the Cauchy-Schwarz inequality. The map $\theta \mapsto \psi(\cdot;\theta)$ is continuous in L_2, so suprema over $\mathcal{D}$ are attained. Let $\mathbf{r}_k$ denote the residual after k greedy iterations ($\mathbf{r}_0 = \mathbf{y}$), and let $\boldsymbol{\varphi}_k \in \mathcal{D}$ be the unit-norm render of the atom accepted at iteration k. The quantity we track is the realized per-step ratio

$$t'_k \;=\; \frac{|\langle \mathbf{r}_{k-1}, \boldsymbol{\varphi}_k \rangle|}{\|\mathbf{r}_{k-1}\|_2^2}. \tag{20}$$

which measures how strongly the accepted atom correlates with the residual it was fitted to.

Lemma (per-step decrease). The first step of the analysis is to show that every accepted atom is guaranteed to reduce the residual by a quantifiable amount. After acceptance of atom k, the residual satisfies

$$\|\mathbf{r}_k\|_2^2 \;\le\; \|\mathbf{r}_{k-1}\|_2^2 - \langle \mathbf{r}_{k-1}, \boldsymbol{\varphi}_k \rangle^2. \tag{21}$$

Proof. The right-hand side of (21) is exactly the error left after the best least-squares fit of the single direction $\boldsymbol{\varphi}_k$ to the residual, so the claim is that every accepted atom does at least as well as this fit. This holds for a simple reason. Amplitude and bias enter every atom render linearly, so setting the gradient of the fitting objective to zero in these two coordinates is the same as solving their least-squares normal equations. An accepted atom whose amplitude and bias are first-order stationary is therefore the exact least-squares fit of its own shape to the residual, which achieves the decrease (21) with all earlier atoms unchanged. CDP drives these coordinates to stationarity with L-BFGS under strong Wolfe line search, and the condition can also be enforced exactly by one closed-form rescaling of the accepted atom. All subsequent refinement stages, namely the full refinement of the accepted atom and the interleaved global refinement of all atoms, accept only steps that decrease the objective, so they can only shrink the residual further. □

Theorem (computable residual bound). Accumulating the guaranteed decreases over all iterations yields a bound on the final residual. Under (21), the residual after K iterations satisfies

$$\frac{1}{\|\mathbf{r}_K\|_2^2} \;\ge\; \frac{1}{\|\mathbf{y}\|_2^2} + \sum_{k=1}^{K} (t'_k)^2. \tag{22}$$

Proof. Writing $a_k = \|\mathbf{r}_k\|_2^2$ and substituting (20) into (21) gives $a_k \le a_{k-1}\big(1 - (t'_k)^2 a_{k-1}\big)$, where $(t'_k)^2 a_{k-1} = \langle \mathbf{r}_{k-1}, \boldsymbol{\varphi}_k \rangle^2 / a_{k-1} \le 1$ by Cauchy-Schwarz. Taking reciprocals and using the elementary inequality $\frac{1}{x(1-\varepsilon x)} \ge \frac{1}{x} + \varepsilon$ for $\varepsilon x < 1$ yields $a_k^{-1} \ge a_{k-1}^{-1} + (t'_k)^2$. Summing over k gives (22). □

The point of this bound is that it contains no unknown constants. Only the signal energy and inner products of computed vectors appear, so the bound can be evaluated exactly on every fitted model, and no supremum over the continuous dictionary is involved.

Corollary (a priori rate). The classical WOGA analysis asks how close each greedy step comes to the best atom in the entire dictionary. Call the scouting step τ-approximate, $\tau \in (0, 1]$, if $|\langle \mathbf{r}_{k-1}, \boldsymbol{\varphi}_k \rangle| \ge \tau \sup_{g \in \mathcal{D}} |\langle \mathbf{r}_{k-1}, g \rangle|$ at every iteration, meaning the selected atom captures at least a fraction τ of the correlation that the best atom in $\mathcal{D}$ would achieve. With the per-step decrease of the lemma alone, CDP is a τ-weak greedy algorithm, for which the classical rates are weaker [55]. It becomes a weak orthogonal greedy algorithm with weakness parameter τ under the additional condition that the residual is kept orthogonal to the span of all accepted atoms, which corresponds to a joint least-squares update of all amplitudes and biases and is what the interleaved global refinement approximates. Under this condition, Temlyakov's rate theorem [55, 56] gives

$$\|\mathbf{r}_K\|_2 \;\le\; |\mathbf{y}|_{\mathcal{A}_1(\mathcal{D})} \left(1 + \tau^2 K\right)^{-1/2}, \tag{23}$$

where $|\mathbf{y}|_{\mathcal{A}_1(\mathcal{D})}$ is the atomic norm of $\mathbf{y}$ with respect to $\mathcal{D}$, a measure of how cheaply $\mathbf{y}$ can be assembled from dictionary atoms. The completeness theorem of subsubsection A.1.2 guarantees that this norm is finite for every $\mathbf{y} \in \mathbb{R}^N$, so the bound applies to every signal. In practice the computable form (22) is the more useful statement, since $\|\mathbf{y}\|_2 \le |\mathbf{y}|_{\mathcal{A}_1(\mathcal{D})}$ makes it the tighter of the two. An explicit upper bound on the atomic norm can further be obtained by basis pursuit over any finite sub-dictionary, which can only overestimate it.

Numerical verification. We evaluate the bound (22) post hoc on fitted models at $K = 10$ on all 13 real-world datasets, recomputing all projections with exact least-squares solves on the fitted atom renders, independently of the optimization path. Since every quantity in the bound is computed from the realized greedy trajectory, this is a diagnostic of how tightly the guaranteed per-step decreases account for the achieved training error, not an independent proof of convergence. Three observations support the analysis. First, the assumption of the lemma holds in practice: the certified projection cascade matches the model's own training error to within 9% relative MSE on every dataset, with a median gap below 1%, and the small remaining differences reflect the finite L-BFGS tolerance and the finite sharpness of the training surrogates for discontinuous atoms. Second, the bound is

tight: it tracks the actual training error with a median ratio of 1.12 and a range of 1.00 to 1.67 across datasets, for example 0.23 against 0.19 on PEMS04. Per greedy step, the selected atom captures a median 7% of the remaining residual energy and up to 74%, and 92% of steps capture at least 1%, with the rare near-zero steps occurring when the interleaved global refinement leaves two atoms nearly collinear, so that the second one contributes almost no incremental energy in the exact cascade. Third, the scouting step is close to optimal: comparing each selected atom against a randomized reference search over the continuous dictionary with 100 refined candidates per atom family, the realized weakness parameter τ has a median of 1.0 and exceeds 0.8 in 85% of steps, occasionally beating the reference search itself. The realized ratios t'_k are also logged during fitting.

Scope. The guarantee is conditional: we do not prove global optimality of the non-convex scouting step, and no method in this class does. Formal sparse recovery and identifiability results for mixed parametric families remain open. Existing guarantees from atomic norm minimization [6, 52, 11] cover the sinusoidal case, where the Toeplitz structure applies, and no analogue is known for dictionaries mixing sinusoids, Gaussians, polynomials, and discontinuous waveforms.

A.1.4 Consistency of the Sharpness Annealing

The differentiable surrogates for discontinuous atoms are consistent approximations of their ideal sharp-limit renders, in the following sense.

Lemma (Γ-convergence of the annealed objectives). For the square, sawtooth, and step families, let $g_\kappa(\cdot;\theta)$ denote the annealed surrogate with sharpness κ and $g_\infty(\cdot;\theta)$ the ideal render. Assume that the duty cycle of the square family is bounded away from $\{0,1\}$ and that the transition of the step family lies inside the observation window. Then:

(i) $\|g_\kappa(\cdot;\theta) - g_\infty(\cdot;\theta)\|_{L_2} \le C(\theta)\,\kappa^{-1/2}$, where $C(\theta)$ is explicit and diverges only as the duty cycle approaches 0 or 1.

(ii) The annealed least-squares objectives F_κ Γ-converge to the ideal objective F_∞ on the compact parameter domain. Consequently, every accumulation point of minimizers of F_κ as $\kappa \to \infty$ is a minimizer of F_∞, and the minimal values converge.

Proof. (i) $\tanh(\kappa u)$ differs appreciably from $\operatorname{sign}(u)$ only where $|u| = O(1/\kappa)$. The argument $u(t) = \sin(2\pi f t + \varphi) - \cos(\pi D)$ crosses zero with derivative bounded below precisely when D is bounded away from 0 and 1, so the set where the surrogate and the ideal render differ has measure $O(1/\kappa)$ per period. On this set the difference is bounded by $2A$, which gives an L_2 error of $O(\kappa^{-1/2})$. The constant degrades as the crossing derivative vanishes, i.e., at extreme duty cycles. (ii) Since the objective is quadratic in the render, (i) implies uniform convergence of F_κ to F_∞ on the compact domain, and uniform convergence on a compact set implies Γ-convergence together with convergence of minimizers. □

Empirical confirmation. The finite-sharpness gap predicted by (i) is measured at approximately 1% at the final sharpness $\kappa = 50$ for moderate duty cycles, and grows at extreme duty cycles, exactly the regime the constant $C(\theta)$ identifies. The theory thus predicts the one regime where the annealed approximation is visibly loose.

A.1.5 Data-Driven Proposal Distribution

The main paper states that localization parameters are bounded within the active time window and amplitudes are scaled to the residual energy. The motivation is identifiability: a GaussianAtom with center μ placed far outside $[0, T]$ contributes near-zero variance within the observation window and is functionally indistinguishable from a constant, producing zero-gradient regions that stall L-BFGS. Strictly bounding μ and t_{step} within $[0, T]$ at initialization prevents this degenerate case.

A.1.6 Soft Barrier Penalty Details

To enforce structural validity, the MSE objective is augmented with a soft barrier penalty term:

$$\mathcal{L}(\theta_k) = \text{MSE}(\theta_k) + w_{\text{limit}} \sum_{p \in \theta_k} \mathcal{L}_{\text{limit}}(p) \tag{24}$$

where $p \in \theta_k$ denotes any scalar entry of the parameter tuple θ_k, and $w_{\text{limit}} > 0$ controls the barrier strength. The full set of per-atom constraints enforced by $\mathcal{L}_{\text{limit}}$ is:

- **Frequency** f: penalized as $f \to 0$ (DC) or $f > f_{\text{Nyquist}}$ (aliasing).
- **Gaussian width** σ: penalized as $\sigma \to 0$ to suppress impulse-like artifacts.
- **Temporal anchors** μ, t_{step}: penalized outside $[0, T]$ for the identifiability reason above.
- **Duty cycle** D: bounded to $(0, 1)$ to keep the SquareAtom physically valid.

The L-BFGS i-th step descent direction $p_i = -H_i \nabla \mathcal{L}(\theta_i)$ uses an approximate inverse Hessian H_i built from a rolling window of gradient differences. This second-order information implicitly rescales each parameter dimension by its local curvature, which is critical for the CDP objective where frequency and amplitude sensitivities can differ by several orders of magnitude.

A.2 Synthetic Parameter Recovery

We provide the exact generative equations and recovered parameters for all six synthetic experiments from subsection 4.2. Both input and target signals are corrupted with additive Gaussian noise, $\tilde{y}(t) = y(t) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 0.1$, and all MSE values are computed against the clean ground truth $y(t)$. This setup jointly tests structural decomposition and denoising, providing a stronger evaluation than noise-free reconstruction alone.

Linear Trend:

$$y = 0.5 \cdot t \tag{25}$$

$$\hat{y} = 0.5 \cdot t - 0.0008 \tag{26}$$

MSE: $6.21\text{e}{-7}$. The LinearAtom recovers both slope and bias accurately, confirming that greedy selection correctly identifies and isolates trend parameters even under noise.

Sine Wave:

$$y = 1.5 \cdot \sin(2\pi \cdot 4.0 \cdot t) \tag{27}$$

$$\hat{y} = 1.4996 \cdot \sin(2\pi \cdot 4.0006 \cdot t - 0.0059) - 0.005 \tag{28}$$

MSE: $2.45\text{e}{-5}$. All three parameters, namely amplitude, frequency, and phase, are recovered to high precision, validating the FFT-based spectral proposal mechanism.

Differentiable Square Wave:

$$y = 1.5 \cdot \text{sign}\left(\sin(2\pi \cdot 4.0 \cdot t) - \cos(0.7\pi)\right) \tag{29}$$

$$\hat{y} = 1.514 \cdot \tanh\Big(\frac{\kappa}{2} \cdot \left(\sin(2\pi \cdot 3.996 \cdot t + \varphi) - \cos(0.7001\pi)\right)\Big) \tag{30}$$

MSE: $2.40\text{e}{-4}$. The duty cycle is recovered at $D = 0.7001$, within $\sim 0.01\%$ of the true value, demonstrating that spectral annealing successfully sharpens the waveform edges without trapping the optimizer in early local minima.

Off-Grid Frequency Superposition:

$$y = 1.5 \cdot \sin(2\pi \cdot 5.8 \cdot t) + 1.0 \cdot \sin(2\pi \cdot 10.3 \cdot t + 0.5) \tag{31}$$

$$\hat{y} = 1.5 \cdot \sin(2\pi \cdot 5.8001 \cdot t) + 1.0 \cdot \sin(2\pi \cdot 10.3001 \cdot t + 0.5) \tag{32}$$

MSE: $7.00\text{e}{-5}$. Because both frequencies fall between FFT bins, standard OMP produces a reconstruction MSE of 0.36 due to severe spectral leakage. CDP resolves both components to within 0.0002% through continuous frequency optimization.

Damped Sine Wave:

$$y = 2.0 \cdot \sin(2\pi \cdot 5 \cdot t) \cdot \exp(-1.5 \cdot t) \tag{33}$$

$$\hat{y} = 2.04 \cdot \sin(2\pi \cdot 5.0049 \cdot t + 0.0199) \cdot \exp(-1.3902 \cdot t) - 0.07 \tag{34}$$

MSE: $6.32\text{e}{-4}$. The correct carrier-modulator pairing emerges from the greedy selection without supervision. OMP, whose Fourier dictionary cannot represent the multiplicative carrier-envelope coupling, yields a reconstruction MSE of 0.07 and a forecasting MSE of 0.5.

Non-Stationary Amplitude Shift: Here $H(t)$ denotes the Heaviside step function and $\sigma(t)$ the sigmoid.

$$y = \sin(2\pi \cdot 4 \cdot t) \cdot 1.3 \cdot H(t - 0.5) + 0.2 \tag{35}$$

$$\hat{y} = 1.2875 \cdot \sin(2\pi \cdot 4.0574 \cdot t - 0.15) \cdot \sigma\big(\tfrac{8 \cdot 50}{0.8}(t - 0.505)\big) + 0.19 \tag{36}$$

MSE: $8.85\text{e}{-3}$. The 0.8 in the sigmoid's steepness term $\frac{8 \cdot 50}{0.8}$ equals $T_{\text{train}} = 0.8$, the length of the training interval under an 80/20 train-test-split of the full unit interval $T = 1.0$. CDP reconstructs the regime change with a single ModulatedAtom, placing the transition at $t = 0.505$, within 1% of the true shift point. OMP's static dictionary is ill-suited for non-stationary shifts, resulting in a reconstruction MSE of 0.31 and a forecasting MSE of 1.38.

A.3 Robustness and Stability

This section collects the analyses of CDP's sensitivity to initialization, hyperparameters, noise, and library design.

Stability across random seeds: At an atom budget of $K = 5$ across all 13 real-world datasets and 10 seeds, we rank-match the recovered periodic atoms across seeds and measure the relative standard deviation of their frequencies: 76% of matched atoms vary by less than 1%, and 81% by less than 5%. The selected atom-type multiset is identical across all seeds on 8 of 13 datasets. The recovered decompositions are therefore substantially a property of the signal rather than of the random initialization.

Stability across time windows: To verify that recovered atoms reflect persistent signal structure rather than idiosyncrasies of a particular observation window, we refit CDP at $K = 5$ on ten overlapping sub-windows per dataset. Each sub-window spans 80% of the training portion, the start offsets are evenly spaced over the remaining 20%, and the fit seed is held fixed so that the observation window is the only varying factor. Periodic atoms are matched across windows by clustering their frequencies with a 2% relative tolerance, and a component is called persistent if it appears in at least eight of the ten windows. Transient atoms (Gaussian, step) are localized by construction and are excluded from the matching. Across the 13 datasets this yields 25 persistent periodic components, and their frequencies are essentially window-invariant: 96% vary by less than 1% relative standard deviation across windows (median 0.02%), and all vary by less than 5%. The waveform-type label at a persistent frequency is less stable and matches the modal type in 76% of appearances, since neighboring waveform families, for example square and sawtooth atoms at the same fundamental frequency, can approximate the same component. On Electricity and Weather no component reaches the persistence threshold at this atom budget, as their energy is distributed over many components of comparable strength that trade places across windows.

Noise robustness: On a more complex synthetic dataset with known ground truth, we evaluate recovery under increasing noise levels. CDP recovers the main components for SNR values down to 2. Below that, the atoms with lower power degrade gradually rather than failing abruptly, which indicates graceful behavior at the noise floor.

Hyperparameter sensitivity: We ran additional configurations on all 13 real datasets varying the key hyperparameters jointly: scouting batch size increased $5\times$ (10 to 50), scouting epochs $10\times$ (5 to 50), and global refinements per fit from 2 up to one refinement per atom. The resulting median training reconstruction differs by less than 3% at matched atom budgets, indicating robustness to these choices.

Atom-library ablation: The effect of restricting CDP to a sinusoid-only library is analyzed as part of the morphology breakdown in subsection A.8, which separates library coverage from optimizer quality per dataset.

Out-of-library structure: CDP removes discretization error only for structure expressible in its library. When a signal contains structure outside every atom family, such as a continuously varying instantaneous frequency, the fit degrades toward a best-in-library approximation composed of locally fitted atoms, and the extracted parameters describe that approximation rather than the generating process (cf. section 5).

A.4 Atom-Based Curriculum Learning

This section provides a detailed description of the curriculum learning procedure introduced in subsection 4.5, including the evaluation protocol, the target construction, the curriculum weighting schedule, and the S4D sequence model used as the downstream learner.

A.4.1 Overview

The core idea is to use the K atoms fitted by CDP as a structured sequence of learning targets for a neural sequence model. Rather than training directly on the raw signal, which presents the model with the full complexity of all components simultaneously, the curriculum introduces atoms one at a time. Each atom is smoothly blended into the training target via a sigmoid-shaped weight that transitions from zero to one over a configurable number of iterations. In the final phase, the model is fine-tuned on the normalized real signal.

A.4.2 Evaluation Protocol

The evaluation protocol, the compared curriculum targets, and the aggregation statistics are described in subsection 4.5. For NOMP and SFW the stored constant offset enters as an additional first component of the staged decomposition. The confidence intervals of the curriculum and standard training conditions overlap on most individual datasets, so the per-dataset effect is modest but consistent in direction across the benchmark.

A.4.3 Curriculum Target Construction

Given a decomposition into K components $\{\psi_k\}_{k=1}^{K}$, the curriculum target at iteration i is defined as the weighted sum:

$$y^{\text{curr}}(\mathbf{t}; i) = \sum_{k=1}^{K} w_k(i) \cdot \psi_k(\mathbf{t}; \theta_k) \tag{37}$$

where $w_k(i) \in [0, 1]$ is the curriculum weight of component k at iteration i, computed from the current iteration index as described below. In practice each component enters as its stored per-component reconstruction from the fitted decomposition, which lets the non-parametric baseline components be staged through the same formula.

A.4.4 Curriculum Weighting Schedule

The total training budget is divided into $T_{\text{curr}} = T \cdot (1 - \rho)$ curriculum iterations and $T_{\text{real}} = T \cdot \rho$ fine-tuning iterations on the real signal, where T is the total number of iterations and ρ is the real-data ratio hyperparameter. The curriculum iterations are divided equally among K atoms, giving $T_k = T_{\text{curr}}/K$ iterations per atom. The activation shift for atom k is $s_k = k \cdot T_k$ for $k = 0, \ldots, K-1$, so the first atom is active from the start of training.

The weight of atom k at iteration i is computed as:

$$\tilde{w}_k(i) = \sigma\left(\frac{6(i - s_k)}{\alpha \cdot T_k} + 6\right) \tag{38}$$

$$w_k(i) = \begin{cases} 1 & \text{if } \tilde{w}_k(i) > \sigma(\alpha \cdot T_k) \\ 0 & \text{if } \tilde{w}_k(i) < \sigma(-\alpha \cdot T_k) \\ \tilde{w}_k(i) & \text{otherwise} \end{cases} \tag{39}$$

where σ is the logistic sigmoid and α is the `curriculum_smoothness` hyperparameter. Weights above $\sigma(\alpha \cdot T_k)$ or below $\sigma(-\alpha \cdot T_k)$ are clamped to exactly 1 or 0. At the settings used, these thresholds are numerically indistinguishable from 1 and 0, so the clamping acts as a numerical saturation guard rather than an active mechanism.

A small α produces a sharp step-like transition, while a larger α yields a gradual blend.

A.4.5 Training Phases

Training proceeds in two phases. During the curriculum phase (iterations 0 to T_{curr}), the model is trained on $y^{\text{curr}}(\mathbf{t}; i)$ computed from the weighted atom sum. During the fine-tuning phase (iterations T_{curr} to T), the target is replaced by the z-score normalized ground truth signal $\tilde{y}(t) = (y(t) - \mu_{\text{train}})/\sigma_{\text{train}}$, allowing the model to adapt to residual structure not captured by the CDP decomposition. The loss in both phases is the mean squared error between the model output and the current target. The model maps the normalized time index to the signal value, so out-of-sample predictions are obtained by evaluating the trained model on the continuation of the time axis, and the reported test MSE is computed on the segment beyond the training portion.

A.4.6 The S4D Sequence Model

The downstream learner is a vanilla S4D model, a stack of L S4D blocks [25] with a pointwise input projection and a linear output projection. The model maps a univariate time series $\mathbf{x} \in \mathbb{R}^{N\times 1}$ to a reconstruction $\hat{\mathbf{x}} \in \mathbb{R}^{N\times 1}$.

Each S4D block contains a structured state space kernel, a local depthwise convolution, a GELU activation followed by a gated linear unit output projection, and a residual connection with layer normalization. The S4D kernel parameterizes a diagonal linear time-invariant system. For a block with model dimension H and state dimension N, the kernel at sequence length $\ell \in \{0, 1, ..., L-1\}$ is computed as:

$$k_h[\ell] = 2\,\mathrm{Re}\left(\sum_{n=1}^{N/2} \tilde{C}_{hn} \cdot e^{A_{hn}\,\ell}\right), \qquad \tilde{C}_{hn} = C_{hn} \cdot \frac{e^{A_{hn}\Delta_h} - 1}{A_{hn}} \tag{40}$$

where $C_{hn} \in \mathbb{C}$ are the output projection weights and $\Delta_h \in \mathbb{R}_{>0}$ is the per-channel time step. To enforce stability and positivity through unconstrained optimisation, both the time step and the magnitude of the real part of the state matrix are parameterised in log-space,

$$A_{hn} = -\exp(\log A^{\text{re}}_{hn}) + i\pi n, \qquad \Delta_h = \exp(\log \Delta_h) \tag{41}$$

so that the learned variables $\log A^{\text{re}}_{hn}$ and $\log \Delta_h$ are real-valued and free of constraints. The imaginary part $i\pi n$ of A_{hn} assigns each state a fixed oscillatory frequency, allowing the kernel to represent periodic patterns efficiently. The real part is held strictly negative by construction, which guarantees that the underlying continuous-time system is stable and that the kernel decays over ℓ.

The kernel is applied to the input via FFT convolution:

$$\mathbf{y}_h = \text{iFFT}(\text{FFT}(\mathbf{x}_h) \odot \text{FFT}(\mathbf{k}_h)) + \mathbf{x}_h \cdot D_h \tag{42}$$

where D_h is a learned skip connection weight. A local depthwise convolution with kernel size 3 is applied after the global convolution to help capture sharp local transitions such as those produced by SquareAtom and StepAtom. The output then passes through a gated linear unit and a residual connection.

The full model architecture and the hyperparameters for the curriculum learning are summarized in Table 3 and Table 4 respectively.

A.5 Extrapolation and Forecasting Performance

Because CDP fits closed-form parametric functions to the training signal, it can extrapolate deterministic periodicities over long horizons without the autoregressive error accumulation typical of deep learning models. As shown in Figure 7, CDP and CDP (Modulated) maintain a favorable position on the forecasting Pareto frontier across the LTSF benchmark suite with a stable forecasting MSE of approximately 0.8–1.0.

Prophet is the most unstable baseline, with forecasting MSE peaking at approximately 4.6 in the 10^2 parameter range before declining, reflecting its inability to reliably capture non-sinusoidal periodicities. OMP starts near 1.0 but drifts upward as more atoms are added, since spectral leakage from off-grid frequencies accumulates rather than cancels across greedy steps. MSTL and SSA enter only beyond 10^4 parameters and converge to MSEs of 1.5–2.0, well above CDP. N-BEATS enters at 10^4 parameters with high variance and does not consistently improve beyond 1.3, likely because its

Table 3: **AtomS4DModel architecture.** Default hyperparameters used in all curriculum learning experiments.

Component	Details	Default
Input projection	Conv1d($1 \to d_{\text{model}}$, kernel 1)	$d_{\text{model}} = 128$
S4D blocks	n_{layers} stacked S4DBlocks	$n_{\text{layers}} = 4$
State dimension	Per-block S4D state size	$d_{\text{state}} = 64$
Local conv	Depthwise Conv1d(kernel 3, padding 1)	per block
Output gate	GLU via Conv1d($d_{\text{model}} \to 2d_{\text{model}}$)	per block
Dropout	Applied after GLU	$p = 0.1$
Output projection	Linear($d_{\text{model}} \to 1$)	
Time step range	log-spread for different time resolutions	$\Delta \in [10^{-3}, 10^{-1}]$
Optimizer	Adam	lr $= 10^{-3}$

Table 4: **Curriculum learning hyperparameters** used in all experiments.

Hyperparameter	Description	Default
T	Total training iterations	1000
K	Total number of atoms	10
ρ	Fraction of iterations on real data	0.1
α	Curriculum transition smoothness	0.2

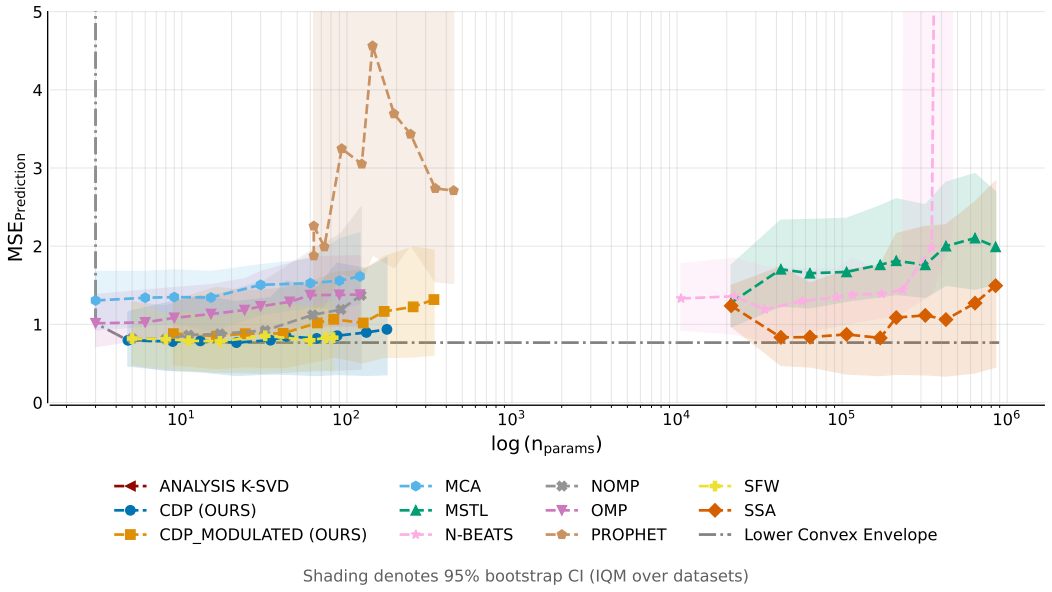

Figure 7: **Pareto frontier of model complexity versus out-of-sample forecasting error.** The x-axis indicates the parameter count on a logarithmic scale. The y-axis shows the out-of-sample forecasting MSE. CDP and CDP (Modulated) maintain their performance at the sparse end of the frontier, achieving competitive forecasting accuracy with far fewer parameters than deep learning baselines. The continuous and structured baselines (NOMP, SFW, MCA, Analysis K-SVD) are included, with Analysis K-SVD lying outside the displayed range (median test error above 28, see subsection A.8).

non-autoregressive architecture is forced to generate long-range predictions without iterative feedback in this evaluation setting.

Performance degrades when non-stationary domain shifts occur between the train and test sets, since the fitted equations extrapolate the historically identified structure without adapting to unforeseen changes. This is an inherent property of any model that performs decomposition by fitting analytical structure to historical observations rather than a deficiency specific to CDP.

A.6 SHAP Analysis

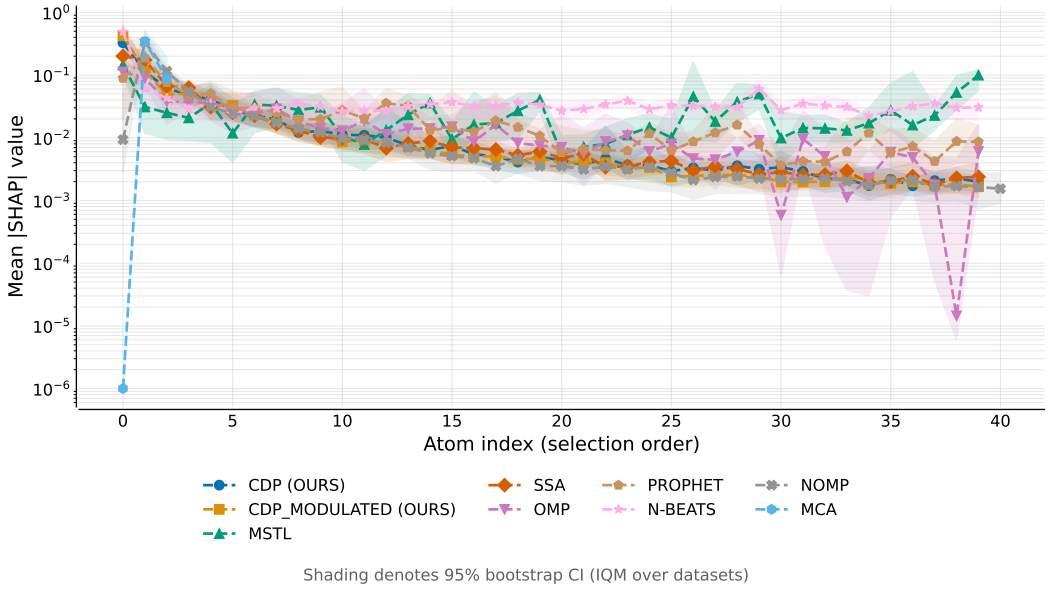

Figure 8: **Mean absolute SHAP value per atom index** in greedy selection order, averaged over all experiments and datasets (log scale). Shading denotes 95% bootstrap confidence intervals. SFW is excluded from the figure because its per-atom attributions diverge at larger budgets (see text).

We apply SHapley Additive exPlanations (SHAP) [31] to quantify how explanatory weight is distributed across the atoms of each fitted model. This analysis quantifies the *separation* of components, that is, whether each atom captures a distinct portion of the signal's explainable variance. It does not verify the *physical identity* of the components, which cannot be established without ground-truth decompositions (cf. section 5). For atom k, the Shapley value ν_k is its average marginal contribution to reconstruction quality over all possible subsets of the active atom set, approximated via Monte Carlo sampling [51]:

$$\nu_k = \sum_{S \subseteq \{1,\dots,K\} \setminus \{k\}} \frac{|S|!(K - |S| - 1)!}{K!} \left[v(S \cup \{k\}) - v(S) \right] \tag{43}$$

where $v(S)$ denotes the reduction in MSE achieved by subset S alone. All SHAP values are computed post hoc from the stored component reconstructions of the final fitted models under this single definition. Computed this way, each model's values sum exactly to the explained variance of its final fit, which makes the attributions comparable across methods. Figure 8 shows the mean absolute SHAP values as a function of the atom index in greedy selection order.

Atom Definition Across Methods: Because SHAP requires a fixed set of additive components, we define an "atom" consistently across methods as one column of each model's component reconstruction. For CDP and OMP this is unambiguous, since both methods are explicitly additive over their dictionary entries. For the structural and learned baselines we adopt the following conventions. MSTL contributes one trend atom from the LoESS-smoothed level component and $K - 1$ seasonal atoms from periods detected via the FFT of the input signal, matching the standard MSTL output

decomposition [4]. SSA contributes K rank-one atoms, each obtained by Hankel-averaging a single singular triplet $\sigma_i u_i v_i^\top$ from the trajectory matrix decomposition, sorted by descending singular value. N-BEATS contributes one atom per residual block, with the first block corresponding to the trend stack (polynomial basis) and the remaining $K-1$ blocks to the seasonality stack (Fourier basis), following the doubly-residual interpretation introduced by Oreshkin et al. [38]. Prophet contributes one trend atom and $K-1$ Fourier seasonality atoms from its native additive model. NOMP contributes one atom per refined sinusoid, and MCA contributes one component per morphological sub-dictionary (trend, sinusoidal, transient), so MCA has exactly three components at every budget. For NOMP and MCA the first stored component is the constant offset, which carries no variance after normalization and appears as a near-zero first index. In all cases the atoms sum to the model's full reconstruction, so SHAP values ν_k are directly comparable across methods. Two of the added baselines are excluded. Analysis K-SVD is not additive over an atom set and admits no comparable attribution. SFW is additive, but at larger budgets it places near-duplicate atoms with large cancelling weights, so individual atoms carry attributions that diverge by several orders of magnitude and are meaningful only jointly. Additivity alone therefore does not make components individually interpretable, and we characterize both methods through the Pareto, morphology, and cost analyses in subsection A.8 instead.

Concentration of Explanatory Weight: CDP concentrates the most explanatory weight in its leading atoms: the first CDP atom carries roughly three times the contribution of the first OMP atom (0.33 against 0.11 in the IQM over datasets), since a single continuous atom can settle at the exact frequency, amplitude, and phase of a physical component that OMP must spread across neighboring grid atoms. Along the selection sequence, the continuous methods CDP, CDP (Modulated), and NOMP decay smoothly toward the lowest tail values. NOMP matches CDP's decay, which is consistent with the morphology analysis in subsection A.8: clean per-atom concentration is a property of continuous parameter optimization, and CDP's advantage over single-family continuous methods lies in library coverage rather than concentration.

The grid-based and smoothing-based methods behave differently. OMP exhibits non-monotone contributions along the selection sequence, where an atom selected later contributes more than atoms selected earlier, reflecting the leakage corrections of the grid mismatch problem [19, 41]. MSTL shows an irregular profile with upward excursions across the full index range, and N-BEATS and Prophet maintain elevated tails, indicating that their explanatory weight remains distributed rather than isolated in a few components. SSA tracks the continuous methods closely. We note that the differences between the parametric methods are moderate rather than spanning orders of magnitude, and the qualitative distinction lies in the shape of the profiles: smooth decay for the continuous methods against irregular, non-monotone contributions for the grid-based and smoothing-based ones.

A.7 Computational Cost and Parameter-Count Definitions

Hardware: CPU-based baselines (OMP, SSA, Prophet, MSTL, MCA, Analysis K-SVD, NOMP, SFW) were run on Intel Xeon Gold and AMD EPYC nodes, while gradient-based methods (CDP, N-BEATS) were accelerated on NVIDIA RTX GPUs.

Parameter counting: Comparing parameter efficiency across disparate model classes requires an explicit definition of what is counted for each method. Table 5 defines the parameter count used in all Pareto analyses: for every method, we count exactly the set of scalars that must be stored to reproduce its decomposition or reconstruction.

Training cost: Panel (a) of Figure 9 reports CDP fitting times of approximately 10s at a single atom rising to 500s at 40 atoms. At matched atom counts this places CDP one to two orders of magnitude above OMP, a gap that reflects the structural difference in per-atom work. OMP evaluates a single inner product against a fixed dictionary at each step, while CDP runs L-BFGS over a continuous parameter manifold initialized from a scout batch of candidates. The additional cost is what enables CDP to resolve grid mismatch and recover off-grid parameters, a capability whose impact is documented in subsection A.2 on signals where OMP fails by construction. CDP remains substantially below high-parameter baselines such as MSTL, which approaches 10^3s at 10^6 parameters. Within the class of continuous pursuit algorithms, CDP is the fastest at matched atom budgets (Table 6): at $K = 40$ its median fitting time is 417s, against 1722s for NOMP and 1879s

Table 5: **Parameter-count definitions** used in all Pareto analyses. Each method's count comprises exactly the scalars required to reproduce its output.

Method	Counted parameters
CDP / CDP (Mod.)	Every continuously optimized scalar: 3–5 atom-specific parameters plus one bias per atom
OMP	Frequency index and linear coefficient per selected dictionary atom
NOMP	Refined continuous frequency plus complex amplitude per sinusoid
SFW	Continuous atom position plus weight per atom
MCA	Selected linear coefficients over the grid-discretized sub-dictionaries
Analysis K-SVD	All entries of the learned analysis operator rows
MSTL	All stored component values (trend and per-period seasonal series) needed to reproduce the decomposition
SSA	All entries of the retained singular triples (σ_i, u_i, v_i)
Prophet	All fitted coefficients (changepoints, trend, and Fourier seasonality terms)
N-BEATS	All trainable network weights

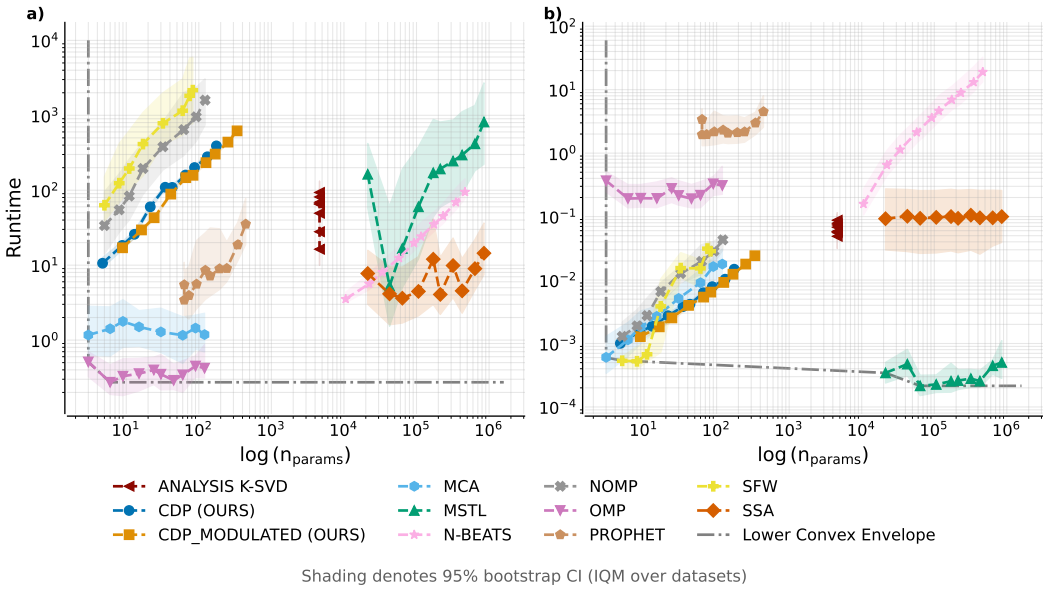

Figure 9: **Training and inference cost versus model complexity.** *(a)* Fitting time in seconds (log–log scale). *(b)* Evaluation (inference) time in seconds. Shading denotes 95% bootstrap confidence intervals over datasets.

for SFW, whose standard algorithms re-optimize all atoms in every iteration. CDP's fitting cost is therefore high relative to classical grid methods but competitive within the class of continuous methods, and it is a one-time cost that yields a closed-form model. MCA is cheap because it optimizes only linear coefficients on fixed grids, and the cost of the classical and deep baselines is shown in Figure 9.

Table 6: **Median fitting time in seconds** per atom budget K for CDP and the continuous and structured baselines, taken over 13 datasets and 10 seeds.

Method	K=1	K=2	K=3	K=5	K=10	K=20	K=30	K=40
CDP	12	20	29	63	129	223	329	417
NOMP	22	44	66	125	441	462	695	1722
SFW	46	91	141	216	451	858	1174	1879
MCA	1	1	1	2	1	1	1	1
Analysis K-SVD	95	86	60	73	76	46	30	14

Inference cost: Panel (b) shows that CDP, once fitted, consists of a small set of closed-form equations and requires only a single forward pass to evaluate, resulting in inference times of 10^{-3} to 10^{-2}s across all atom counts. This is the lowest inference time of any evaluated method. OMP requires approximately 0.3s per query because it must reproject onto its dictionary for longer prediction horizons. Prophet requires 1–3s per evaluation. N-BEATS scales from 10^{-1}s to 10^{1}s as parameter count grows, making it by far the most expensive at inference despite offering no interpretability advantage.

In practice, a decomposition model is fitted once and then used for many downstream queries. Under this regime, CDP's training cost is a one-time investment, and its near-zero inference latency makes it particularly suitable for real-time or repeated-query applications.

A.8 Baselines

This section describes the implementation of all baselines, grouped into classical and neural methods on the one hand and continuous-pursuit and structured-dictionary methods on the other, and reports the analyses that separate dictionary coverage from optimization quality. For every method, the atom budget K controls the number of components in the decomposition as described below, and the counted parameters are defined in Table 5.

MSTL [4]: MSTL decomposes the signal by iterative LoESS smoothing into one trend component and $K-1$ seasonal components, whose periods are set to the $K-1$ dominant frequencies detected via the FFT of the input signal.

SSA [59]: SSA embeds the signal into a trajectory matrix and retains the K leading singular triples. Each component is obtained by Hankel-averaging one rank-one term, and forecasts follow the linear recurrence induced by the retained subspace.

OMP [40]: OMP greedily selects K atoms from a discrete Fourier dictionary and solves for their coefficients by orthogonal projection at each step.

Prophet [54]: Prophet fits a piecewise linear trend and $K-1$ Fourier seasonalities, whose periods are set to the $K-1$ dominant FFT-detected frequencies of the input signal.

N-BEATS [38]: We use the interpretable N-BEATS configuration with one trend block (polynomial basis) and $K-1$ seasonality blocks (Fourier basis), trained for point-by-point forecasting on a sliding input window.

Newtonized OMP (NOMP) [33]: NOMP is a continuous matching pursuit for sinusoids. Each greedy step selects the strongest frequency on an oversampled FFT grid and refines it over the continuum via Newton steps. All previously selected frequencies are then refined cyclically, and all amplitudes are updated by an exact least-squares solve.

Sliding Frank-Wolfe (SFW) [17]: SFW is the standard algorithm for the continuous sparse approximation problem (BLASSO). Each iteration inserts the atom that maximizes correlation with the residual and then jointly moves all atom positions and weights by local optimization. Like NOMP, it operates on a single chosen analytic family.

Morphological Component Analysis (MCA) [50, 7]: We instantiate MCA with the same primitive families as CDP (polynomial trend, sinusoids at FFT-detected frequencies, Gaussian transients), but with all atom parameters frozen on discrete grids. Only the linear coefficients are optimized, by block-coordinate thresholding over the sub-dictionaries. MCA is therefore a *discretized ablation* of CDP, and any performance gap at matched atom families is attributable to continuous parameter optimization.

Analysis K-SVD [45]: Analysis K-SVD performs analysis-model dictionary learning with backward-greedy pursuit and eigenvector-based row updates on overlapping patches. Forecasts are produced by extrapolating the learned annihilation recurrence, protected by divergence guards. Since the co-sparse model stores the full analysis operator rather than an additive atom set, its parameter count is independent of the atom budget, which only sets the dimension of the per-patch signal subspace. Analysis K-SVD therefore appears at a single parameter count in the Pareto plots.

Results: two regimes. In the sparse regime ($K = 2$ to 5), CDP attains the best median test error of all methods across the 13 datasets, with 0.65 against 0.97 for NOMP and 0.87 for SFW at $K{=}2$, 0.62 against 1.05 and 0.84 at $K{=}3$, and 0.64 against 0.93 for both at $K{=}5$. At $K = 10$ the continuous methods are close, with 0.76 for CDP against 0.73 for NOMP and SFW. At large budgets ($K \geq 20$), the single-family continuous methods profit from exact update steps available for pure sinusoid dictionaries: NOMP reaches lower training error (0.23 against 0.25 at $K{=}20$) and SFW lower test error (0.68 against 0.89 at $K{=}40$).

Results: matched-family MCA ablation. Across all 13 datasets, CDP dominates MCA at every atom budget on training reconstruction (median 0.69 against 0.73 at $K{=}1$, 0.43 against 0.59 at $K{=}5$, and 0.16 against 0.34 at $K{=}40$) and on test error (0.62–0.94 against 1.3–1.7 across budgets). Since the atom families are matched by construction, this gap isolates the contribution of continuous parameter optimization.

Results: Analysis K-SVD. As expected for a co-sparse patch model, Analysis K-SVD excels at in-sample denoising and reaches a near-zero training residual. Its learned operator rows are however numerical vectors without closed form, and extrapolation through the learned recurrence is unstable, with a median test error above 28 across datasets even with divergence guards. It illustrates what is gained (flexibility) and lost (interpretability, forecastability) when parametric structure is abandoned.

Morphology breakdown: coverage versus optimization quality. To separate library coverage from optimizer quality we use CDP restricted to a sinusoid-only library (CDP-sine), which shares its optimizer with full CDP and its dictionary family with NOMP and SFW. We compare training reconstruction loss at a budget of 5 atoms (Table 1). First, *optimization quality*: on the strongly periodic datasets (PEMS03, PEMS04, PEMS08, Solar, Traffic), CDP-sine matches the best single-family baseline to within 1.2% on four of the five, with 3.7% on Solar as the worst case (0.167 against 0.161 for NOMP). At a matched dictionary the continuous optimizers are therefore equivalent. Second, *coverage*: the full library improves reconstruction over CDP-sine on 9 of 13 datasets, by 13–28% on datasets with switching or spiky structure (Electricity $0.593 \rightarrow 0.514$, PEMS07 $0.436 \rightarrow 0.313$, Weather $0.599 \rightarrow 0.496$). With this, full CDP attains the best reconstruction on five datasets (Electricity, ETTm2, PEMS04, PEMS07, Weather), excluding the in-sample-only Analysis K-SVD, and lies within 2.2% of NOMP on ETTh1 and ETTm1. CDP's advantage is thus a coverage effect on mixed-morphology data on top of an optimizer that matches the single-family methods on their home ground.

A.9 CDP Hyperparameters

Table 7 lists all hyperparameters used in the CDP experiments. The two configurations correspond to CDP and CDP (Modulated), which differ only in the set of atom types available during greedy selection.

Table 7: **CDP hyperparameters** used in all experiments. CDP uses all atom types excluding the ModulatedAtom, while CDP (Modulated) includes it.

Hyperparameter	Value	Description
`scout_batch_size`	10	Number of candidate atoms evaluated in parallel during truncated optimization
`scout_epochs`	5	Number of L-BFGS steps per candidate during the truncated scouting phase
`global_epochs`	100	Number of L-BFGS steps during full optimization of the selected atom
`num_global_optims`	5	Number of joint global refinement passes over all active atoms
`n_atoms`	{1,2,3,5,8,10,15,20,30,40}	Atom budgets evaluated for the Pareto frontier
`atom_types` (CDP)	All except ModulatedAtom	Atom families available for greedy selection
`atom_types` (CDP Mod.)	All	Atom families available for greedy selection
`lbfgs_lr`	1.0	Learning rate for the L-BFGS optimizer
`lbfgs_max_iter`	20	Maximum number of iterations per L-BFGS call
`lbfgs_line_search_fn`	Strong Wolfe	Line search method used by L-BFGS
`limit_loss_weight`	10^3	Weight applied to soft barrier penalty terms
`start_sharpness`	1.0	Starting κ for sharpness annealing
`end_sharpness`	50.0	End κ for sharpness annealing
`sharpness_growth_rate`	$\log(\frac{\kappa_{\text{end}}}{\kappa_{\text{start}}})$	Growth rate γ for sharpness annealing

A.10 Individual Evaluations per Dataset

The following figures report reconstruction and forecasting MSE as a function of parameter count for each individual dataset in the LTSF benchmark suite, including all baselines described in subsection A.8. These per-dataset results underlie the aggregated Pareto frontiers shown in the main paper.

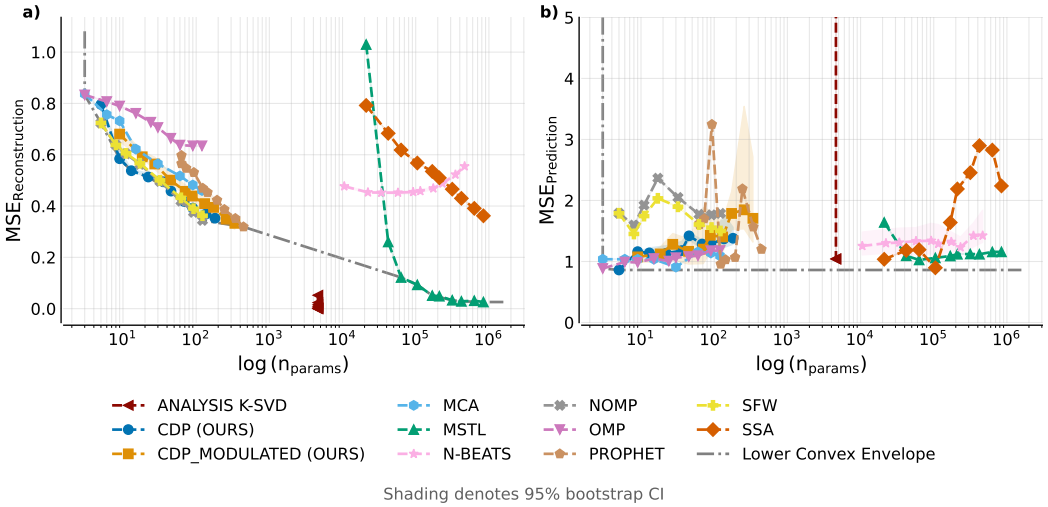

Figure 10: **Electricity.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

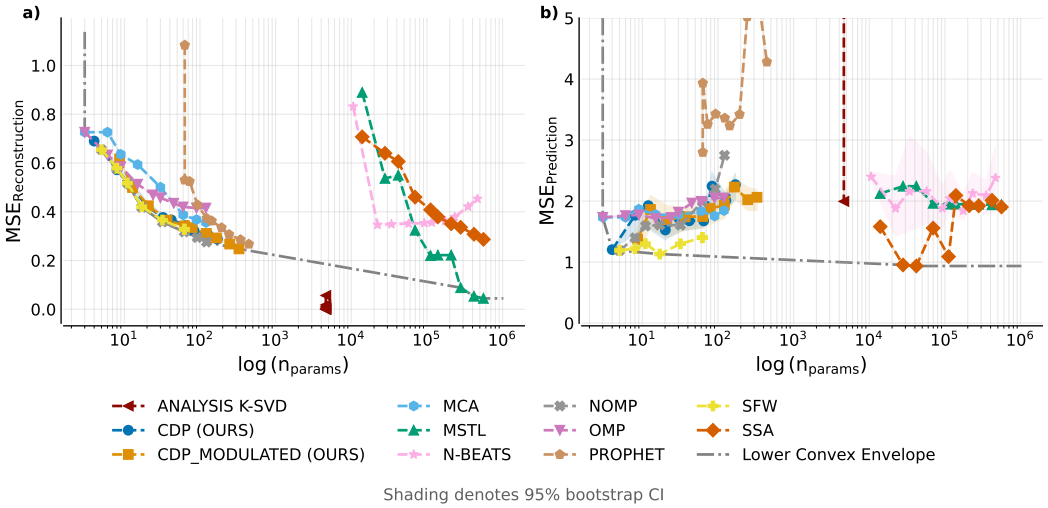

Figure 11: **ETTh1.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

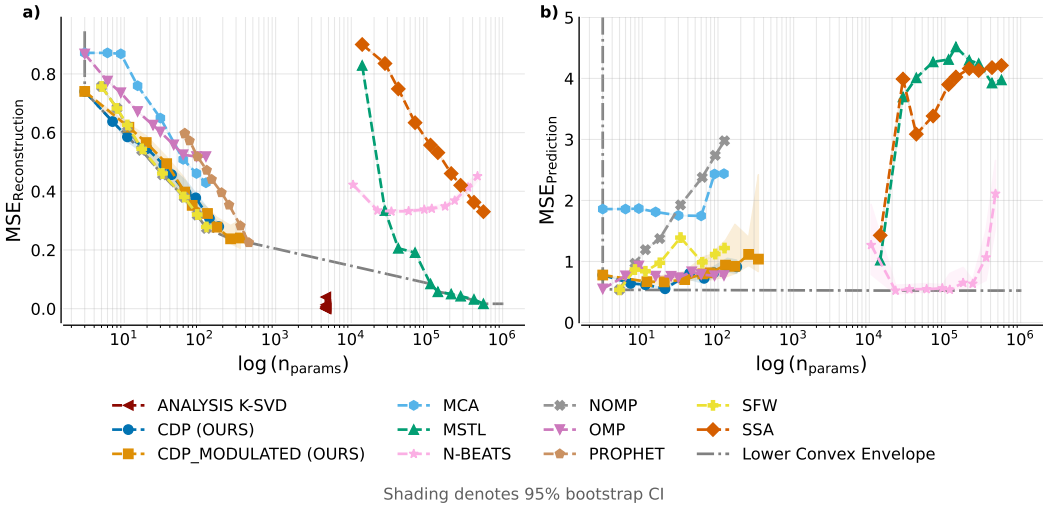

Figure 12: **ETTh2.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

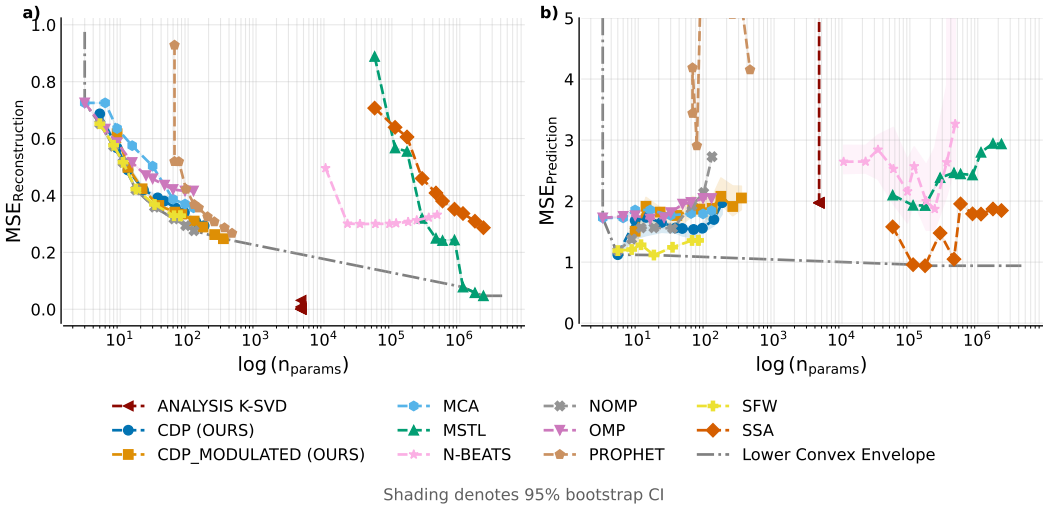

Figure 13: **ETTm1.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

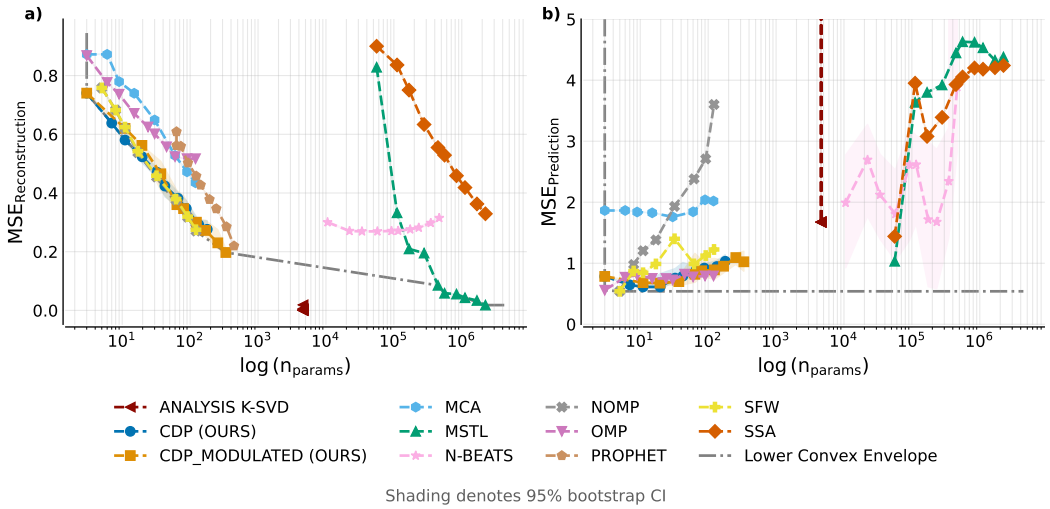

Figure 14: **ETTm2.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

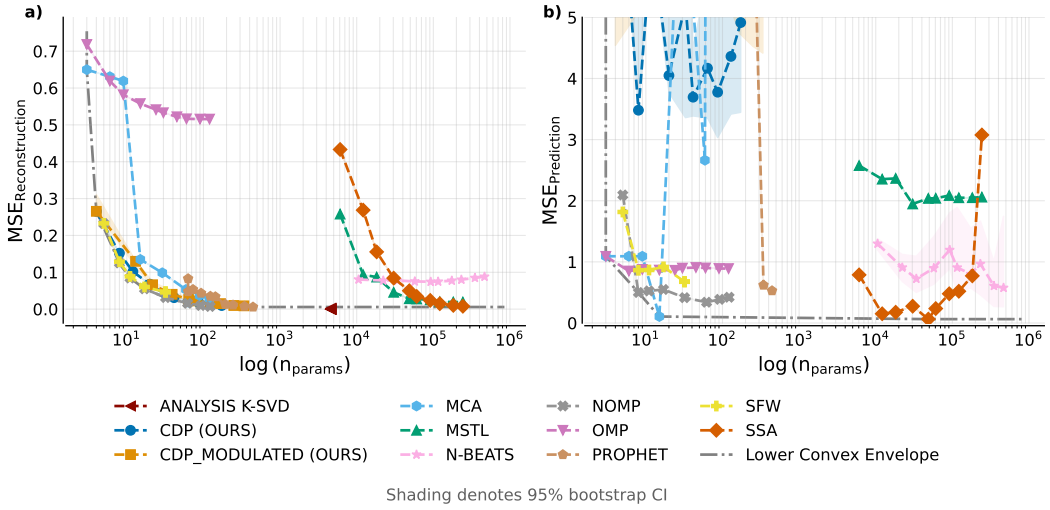

Figure 15: **Exchange.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

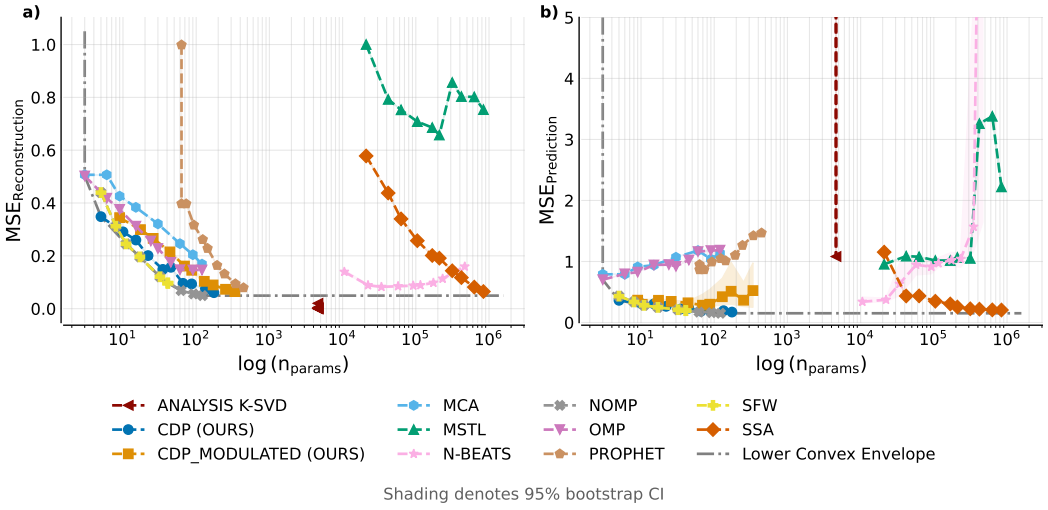

Figure 16: **PEMS03.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

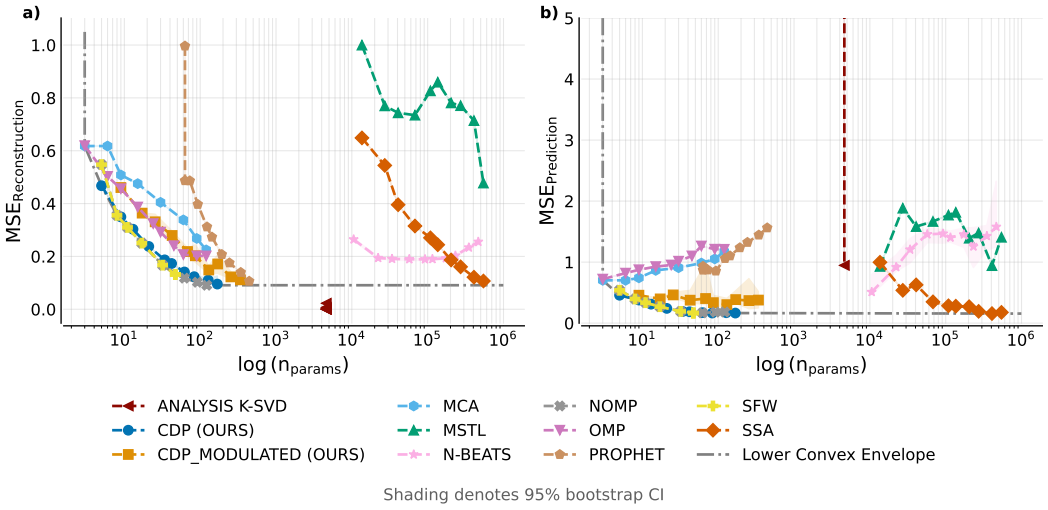

Figure 17: **PEMS04.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

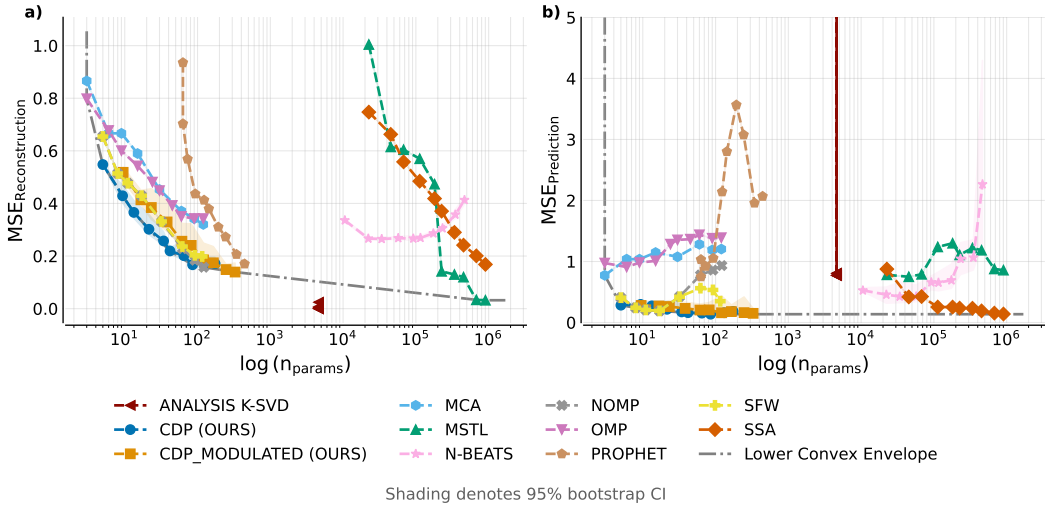

Figure 18: **PEMS07.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

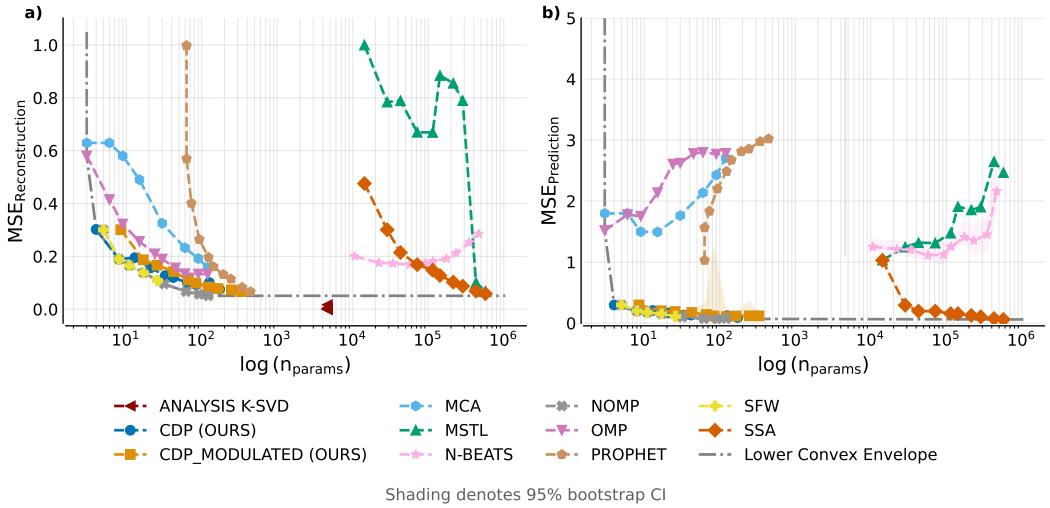

Figure 19: **PEMS08.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

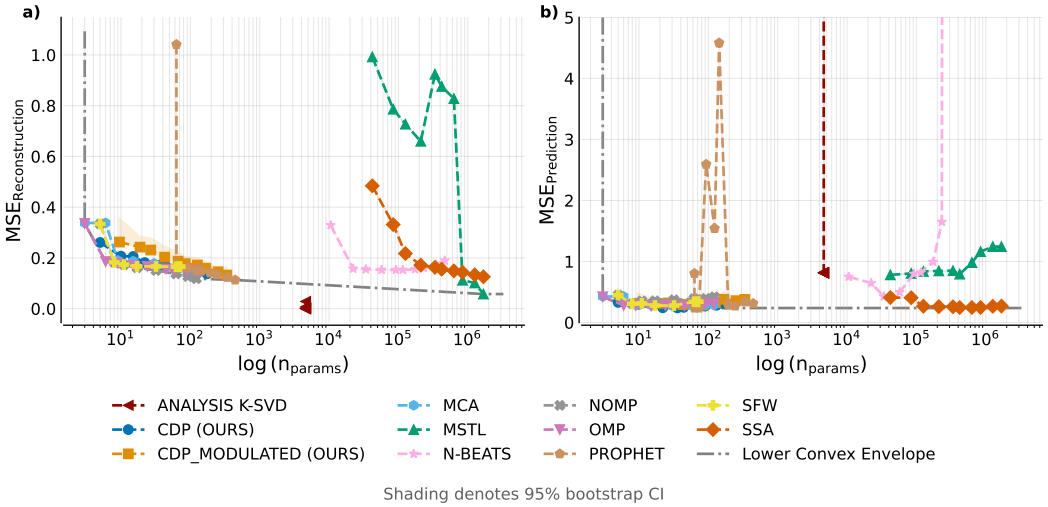

Figure 20: **Solar.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

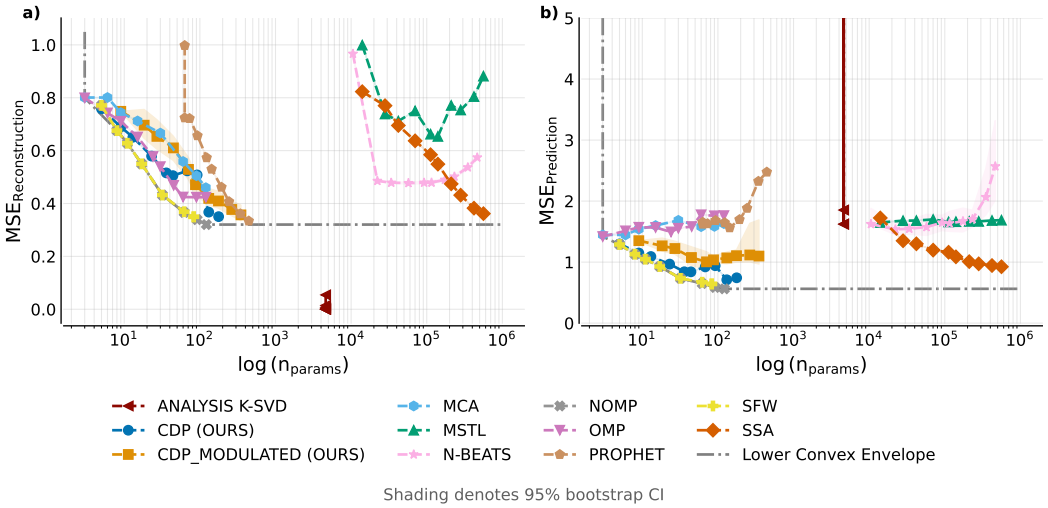

Figure 21: **Traffic.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

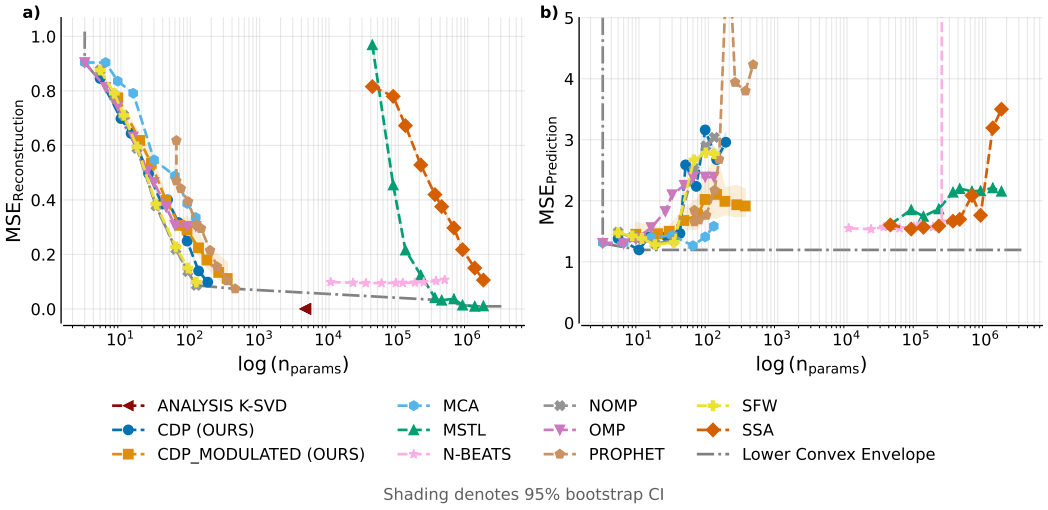

Figure 22: **Weather.** Reconstruction (a) and forecasting (b) MSE versus parameter count.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state five concrete contributions (the CDP framework, the greedy forward selection strategy, Pareto-efficient reconstruction in the sparse regime, atom-based curriculum learning, and the open-source release) and each is supported by corresponding experiments and derivations in section 3, section 4 and Appendix A. All claims are explicitly scoped to atom families contained in the library, and the univariate restriction is acknowledged in subsection 4.1 and section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A dedicated Limitations section (section 5) discusses the non-convexity of the optimization and the conditional nature of the convergence guarantee, the dependence on the predefined atom library and the risk of over-interpreting extracted atoms, the higher one-time fitting cost, and the restriction to univariate signals. The robustness of the method to seeds, time windows, and hyperparameters is analyzed in subsection A.3, and the forecasting appendix (subsection A.5) additionally discusses degradation under non-stationary domain shifts.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper states three theoretical results with assumptions and proofs. The completeness of the dictionary on $\mathbb{R}^N$ is stated in subsection 3.1 and proven in subsubsection A.1.2. The conditional convergence of the greedy loop as a weak orthogonal greedy algorithm, including the per-step condition it relies on and its numerical verification, is given in subsubsection A.1.3. The Γ-convergence of the annealed surrogate objectives, with the duty-cycle assumption stated, is proven in subsubsection A.1.4. The open status of global recovery and identifiability guarantees is stated explicitly.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The algorithm is fully specified in section 3, including the objective function (Equation 1), all atom parameterizations (Eqs. (4) to (11)), the greedy selection and refinement strategy (subsection 3.2), and the soft barrier penalty (subsubsection A.1.6). The synthetic ground-truth equations and recovered parameters are listed in Appendix A.2. Implementation details for all baselines are given in subsection A.8. The S4D architecture and all curriculum learning hyperparameters are tabulated in Table 3 and Table 4. Hardware details are given in Appendix A.7. CDP hyperparameters are provided in Appendix A.9.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case

of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The full implementation is publicly available on GitHub, with the repository linked in a footnote in section 1. All evaluation datasets are publicly available benchmarks (ETT, Electricity, Exchange, PEMS, Solar, Traffic, Weather) cited with their original sources in subsection 4.1.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The subsection 4.1 describes the dataset selection procedure and baseline configurations, with hardware details in Appendix A.7. The optimizer (L-BFGS for CDP, Adam for S4D), all architecture hyperparameters (Table 3 in Appendix A.4), all curriculum

hyperparameters (Table 4 in Appendix A.4), and the CDP hyperparameters (Appendix A.9) are reported. The train/test split follows established LTSF benchmark practice as cited in subsection 4.1.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All figures report the Interquartile Mean (IQM) together with bootstrapped 95% confidence intervals, providing a robust summary of central tendency and variability that is resistant to outliers. All experiments were done with 10 different seeds except for the evaluations of the deterministic models. This applies to the Pareto frontier plots, the curriculum learning results, the runtime figures, and the SHAP analysis. Per-dataset breakdowns underlying the aggregated figures are provided in Appendix A.10.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix A.7 states that CPU-based baselines were run on Intel Xeon Gold and AMD EPYC nodes and that gradient-based methods including CDP were accelerated on NVIDIA RTX GPUs, and it provides fitting and inference times as a function of parameter count for all methods, including a wall-clock table for the continuous and structured baselines.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper presents a signal decomposition algorithm evaluated on publicly available time series benchmarks. No human subjects, sensitive personal data, or dual-use risks are involved, and the work conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: CDP produces closed-form, physically interpretable decompositions of time series signals. This property is directly relevant to safety-critical application domains such as healthcare, autonomous systems, and industrial monitoring, where regulatory frameworks such as the EU GDPR require that model outputs be explainable. By making signal structure transparent, CDP can help practitioners demonstrate compliance with such requirements.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper proposes a signal decomposition algorithm. It does not constitute a pre-trained generative model, a scraped dataset, or any asset that poses a high risk for misuse, so no additional safeguards are required.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets used in the evaluation (ETT, Electricity, Exchange, PEMS, Solar, Traffic, Weather) are cited with their original publications in subsection 4.1. ETT is distributed under the CC BY-ND 4.0 license, Electricity originates from the UCI Machine Learning Repository (CC BY 4.0), PEMS derives from the public Caltrans Performance Measurement System, and the remaining series are distributed through the public benchmark repositories of their original publications. The S4D model and all baseline methods (MSTL, SSA, OMP, Prophet, N-BEATS, NOMP, SFW, MCA, Analysis K-SVD) are cited with their original publications. The datasets are not redistributed in the repository but can be sourced directly from the repositories accompanying their respective original papers.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

Justification: The primary new asset is the CDP codebase, which is publicly available on GitHub. The paper provides full algorithmic specification (section 3), all atom parameterizations with equations, hyperparameter tables (subsection A.4, subsection A.9), and architectural details sufficient for others to re-implement or extend the framework.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects. All experiments are conducted on publicly available sensor and measurement datasets.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve human subjects, so IRB approval is not required.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not a component of the CDP methodology. LLM tools were used only in an assistive role for writing, evaluation and visualization code, and proof drafting. All AI-assisted content was reviewed and verified by the authors.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.