

Bayesian Optimisation for Bayesian Evidence

(BOBE)

Nathan Cohen

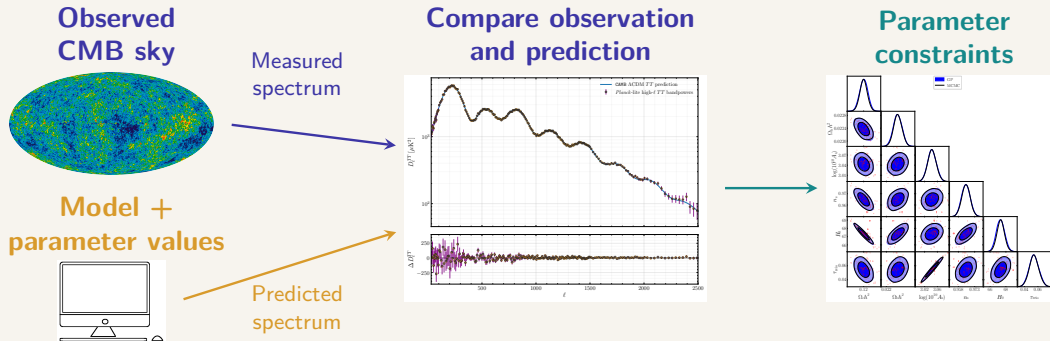
with Jan Hamann and Ameek Malhotra

School of Physics, University of New South Wales

PPC 2026 / 4th Gordon Godfrey Workshop on Astroparticle Physics

01 September 2026

From observation to parameter constraints



Parameter inference: Where can the parameters of a **chosen model** live?

CMB sky map: NASA / WMAP Science Team.

The Likelihood

Gaussian likelihood, with fixed covariance

$$-2 \ln \mathcal{L}(D \mid \boldsymbol{\theta}, M) = [\mathbf{d} - \mathbf{t}(\boldsymbol{\theta}, M)]^T \mathbf{C}^{-1} [\mathbf{d} - \mathbf{t}(\boldsymbol{\theta}, M)] + \text{constant}$$

$\mathbf{C}$ encodes uncertainties and correlations.

Bayes' law

$$\underbrace{p(\boldsymbol{\theta} \mid D, M)}_{\text{posterior}} \propto \underbrace{\mathcal{L}(D \mid \boldsymbol{\theta}, M)}_{\text{likelihood}} \underbrace{\pi(\boldsymbol{\theta} \mid M)}_{\text{prior}}$$

Parameter inference needs relative posterior density.

Generating the **prediction** can dominate the cost.

Which model?

The same data can be interpreted under different models:

$$M_i \in \{\Lambda\text{CDM}, \Lambda\text{CDM} + \Omega_k, w\text{CDM}\}$$

$$p(\boldsymbol{\theta}_i \mid D, M_i) = \frac{\mathcal{L}(D \mid \boldsymbol{\theta}_i, M_i) \pi(\boldsymbol{\theta}_i \mid M_i)}{Z_i}$$

$$Z_i = p(D \mid M_i) = \int_{\Theta_i} \mathcal{L}(D \mid \boldsymbol{\theta}_i, M_i) \pi(\boldsymbol{\theta}_i \mid M_i) d\boldsymbol{\theta}_i$$

$$\underbrace{\frac{p(M_0 \mid D)}{p(M_1 \mid D)}}_{\text{posterior odds}} = \underbrace{\frac{Z_0}{Z_1}}_{\text{Bayes factor}} \times \underbrace{\frac{p(M_0)}{p(M_1)}}_{\text{prior odds}}$$

Occam's razor: extra flexibility must earn its place.

Standard samplers can solve both inference problems

MCMC

$$p(\boldsymbol{\theta} \mid D, M)$$

Samples the parameter posterior.

Nested sampling

$$Z \quad \text{and} \quad p(\boldsymbol{\theta} \mid D, M)$$

Estimates evidence and the posterior.

Evidence requires a global integral for every model, making model selection more expensive than parameter inference.

For both inference tasks, runtime is approximately

$$\underbrace{t_{\text{total}}}_{\text{total runtime}} \simeq \underbrace{N_{\text{eval}}}_{\text{evaluations}} \left(\underbrace{t_{\text{like}}}_{\text{likelihood cost}} + \underbrace{t_{\text{over}}}_{\text{overhead}} \right)$$

Cheap likelihoods make stochastic sampling effective

$$t_{\text{total}} \simeq \underbrace{N_{\text{eval}}}_{\text{very large}} \left(\underbrace{t_{\text{like}}}_{\text{small}} + \underbrace{t_{\text{over}}}_{\text{very small}} \right)$$

Each point is cheap to choose

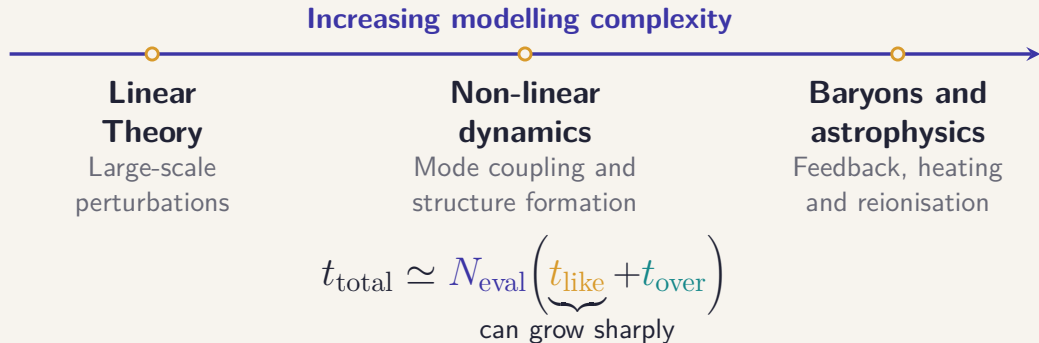
Almost no computation is spent deciding where to evaluate next.

Many points are required

Stochastic exploration instead uses a very large number of samples.

When t_{like} is small, this is an excellent computational trade.

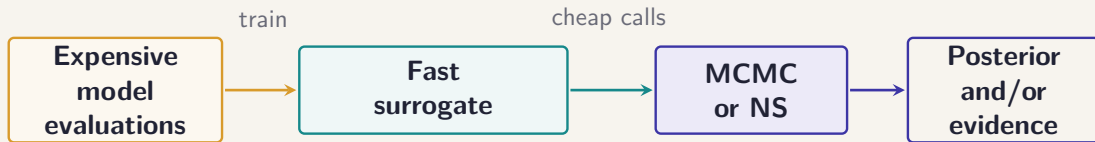
Higher precision demands more accurate predictions



How do we keep parameter inference and model comparison affordable?

Emulators make expensive functions cheap to sample

Learn a cheap surrogate from expensive evaluations.



The sampler evaluates the surrogate, rather than repeatedly calling the expensive model.

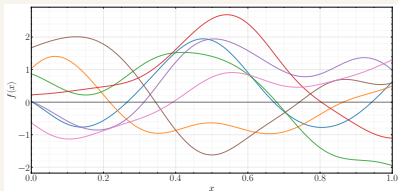
Which emulator can learn from **few expensive evaluations** of an **unknown objective function** and quantify **what it does not know**?

A Gaussian process is a distribution over functions

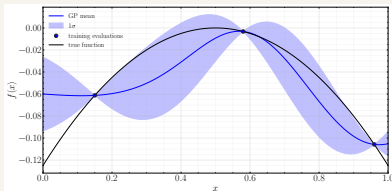
$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

A distribution over functions, defined by its mean and covariance.

Before evaluating: GP prior



After 3 evaluations: GP posterior



Non-parametric

No fixed functional shape.

Predictive uncertainty

Can be propagated into inference.

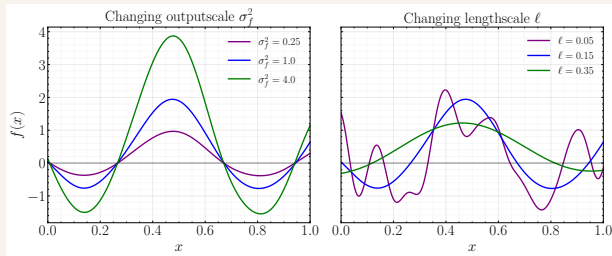
Data-efficient

Uses prior structure to learn from few evaluations.

The kernel controls how functions can vary

$$k_{\text{RBF}}(x, x') = \sigma_f^2 \exp \left[-\frac{(x - x')^2}{2\ell^2} \right]$$

The kernel encodes prior assumptions about function structure.



Amplitude σ_f^2

Sets the typical vertical scale of function variation.

Lengthscale ℓ

Sets how far information propagates and how rapidly the function can vary.

The GP lets the data decide

Learn the kernel's **lengthscales** and **amplitude** from the evaluations.

GP marginal likelihood (MLL)

$$\log p(\mathbf{y} \mid X, \phi) = \underbrace{-\frac{1}{2} \mathbf{r}^\top K_\phi^{-1} \mathbf{r}}_{\text{fit to evaluations}} \underbrace{-\frac{1}{2} \log |K_\phi|}_{\text{Occam penalty}} - \underbrace{\frac{n}{2} \log(2\pi)}_{\text{Normalisation}}$$

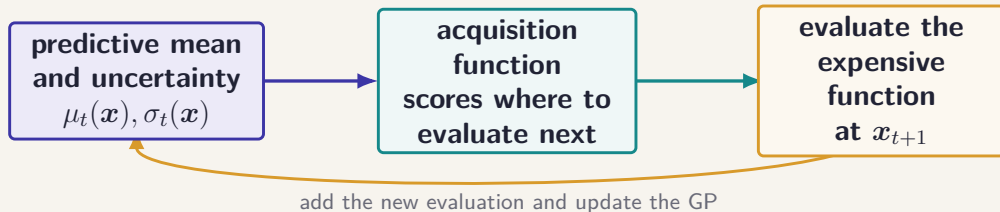
$$\hat{\phi} = \arg \max_{\phi} \log p(\mathbf{y} \mid X, \phi)$$

Like a human agent, the GP **updates its expectations** as data arrive.

Bayesian optimisation turns predictions into decisions

$$\mathbf{x}_{t+1} = \arg \max_x / \min \mathcal{A}_t(\mathbf{x}; \mu_t, \sigma_t)$$

Use the GP to score candidate locations, then evaluate the most promising.

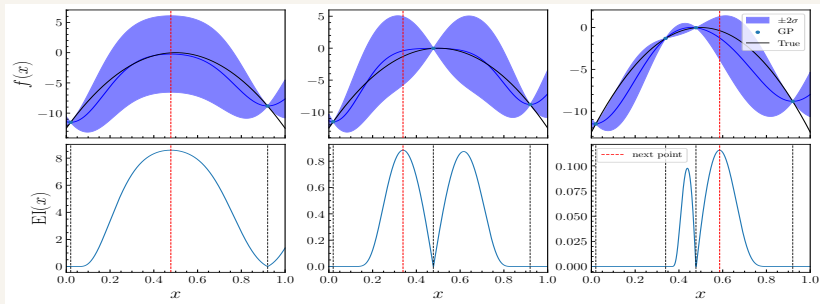


The acquisition function is a **decision rule**. Different rules encode different objectives.

Acquisition functions encode different goals

Expected improvement

$$\mathcal{A}_{\text{EI}}(\mathbf{x}) = \mathbb{E}[\max(f(\mathbf{x}) - f_{\text{best}}, 0)]$$



Asks “where is the **maximum?**”

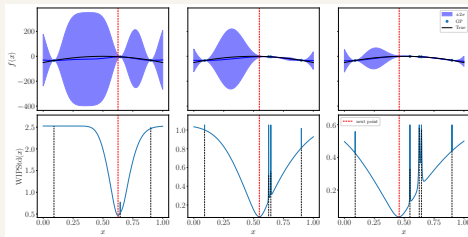
WIPStd reduces uncertainty in the evidence

Weighted Integrated log-Posterior Standard Deviation

$$\text{WIPStd}_t(\mathbf{x}') = \int_V d\mathbf{x} \mathcal{P}_t(\mathbf{x}) \sigma_{t+1}(\mathbf{x} | \mathbf{x}')$$

posterior weight $\times$ predicted remaining GP uncertainty if the next point is at $\mathbf{x}'$

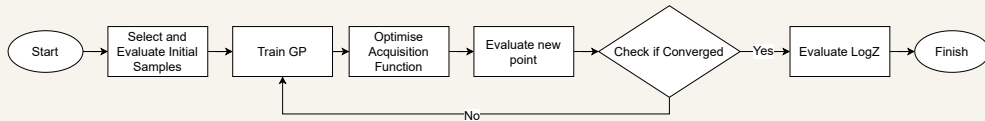
$$\mathbf{x}_{t+1} = \arg \min_{\mathbf{x}'} \text{WIPStd}_t(\mathbf{x}')$$



The minimum selects the next evaluation; note the changing vertical scale.

Asks “which next evaluation most reduces uncertainty in $\ln \mathcal{Z}$?”

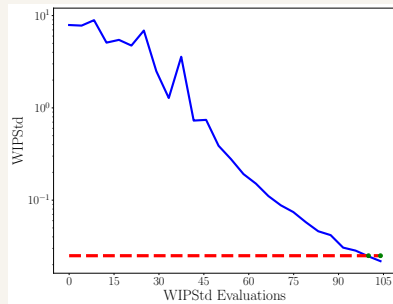
BOBE stops at the target evidence precision



The acquisition function also tells us **when to stop**.

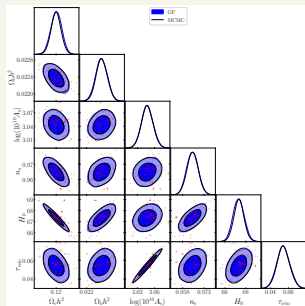
When WIPStd remains below the accuracy threshold, further likelihood evaluations make little difference to the inferred evidence.

Final check: run nested sampling on the GP mean and $\mu(x) \pm \sigma(x)$ surfaces.



BOBE matches standard samplers with fewer calls

Λ CDM-lite (6D)

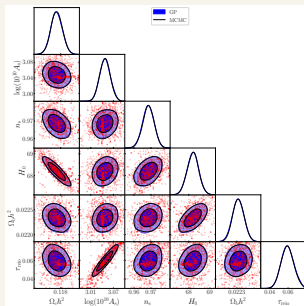


116 evaluations

-519.84 ± 0.15 vs. -520.18 ± 0.32

3 min vs. 10 h

Λ CDM (15D)

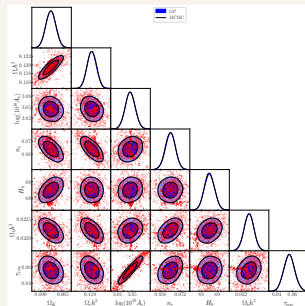


1213 evaluations

-5529.84 ± 0.28 vs. -5529.65 ± 0.45

54 min vs. ~1 day

Λ CDM + Ω_k (16D)



1713 evaluations

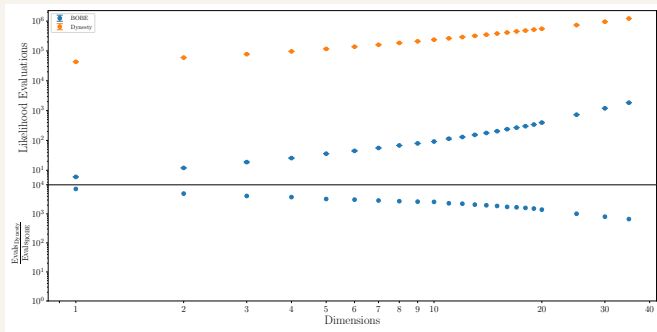
-5529.93 ± 0.33 vs. -5529.80 ± 0.37

2.5 h vs. >1 day

In every case, $\ln \mathcal{Z}_{\text{BOBE}}$ agrees with $\ln \mathcal{Z}_{\text{PolyChord}}$ within uncertainty. Wall-time gains grow with **likelihood cost**.

GP training sets BOBE's high-dimensional limit

Gaussian test functions: BOBE vs Dynesty



Fewer expensive calls

Around 10^3 fewer evaluations than direct Dynesty on these Gaussian tests.

The remaining bottleneck

Dense GP factorisation costs

$$\mathcal{O}(N_{\text{train}}^3)$$

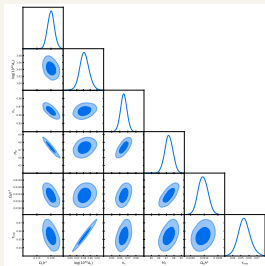
Larger training sets make each refit more expensive.

Can we make the same likelihood **easier for the GP to learn?**

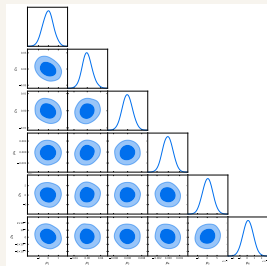
Rotation aligns the GP with posterior degeneracies

A fixed length scale per parameter cannot efficiently follow tilted degeneracy directions.

Original parameters



Rotated and scaled parameters



Two ways to use this geometry

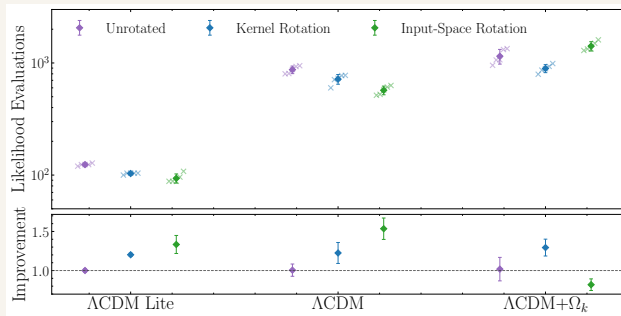
Rotate the kernel metric

Only the GP's notion of distance changes. The likelihood, prior and acquisition domain remain unchanged.

Rotate the input space

All components see simpler coordinates, and the prior volume shrinks. This is more powerful, but assumes more.

Rotation reduces evaluations—but not uniformly



All rotated runs remain consistent with the corresponding PolyChord evidence and unrotated posterior constraints.

Kernel rotation

A modest, consistent reduction in evaluations across all three likelihoods.

Input-space rotation

Reduction in evaluations in two tested cases; more in Λ CDM + Ω_k case.

Marginalisation trades dimension for evaluation cost

Each time BOBE requests a cosmological point, integrate over its nuisance parameters.

$$\mathcal{L}_{\text{marg}}(D \mid \boldsymbol{\theta}) = \int \mathcal{L}(D \mid \boldsymbol{\theta}, \boldsymbol{\eta}) \pi(\boldsymbol{\eta} \mid \boldsymbol{\theta}) \mathrm{d}\boldsymbol{\eta}$$

$\boldsymbol{\theta}$: cosmological parameters

$\boldsymbol{\eta}$: nuisance parameters

Train on the returned marginal likelihood

The GP only sees the lower-dimensional cosmological parameter space.

More work in every likelihood call

Every requested $\boldsymbol{\theta}$ requires an internal integral over $\boldsymbol{\eta}$.

A complementary strategy for nuisance-heavy likelihoods.

21cmFAST: an expensive-likelihood test

8

astrophysical parameters

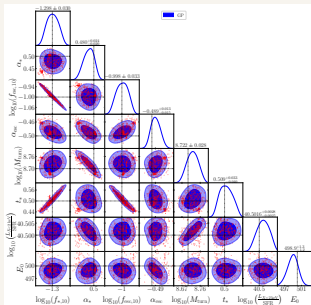
~ 40 minutes

per direct evaluation

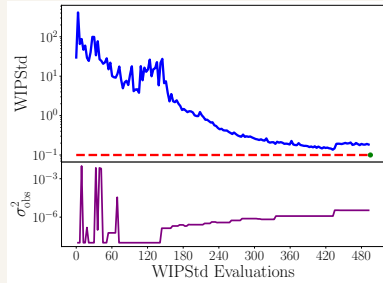
500

direct evaluations

Posterior constraints



Training history



$\Delta \ln Z \simeq 0.3$ after approximately **130 hours wall time**

Feasibility run capped at 500 calls; like-for-like 21cmEMU v3 validation is ongoing.

BOBE makes expensive model comparison practical

Up to 10^3 fewer
direct likelihood evaluations

on tested problems, while recovering consistent posterior constraints and evidence estimates

Evidence-aware

WIPStd directs evaluations
toward uncertainty that
matters for $\ln \mathcal{Z}$.

Built-in stopping

The same acquisition
function supplies a natural
convergence diagnostic.

Extensible

Rotation and nuisance
marginalisation can extend
BOBE to harder parameter
spaces.

Next: online geometry · nuisance marginalisation · validated 21 cm benchmark

github.com/BOBE-Sampler/BOBE Arxiv:2601.11150