LEARNING WHAT TO EVALUATE: CORRELATION-AWARE DECOUPLING FOR MULTIOBJECTIVE BAYESIAN OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multiobjective Bayesian optimization (MOBO) with Gaussian process (GP) surrogates is a sample efficient approach to solving multiobjective optimization problems. In MOBO, a Bayesian decision theoretic acquisition function guides the adaptive selection of new candidate inputs, on which objectives and constraints are evaluated to update the surrogate model sequentially. Existing approaches maintain independent GP models for the objectives and constraints, with new observations evaluating all objectives and constraints in a coupled fashion. However, the objectives and constraints often contain inherent correlations which, if exploited, can enable *decoupled* evaluations where only a subset of them are evaluated at each round. We present a new approach that leverages a multitask GP model to jointly learn all objectives and constraints, and propose a *total correlation* metric that enables identifying an *optimal* subset of objectives and constraints to be evaluated at every round, even under uniform evaluation costs. Theoretically, we show that our acquisition policy is asymptotically consistent despite decoupling and that our proposed decoupled subset selection rule maximizes the expected posterior entropy reduction about unevaluated tasks under mild conditions. Empirically, we show that our approach outperforms coupled and decoupled baselines in the state of the art.

1 INTRODUCTION

Many scientific and engineering design problems require optimizing several expensive objectives while satisfying expensive constraints. Standard MOBO methods, with GP [29] surrogates, usually assume that evaluating a design returns all objectives and constraints together. In many applications, however, separate simulators, tests, operating conditions, or analyses produce these quantities; their evaluations can therefore be "decoupled". This creates a scalar-oracle budget: the algorithm may choose both a design and an objective or constraint oracle to query. For instance, aircraft design uses computational aerodynamic simulators to evaluate performance at multiple operating conditions, each evaluated independently [8]. Neural architecture search (NAS) requires simultaneously optimizing predictive accuracy and model latency on edge devices [22], which are separately tested.

We consider optimizing K objectives subject to C constraints, all independently evaluable:

$$\begin{aligned} \max_{\mathbf{x}\in\mathcal{X}} \; & \{f_1(\mathbf{x}), \ldots, f_K(\mathbf{x})\} \\ \text{s.t. } & c_i(\mathbf{x}) = 0, \; i \in \mathcal{E} \\ & c_i(\mathbf{x}) \geq 0, \; i \in \mathcal{I}, \end{aligned} \tag{1}$$

where $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^d$ is the design variable, $f_k : \mathcal{X} \to \mathbb{R}$, $\forall k = 1, \ldots, K$ and $c_i : \mathcal{X} \to \mathbb{R}$, the objectives and constraints, respectively, are expensive zeroth-order oracles. $\mathcal{E}$ and $\mathcal{I}$ index the equality and inequality constraints, respectively, and $C = |\mathcal{E} \cup \mathcal{I}|$. $\mathcal{X}$ is the domain of f and c. We seek a *Pareto* optimal set of solutions that "Pareto dominates" all other points. Let $\boldsymbol{f}(\mathbf{x}) = [f_1(\mathbf{x}), \ldots, f_K(\mathbf{x})]^\top$; if $\boldsymbol{f}(\mathbf{x})$ Pareto dominates $\boldsymbol{f}(\mathbf{x}')$, then $f_k(\mathbf{x}) \geq f_k(\mathbf{x}')$, $\forall k = 1, \ldots, K$ and $\exists k \in [K]$ such that $f_k(\mathbf{x}) > f_k(\mathbf{x}')$. We write Pareto dominance as $\boldsymbol{f}(\mathbf{x}) \succ \boldsymbol{f}(\mathbf{x}')$. The set $\mathcal{Y}^* = \{\boldsymbol{f}(\mathbf{x}) \; : \; \nexists \mathbf{x}' \in \mathcal{X} : \boldsymbol{f}(\mathbf{x}') \succ \boldsymbol{f}(\mathbf{x})\}$ is the *Pareto frontier*, and the set $\mathcal{X}^* = \{\mathbf{x} \in \mathcal{X} \; : \; \boldsymbol{f}(\mathbf{x}) \in \mathcal{Y}^*\}$ is the *Pareto set*.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The hypervolume (HV) of a Pareto frontier is the $K-$dimensional Lebesgue measure λ of the region dominated by $\mathcal{Y}^*$ and bounded below by a reference point $\mathbf{r} \in \mathbb{R}^K$. This monotonically increasing metric is typically used to assess convergence in MOBO.

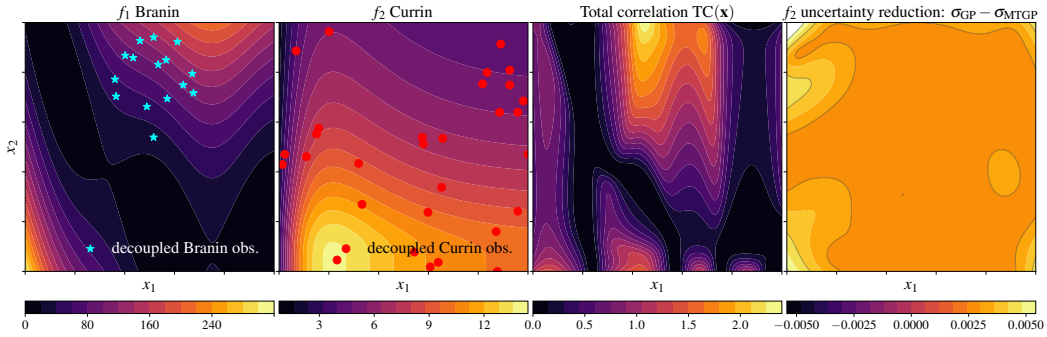


Figure 1: The Branin and Currin test functions (left and middle left panels) and their total correlation $\mathrm{TC}(\mathbf{x})$ (middle right panel). Knowing the $\mathrm{TC}(\mathbf{x})$ structure allows us to decouple the two functions' evaluations – f_1 is evaluated at 15 points ($\star$, left) with *high* $\mathrm{TC}(\mathbf{x})$ and f_2 is evaluated on 30 points ($\bullet$, middle left) with *low* $\mathrm{TC}(\mathbf{x})$. Then, we fit two independent GPs using all 45 points for both f_1 and f_2 (90 evaluations total) and one multitask GP (MTGP) with only the 45 *decoupled* evaluations. The right panel shows the MTGP's reduction of posterior uncertainty in f_2. Thus, appropriately chosen observation sites allow MTGP to transfer information between tasks, which can improve sample efficiency in MOBO.

In MOBO, a decision-theoretic rule (the acquisition function) typically selects the next evaluation point [23, 31, 16, 24]. When objectives or constraints are statistically correlated, a query to one oracle can reduce posterior uncertainty about others. Evaluating every oracle at every selected design may then be redundant. The key question is therefore not only where to evaluate, but also which subset of objectives and constraints to evaluate at each design. Multitask Gaussian process (MTGP) models [5] offer an effective way to jointly learn correlated tasks. Figure 1 illustrates the mechanism we exploit: regions with high "total correlation" (defined in Section 3) indicate where observing one task can inform another. In the Branin-Currin example, a multitask posterior trained on selectively decoupled observations has lower uncertainty for one objective than independent, "single task", models unable to transfer information across tasks.

Traditionally, independent GPs emulate objectives and constraints in MOBO. For instance, the expected hypervolume improvement (EHVI) [14], its stochastic extension [12], and predictive entropy search (PESMO) [18] all assume independent GP priors across objectives and constraints. This independence assumption permits simple product factorizations yielding closed-form acquisition functions. However, MTGPs complicate acquisition function construction because cross-output correlations prevent product factorizations possible with independent GPs. Standard acquisitions thus become high-dimensional integrals over correlated Gaussians (truncated Gaussians under constraints), preventing their widespread use in MOBO. Despite this limitation, MTGPs have been used in single-objective BO to show that sharing structure across tasks can substantially reduce the evaluations needed for good solutions [34]. In this work, we combine MTGPs with a total-correlation-based metric to propose an optimal way to decouple objective and constraint evaluations in MOBO. A constrained MOBO method is needed to exploit inter-task correlation to decouple objective and constraint evaluations *even under uniform oracle evaluation costs*, while avoiding intractable MTGP acquisition computations.

Related work. Recently, Renganathan & Carlson [30] extended Thompson sampling [37] to constrained MOBO as Pareto optimal Thompson sampling (qPOTS). qPOTS is attractive here because it replaces acquisition integration over correlated predictive distributions with optimization over draws of posterior sample paths. Drawing joint sample paths from an MTGP is straightforward, whereas adapting EHVI or PESMO style acquisitions to correlated objective/constraint posteriors leads to difficult high-dimensional integrals. Therefore, we build on qPOTS [30] in this work.

MOBO with decoupled evaluations has been addressed before, albeit rarely. These include the entropy-based approaches Predictive Entropy Search (PESMO) [18], Pareto Frontier Entropy Search

(PFES) [33], and more recently the Joint Entropy Search (JES) [38]. Other approaches include the hypervolume knowledge gradient (HVKG) [11] and another variant of knowledge gradient [7]. Outside of these two categories, FlexiBO [20] uses the uncertainty reduction in the hypervolume to accommodate decoupled evaluations. However, all of these methods share the following limiting characteristics. (1) They require the cost of evaluating each objective to be different for the decoupling to have a value proposition. Under uniform evaluation costs, they may not provide benefit. (2) They do not consider the decoupling of constraint evaluations. (3) They do not exploit the inherent correlation between objectives; instead, decoupling is achieved as a consequence of cost weighting. (4) They are not directly compatible with MTGPs, which would necessitate appropriate modifications to their acquisition function computation. On the other hand, our proposed approach overcomes all of these limitations, while enabling decoupling *under uniform evaluation costs* – this crucial benefit broadens decoupling to scenarios where cost is not a factor or unknown. Our take is that, as long as objectives and constraints are correlated and decoupled evaluations are feasible, it should be exploited for better sample efficiency in MOBO regardless of any difference in evaluation costs. Our contributions are summarized as follows.

1. **Method.** We introduce q`POTS-DOE`: "batch Pareto optimal Thompson sampling with decoupled oracle evaluations" for constrained MOBO with partial objective and constraint observations under uniform scalar-oracle costs.
2. **Decoupling rule.** We propose a two-stage decoupling mechanism: a total correlation (Section 3.1) gate to decide whether decoupling is warranted at a candidate design, followed by a Gaussian mutual-information criterion to choose the optimal subset of objectives and constraints to evaluate.
3. **Theory.** We show that despite decoupling our acquisition policy is asymptotically consistent, and that the chosen subset of decoupled evaluations is optimal.
4. **Empirics.** We demonstrate improved hypervolume performance and evaluation efficiency on synthetic and real-world constrained MOBO benchmarks, including ablations over the TC threshold and batch size.

The rest of the paper is organized as follows. In Section 2 we briefly review preliminaries. Our overall method is summarized in Section 3. Our theoretical and empirical results are presented in Section 4 and Section 5, respectively. We provide concluding remarks in Section 6. Our software implementation is openly available at `Link` (and currently attached with this submission).

2 PRELIMINARIES

Multitask Gaussian process models. A review of "single task" GP models is provided in Appendix A.2. For MTGP models, the output at each input $\mathbf{x} \in \mathbb{R}^d$ is K-dimensional. We define the vector-valued process

$$\mathbf{Y}(\mathbf{x}) \;=\; \big[Y_1(\mathbf{x}),\, \ldots,\, Y_K(\mathbf{x})\big]^\top \in \mathbb{R}^K,$$

where $Y_k(\mathbf{x})$ denotes the latent function for task $k \in \{1, \ldots, K\}$. An MTGP assumes that any finite collection of stacked outputs is jointly Gaussian [5, 1]. We write $\mathbf{Y}(\mathbf{x}) \sim \mathrm{MTGP}\big(\boldsymbol{\mu}(\mathbf{x}),\, \mathbf{K}(\mathbf{x}, \mathbf{x}')\big)$, where $\boldsymbol{\mu}(\mathbf{x}) \in \mathbb{R}^K$ is the mean vector and $\mathbf{K}(\mathbf{x}, \mathbf{x}') \;=\; \mathrm{cov}\big(\mathbf{Y}(\mathbf{x}), \mathbf{Y}(\mathbf{x}')\big) \in \mathbb{R}^{K\times K}$ is a *matrix-valued* kernel that encodes both input similarity and inter-task correlations [1]. Given inputs $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top$, define the stacked output vector

$$\mathbf{y} \;=\; \big[\mathbf{Y}(\mathbf{x}_1)^\top, \ldots, \mathbf{Y}(\mathbf{x}_n)^\top\big]^\top \in \mathbb{R}^{nK}.$$

Then $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, where $\mathbf{m} = [\boldsymbol{\mu}(\mathbf{x}_1)^\top, \ldots, \boldsymbol{\mu}(\mathbf{x}_n)^\top]^\top$ and $\Sigma \in \mathbb{R}^{nK\times nK}$ is a block covariance matrix with $K \times K$ blocks $\Sigma_{ij} \;=\; \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j),\; i, j \in \{1, \ldots, n\}$. A common observation model is additive Gaussian noise per input: $\tilde{\mathbf{y}}_i = \mathbf{Y}(\mathbf{x}_i) + \boldsymbol{\varepsilon}_i,\; \boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \Sigma_\varepsilon)$, with $\Sigma_\varepsilon \in \mathbb{R}^{K\times K}$ often chosen diagonal (independent task noise) or full (correlated noise). Stacking all observations gives

$$\tilde{\mathbf{y}} \sim \mathcal{N}\big(\mathbf{m},\; \Sigma + \mathbf{I}_n \otimes \Sigma_\varepsilon\big),$$

where $\otimes$ denotes the Kronecker product. For a test input $\mathbf{x}_*$, define $\Sigma_{X*} \;=\; \big[\mathbf{K}(\mathbf{x}_i, \mathbf{x}_*)\big]_{i=1}^{n} \in \mathbb{R}^{nK\times K},\; \Sigma_{*X} = \Sigma_{X*}^\top,\; \Sigma_{**} = \mathbf{K}(\mathbf{x}_*, \mathbf{x}_*) \in \mathbb{R}^{K\times K}$. Conditioning the joint Gaussian on observa-

tions yields [5, 1]:

$$\mathbf{Y}(\mathbf{x}_*) \mid \mathbf{X}, \tilde{\mathbf{y}} \;\sim\; \mathcal{N}\Big(\mathbf{m}(\mathbf{x}_*) + \Sigma_{*X}\big(\Sigma + \mathbf{I}_n \otimes \Sigma_\varepsilon\big)^{-1}\big(\tilde{\mathbf{y}} - \mathbf{m}\big), \\ \Sigma_{**} - \Sigma_{*X}\big(\Sigma + \mathbf{I}_n \otimes \Sigma_\varepsilon\big)^{-1}\Sigma_{X*}\Big). \tag{2}$$

A central modeling choice is the form of the matrix-valued kernel $\mathbf{K}(\mathbf{x}, \mathbf{x}')$. A widely used parameterization is the *intrinsic coregionalization model* (ICM), $\mathbf{K}(\mathbf{x}, \mathbf{x}') = \mathbf{B}\, k(\mathbf{x}, \mathbf{x}')$, where $k(\cdot, \cdot)$ is a scalar kernel over inputs and $\mathbf{B} \in \mathbb{R}^{K \times K}$ is a positive semidefinite *coregionalization* matrix capturing task relatedness [5, 1]. Under the ICM kernel, the stacked covariance takes the Kronecker form $\Sigma = \Sigma_{\mathbf{x}} \otimes \mathbf{B}$, $(\Sigma_{\mathbf{x}})_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, which can be exploited computationally when n and K are large. More generally, the *linear model of coregionalization* (LMC) uses a sum of separable components, allowing multiple latent correlation structures across tasks and inputs [1, 36]. This flexibility often improves transfer across related tasks by sharing statistical strength while preserving task-specific variation.

3 qPOTS−DOE: BATCH PARETO OPTIMAL THOMPSON SAMPLING WITH DECOUPLED ORACLE EVALUATIONS

Instead of fitting K independent posterior GP models in [30], we fit a single K-task MTGP model for all objectives. We directly address the constrained setting; but review the unconstrained setting in Appendix A.2.1.

In addition to K objectives, let there be a total of C constraints $\{c_1, \ldots, c_C\}$, where $\mathcal{I} \cup \mathcal{E} = [C]$. We fit a single $(K + C)$-task multitask GP model for all objectives and constraints. Let

$$\mathbf{Y}_{1:K+C}(\mathbf{x}) \equiv [Y_1(\mathbf{x}) \quad \cdots \quad Y_K(\mathbf{x}) \quad Y_{K+1}(\mathbf{x}) \quad \cdots \quad Y_{K+C}(\mathbf{x})]^\top,$$

where, for each $i \in [C]$, Y_{K+i} denotes the posterior task corresponding to constraint c_i. The joint multitask posterior, with hyperparameters Ω, is

$$\mathbf{Y}_{1:K+C}(\cdot) \mid D_n^{1:K+C}, \Omega \sim \mathrm{MTGP}(\boldsymbol{\mu}_n(\cdot), \boldsymbol{\Sigma}_n(\cdot, \cdot)),$$

with

$$[\boldsymbol{\Sigma}_n(\mathbf{x}, \mathbf{x}')]_{ij} = \mathrm{cov}\left(Y_i(\mathbf{x}), Y_j(\mathbf{x}') \mid D_n^{1:K+C}, \Omega\right), \qquad i, j \in [K + C].$$

Hence, the objectives and constraints are learned jointly, and the posterior cross-covariances between objective tasks and constraint tasks are retained. For a posterior sample path, define the sampled feasible region as

$$\mathcal{X}_{\mathbf{Y}}^{\mathrm{feas}} = \left\{\mathbf{x} \in \mathcal{X} : \mathbb{1}\left\{\bigcap_{i \in \mathcal{E}} Y_{K+i}(\mathbf{x}) = 0\right\} \mathbb{1}\left\{\bigcap_{i \in \mathcal{I}} Y_{K+i}(\mathbf{x}) \geq 0\right\} = 1\right\}.$$

Then, we choose points according to

$$p_{X^*}(\mathbf{x}) = \int \delta\left(\mathbf{x} - \underset{\mathbf{x} \in \mathcal{X}_{\mathbf{Y}}^{\mathrm{feas}}}{\arg\max}\{Y_1(\mathbf{x}), \ldots, Y_K(\mathbf{x})\}\right) p\big(\mathbf{Y}_{1:K+C} \mid \mathcal{D}_n^{1:K+C}\big)\, d\mathbf{Y}_{1:K+C},$$

where δ is the Dirac's delta function and $d\mathbf{Y}_{1:K+C} \equiv dY_1 \cdots dY_K\, dY_{K+1} \cdots dY_{K+C}$. Equivalently, we draw one joint sample path from the $(K + C)$-task multitask GP posterior,

$$\mathbf{Y}_{1:K+C}(\cdot, \omega) = \boldsymbol{\mu}_n(\cdot) + \boldsymbol{\Sigma}_n^{1/2}(\cdot)\, Z(\omega), \qquad Z(\omega) \sim \mathcal{N}(0, I),$$

filter the sampled design space by the same feasibility indicators, and solve

$$X^* = \underset{x \in \mathcal{X}_\omega^{\mathrm{feas}}}{\arg\max}\{Y_1(\mathbf{x}, \omega), \ldots, Y_K(\mathbf{x}, \omega)\},$$

where

$$\mathcal{X}_\omega^{\mathrm{feas}} = \left\{\mathbf{x} \in \mathcal{X} : \mathbb{1}\left\{\bigcap_{i \in E} Y_{K+i}(\mathbf{x}, \omega) = 0\right\} \mathbb{1}\left\{\bigcap_{i \in I} Y_{K+i}(\mathbf{x}, \omega) \geq 0\right\} = 1\right\}.$$

This cheap multiobjective optimization problem is solved via evolutionary approaches e.g., NSGA-II [13]. Let $\gamma(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ denote the Euclidean distance between two points in $\mathcal{X}_\omega^{\text{feas}}$. Then, a batch of q acquisitions is chosen per

$$\begin{aligned}
\mathbf{x}_{n+1} &= \arg\max_{\mathbf{x}^* \in X^*} \min_{\mathbf{x}_i \in \mathbf{X}} \gamma(\mathbf{x}^*, \mathbf{x}_i), \\
\mathbf{x}_{n+2} &= \arg\max_{\mathbf{x}^* \in X^*} \min_{\mathbf{x}_i \in \mathbf{X} \cup \{\mathbf{x}_{n+1}\}} \gamma(\mathbf{x}^*, \mathbf{x}_i), \\
&\vdots \\
\mathbf{x}_{n+q} &= \arg\max_{\mathbf{x}^* \in X^*} \min_{\mathbf{x}_i \in \mathbf{X} \cup \{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+q-1}\}} \gamma(\mathbf{x}^*, \mathbf{x}_i).
\end{aligned} \tag{3}$$

Once a batch of q points is chosen per (3), we then decide if evaluations must be decoupled, followed by choosing the optimal subset of them to evaluate – this is presented next.

3.1 The total correlation metric $\mathrm{TC}(\mathbf{x})$ thresholding for decoupling

A key advantage of MTGPs is their ability to model statistical dependence across tasks. After conditioning on data $\mathcal{D}$, the predictive distribution at a fixed input $\mathbf{x}$ is Gaussian, $\mathbf{Y}(\mathbf{x}) \mid \mathcal{D} \sim \mathcal{N}\big(\boldsymbol{\mu}(\mathbf{x}),\, \Sigma(\mathbf{x})\big),\ \Sigma(\mathbf{x}) \in \mathbb{R}^{K \times K}$, where $\Sigma(\mathbf{x})$ is the *predictive* covariance matrix across tasks at $\mathbf{x}$. Then, a standard normalized measure of pairwise coupling is the (posterior) correlation coefficient

$$\rho_{k\ell}(\mathbf{x}) \;=\; \frac{\Sigma_{k\ell}(\mathbf{x})}{\sqrt{\Sigma_{kk}(\mathbf{x})\,\Sigma_{\ell\ell}(\mathbf{x})}}, \qquad k \neq \ell,$$

and the full correlation matrix can be written compactly as

$$\mathbf{R}(\mathbf{x}) \;=\; \mathbf{D}(\mathbf{x})^{-\frac{1}{2}}\,\Sigma(\mathbf{x})\,\mathbf{D}(\mathbf{x})^{-\frac{1}{2}}, \quad \mathbf{D}(\mathbf{x}) = \mathrm{diag}\big(\Sigma(\mathbf{x})\big). \tag{4}$$

While pairwise correlations $\rho_{k\ell}(\mathbf{x})$ are intuitive, they do not provide a single scalar summary of the *overall* dependence among all K tasks at $\mathbf{x}$ – we overcome this with a "total correlation" metric (also called *multi-information*) [39, 9], defined as

$$\mathrm{TC}(\mathbf{x}) \;\triangleq\; D_{\mathrm{KL}}\Big(p\big(\mathbf{Y}(\mathbf{x}) \mid \mathcal{D}\big) \,\Big\|\, \prod_{k=1}^{K} p\big(Y_k(\mathbf{x}) \mid \mathcal{D}\big)\Big), \tag{5}$$

which measures how far the joint predictive distribution across tasks deviates from the product of its (task-wise) marginals. Equivalently, in entropy form, $\mathrm{TC}(\mathbf{x}) \;=\; \sum_{k=1}^{K} H\big(Y_k(\mathbf{x}) \mid \mathcal{D}\big) \;-\; H\big(\mathbf{Y}(\mathbf{x}) \mid \mathcal{D}\big)$, so $\mathrm{TC}(\mathbf{x}) \geq 0$, and $\mathrm{TC}(\mathbf{x}) = 0$ if and only if the tasks are independent under the predictive distribution at $\mathbf{x}$ [9]. Because $\mathbf{Y}$ is multivariate Gaussian, $\mathrm{TC}(\mathbf{x})$ has a simple closed form in terms of $\Sigma(\mathbf{x})$:

$$\mathrm{TC}(\mathbf{x}) \;=\; \frac{1}{2}\log\frac{\big|\mathrm{diag}\big(\Sigma(\mathbf{x})\big)\big|}{\big|\Sigma(\mathbf{x})\big|} \;=\; \frac{1}{2}\log\frac{\prod_{k=1}^{K}\Sigma_{kk}(\mathbf{x})}{\big|\Sigma(\mathbf{x})\big|} = -\frac{1}{2}\log\big|\mathbf{R}(\mathbf{x})\big|,$$

where the last equality uses the identity $|\Sigma(\mathbf{x})| = |\mathbf{D}(\mathbf{x})|\,|\mathbf{R}(\mathbf{x})|$. This highlights that (for Gaussian predictive distributions) total correlation depends only on the correlation structure, not on the marginal scales.

Finally, we use $\mathrm{TC}(\mathbf{x}) \geq \tau$ as a gate to decide if decoupling is warranted at a candidate $\mathbf{x}$ or not, where $\tau > 0$ is a threshold the user must specify. As $\tau \to \infty$, qPOTS-DOE reduces to qPOTS [30]. Note that τ represents the amount of information (in nats) that can be transferred at $\mathbf{x}$ between tasks; thus when $\mathrm{TC}(\mathbf{x}) \geq \tau$, $\mathrm{TC}(\mathbf{x})$ is an upper bound on the maximum possible information transfer from any decoupled subset. In practice we observe (and as shown in the ablation in Figure 5) that choosing $\tau \in [10^{-6}, 10^{0}]$ has negligible impact on performance. Once a candidate $\mathbf{x}$ is chosen, and the total correlation gate ($\mathrm{TC}(\mathbf{x}) > \tau$) passes, an optimal subset of objectives and constraints for decoupled evaluations is chosen by solving an inner optimization problem – this is described next.

3.2 OPTIMAL SUBSET FOR DECOUPLED EVALUATION

Without loss of generality, we introduce an inner combinatorial optimization problem that selects a subset of M objectives out of K.[1] Let $S \subseteq \mathcal{K} \triangleq \{1, \dots, K\}$ and $\bar{S} \triangleq \mathcal{K} \setminus S$, so that the selected objectives $\mathbf{Y}_S(\mathbf{x})$ are maximally informative about the remaining objectives $\mathbf{Y}_{\bar{S}}(\mathbf{x})$. With a cardinality budget $|S| = M$ (that is, we can evaluate M objectives per candidate $\mathbf{x}$), we define the inner problem as

$$S^\star(\mathbf{x}) \;\in\; \arg\max_{S \subseteq \mathcal{K} \,:\, |S| = M} I\big(\mathbf{Y}_S(\mathbf{x}); \mathbf{Y}_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}\big). \tag{6}$$

This criterion optimally selects M objectives that are (i) strongly coupled with the rest while (ii) avoiding redundancy among selected objectives, since redundant tasks provide little incremental information. We later show how to determine the cardinality budget M. Using the identity $I(A; B) = H(A) + H(B) - H(A, B)$ and the Gaussian differential entropy $H(\mathbf{Z}) = \frac{1}{2} \log\big((2\pi e)^m |\Sigma_{\mathbf{Z}}|\big)$ for $\mathbf{Z} \in \mathbb{R}^m$ [9], we obtain

$$\begin{aligned} I\big(\mathbf{Y}_S(\mathbf{x}); \mathbf{Y}_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}\big) &= H\big(\mathbf{Y}_S(\mathbf{x}) \mid \mathcal{D}\big) + H\big(\mathbf{Y}_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}\big) - H\big(\mathbf{Y}(\mathbf{x}) \mid \mathcal{D}\big) \\ &= \frac{1}{2} \log \frac{\big|\Sigma_{SS}(\mathbf{x})\big| \, \big|\Sigma_{\bar{S}\bar{S}}(\mathbf{x})\big|}{\big|\Sigma(\mathbf{x})\big|}. \end{aligned} \tag{7}$$

Here $\Sigma_{SS}(\mathbf{x})$ and $\Sigma_{\bar{S}\bar{S}}(\mathbf{x})$ are principal submatrices of $\Sigma(\mathbf{x})$ indexed by S and $\bar{S}$, respectively. Since $\log |\Sigma(\mathbf{x})|$ does not depend on S, maximizing (7) is equivalent to

$$S^\star(\mathbf{x}) \in \arg\max_{S \subseteq \mathcal{K}:\, |S| = M} \Big\{ \log \big|\Sigma_{SS}(\mathbf{x})\big| + \log \big|\Sigma_{\bar{S}\bar{S}}(\mathbf{x})\big| \Big\},$$

which can be interpreted as finding a partition (of fixed size) that makes both sides "internally uncertain" while being as inter-dependent as possible. In Theorem A.3, we show that this rule selects the *best* M out of K oracles to evaluate, in terms of the one-step expected entropy reduction.

Relationship to total correlation. Define the within-subset total correlations $\mathrm{TC}_S(\mathbf{x})$ and $\mathrm{TC}_{\bar{S}}(\mathbf{x})$ analogously but restricted to objectives in S and $\bar{S}$. Using $\mathrm{TC} = \sum_k H(Y_k) - H(\mathbf{Y})$ and elementary cancellations,

$$I\big(\mathbf{Y}_S(\mathbf{x}); \mathbf{Y}_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}\big) = \mathrm{TC}(\mathbf{x}) - \mathrm{TC}_S(\mathbf{x}) - \mathrm{TC}_{\bar{S}}(\mathbf{x}),$$

so maximizing cross-subset mutual information (Equation (6)) favors partitions where the *between-group* dependence is large relative to the *within-group* dependence.

The inner problem (6) is combinatorial: it requires first choosing a subset size M, and then choosing the optimal subset among $\binom{K}{M}$ combinations. Related mutual-information subset selection problems are known to be NP-hard in GP settings [25]. Accordingly, we consider two practical approaches:

(1) Exact maximization (small K). When K is small (typical in multiobjective optimization), we can evaluate (7) for all subsets of size M, for any M, and select the maximizer. Each evaluation requires computing log-determinants of principal submatrices; this can be done stably via Cholesky factorization. If $\Sigma_{SS}(\mathbf{x}) = \mathbf{L}\mathbf{L}^\top$, $\log \big|\Sigma_{SS}(\mathbf{x})\big| = 2 \sum_i \log L_{ii}$, and similarly for $\Sigma_{\bar{S}\bar{S}}(\mathbf{x})$.

(2) Approximate maximization (moderate/large K). Define the set function

$$f_{\mathbf{x}}(S) \;\triangleq\; I\big(\mathbf{Y}_S(\mathbf{x}); \mathbf{Y}_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}\big).$$

Because $f_{\mathbf{x}}(S) = H(\mathbf{Y}_S) + H(\mathbf{Y}_{\bar{S}}) - H(\mathbf{Y})$ and entropy is submodular, $f_{\mathbf{x}}$ is a (typically non-monotone) *symmetric submodular* function of S [9, 25]. This enables the use of standard submodular maximization heuristics and approximation algorithms [15, 6]. A simple and effective heuristic under the cardinality constraint $|S| = M$ is a *greedy* construction followed by *swap* local search (starting with $S = \varnothing$):

$$\begin{aligned} \Delta_{\mathbf{x}}(s \mid S) &\triangleq f_{\mathbf{x}}(S \cup \{s\}) - f_{\mathbf{x}}(S), \\ s^\star &\in \arg\max_{s \in \mathcal{K} \setminus S} \Delta_{\mathbf{x}}(s \mid S), \end{aligned} \tag{8}$$

iterated until $|S| = M$, and then refined by swapping $i \in S$ with $j \in \bar{S}$ whenever $f_{\mathbf{x}}(S \setminus \{i\} \cup \{j\})$ improves. Then, we repeat Equation (8) for $M = 1, \dots, K - 1$ to pick M that gives the smallest

[1] In the constrained setting, we would select from $K + C$ instead.

$f_{\mathbf{x}}(S)$. More sophisticated choices (e.g., randomized methods with constant-factor guarantees for non-monotone submodular maximization) can be used when K is large [15, 6]. Our experiments use the exact maximization approach, but we perform an ablation on exact vs approximate maximization in Figure 13, showing that under moderate $K + C$, the accuracy difference is not substantial.

If objective evaluations at $\mathbf{x}$ are noisy, one can compute the mutual information between the *noisy* observed subset and the *latent* unobserved objectives by replacing the observed covariance block with $\Sigma_{SS}(\mathbf{x}) + \Sigma_{\varepsilon,SS}$ in (7), where $\Sigma_{\varepsilon,SS}$ is the noise covariance for the selected objectives. This makes the subset selection explicitly account for heteroscedastic or task-dependent noise. All the steps involved are summarized in Algorithm 1.

4 Theoretical results

We prove two key results: (1) that qPOTS−DOE is asymptotically consistent and (2) under a finite sample $\mathcal{D}_n$, any subset of M oracle evaluations chosen per Equation (6) maximizes the one-step expected entropy reduction about the omitted oracles. W.l.o.g, we consider K objectives without constraints. The informal assumptions and results are presented below; but detailed versions, with proofs, are in Appendix A.3.

Assumptions. We assume that the design space $\mathcal{X}$ is compact and that the K latent oracles are generated by the MTGP model used by the algorithm, with continuous sample paths. Observations may be decoupled: at round i, the algorithm queries a design $\mathbf{x}_i$ but observes only a subset $S_i \subseteq [K]$, with independent Gaussian observation noise of finite, nonzero variance for each observed oracle. The only exploration condition is a no-starvation requirement: every oracle must be directly evaluated on design points that become dense in $\mathcal{X}$. Equivalently, decoupling is allowed, but no task may be permanently ignored – this condition can be enforced by a lightweight safeguard such as occasional coupled evaluations (see line 16 in Algorithm 1).

Theorem 4.1 (Informal: consistency of qPOTS−DOE)**.** *Under Assumption A.1, decoupled evaluations do not prevent posterior consistency. In particular, if every task is directly evaluated on a set of design points that becomes dense in $\mathcal{X}$, then the multitask GP posterior learned from the partially observed data $\mathcal{D}_n$ concentrates on the true task values. Equivalently, for any fixed design point $\mathbf{x} \in \mathcal{X}$ and every task $k \in [K]$, the posterior mean converges to the true value $Y_{0,k}(\mathbf{x})$ and the posterior uncertainty vanishes as $n \to \infty$.*

Theorem 4.2 (Informal: finite-sample value of decoupling)**.** *Fix the current data $\mathcal{D}_n$ and a candidate design $\mathbf{x}$. If only $M < K$ oracles can be evaluated at $\mathbf{x}$, then the qPOTS−DOE subset rule*

$$S_n^\star(\mathbf{x}) \in \arg\max_{S \subseteq [K]:\, |S| = M} I(Y_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}_n)$$

selects the M oracles whose observation gives the largest expected one-step reduction in posterior uncertainty about the unevaluated oracles $Y_{\bar{S}}(\mathbf{x})$.

5 Empirical results

We demonstrate our methodology on several synthetic and real-world benchmarks varying in input (d), output (K), and constraint (C) dimensionality. We focus on evaluating high $K + C$ dimensionality since the benefit of decoupling is the primary objective of this paper; high d scaling is focus of future work. We compare qPOTS−DOE against qPOTS [30] (no decoupling) and qLogNEHVI [12, 2] (a classical baseline, also with no decoupling) and four decoupled baselines: HVKG [11] (our main competitor), PESMO [18], MESMO [4], and JESMO [38], all under uniform costs.

For **synthetic experiments**, we consider the Branin-Currin, ZDT1/2, DTLZ4/5, Gaussian mixture model (GMM), BNH [19], OSY [28], and CONSTR [19] test functions. Additional test functions are included in the supplementary material. For **real-world experiments**, we consider the Vehicle Safety problem [35], the Penicillin production problem [26], the Car side-impact problem [21], and the disc brake problem [35]. Further details on all experiments, including the reference point $\mathbf{r}$ (Table 1), are provided in the Appendix.

We use MTGPs with an ICM kernel [5] for task correlation and the anisotropic Matern for input correlation, for qPOTS and qPOTS−DOE, and independent "single task" GPs for all other methods

(as they are designed to be). We provide each experiment with a set of $n = 10 \times d$ seed samples chosen uniformly at random from $\mathcal{X}$, and *fully coupled* observations to start the algorithm; then they are repeated 10 times to observe mean and $\max / \min$ performance. Our metric for comparison is the hypervolume indicator [41] and the number of oracle evaluations. Our implementation primarily builds on GPyTorch [17] and BoTorch [3]; we leverage MPI4Py [10] to parallelize our repetitions on CPUs. The full source code is available at `Link`; currently attached with this submission.

The results of the synthetic experiments are shown in Figure 3 – notice that qPOTS−DOE is the clear winner in $6/8$ experiments (BNH, DTLZ4-5, ZDT2, OSY, CONSTR); in the remaining 2, it is still no worse than others; crucially, qPOTS−DOE outperforms HVKG in all of them. Note that PESMO/JESMO/MESMO were run until we ran out of memory, and hence are missing in some experiments. The real-world experiments are shown in Figure 4, where qPOTS−DOE beats all competitors in all cases. Figure 2 shows how qPOTS−DOE picks the objective and constraint evaluation at each round, for the OSY problem ($K = 2, C = 6$) at $q = 1$ (additional illustrations are shown in Figure 9). In this example, if decoupling were not allowed, then the total objective and constraint evaluations would have been $160 \times 8 = 1280$ – on the other hand, qPOTS−DOE required only 692 evaluations total (46% fewer evaluations).

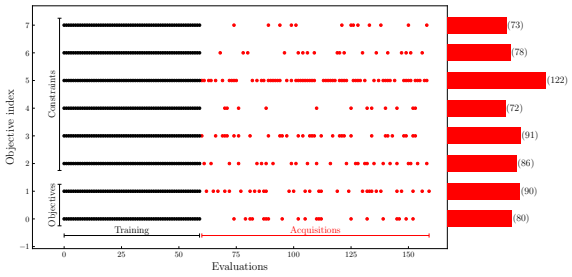


Figure 2: qPOTS−DOE decoupled evaluations (OSY)

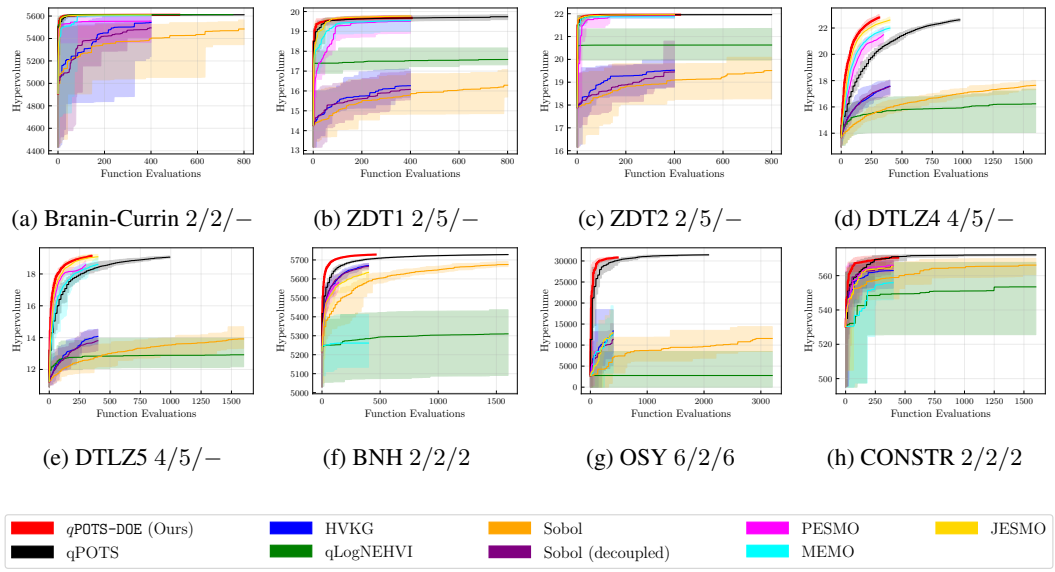
Ablations. We set $q = 4$ for all experiments; however, an ablation on q (see Appendix A.1) shows that qPOTS−DOE maintains its performance across q [2]. We also did an ablation on τ, varying it in $[10^{-6}, 1]$ for the OSY and Branin-Currin experiments (Figure 5); note that when $\tau = \infty$, qPOTS−DOE reduces to qPOTS as expected. We found that any choice between $[10^{-6}, 1)$ had a marginal impact on performance; as a default choice, we set (and recommend) $\tau = 10^{-4}$ in our method. We observe that identifying an optimal τ for each problem is not critical for the overall performance. We include additional ablations on the kernel choice, run-time comparisons, and the benefit of multitask over single-task GPs, in the Appendix.

6 Conclusions

We introduced qPOTS−DOE, a batch Thompson sampling framework for constrained multiobjective Bayesian optimization that exploits correlations among objectives and constraints to support decoupled oracle evaluations under uniform evaluation costs. By replacing independent surrogate models with a multitask GP, qPOTS−DOE preserves cross-task posterior structure; by using total correlation as a decoupling gate and Gaussian mutual information to select which objectives or constraints to evaluate, it turns correlation into a practical mechanism for reducing redundant evaluations. Theoretical analysis shows that the resulting policy remains asymptotically consistent under dense-design and no-starvation , and that our subset rule for decoupling is optimal, while the empirical results on synthetic and real-world benchmarks indicate that qPOTS−DOE improves hypervolume performance relative to several existing coupled and decoupled baselines. Overall, the method broadens the role of decoupled evaluations in MOBO: decoupling need not rely only on unequal costs, but can also arise from statistical dependence among expensive oracles. Future work should focus on high (input) dimensional scaling, testing on additional gains due to variable costs, and robustness to misspecified task correlations.

Limitations. qPOTS−DOE is most effective when objectives and constraints exhibit exploitable posterior correlations; when such correlations are weak or poorly captured by the MTGP kernel, the gains from decoupling may diminish, although the TC threshold allows the method to revert toward coupled evaluation. The method also introduces hyperparameters, notably the TC threshold and the decoupled task budget, and while our ablations suggest limited sensitivity to the threshold, adaptive selection remains a direction to explore.

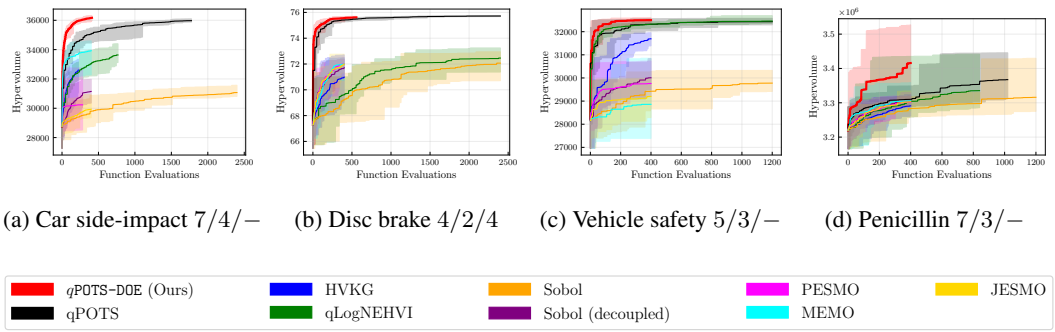
[2] Note that the perceived difference in the plots is an artifact of the changing x-axis scale.



(a) Branin-Currin $2/2/-$ (b) ZDT1 $2/5/-$ (c) ZDT2 $2/5/-$ (d) DTLZ4 $4/5/-$

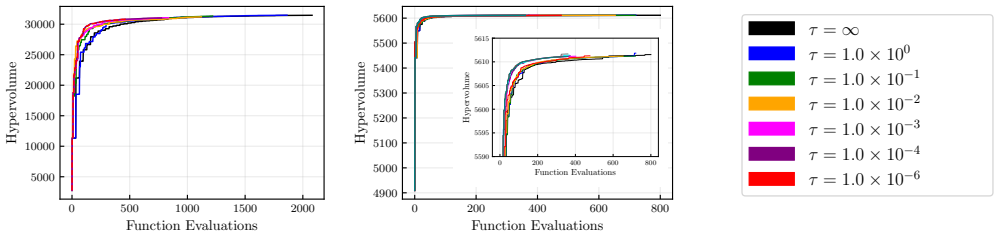
(e) DTLZ5 $4/5/-$ (f) BNH $2/2/2$ (g) OSY $6/2/6$ (h) CONSTR $2/2/2$

Figure 3: Synthetic experiments. The numbers in the labels follow $d/K/C$ dimensions. Solid lines are the mean and filled colors are the $\min$ / $\max$ range over 10 repetitions.



(a) Car side-impact $7/4/-$ (b) Disc brake $4/2/4$ (c) Vehicle safety $5/3/-$ (d) Penicillin $7/3/-$

Figure 4: Real-world experiments. Solid lines are the mean and filled colors are the $\min$ / $\max$ range over 10 repetitions.



OSY Branin-Currin (inset: zoom-in)

Figure 5: **Ablation studies.** Top row: ablation on q (OSY problem); bottom row: ablation on τ.

AI USE STATEMENT

No generative AI was used.

ETHICS STATEMENT

N/A

REPRODUCIBILITY STATEMENT

The code is attached as supplementary material to reproduce all experimental results.

REFERENCES

[1] Mauricio A Alvarez, Lorenzo Rosasco, and Neil D Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266, 2012.

[2] Sebastian Ament, Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy. Unexpected improvements to expected improvement for bayesian optimization. *Advances in neural information processing systems*, 36:20577–20612, 2023.

[3] Maximilian Balandat, Brian Karrer, Daniel Jiang, Samuel Daulton, Ben Letham, Andrew G Wilson, and Eytan Bakshy. Botorch: A framework for efficient monte-carlo bayesian optimization. *Advances in neural information processing systems*, 33:21524–21538, 2020.

[4] Syrine Belakaria, Aryan Deshwal, and Janardhan Rao Doppa. Max-value entropy search for multi-objective bayesian optimization. *Advances in neural information processing systems*, 32, 2019.

[5] Edwin V Bonilla, Kian Chai, and Christopher Williams. Multi-task gaussian process prediction. In *Advances in neural information processing systems*, volume 20, 2007.

[6] Niv Buchbinder, Moran Feldman, Joseph Seffi, and Roy Schwartz. A tight linear time (1/2)-approximation for unconstrained submodular maximization. *SIAM Journal on Computing*, 44 (5):1384–1402, 2015.

[7] Jack M Buckingham, Sebastian Rojas-Gonzalez, and Juergen Branke. Knowledge gradient for multi-objective bayesian optimization with decoupled evaluations. In *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 117–132. Springer, 2025.

[8] Kade Carlson and Ashwin Renganathan. Multiobjective aerodynamic design optimization of the nasa common research model. *Aerospace Science and Technology*, pp. 111120, 2025.

[9] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2 edition, 2006. ISBN 978-0471241959.

[10] Lisandro Dalcin and Yao-Lung L Fang. mpi4py: Status update after 12 years of development. *Computing in Science & Engineering*, 23(4):47–54, 2021.

[11] Sam Daulton, Maximilian Balandat, and Eytan Bakshy. Hypervolume knowledge gradient: a lookahead approach for multi-objective bayesian optimization with partial information. *International Conference on Machine Learning*, pp. 7167–7204, 2023.

[12] Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. Differentiable expected hypervolume improvement for parallel multi-objective bayesian optimization. In *Advances in neural information processing systems*, volume 33, pp. 9851–9864, 2020.

[13] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6 (2):182–197, 2002.

[14] Michael Emmerich and Jan-willem Klinkenberg. The computation of the expected improvement in dominated hypervolume of pareto front approximations. Technical report, Leiden University, 2008.

[15] Uriel Feige, Vahab S Mirrokni, and Jan Vondrák. Maximizing non-monotone submodular functions. *SIAM Journal on Computing*, 40(4):1133–1153, 2011.

[16] Peter I. Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

[17] Jacob Gardner, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems*, pp. 7576–7586, 2018.

[18] Eduardo C Garrido-Merchán and Daniel Hernández-Lobato. Predictive entropy search for multi-objective bayesian optimization with constraints. *Neurocomputing*, 361:50–68, 2019.

[19] Eduardo C Garrido-Merchán and Daniel Hernández-Lobato. Parallel predictive entropy search for multi-objective bayesian optimization with constraints. *arXiv preprint arXiv:2004.00601*, 2020.

[20] Md Shahriar Iqbal, Jianhai Su, Lars Kotthoff, and Pooyan Jamshidi. Flexibo: a decoupled cost-aware multi-objective optimization approach for deep neural networks. *Journal of Artificial Intelligence Research*, 77:645–682, 2023.

[21] Himanshu Jain and Kalyanmoy Deb. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: Handling constraints and extending to an adaptive approach. *IEEE Transactions on evolutionary computation*, 18(4): 602–622, 2013.

[22] Vijay Janapa Reddi, David Kanter, Peter Mattson, Jared Duke, Thai Nguyen, Ramesh Chukka, Ken Shiring, Koan-Sin Tan, Mark Charlebois, William Chou, et al. Mlperf mobile inference benchmark: An industry-standard open-source machine learning benchmark for on-device ai. *Proceedings of Machine Learning and Systems*, 4:352–369, 2022.

[23] Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

[24] Joshua Knowles. Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66, 2006.

[25] Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(2), 2008.

[26] Qiaohao Liang and Lipeng Lai. Scalable bayesian optimization accelerates process optimization of penicillin production. In *NeurIPS 2021 AI for Science Workshop*, 2021.

[27] Xingtao Liao, Qing Li, Xujing Yang, Weigang Zhang, and Wei Li. Multiobjective optimization for crash safety design of vehicles using stepwise regression model. *Structural and multidisciplinary optimization*, 35:561–569, 2008.

[28] Andrzej Osyczka and Sourav Kundu. A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm. *Structural optimization*, 10(2):94–99, 1995.

[29] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

[30] Ashwin Renganathan and Kade Carlson. qpots: Efficient batch multiobjective bayesian optimization via pareto optimal thompson sampling. In *International Conference on Artificial Intelligence and Statistics*, pp. 4051–4059. PMLR, 2025.

[31] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104 (1):148–175, 2015.

[32] Ilya M Sobol. Distribution of points in a cube and approximate evaluation of integrals. *USSR Computational mathematics and mathematical physics*, 7:86–112, 1967.

[33] Shinya Suzuki, Shion Takeno, Tomoyuki Tamura, Kazuki Shitara, and Masayuki Karasuyama. Multi-objective bayesian optimization using pareto-frontier entropy. In *International conference on machine learning*, pp. 9279–9288. PMLR, 2020.

[34] Kevin Swersky, Jasper Snoek, and Ryan P Adams. Multi-task bayesian optimization. In *Advances in neural information processing systems*, volume 26, 2013.

[35] Ryoji Tanabe and Hisao Ishibuchi. An easy-to-use real-world multi-objective optimization problem suite. *Applied Soft Computing*, 89:106078, 2020.

[36] Yee Whye Teh, Matthias Seeger, and Michael I Jordan. Semiparametric latent factor models. In *International Workshop on Artificial Intelligence and Statistics*, pp. 333–340. PMLR, 2005.

[37] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.

[38] Ben Tu, Axel Gandy, Nikolas Kantas, and Behrang Shafei. Joint entropy search for multi-objective bayesian optimization. *Advances in Neural Information Processing Systems*, 35: 9922–9938, 2022.

[39] Satosi Watanabe. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development*, 4(1):66–82, 1960.

[40] Christopher KI Williams. Prediction with gaussian processes: From linear regression to linear prediction and beyond. In *Learning in graphical models*, pp. 599–621. Springer, 1998.

[41] Kaifeng Yang, Michael Emmerich, André Deutz, and Thomas Bäck. Efficient computation of expected hypervolume improvement using box decomposition algorithms. *Journal of Global Optimization*, 75:3–34, 2019.

Algorithm 1 qPOTS-DOE: Batch Pareto optimal Thompson sampling with Decoupled Oracle Evaluations

Require: Initial data $\mathcal{D}_0$; objective and constraint oracles $f_k,\ k = 1, 2, 3, \ldots$ and $c_j,\ j \in \mathcal{I} \cup \mathcal{E}$; batch size q; decoupling threshold τ; rounds T

for $i = 0, \ldots, T-1$ **do**
  Fit one multitask Gaussian process to all observations in $\mathcal{D}_i$
  Draw one joint Thompson sample of all objectives and constraints
  Find the feasible Pareto set under this sampled model
  Select q diverse candidate designs from the sampled Pareto set
  for each candidate design $\mathbf{x}$ **do**
    Compute a dependence score $\mathrm{TC}(\mathbf{x})$ between the objective/constraint tasks
    if $\mathrm{TC} < \tau$ **then**
      Evaluate all objectives and constraints at $\mathbf{x}$ ▷ coupled evaluation
    else
      Identify M, and choose M tasks that are most informative about the remaining tasks
      Evaluate only those selected tasks at $\mathbf{x}$ ▷ decoupled evaluation
    end if
    Add the new observations to $\mathcal{D}_{i+1}$
  end for
  (optional) Occasionally force rarely observed tasks to be evaluated ▷ prevents task starvation
end for
return The feasible designs and estimated Pareto frontier from the final model

A Supplementary material

A.1 Additional experiments and ablations

Details on experiments. The Penicillin production problem ($d = 7, K = 3$) seeks to maximize Penicillin yield while minimizing both the production time and the carbon dioxide byproduct emission. The $d = 7$ input variables include the culture medium volume, biomass concentration, temperature (in K), glucose substrate concentration, substrate feed rate, substrate feed concentration and the $H+$ concentration [26]. The Vehicle Safety problem ($d = 5, K = 3$) [35] concerns the design of automobiles for enhanced safety. Specifically, we consider the minimization of vehicle mass, the vehicle acceleration during a full-frontal crash, and the toe-board intrusion—a measure of the mechanical damage to the vehicle during an off-frontal crash, with respect to $d = 5$ design variables which represent the reinforced parts of the frontal frame; further details are found in [27]. The Car side-impact problem is aimed at minimizing the weight of the car while minimizing the pubic force experienced by a passenger and the average velocity of the V-pillar responsible for withstanding the impact load. Additionally, as the fourth objective, the maximum constraint violation of 10 constraints which include limiting values of abdomen load, pubic force, velocity of V-pillar, and rib deflection is also added [21].

Experimental settings. Additionally, we also include Sobol sampling [32] with and without decoupling. We implemented decoupled variants of PESMO, MESMO, and JESMO by introducing an evaluation mask that specifies which objective is observed at a candidate design, and by optimizing over the input ($\mathbf{x}$) - objective index (k) pairs rather than over $\mathbf{x}$ alone. For PESMO, the entropy reduction is restricted to the masked objective term, while for MESMO and JESMO we use a projection-based approximation that conditions only on the observed objective by projecting the posterior and sampled Pareto geometry onto the corresponding subspace and recomputing the lower-dimensional box decomposition.

Table 1: Summary of all experiments

Test Case	**Dim**	**Obj**	**Cons**	**Ref Point**	**Ablations:**
BNH	2	2	2	$(-150, -50)$	–
BraninCurrin	2	2	0	$(-300, -20)$	$q = 1, 2, 4, 8$
C2DTLZ2	5	4	1	$(-10, -10, -10, -10)$	–
CarsideImpact	7	4	0	$(-50, -10, -20, -20)$	$q = 1, 2, 4, 8$
CONSTR	2	2	2	$(-10, -60)$	–
Constrained Branin-Currin	2	2	1	$(-300, -20)$	–
DiscBrake	4	2	4	$(-10, -10)$	$q = 1, 2, 4, 8$
DTLZ2	5	4	0	$(-10, -10, -10, -10)$	–
DTLZ3	3	2	0	$(-100, -100)$	–
DTLZ4	5	4	0	$(-1, -1, -1, -1)$	–
DTLZ5	5	4	0	$(-1, -1, -1, -1)$	–
DTLZ7	20	2	0	$(-25, -25)$	–
GMM	2	4	0	$(-1, -1, -1, -1)$	–
MW7	2	2	2	$(-10, -10)$	–
OSY	6	2	6	$(-100, -100)$	$q = 1, 2, 4, 8$
Penicillin	7	3	0	$(-40, -70, -1000)$	–
SRN	2	2	2	$(-1000, -200)$	–
VehicleSafety	5	3	0	$(-1900, -20, -10)$	–
ZDT1	5	2	0	$(-1, -1)$	–
ZDT2	5	2	0	$(-1, -1)$	–

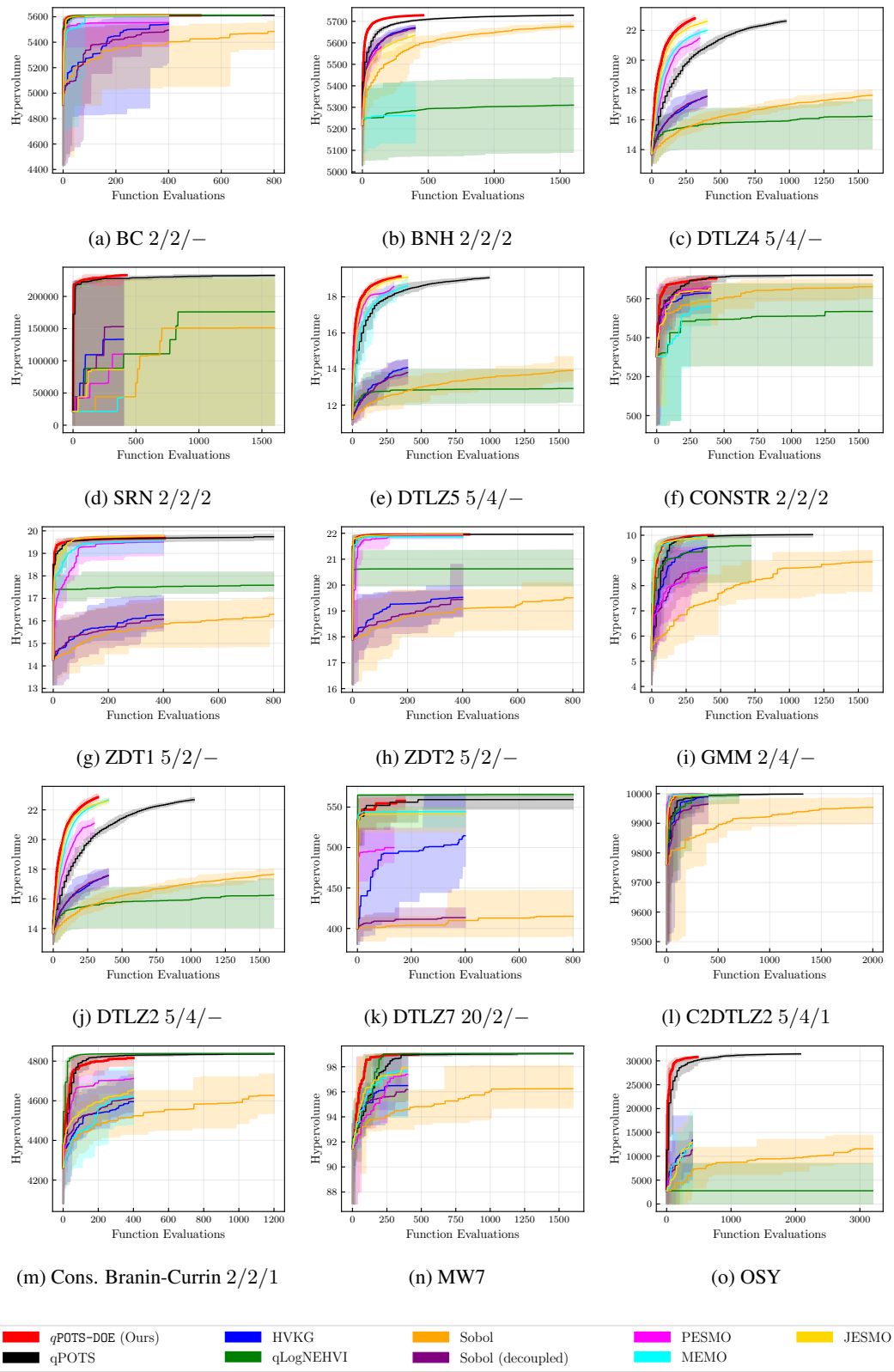


Figure 6: Synthetic experiments

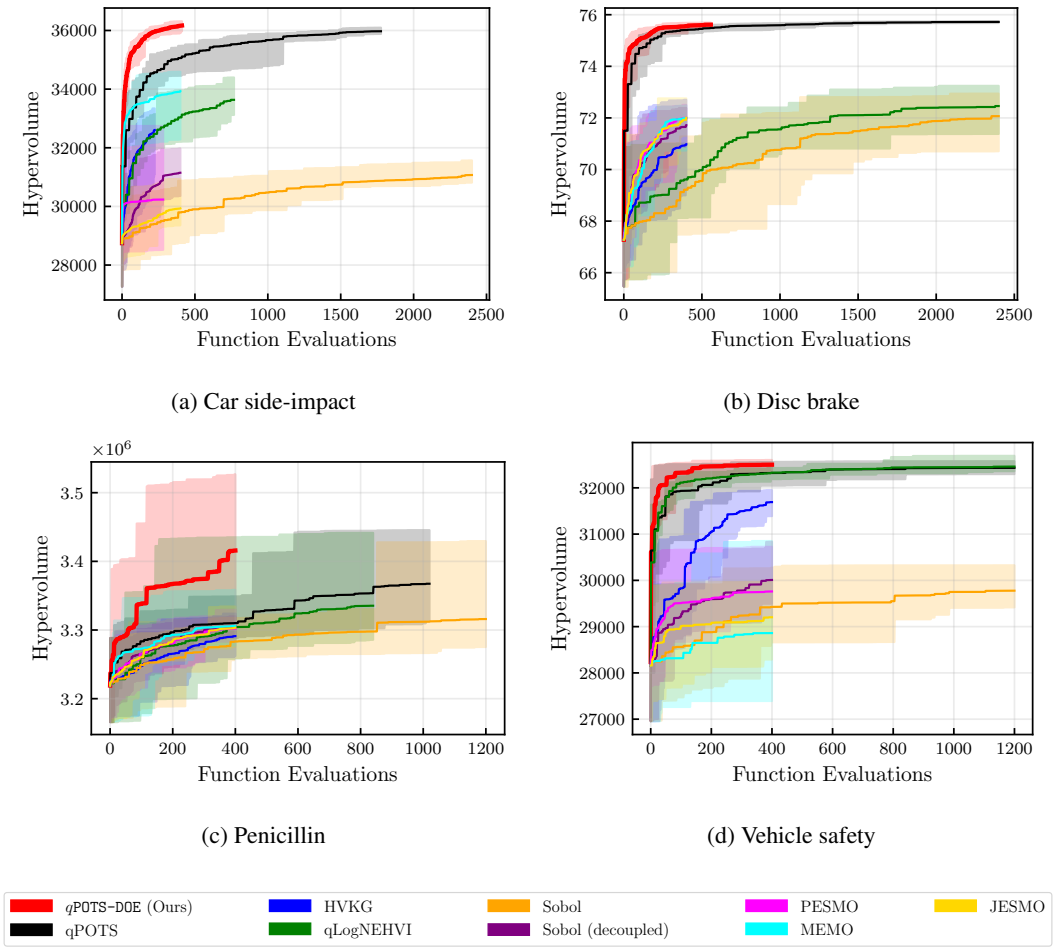


Figure 7: Real-world experiments.

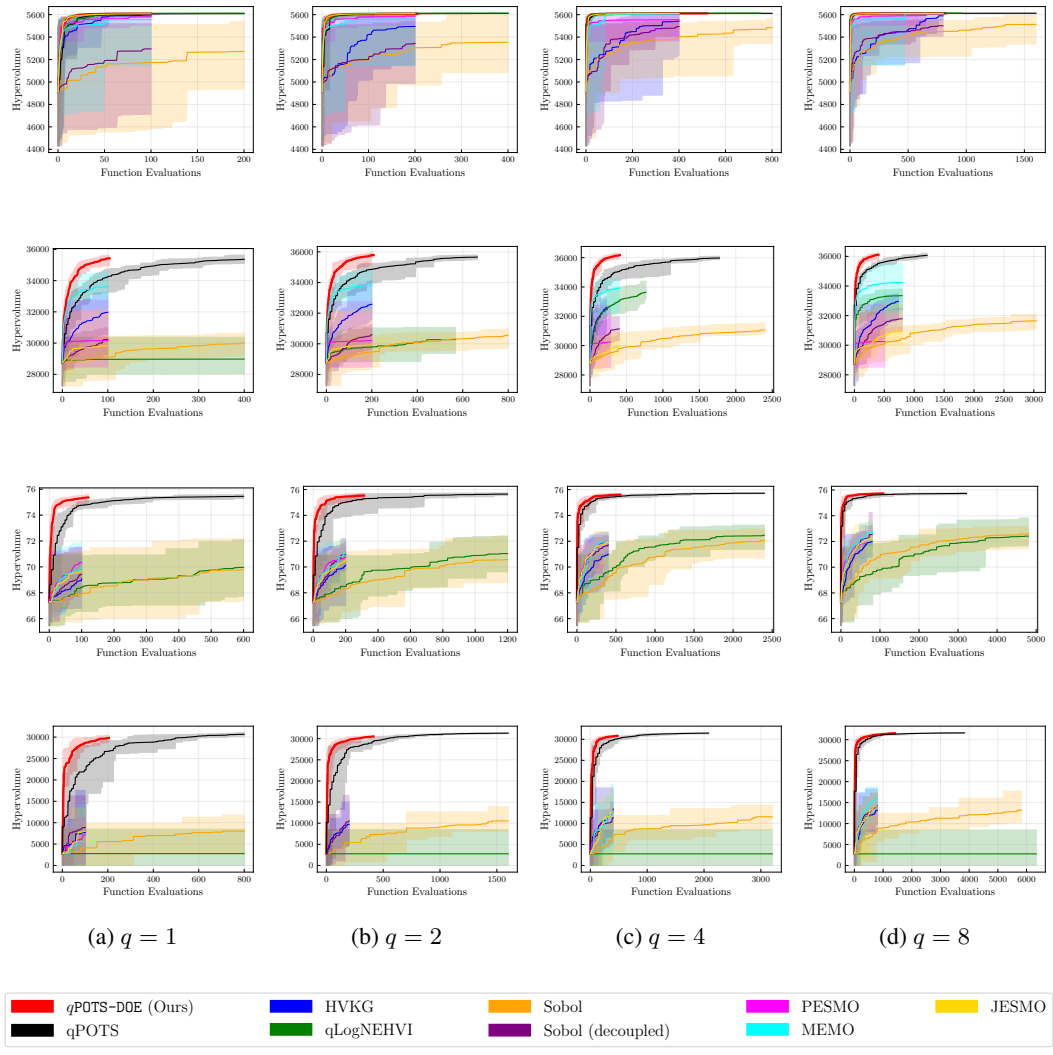
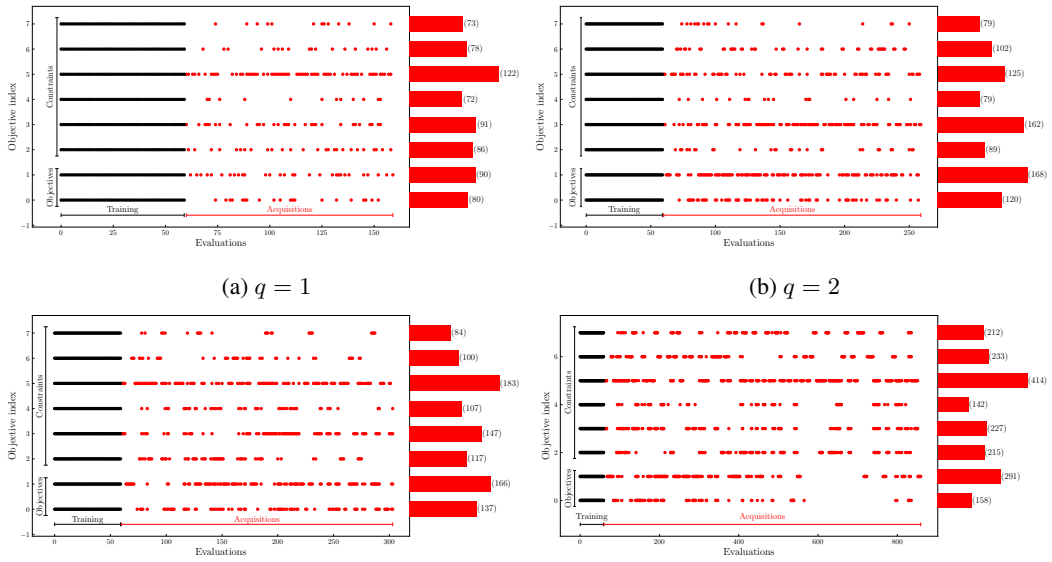


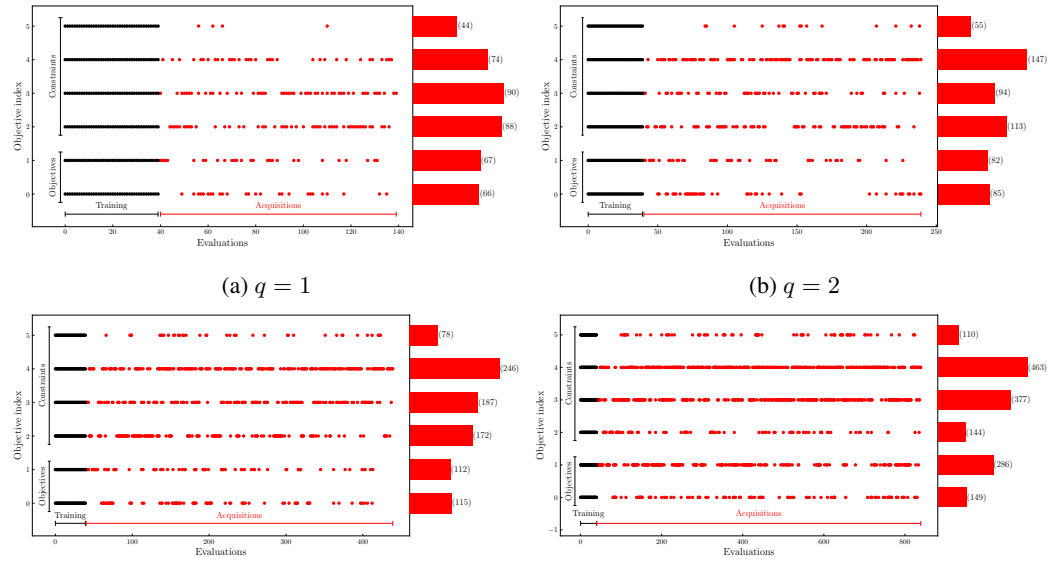
Figure 8: Ablation on batch size q. Row 1: Branin-Currin, row-2: Car side-impact, row-3: Disc brake, row-4: OSY.



(a) $q = 1$ (b) $q = 2$

(c) $q = 4$ (d) $q = 8$

Figure 9: Distribution of decoupled oracle evaluations (OSY)



(a) $q = 1$ (b) $q = 2$

(c) $q = 4$ (d) $q = 8$

Figure 10: Distribution of decoupled oracle evaluations (Disc brake)

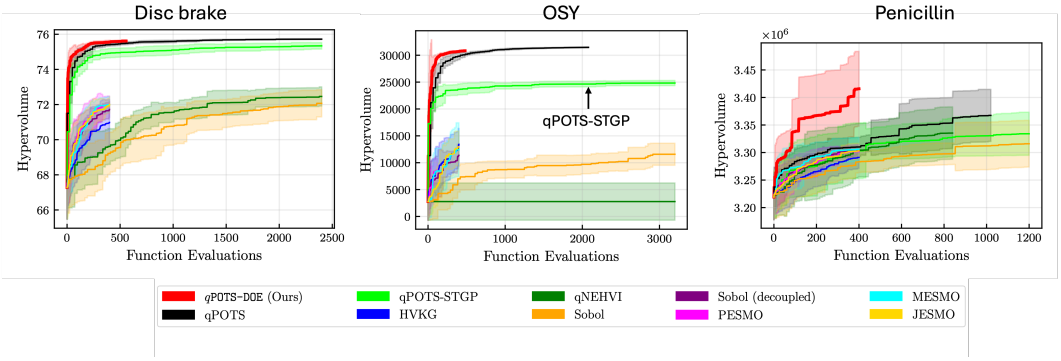


Figure 11: Coupled single-task GP ablation on Disc brake, OSY, and Penicillin.

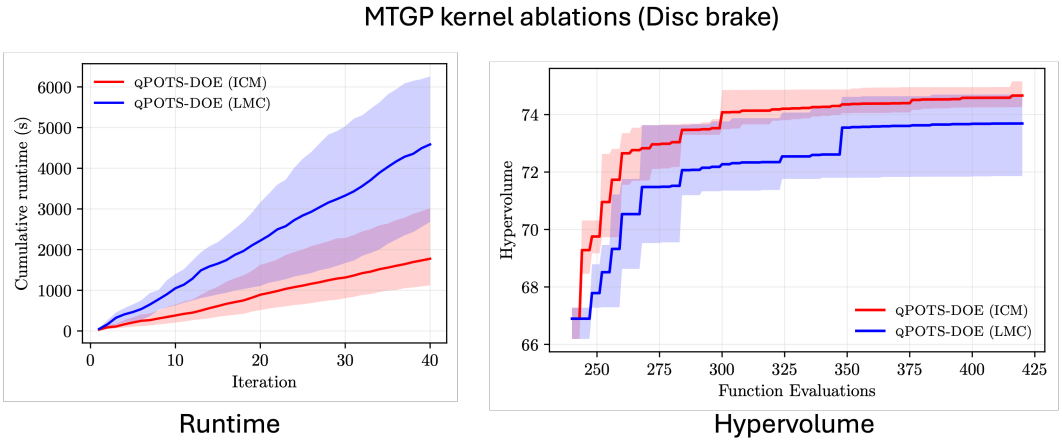


Figure 12: ICM versus LMC: cumulative runtime and hypervolume.

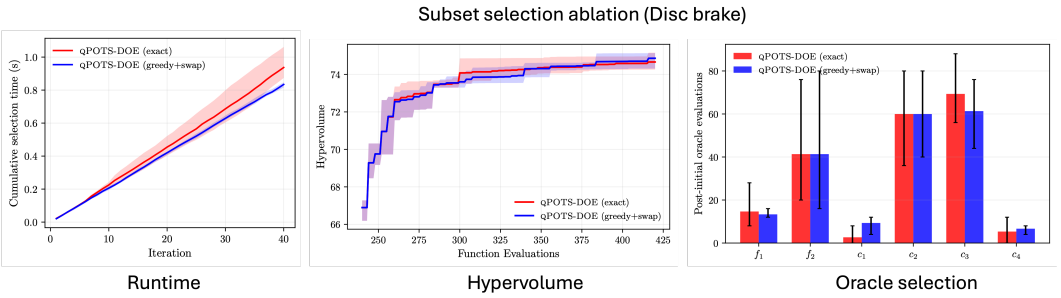


Figure 13: Exact versus greedy-plus-swap subset selection: cumulative selection time, hypervolume, and oracle evaluations.

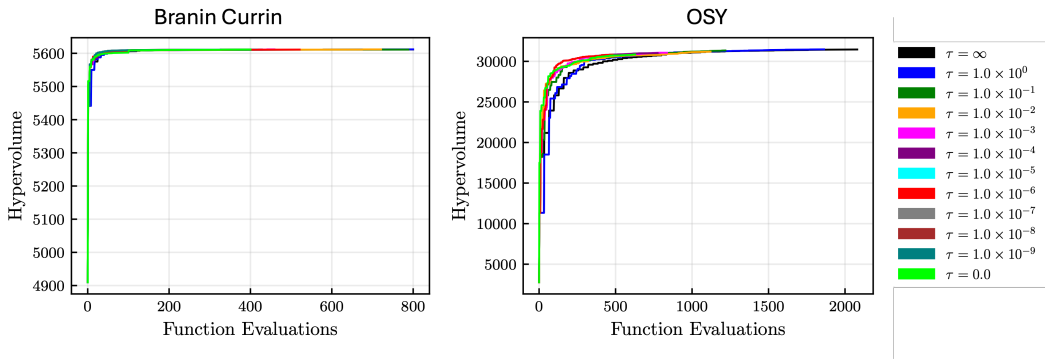


Figure 14: Ablation on the decoupling threshold τ for Branin-Currin and OSY.

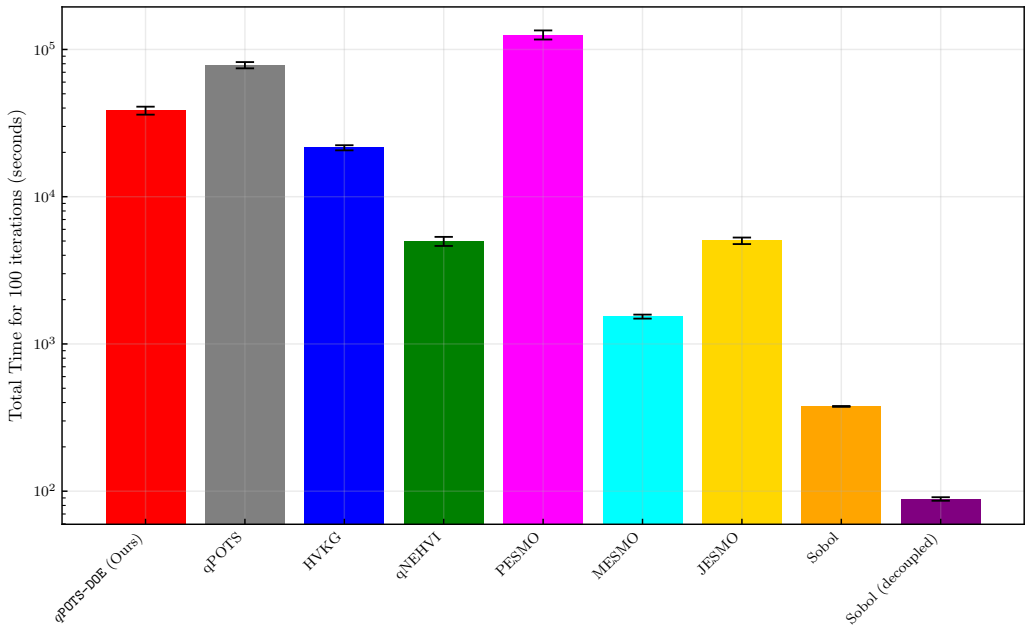


Figure 15: Runtime comparison on OSY.

A.2 Additional detail on the method

Gaussian process models. Let $f : \mathcal{X} \to \mathbb{R}$ be an expensive zeroth-order oracle, and the input $\mathbf{x}$ be a d-dimensional vector $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^d$, and let $Y(\mathbf{x})$ denote a real-valued random function evaluated at $\mathbf{x}$. A GP is a stochastic process such that any finite collection of function values is jointly Gaussian [29, 40]. We write the GP prior as

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

where $m(\mathbf{x}) = \mathbb{E}[Y(\mathbf{x})]$ is the mean function and $k(\mathbf{x}, \mathbf{x}') = \operatorname{cov}(Y(\mathbf{x}), Y(\mathbf{x}'))$ is a positive semidefinite covariance (kernel) function. Given a set of inputs $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n \times d}$, define the vector of latent function values $\mathbf{y} = \big[Y(\mathbf{x}_1), \ldots, Y(\mathbf{x}_n)\big]^\top \in \mathbb{R}^n$. By the definition of a GP, $\mathbf{y}$ is multivariate Gaussian: $\mathbf{y} \sim \mathcal{N}(\mathbf{m}, \Sigma), \quad \mathbf{m}_i = m(\mathbf{x}_i), \quad \Sigma_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. Let there be noisy observations of f of the form $\tilde{y}_i = Y(\mathbf{x}_i) + \varepsilon_i, \qquad \varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$, so that the observation vector $\tilde{\mathbf{y}} = [\tilde{y}_1, \ldots, \tilde{y}_n]^\top$ satisfies $\tilde{\mathbf{y}} \sim \mathcal{N}\big(\mathbf{m},\, \Sigma + \sigma^2 \mathbf{I}_n\big)$. Then, for a test input $\mathbf{x}_* \in \mathbb{R}^d$, define the cross-covariances $\Sigma_{X*} = \big[k(\mathbf{x}_i, \mathbf{x}_*)\big]_{i=1}^n \in \mathbb{R}^{n \times 1}, \quad \Sigma_{*X} = \Sigma_{X*}^\top, \quad \Sigma_{**} = k(\mathbf{x}_*, \mathbf{x}_*) \in \mathbb{R}$. Conditioning the joint Gaussian of $(\tilde{\mathbf{y}},\, Y(\mathbf{x}_*))$ yields the standard GP regression posterior [29]:

$$Y(\mathbf{x}_*) \mid \mathbf{X}, \tilde{\mathbf{y}} \;\sim\; \mathcal{N}\Big(m(\mathbf{x}_*) + \Sigma_{*X}\big(\Sigma + \sigma^2 \mathbf{I}_n\big)^{-1}\big(\tilde{\mathbf{y}} - \mathbf{m}\big), \Sigma_{**} - \Sigma_{*X}\big(\Sigma + \sigma^2 \mathbf{I}_n\big)^{-1}\Sigma_{X*}\Big).$$

This posterior provides both a predictive mean and a calibrated uncertainty (predictive variance) at $\mathbf{x}_*$.

A.2.1 Unconstrained qPOTS−DOE with MTGPs

Let $\mathbf{Y}_{1:K}(\mathbf{x}) \equiv [Y_1(\mathbf{x}) \quad \cdots \quad Y_K(\mathbf{x})]^\top$. Then, the multitask posterior is written as

$$\mathbf{Y}_{1:K}(\cdot) \mid D_n^{1:K}, \Omega \sim \mathrm{MTGP}(\boldsymbol{\mu}_n(\cdot), \boldsymbol{\Sigma}_n(\cdot, \cdot))\,,$$

where

$$[\boldsymbol{\Sigma}_n(\mathbf{x}, \mathbf{x}')]_{ij} = \operatorname{cov}\left(Y_i(\mathbf{x}), Y_j(\mathbf{x}') \mid D_n^{1:K}, \Omega\right), \qquad i, j \in [K].$$

Thus, the posterior over the K objective sample paths is no longer factorized across objectives. We choose $\mathbf{x}$ according to

$$p_{X^*}(\mathbf{x}) = \int \delta\left(\mathbf{x} - \arg\max_{\mathbf{x} \in \mathcal{X}} \{Y_1(\mathbf{x}), \ldots, Y_K(\mathbf{x})\}\right) p\big(\mathbf{Y}_{1:K} \mid D_n^{1:K}\big)\; d\mathbf{Y}_{1:K}, \tag{9}$$

where $d\mathbf{Y}_{1:K} \equiv dY_1 \cdots dY_K$. If $\boldsymbol{\Sigma}_n(\cdot, \cdot)$ is block diagonal across the K tasks, then (9) reduces to the independent-GP factorization as in [30]. In practice, this is equivalent to drawing one joint sample path from the multitask posterior,

$$\mathbf{Y}_{1:K}(\cdot, \omega) = \boldsymbol{\mu}_n(\cdot) + \boldsymbol{\Sigma}_n^{1/2}(\cdot)\, Z(\omega), \qquad Z(\omega) \sim \mathcal{N}(0, I),$$

and choosing the next point(s) from the Pareto set of the jointly sampled objective paths:

$$X^* = \arg\max_{\mathbf{x} \in \mathcal{X}} \{Y_1(\mathbf{x}, \omega), \ldots, Y_K(\mathbf{x}, \omega)\}.$$

As in [30], this cheap multiobjective optimization problem is solved via evolutionary approaches e.g., NSGA-II.

Let $\gamma(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ denote the Euclidean distance between two points in $\mathcal{X}$. Then, the next batch of q points is chosen per

$$\begin{aligned}
\mathbf{x}_{n+1} &= \arg\max_{\mathbf{x}^* \in \mathcal{X}^*} \min_{\mathbf{x}_i \in X_n} \gamma(\mathbf{x}^*, \mathbf{x}_i), \\
\mathbf{x}_{n+2} &= \arg\max_{\mathbf{x}^* \in X^*} \min_{\mathbf{x}_i \in X_n \cup \{\mathbf{x}_{n+1}\}} \gamma(\mathbf{x}^*, \mathbf{x}_i), \\
&\vdots \\
\mathbf{x}_{n+q} &= \arg\max_{\mathbf{x}^* \in X^*} \min_{\mathbf{x}_i \in X_n \cup \{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+q-1}\}} \gamma(\mathbf{x}^*, \mathbf{x}_i).
\end{aligned} \tag{10}$$

Equation (10) is unchanged from the independent-GP setting; only the posterior sample path used to construct X^* is now drawn jointly from the multitask GP posterior.

A.3 PROOFS

Assumption A.1 (Regularity and no-starvation)**.** Let $\mathcal{X} \subset \mathbb{R}^d$ be compact with nonempty interior, and let

$$\mathbf{Y}(\mathbf{x}) = \big(Y_1(\mathbf{x}), \dots, Y_K(\mathbf{x})\big)^\top$$

be a correctly specified K-task Gaussian process prior with mean $\boldsymbol{\mu}_0(\mathbf{x})$ and matrix-valued covariance kernel

$$\mathbf{K}_0(\mathbf{x}, \mathbf{x}') = \mathrm{Cov}(\mathbf{Y}(\mathbf{x}), \mathbf{Y}(\mathbf{x}')) \in \mathbb{R}^{K \times K}.$$

Assume that the process admits an almost surely continuous version on $\mathcal{X}$. At round i, the algorithm queries a design point $\mathbf{x}_i \in \mathcal{X}$ and observes only a subset of tasks $S_i \subseteq [K]$. For each $k \in S_i$, the observation model is

$$y_{i,k} = Y_k(\mathbf{x}_i) + \varepsilon_{i,k}, \qquad \varepsilon_{i,k} \sim \mathcal{N}(0, \sigma_k^2),$$

where the noise variables are independent across rounds and tasks, with $0 < \sigma_k^2 < \infty$. Let

$$\mathcal{D}_n = \{(\mathbf{x}_i, S_i, \{y_{i,k} : k \in S_i\})\}_{i=1}^n$$

denote the partial-observation data after n rounds.

Finally, assume no starvation under decoupling: for every task $k \in [K]$, the set of design points at which task k is directly observed,

$$\mathcal{X}_{n,k} := \{\mathbf{x}_i : i \le n,\ k \in S_i\},$$

has fill distance

$$h_{n,k} := \sup_{\mathbf{x} \in \mathcal{X}} \min_{\mathbf{z} \in \mathcal{X}_{n,k}} \|\mathbf{x} - \mathbf{z}\|_2$$

satisfying

$$h_{n,k} \longrightarrow 0 \qquad \text{as } n \to \infty, \qquad \forall k \in [K].$$

Note that the no-starvation assumption is only necessary to prove Theorem A.2; but, in practice, it may not be necessary. Indeed, all experiments reported in this paper don't perform line 16 of Algorithm 1 (occasional evaluations of all objectives and constraints).

Theorem A.2 (Posterior consistency under decoupled oracle evaluations)**.** *Let $\mathcal{X} \subset \mathbb{R}^d$ be compact with nonempty interior, and let*

$$\mathbf{Y}(\mathbf{x}) = \big(Y_1(\mathbf{x}), \dots, Y_K(\mathbf{x})\big)^\top$$

denote a K-task Gaussian process prior with mean $\boldsymbol{\mu}_0 : \mathcal{X} \to \mathbb{R}^K$ and matrix-valued covariance kernel

$$\mathbf{K}_0(\mathbf{x}, \mathbf{x}') = \mathrm{Cov}(\mathbf{Y}(\mathbf{x}), \mathbf{Y}(\mathbf{x}')) \in \mathbb{R}^{K \times K}.$$

Assume that the latent objective vector

$$\mathbf{Y}_0(\mathbf{x}) = \big(Y_{0,1}(\mathbf{x}), \dots, Y_{0,K}(\mathbf{x})\big)^\top$$

is generated from this correctly specified K-task GP model, and that the process admits an almost surely continuous version on $\mathcal{X}$.

At round i, the algorithm selects a design point $\mathbf{x}_i \in \mathcal{X}$ and evaluates only a subset of objectives

$$S_i \subseteq [K], \qquad \bar{S}_i := [K] \setminus S_i.$$

For every $k \in S_i$, the observation model is

$$y_{i,k} = Y_{0,k}(\mathbf{x}_i) + \varepsilon_{i,k}, \qquad \varepsilon_{i,k} \sim \mathcal{N}(0, \sigma_k^2),$$

where the noise variables are independent across rounds and objectives, with $0 < \sigma_k^2 < \infty$. No observation is collected for objectives $k \in \bar{S}_i$. Let

$$\mathcal{D}_n = \{(\mathbf{x}_i, S_i, \{y_{i,k} : k \in S_i\})\}_{i=1}^n$$

denote the partial-observation data after n rounds, and write the resulting multitask GP posterior as

$$\mathbf{Y} \mid \mathcal{D}_n \sim \mathcal{GP}(\boldsymbol{\mu}_n(\cdot), \mathbf{K}_n(\cdot, \cdot)).$$

Assume no starvation under decoupling: for every objective $k \in [K]$, the set of design points at which objective k is directly evaluated,

$$\mathcal{X}_{n,k} := \{\mathbf{x}_i : i \le n,\ k \in S_i\},$$

has fill distance

$$h_{n,k} := \sup_{\mathbf{x}\in\mathcal{X}} \min_{\mathbf{z}\in\mathcal{X}_{n,k}} \|\mathbf{x}-\mathbf{z}\|_2$$

satisfying

$$h_{n,k} \longrightarrow 0 \qquad \text{as } n\to\infty, \qquad \forall k \in [K].$$

Then, for any finite set

$$\mathcal{X}^\star = \{\mathbf{x}^{(1)},\dots,\mathbf{x}^{(m)}\} \subset \mathcal{X}$$

and any $\epsilon > 0$,

$$\Pi(\|\mathbf{Y}(\mathcal{X}^\star) - \mathbf{Y}_0(\mathcal{X}^\star)\|_2 > \epsilon \mid \mathcal{D}_n) \longrightarrow 0 \qquad \textit{almost surely under the true data-generating law.}$$

Equivalently,

$$\boldsymbol{\mu}_n(\mathcal{X}^\star) \longrightarrow \mathbf{Y}_0(\mathcal{X}^\star), \qquad \mathbf{K}_n(\mathcal{X}^\star,\mathcal{X}^\star) \longrightarrow \mathbf{0}$$

almost surely. In particular, for every fixed $\mathbf{x} \in \mathcal{X}$,

$$\boldsymbol{\mu}_n(\mathbf{x}) \longrightarrow \mathbf{Y}_0(\mathbf{x}), \qquad \mathbf{K}_n(\mathbf{x},\mathbf{x}) \longrightarrow \mathbf{0}_{K\times K}.$$

Proof. Let $\mathcal{F}_n := \sigma(\mathcal{D}_n)$ denote the filtration generated by the partial-observation data. For each task $k \in [K]$, define the task-specific data

$$\mathcal{D}_{n,k} := \{(\mathbf{x}_i, y_{i,k}) : i \le n,\ k \in S_i\}.$$

Thus $\mathcal{D}_{n,k}$ contains only the direct observations of objective k, whereas $\mathcal{D}_n$ contains all observations from the evaluated subsets S_i and no observations from the omitted subsets $\bar{S}_i = [K] \setminus S_i$.

We first show that the posterior variance of each task vanishes at every fixed input. Fix $k \in [K]$ and $\mathbf{x} \in \mathcal{X}$. Consider the scalar GP regression problem for Y_k using only $\mathcal{D}_{n,k}$, and write

$$\sigma^2_{n,k}(\mathbf{x}) := \mathrm{Var}(Y_k(\mathbf{x}) \mid \mathcal{D}_{n,k})\,.$$

Let μ_k denote the prior mean of Y_k, and define the centered process $\widetilde{Y}_k(\mathbf{x}) := Y_k(\mathbf{x}) - \mu_k(\mathbf{x})$. Since the covariance kernel is continuous and the process admits an almost surely continuous version, Y_k is mean-square continuous. Hence

$$\omega_k(r) := \sup_{\substack{\mathbf{u},\mathbf{v}\in\mathcal{X}\\ \|\mathbf{u}-\mathbf{v}\|_2\le r}} \left(\mathbb{E}\left[\left(\widetilde{Y}_k(\mathbf{u}) - \widetilde{Y}_k(\mathbf{v})\right)^2\right]\right)^{1/2} \longrightarrow 0 \qquad \text{as } r \downarrow 0.$$

By the no-starvation condition, the task-k design set

$$\mathcal{X}_{n,k} := \{\mathbf{x}_i : i \le n,\ k \in S_i\}$$

has fill distance $h_{n,k} \to 0$. Therefore, for every fixed $\mathbf{x} \in \mathcal{X}$, there exists a sequence $r_n \downarrow 0$ such that the number of task-k observations in the local ball

$$I_{n,k}(\mathbf{x}) := \{i \le n : k \in S_i,\ \|\mathbf{x}_i - \mathbf{x}\|_2 \le r_n\}$$

satisfies

$$N_{n,k}(\mathbf{x}) := |I_{n,k}(\mathbf{x})| \longrightarrow \infty.$$

For example, on a compact box or any compact domain with the usual local covering regularity, this follows from the fill-distance condition by taking $r_n/h_{n,k} \to \infty$ and $r_n \to 0$.

Define the local linear estimator

$$\widehat{Y}_{n,k}(\mathbf{x}) := \mu_k(\mathbf{x}) + \frac{1}{N_{n,k}(\mathbf{x})} \sum_{i\in I_{n,k}(\mathbf{x})} \big(y_{i,k} - \mu_k(\mathbf{x}_i)\big).$$

Since

$$y_{i,k} = Y_k(\mathbf{x}_i) + \varepsilon_{i,k}, \qquad \varepsilon_{i,k} \sim \mathcal{N}(0,\sigma_k^2),$$

with independent noise, we have

$$Y_k(\mathbf{x}) - \widehat{Y}_{n,k}(\mathbf{x}) = \frac{1}{N_{n,k}(\mathbf{x})} \sum_{i \in I_{n,k}(\mathbf{x})} \big(\widetilde{Y}_k(\mathbf{x}) - \widetilde{Y}_k(\mathbf{x}_i)\big) - \frac{1}{N_{n,k}(\mathbf{x})} \sum_{i \in I_{n,k}(\mathbf{x})} \varepsilon_{i,k}.$$

Using Cauchy's inequality in L^2 and the independence of the observation noise,

$$\mathbb{E}\Big[\big(Y_k(\mathbf{x}) - \widehat{Y}_{n,k}(\mathbf{x})\big)^2\Big] \le \omega_k(r_n)^2 + \frac{\sigma_k^2}{N_{n,k}(\mathbf{x})} \longrightarrow 0.$$

The scalar GP posterior variance is the minimum mean-squared error among estimators measurable with respect to $\mathcal{D}_{n,k}$. Therefore,

$$\sigma^2_{n,k}(\mathbf{x}) = \mathrm{Var}(Y_k(\mathbf{x}) \mid \mathcal{D}_{n,k}) \le \mathbb{E}\Big[\big(Y_k(\mathbf{x}) - \widehat{Y}_{n,k}(\mathbf{x})\big)^2\Big] \longrightarrow 0.$$

Now compare the scalar posterior to the full multitask posterior. Since $\mathcal{D}_{n,k} \subseteq \mathcal{D}_n$, conditioning on the full multitask data cannot increase conditional variance. Hence

$$\big[\mathbf{K}_n(\mathbf{x},\mathbf{x})\big]_{kk} = \mathrm{Var}(Y_k(\mathbf{x}) \mid \mathcal{D}_n) \le \mathrm{Var}(Y_k(\mathbf{x}) \mid \mathcal{D}_{n,k}) = \sigma^2_{n,k}(\mathbf{x}) \longrightarrow 0.$$

This holds for every $k \in [K]$. Since $\mathbf{K}_n(\mathbf{x},\mathbf{x})$ is positive semidefinite,

$$\big|\big[\mathbf{K}_n(\mathbf{x},\mathbf{x})\big]_{k\ell}\big| \le \sqrt{\big[\mathbf{K}_n(\mathbf{x},\mathbf{x})\big]_{kk}\big[\mathbf{K}_n(\mathbf{x},\mathbf{x})\big]_{\ell\ell}},$$

so every off-diagonal entry also converges to zero. Therefore,

$$\mathbf{K}_n(\mathbf{x},\mathbf{x}) \longrightarrow \mathbf{0}_{K\times K} \qquad \text{for every fixed } \mathbf{x} \in \mathcal{X}.$$

Next, let

$$\mathcal{X}^\star = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\} \subset \mathcal{X}$$

be any finite set, and define the stacked latent vector

$$\mathbf{Z} := \mathbf{Y}(\mathcal{X}^\star) := \big(\mathbf{Y}(\mathbf{x}^{(1)})^\top, \dots, \mathbf{Y}(\mathbf{x}^{(m)})^\top\big)^\top \in \mathbb{R}^{mK}.$$

Under the multitask GP posterior,

$$\mathbf{Z} \mid \mathcal{D}_n \sim \mathcal{N}(\boldsymbol{\mu}_n^\star, \mathbf{K}_n^\star),$$

where

$$\boldsymbol{\mu}_n^\star := \mathbb{E}[\mathbf{Z} \mid \mathcal{F}_n], \qquad \mathbf{K}_n^\star := \mathrm{Cov}(\mathbf{Z} \mid \mathcal{F}_n).$$

Applying the previous variance argument to each component $Y_k(\mathbf{x}^{(j)})$ gives

$$\mathrm{tr}(\mathbf{K}_n^\star) \longrightarrow 0.$$

Since $\mathbf{K}_n^\star$ is positive semidefinite, this also implies

$$\mathbf{K}_n^\star \longrightarrow \mathbf{0}_{mK\times mK}.$$

It remains to show that the posterior mean converges to the true latent values. By correct specification, the true objective vector $\mathbf{Y}_0(\mathcal{X}^\star)$ is the realized value of $\mathbf{Z}$ under the same GP data-generating law. Moreover,

$$\boldsymbol{\mu}_n^\star = \mathbb{E}[\mathbf{Z} \mid \mathcal{F}_n]$$

is an L^2 martingale with respect to $(\mathcal{F}_n)_{n\ge1}$. Since

$$\mathbb{E}\big[\|\mathbf{Z} - \boldsymbol{\mu}_n^\star\|_2^2 \mid \mathcal{F}_n\big] = \mathrm{tr}(\mathbf{K}_n^\star) \longrightarrow 0,$$

and the conditional variances are bounded above by the prior second moment, dominated convergence gives

$$\mathbb{E}\big[\|\mathbf{Z} - \boldsymbol{\mu}_n^\star\|_2^2\big] \longrightarrow 0.$$

The martingale convergence theorem implies that $\boldsymbol{\mu}_n^\star$ converges almost surely and in L^2 to $\mathbb{E}[\mathbf{Z} \mid \mathcal{F}_\infty]$, where $\mathcal{F}_\infty := \sigma(\cup_{n\ge1}\mathcal{F}_n)$. Since the L^2 limit is unique and $\boldsymbol{\mu}_n^\star \to \mathbf{Z}$ in L^2, we obtain

$$\boldsymbol{\mu}_n^\star \longrightarrow \mathbf{Z} = \mathbf{Y}_0(\mathcal{X}^\star) \qquad \text{almost surely.}$$

Finally, for any $\epsilon > 0$,

$$\Pi(\|\mathbf{Y}(\mathcal{X}^\star) - \mathbf{Y}_0(\mathcal{X}^\star)\|_2 > \epsilon \mid \mathcal{D}_n) \\ \leq \Pi(\|\mathbf{Y}(\mathcal{X}^\star) - \boldsymbol{\mu}_n^\star\|_2 > \epsilon/2 \mid \mathcal{D}_n) + \mathbf{1}\{\|\boldsymbol{\mu}_n^\star - \mathbf{Y}_0(\mathcal{X}^\star)\|_2 > \epsilon/2\}.$$

By Chebyshev's inequality,

$$\Pi(\|\mathbf{Y}(\mathcal{X}^\star) - \boldsymbol{\mu}_n^\star\|_2 > \epsilon/2 \mid \mathcal{D}_n) \leq \frac{4\ \mathrm{tr}(\mathbf{K}_n^\star)}{\epsilon^2} \longrightarrow 0,$$

and the indicator term converges to zero almost surely because $\boldsymbol{\mu}_n^\star \to \mathbf{Y}_0(\mathcal{X}^\star)$ almost surely. Therefore,

$$\Pi(\|\mathbf{Y}(\mathcal{X}^\star) - \mathbf{Y}_0(\mathcal{X}^\star)\|_2 > \epsilon \mid \mathcal{D}_n) \longrightarrow 0 \qquad \text{almost surely.}$$

Since $\mathcal{X}^\star$ was arbitrary, the same argument applied to a singleton $\mathcal{X}^\star = \{\mathbf{x}\}$ gives, for every fixed $\mathbf{x} \in \mathcal{X}$,

$$\boldsymbol{\mu}_n(\mathbf{x}) \longrightarrow \mathbf{Y}_0(\mathbf{x}), \qquad \mathbf{K}_n(\mathbf{x}, \mathbf{x}) \longrightarrow \mathbf{0}_{K\times K}.$$

This proves posterior consistency under decoupled oracle evaluations. □

Theorem A.3 (Finite-sample information transfer under decoupling)**.** *Fix a round n, a candidate design $\mathbf{x} \in \mathcal{X}$, and suppose that the K-oracle multitask posterior at $\mathbf{x}$ satisfies*

$$Y_{1:K}(\mathbf{x}) \mid D_n \sim \mathcal{N}(\mu_n(\mathbf{x}), \Sigma_n(\mathbf{x})), \qquad \Sigma_n(\mathbf{x}) \succ 0.$$

Let $\mathcal{K} \triangleq \{1, \ldots, K\}$. For any nonempty proper subset $S \subsetneq \mathcal{K}$, write

$$\bar{S} \triangleq \mathcal{K} \setminus S.$$

Suppose that a decoupled evaluation observes the selected oracle values through

$$Z_S(\mathbf{x}) = Y_S(\mathbf{x}) + \varepsilon_S, \qquad \varepsilon_S \sim \mathcal{N}(0, \Sigma_{\varepsilon,SS}),$$

where ε_S is independent of $Y_{1:K}(\mathbf{x}) \mid D_n$ and $\Sigma_{\varepsilon,SS} \succeq 0$. Define the expected one-step posterior entropy reduction about the unevaluated oracles $Y_{\bar{S}}(\mathbf{x})$ by

$$\Delta_n(S; \mathbf{x}) \triangleq H(Y_{\bar{S}}(\mathbf{x}) \mid D_n) - \mathbb{E}_{Z_S(\mathbf{x})|D_n}\left[H(Y_{\bar{S}}(\mathbf{x}) \mid D_n, Z_S(\mathbf{x}))\right].$$

Then

$$\Delta_n(S; \mathbf{x}) = I(Z_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid D_n),$$

and this quantity has the closed form

$$\Delta_n(S; \mathbf{x}) = \frac{1}{2} \log \frac{\left|\Sigma_{n,\bar{S}\bar{S}}(\mathbf{x})\right|}{\left|\Sigma_{n,\bar{S}\bar{S}}(\mathbf{x}) - \Sigma_{n,\bar{S}S}(\mathbf{x}) \left(\Sigma_{n,SS}(\mathbf{x}) + \Sigma_{\varepsilon,SS}\right)^{-1} \Sigma_{n,S\bar{S}}(\mathbf{x})\right|}.$$

Equivalently,

$$\Delta_n(S; \mathbf{x}) = \frac{1}{2} \log \frac{\left|\Sigma_{n,SS}(\mathbf{x}) + \Sigma_{\varepsilon,SS}\right| \left|\Sigma_{n,\bar{S}\bar{S}}(\mathbf{x})\right|}{\left|\begin{bmatrix} \Sigma_{n,SS}(\mathbf{x}) + \Sigma_{\varepsilon,SS} & \Sigma_{n,S\bar{S}}(\mathbf{x}) \\ \Sigma_{n,\bar{S}S}(\mathbf{x}) & \Sigma_{n,\bar{S}\bar{S}}(\mathbf{x}) \end{bmatrix}\right|}.$$

Consequently, for any fixed decoupled oracle budget $M \in \{1, \ldots, K-1\}$, the subset

$$S_n^\star(\mathbf{x}) \in \arg\max_{S \subseteq \mathcal{K}:\ |S| = M} I(Z_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid D_n)$$

maximizes the expected one-step posterior entropy reduction about the unevaluated oracles.

In the noiseless latent-oracle case $\Sigma_{\varepsilon,SS} = 0$, this reduces to

$$\Delta_n(S; \mathbf{x}) = I(Y_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid D_n) = \frac{1}{2} \log \frac{\left|\Sigma_{n,SS}(\mathbf{x})\right| \left|\Sigma_{n,\bar{S}\bar{S}}(\mathbf{x})\right|}{\left|\Sigma_n(\mathbf{x})\right|},$$

which is exactly the Gaussian mutual-information criterion used for subset selection.

Moreover, let

$$\mathrm{TC}_n(\mathbf{x}) \triangleq \sum_{k=1}^{K} H(Y_k(\mathbf{x}) \mid D_n) - H(Y_{1:K}(\mathbf{x}) \mid D_n),$$

and define $\mathrm{TC}_{n,S}(\mathbf{x})$ *and* $\mathrm{TC}_{n,\bar{S}}(\mathbf{x})$ *analogously for the subvectors* $Y_S(\mathbf{x})$ *and* $Y_{\bar{S}}(\mathbf{x})$*. Then, for every* $S \subsetneq \mathcal{K}$*,*

$$0 \le \Delta_n(S;\mathbf{x}) \le I(Y_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid D_n) = \mathrm{TC}_n(\mathbf{x}) - \mathrm{TC}_{n,S}(\mathbf{x}) - \mathrm{TC}_{n,\bar{S}}(\mathbf{x}) \le \mathrm{TC}_n(\mathbf{x}).$$

Thus, if $\mathrm{TC}_n(\mathbf{x}) < \tau$*, then no decoupled* M*-oracle evaluation at* $\mathbf{x}$ *can reduce the posterior entropy of the unevaluated oracles by more than* τ *nats. Finally,*

$$\Delta_n(S;\mathbf{x}) = 0 \quad \Longleftrightarrow \quad \Sigma_{n,S\bar{S}}(\mathbf{x}) = 0,$$

so a decoupled evaluation transfers strictly positive information to the unevaluated oracles exactly when the selected and unevaluated oracle blocks have nonzero posterior cross-covariance at $\mathbf{x}$*.*

Proof. Condition throughout on $\mathcal{D}_n$. Since

$$\begin{bmatrix} Z_S(\mathbf{x}) \\ Y_{\bar{S}}(\mathbf{x}) \end{bmatrix}$$

is jointly Gaussian, the conditional covariance of $Y_{\bar{S}}(\mathbf{x})$ after observing $Z_S(\mathbf{x})$ is

$$\Sigma_{n,\bar{S}\bar{S}}(\mathbf{x}) - \Sigma_{n,\bar{S}S}(\mathbf{x}) \left(\Sigma_{n,SS}(\mathbf{x}) + \Sigma_{\varepsilon,SS}\right)^{-1} \Sigma_{n,S\bar{S}}(\mathbf{x}).$$

This covariance does not depend on the realized value of $Z_S(\mathbf{x})$. Applying the Gaussian entropy formula

$$H(W) = \frac{1}{2} \log((2\pi e)^m |\Sigma_W|)$$

therefore gives the stated expression for $\Delta_n(S;\mathbf{x})$. Since expected entropy reduction is mutual information,

$$\Delta_n(S;\mathbf{x}) = I(Z_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}_n)\,.$$

Maximizing this quantity over all subsets $S \subseteq \mathcal{K}$ with $|S| = M$ therefore maximizes the expected one-step posterior entropy reduction about the unevaluated oracles.

For the total-correlation bound, the observation model gives the Markov relation

$$Y_{\bar{S}}(\mathbf{x}) \longrightarrow Y_S(\mathbf{x}) \longrightarrow Z_S(\mathbf{x})$$

conditional on $\mathcal{D}_n$. Hence, by the data-processing inequality,

$$I(Z_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}_n) \le I(Y_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}_n)\,.$$

Using

$$\mathrm{TC}_n(\mathbf{x}) = \sum_{k=1}^{K} H(Y_k(\mathbf{x}) \mid \mathcal{D}_n) - H(Y_{1:K}(\mathbf{x}) \mid \mathcal{D}_n)\,,$$

and the analogous definitions for $\mathrm{TC}_{n,S}(\mathbf{x})$ and $\mathrm{TC}_{n,\bar{S}}(\mathbf{x})$, direct cancellation yields

$$I(Y_S(\mathbf{x}); Y_{\bar{S}}(\mathbf{x}) \mid \mathcal{D}_n) = \mathrm{TC}_n(\mathbf{x}) - \mathrm{TC}_{n,S}(\mathbf{x}) - \mathrm{TC}_{n,\bar{S}}(\mathbf{x}).$$

Nonnegativity of total correlation gives the final upper bound by $\mathrm{TC}_n(\mathbf{x})$. Finally, for jointly Gaussian random vectors, zero mutual information is equivalent to zero cross-covariance. Since

$$\mathrm{Cov}(Z_S(\mathbf{x}), Y_{\bar{S}}(\mathbf{x}) \mid D_n) = \Sigma_{n,S\bar{S}}(\mathbf{x}),$$

we have $\Delta_n(S;\mathbf{x}) = 0$ if and only if $\Sigma_{n,S\bar{S}}(\mathbf{x}) = 0$. □