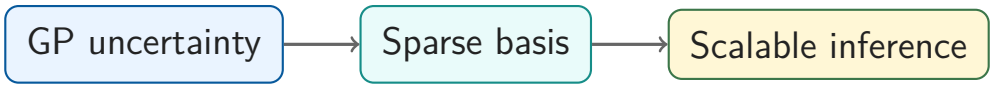
SIKA-GP

Accelerating Gaussian Process Inference with Sparse Inducing Kernel Approximations



Wenyuan Zhao, Rui Tuo, Chao Tian

Texas A&M University

Code: `https://github.com/warrenzha/sika-gp`

Motivation: uncertainty is useful, but GP inference is expensive

Why GPs?

Calibrated uncertainty

Bayesian predictions

Classic bottleneck

Exact GP

$\mathcal{O}(N^3)$

Kernel matrix inverse

BDL bottleneck

Repeated cost

D features $\times$ layers $\times$ MC samples

Main question

Can we use a dense inducing grid for expressive GPs, **but touch only a few basis functions for each input**?

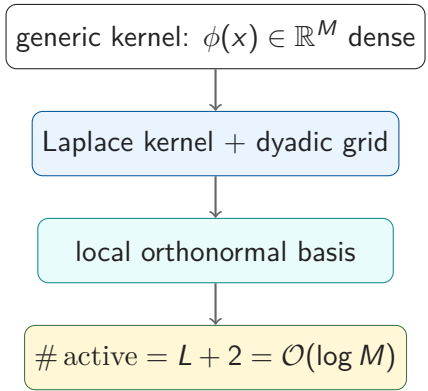
Dense forward: M coordinates → sparsify coordinates → SIKA-GP: $L + 2$ active, $M = 2^L + 1$

Key starting point: inducing GP as a Bayesian forward layer

$$f(x) = K(x,U)\,[K(U,U)]^{-1} f(U)$$
$$= \underbrace{K(x,U)P^{-\top}}_{\phi(x)}\ \underbrace{P^{-1}f(U)}_{w}, \quad w \sim \mathcal{N}(0,I)$$

Interpretation

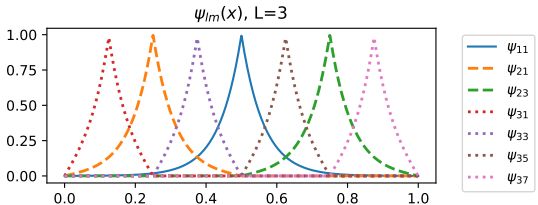
A finite inducing GP can be evaluated like a BNN layer: activation $\phi(x)$ with random Bayesian weight w.



SIKA-GP idea

Use sparse coordinates for the same fixed-grid inducing representation: the gain comes from evaluating the finite-dimensional span sparsely.

High-level method: dyadic local basis functions



Example for $L = 3$: basis functions are local, so most activations are zero.

Construction intuition

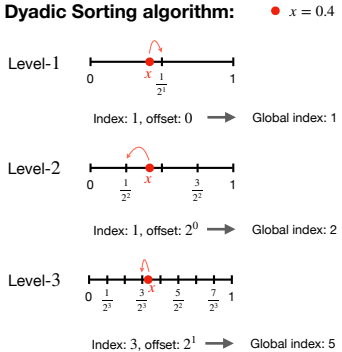
- Dyadic inducing levels:

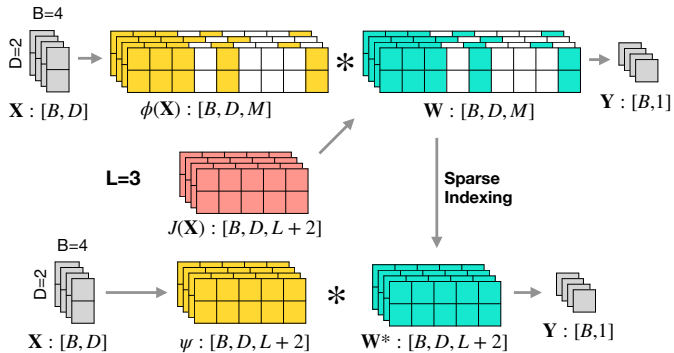
$$U_\ell = \{m2^{-\ell} : m \text{ odd}\}.$$

- Laplace/Markov structure gives compactly supported basis functions.
- For each level ℓ, only the basis around the closest dyadic point can be active.

2 global + 1 per level $= L + 2$ active, while $M = 2^L + 1$

Tensorized Sparse Indexing: making sparsity GPU-friendly





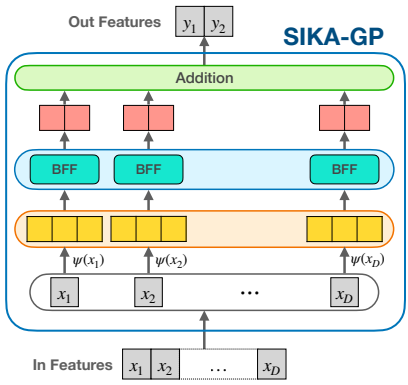
1. Index
compute active indices $J(X)$ for the whole mini-batch

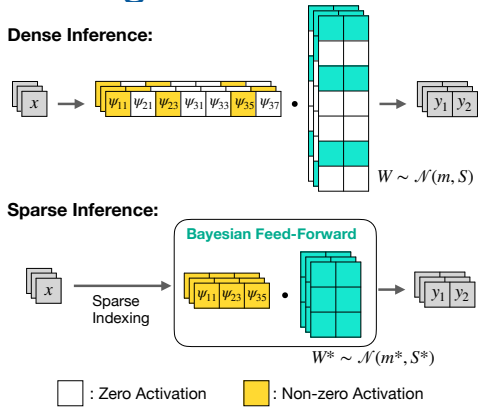
2. Gather
squeeze W into $W^* = W[J(X)]$

3. Forward
multiply tensors of size $B \times D \times (L+2)$

Dense: $\mathcal{O}(SDM)$ $\Longrightarrow$ SIKA-GP: $\mathcal{O}(SD \log M)$

Drop-in block for Bayesian deep learning



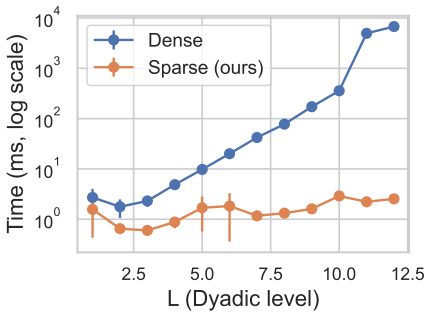


VI learns mean-field Gaussian weights;
Prediction averages MC samples from the variational posterior.

Drop-in replacement: dense GP forward $\rightarrow$ sparse SIKA forward.

What changes in practice? Complexity and empirical highlights

Model	Training	Inference
SVGP	$\mathcal{O}(SDNM + NM^2 + M^3)$	$\mathcal{O}(SNM^2)$
KISS-GP	$\mathcal{O}(SDNM + DM^{3/D})$	$\mathcal{O}(SDM^{1+1/D})$
DAK	$\mathcal{O}(SDNM)$	$\mathcal{O}(SDM)$
SIKA-GP	$\mathcal{O}(SDN \log M + DM)$	$\mathcal{O}(SD \log M)$



DGP regression: comparable RMSE/NLPD; up to 7× faster speed.

CIFAR-100 DKL: best accuracy, ECE/NLL, about 2× faster than SVDKL.

CLINC-150 LM: best AUROC, about 3× faster than SVDKL.

Summary

SIKA-GP in one sentence

SIKA-GP makes inducing-kernel GP inference fast by evaluating an inducing-grid GP in a sparse dyadic basis.

Active basis size
$L + 2$
instead of $M = 2^L + 1$

- **Structured sparsity:** Laplace kernel + dyadic inducing grid.
- **Efficient execution:** Tensorized Sparse Indexing gathers only active weights.
- **BDL compatibility:** works in DGP, DKL, and transformer-based model settings.

Limitations and future work

Laplace kernel and fixed grid are structured choices. Future directions include adaptive/hybrid dyadic grids and broader kernel classes.

Thank you! `https://github.com/warrenzha/sika-gp`