

# RANDOMIZED MAXIMUM LIKELIHOOD VIA HIGH-DIMENSIONAL BAYESIAN OPTIMIZATION

Valentin Breaz

Richard Wilkinson

Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP\*  
LJK, Grenoble, France

School of Mathematical Sciences  
University of Nottingham, UK

## ABSTRACT

Posterior sampling for high-dimensional Bayesian inverse problems is a common challenge in real-world applications. Randomized Maximum Likelihood (RML) is an optimization based methodology that gives samples from an approximation to the posterior distribution. We develop a high-dimensional Bayesian Optimization (BO) approach based on Gaussian Process (GP) surrogate models to solve the RML problem. We demonstrate the benefits of our approach in comparison to alternative optimization methods on a variety of synthetic and real-world Bayesian inverse problems, including medical and magnetohydrodynamics applications.

**Index Terms**— Bayesian Optimization, Gaussian Processes, Randomized Maximum Likelihood, Inverse Problems

## 1. INTRODUCTION

We consider Bayesian inverse problems, where the goal is to sample from the posterior distribution  $p(x|\mathcal{D}) = \frac{p(\mathcal{D}|x)p(x)}{p(\mathcal{D})}$  of the unknown parameters  $x \in \mathbb{R}^D$  given the observed data  $\mathcal{D} \in \mathbb{R}^m$ . Typically, the likelihood is assumed to be a Gaussian distribution  $\mathcal{D}|x \sim \mathcal{N}_m(f(x), \Sigma_{\text{obs}})$ , where  $f(x) : \mathbb{R}^D \rightarrow \mathbb{R}^m$  is known as the simulator;  $f$  is often computationally expensive to evaluate and it usually models an underlying physical process. The covariance matrix  $\Sigma_{\text{obs}}$  describes the modelling and observational errors. Bayesian inverse problems are encountered in applications such as climate modelling [1], medical imaging [2], and material sciences [3].

Randomized Maximum Likelihood (RML) was introduced in [4, 5], as an approximate posterior sampling methodology. RML is formulated for the situation where the observations are subject to Gaussian distributed errors (i.e., a Gaussian likelihood) and where the prior distribution is also Gaussian:  $x \sim \mathcal{N}_D(\mu, \Sigma)$ . The algorithm proceeds by first perturbing the data and the prior mean, and then optimizing the unnormalised log-posterior using these perturbed values. See Algorithm 1.

This work was partially funded by TotalEnergies.

---

## Algorithm 1 Randomized Maximum Likelihood (RML)

---

$n_{RML}$  : number of samples required

**for**  $n \in [n_{RML}]$  **do**

1. Sample  $\mathcal{D}_n \sim \mathcal{N}_m(\mathcal{D}, \Sigma_{\text{obs}})$  from the Gaussian likelihood
2. Sample  $\mu_n \sim \mathcal{N}_D(\mu, \Sigma)$  from the Gaussian prior
3. Construct  $O_n(x) := \log p(\mathcal{D}|x)p(x)$  w.r.t. the randomizations  $(\mathcal{D}_n, \mu_n)$

$$O_n(x) = \log \mathcal{N}_m(\mathcal{D}_n|f(x), \Sigma_{\text{obs}}) + \log \mathcal{N}_D(x|\mu_n, \Sigma) \quad (1)$$

4. Obtain  $x_n^*$  as the maximizer  $x_n^* = \arg \max_x O_n(x)$ .

**end for**

---

Here, we use the notation  $[n_{RML}] = \{1, 2, \dots, n_{RML}\}$ . Thus, RML solves  $n_{RML}$  optimization problems, each with a different objective function,  $O_n(x)$ . The resulting solutions  $\{x_n^*\}_{n=1}^{n_{RML}}$  are approximate samples from the posterior distribution  $p(x|\mathcal{D})$ . The samples are exact draws from the posterior if the simulator  $f(x)$  is linear; see Appendix A for a simple proof, or [6] for the original proof treating a more general case. Nonetheless, good practical performance has been observed for nonlinear (deep) neural network parametrized simulators [7]; see also [8] for a modification of RML for multi-modal posteriors with highly nonlinear simulators.

Instead of focusing on the accuracy of the approximate samples  $\{x_n^*\}_{n=1}^{n_{RML}}$  with respect to the true posterior, we address solving the optimization problems (1) efficiently in the challenging case of a high-dimensional input space  $\mathbb{R}^D$ . We focus on the specific scenario where the log-likelihood

$$\log p(\mathcal{D}|x) \propto L(x) := -\|\mathcal{D} - f(x)\|_{\Sigma_{\text{obs}}}^2 \quad (2)$$

has a low-dimensional active subspace [9], where we use the notational convention that  $\|y\|_{\Sigma}^2 = y^T \Sigma^{-1} y$ . In other words, we assume  $L(x) \approx g(A^T x)$ , where  $g : \mathbb{R}^d \rightarrow \mathbb{R}$  with  $d \ll D$ , and  $A \in \mathbb{R}^{D \times d}$  is a semi-orthogonal matrix (i.e.,  $A^T A = I_d$ );  $A$  is known as the active subspace. As pointed out in [9], many log-likelihoods used in inverse problems possess such a low-dimensional structure, e.g., in medical applications [10].

Bayesian Optimization (BO, i.e., finding  $\arg \max_x O(x)$ )

for a generic objective function  $O(x)$  using a Gaussian process surrogate model  $g_{\text{GP}}(x) \approx O(x)$  is based on a standard exploration-exploitation principle. Namely, an acquisition function based on  $g_{\text{GP}}(x)$  is used such that in the exploration phase, the target function  $O(x)$  is explored globally, whereas in the exploitation phase, points  $\tilde{x}$  that are likely to satisfy  $\tilde{x} = \arg \max_x O(x)$  are sampled until the maximum is eventually found. For an introduction, see [11] for BO, and [12] for Gaussian processes (GPs). In our setting, we deal with high-dimensional objectives  $O(x)$  with an active subspace; however, we do not assume access to the active subspace. The most common solution is the use of random projections (embeddings) instead of the true low-dimensional active subspace; see Algorithm 2, and [13] for a seminal work.

---

**Algorithm 2** Standard high-dimensional BO algorithm with random embeddings for a generic objective function  $O(\cdot)$

---

$M$  : number of evaluations possible for a generic objective function  $O(\cdot)$  given the computational budget;

$d_e$  : chosen dimensionality of the embedding  $R$ ;

$R \in \mathbb{R}^{D \times d_e}$  : random embedding;

$m_0$  : initial training points  $\{y_i, O(Ry_i)\}_{i=1}^{m_0}$ ,  $y_i \in \mathbb{R}^{d_e}$

**for**  $m \in \{m_0 + 1, \dots, M\}$  **do**

1. Construct a GP approximation  $O(Ry) \sim \text{GP}$  using the available objective function evaluations  $\{y_i, O(Ry_i)\}_{i=1}^{m-1}$

2. Select  $y_m = \arg \max_y a_m(y)$  as the maximizer of a BO acquisition function  $a_m(y)$  for  $O(Ry) \sim \text{GP}$

3. Update the training data to  $\{y_i, O(Ry_i)\}_{i=1}^m$ .

**end for**

Obtain  $x_\star = Ry_{m_\star}$  as the maximizer

$$m_\star = \arg \max_m O(Ry_m), \quad m \leq M.$$


---

Although there is an extensive literature for high dimensional BO (HD-BO) with random embeddings, which offers theoretical guarantees and good practical performance (see [11] for a recent survey on the topic), there is no methodology designed for posterior sampling. Yet RML is a natural way to use HD-BO for high-dimensional posterior sampling. The closest related work to ours is [14], where BO is compared with alternative gradient-free optimization methods for maximizing the log-likelihood in low-dimensional inverse problems. We extend their work by considering the multi-objective setting ( $n_{\text{RML}}$  randomized log-likelihoods) in order to do posterior sampling, as well as considering high-dimensional inverse problems with high-dimensional priors.

**Contributions.** We introduce a new methodology for high-dimensional posterior sampling via BO (Algorithm 3); we propose a natural way to exploit the shared simulator  $f(x)$  that is present in all of the objective functions (1), as well as an adjustment needed to incorporate a prior distribution without a low-dimensional structure. By using random

embeddings, we sidestep the difficulty of estimating the active subspace, which might be impossible under tight computational budgets. We show that in the limited budget setting, our methodology usually outperforms alternative gradient-free optimization methods, according to a series of synthetic and real-world experiments. Finally, we show that the samples produced by our method are indeed close to ‘true’ RML samples (collected via an infinite computational budget), whilst also covering well the high posterior density regions.

## 2. METHODOLOGY

Firstly, we consider the setting of a uniform prior,  $x \sim U[a_i, b_i]_{i=1}^D$ , where  $[a_i, b_i]_{i=1}^D := [a_1, b_1] \times \dots \times [a_D, b_D]$ . In this case, the posterior distribution is proportional to the likelihood in the region of the prior support, and using our active subspace assumption the RML objectives (1) become

$$O_n(x) \propto L_n(x) := \log \mathcal{N}(f(x) | \mathcal{D}_n, \Sigma_{\text{obs}}) \approx g_n(A_n^T x), \quad (3)$$

where the objective functions now need to be maximized over the prior support  $[a_i, b_i]_{i=1}^D \subseteq \mathbb{R}^D$ . Note that we do not assume knowledge of the active subspaces  $A_n \in \mathbb{R}^{D \times d}$  or the low-dimensional link functions  $g_n : \mathbb{R}^d \rightarrow \mathbb{R}$ .

We propose our procedure HD-BO-RML in Algorithm 3. Following the suggestion of [13], we use a collection of  $K$  interleaved random embeddings. Algorithm 3 is based on a cyclic pass through all the objective functions (3), where for every random embedding  $k \in [K]$ , the simulations  $\{(y_m^k, f(R_k y_m^k))\}_{m=1}^n$  collected up to some iteration  $n$  will serve as the basis for the training set used in the next iteration, where the objective function  $O_{(n+1)'}(x)$  is considered, with  $(n+1)' := n+1 \pmod{n_{\text{RML}}}$ .

Given a budget of  $N$  simulator evaluations, our procedure performs  $T := N/n_{\text{RML}}$  HD-BO iterations for each objective. If  $N$  (and thus  $T$ ) is large enough, then HD-BO-RML benefits from the same theoretical guarantees as a (vanilla) HD-BO algorithm for each objective function (e.g., Theorem 11 in [13]). However, instead of performing the  $T$  HD-BO iterations independently for each objective, we further exploit that the simulator function  $f(x)$  is present in all the objectives  $L_n(x) := \log \mathcal{N}(f(x) | \mathcal{D}_n, \Sigma_{\text{obs}})$ . Namely, with one HD-BO iteration for  $L_j(x)$ , where some point  $y_j^k$  is selected by maximizing the acquisition function, we can use  $(y_j^k, f(R_k y_j^k))$  to obtain (training) data for all the objectives. The next HD-BO iteration for  $L_j(x)$  will come after  $n_{\text{RML}} - 1$  steps, and instead of having only one training point  $(y_j^k, L_j(R_k y_j^k))$ , we will have  $n_{\text{RML}} - 1$  extra points  $\{(y_n^k, L_j(R_k y_n^k))\}_{n=j+1}^{n_{\text{RML}}+j-1}$  obtained from all the other objectives during the cyclic pass.

What we tend to observe in experiments is that during the first few HD-BO iterations (exploration phase), the selected points  $(y_n^k, f(R_k y_n^k))$  are informative of the overall landscape of the posterior space (i.e., they separate the low-posterior density regions from the high-posterior density re-

gions), whereas training points selected during the exploitation phase are informative for the high-posterior density regions, and thus are beneficial for every RML objective.

---

**Algorithm 3** HD-BO-RML (Uniform or Gaussian priors; the additional steps for Gaussian priors are shown in parentheses)

---

$N$  : max possible number of evaluations of  $f(\cdot)$ ;  
 $d_e$  : choice of embedding dimensionality;  
 $R_1, \dots, R_K \in \mathbb{R}^{D \times d_e}$  : random embeddings;  
 $n_0 \times K$  initial points:  $\{y_i^k, f(R_k y_i^k)\}_{i=1}^{n_0}$ , with  $y_i^k \in \mathbb{R}^{d_e}$ ,  $k \in [K]$ ;  
**for**  $k \in \{1, \dots, K\}$  **do**  
  **for**  $n \in \{n_0 + 1, \dots, \lfloor N/K \rfloor\}$  ( $n \in \{n_0 + 1, \dots, \lfloor N/2K \rfloor\}$  in case of a Gaussian prior) **do**  
    1. Let  $n' := n \bmod n_{RML}$   
    2. Construct a GP approximation to  $L_{n'}(R_k y)$  using the existing simulations  $\{y_i^k, f(R_k y_i^k)\}_{i=1}^{n-1}$   
    3. Select  $y_n^k = \arg \max_y a_n^k(y)$  as the maximizer of a BO acquisition function using the GP approximation  
    4. Perform  $f(R_k y_n^k)$  and update the shared simulation ensemble (i.e., training set) to  $\{y_i^k, f(R_k y_i^k)\}_{i=1}^n$   
    (Gaussian prior) 5G. Perform local optimization in  $\mathbb{R}^D$  around  $x_0 = R_k y_n^k$  with respect to the prior  $p_{n'}(x) = \mathcal{N}_D(x|\mu_{n'}, \Sigma)$   
  **end for**  
**end for**  
**for**  $n \in \{1, \dots, n_{RML}\}$  **do**  
  (Uniform prior) 1U. Obtain  $x_n^* = R_{k_*} y_{m_*}^{k_*}$  as the maximizer  

$$k_*, m_* = \arg \max_{k, m} O_n(R_k y_m^k), k \in [K], m \leq \lfloor N/K \rfloor$$
  
  (Gaussian prior) 1G. Writing  $z_m^k$  for the local maxima from step 5G. above, obtain  $x_n^* = z_{m_*}^{k_*}$  via  

$$k_*, m_* = \arg \max_{k, m} O_n(z_m^k), k \in [K], m \leq \lfloor N/2K \rfloor$$
  
**end for**

---

Using a Gaussian prior distribution,  $x \sim \mathcal{N}_D(\mu, \Sigma)$ , rather than a uniform prior, requires a modification of the algorithm. Using again the active subspace assumption, the objective functions (1) become:

$$O_n(x) = L_n(x) + \log p_n(x) \approx g_n(A_n^T x) + \log \mathcal{N}_D(x|\mu_n, \Sigma).$$

Due to the potential lack of a low-dimensional (linear) subspace in the prior, as for example if  $x \sim \mathcal{N}(0, I_D)$ , running HD-BO by directly modelling  $O_n(Ry)$  as a GP might be unsatisfactory, and hence we cannot simply use the algorithm from the uniform prior case. Steps 5G and 1G in Algorithm 3 propose a heuristic solution. While we keep performing HD-BO with respect to the log-likelihood

$L_{n'}(x) \approx g_{n'}(A_{n'}^T x)$  as in the uniform prior case, we try to increase the prior value for the points  $y_n^k = \arg \max_y a_n^k(y)$  of potentially high-likelihood  $L_{n'}(R_k y_n^k)$  by carefully performing local optimization with respect to the prior  $p_{n'}(x) = \mathcal{N}_D(x|\mu_{n'}, \Sigma)$  in the high-dimensional space  $\mathbb{R}^D$ , starting from  $x_0 := R_k y_n^k$ ; the resulting local maxima  $z_n^k$  try to achieve  $L_{n'}(z_n^k) + p_{n'}(z_n^k) > L_{n'}(R_k y_n^k) + p_{n'}(R_k y_n^k)$  (i.e.,  $O_{n'}(z_n^k) > O_{n'}(R_k y_n^k)$ ). This inequality was indeed observed in our experiments; during the exploitation stage,  $z_n^k$  managed to achieve a trade-off between high-likelihood and high-prior. Note that the local maximization (step 5G) does not require simulator evaluations, and any local optimization method can be applied without any significant costs.

### 3. EXPERIMENTS

We now give empirical results showing that the proposed algorithm performs well in comparison with competing methods in a variety of synthetic and real-world Bayesian inverse problems. We use four simulators from the Active Subspaces github page maintained by [15]:

- **Elliptic-PDE**  $f : \mathbb{R}^{100} \rightarrow \mathbb{R}^7$ . This is a widely used exemplar for Bayesian inverse problems.
- **Ebola**:  $f : \mathbb{R}^8 \rightarrow \mathbb{R}$ , an 8-parameter dynamical system model for the spread of Ebola in Liberia [16].
- **MHD**:  $f : \mathbb{R}^5 \rightarrow \mathbb{R}$ , a 5-parameter magnetohydrodynamics power generation model [17].
- **HIV long-term model**:  $f : \mathbb{R}^{27} \rightarrow \mathbb{R}$  is the cell count at time  $t = 24$  days [10].

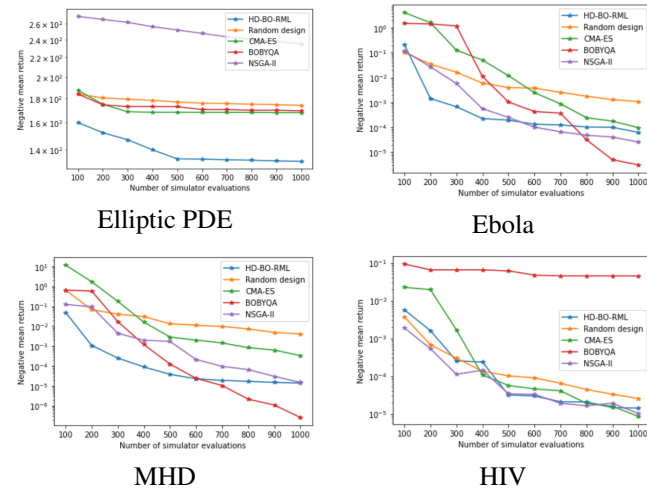
In each case, the log-likelihood has a low-dimensional active subspace  $L(x) \approx g(A^T x)$  with  $g : \mathbb{R} \rightarrow \mathbb{R}$  for the HIV model, and  $g : \mathbb{R}^2 \rightarrow \mathbb{R}$  for the other simulators. We do not assume knowledge of  $A$ . We use a standard Gaussian prior  $x \sim \mathcal{N}_{100}(0, I)$  for the elliptic-PDE, and a uniform prior distribution for the other simulators. For each experiment, we use RML to approximately sample from the posterior distribution, comparing our high dimensional Bayesian optimization (HD-BO) approach **HD-BO-RML** against the following gradient-free optimization methods: **BOBYQA** [18], a trust-region algorithm, used for method comparison versus HD-BO in [19]; **CMA-ES** [20], an evolution strategy algorithm, used for method comparison in [19, 11]; **NSGA-II** [21], an evolutionary algorithm for multi-objective optimization, used for method comparison versus HD-BO in [22]; and **random design** [13], a standard baseline for comparison versus BO. We measure performance of the different methods by comparing the mean return, defined as:  $1/n_{RML} \sum_{n=1}^{n_{RML}} O_n(x_n^*)$ , where  $x_n^*$  is the approximate maximizer of  $O_n(x)$  as selected by the different optimization methods considered.

For HD-BO-RML, we use the Gaussian Process Upper Confidence Bound (GP-UCB) acquisition function [23], i.e.,  $a_n^k(y) = \mu_n^k(y) + \beta \sigma_n^k(y)$ , where  $\mu_n^k(y)$  is the GP predictive mean which tries to approximate the log-likelihood  $L_{n'}(R_k y)$

( $n' := n \bmod n_{RML}$ ), and  $\sigma_n^k(y)$  represents the GP predictive uncertainty. This acquisition function has been studied in the HD-BO literature with random embeddings [13]. We use a standard squared-exponential covariance function (RBF kernel), i.e.,  $\text{Cov}(O(Ry_1), O(Ry_2)) := o^2 \exp(-\frac{\|y_1 - y_2\|^2}{2l^2})$  for each GP model, where  $o$  and  $l$  are trainable hyperparameters known as the outputscale and lengthscale, respectively. For the random embeddings,  $R_k$ , each row is sampled independently from the uniform distribution on the unit hypersphere  $S^{d_e-1}$ , as suggested in [11]. We use  $K = 10$  random embeddings of low dimensionality  $d_e = d + 1$  and  $n_0 = 5$  initial points (where  $d$  is the dimension of the active subspace).

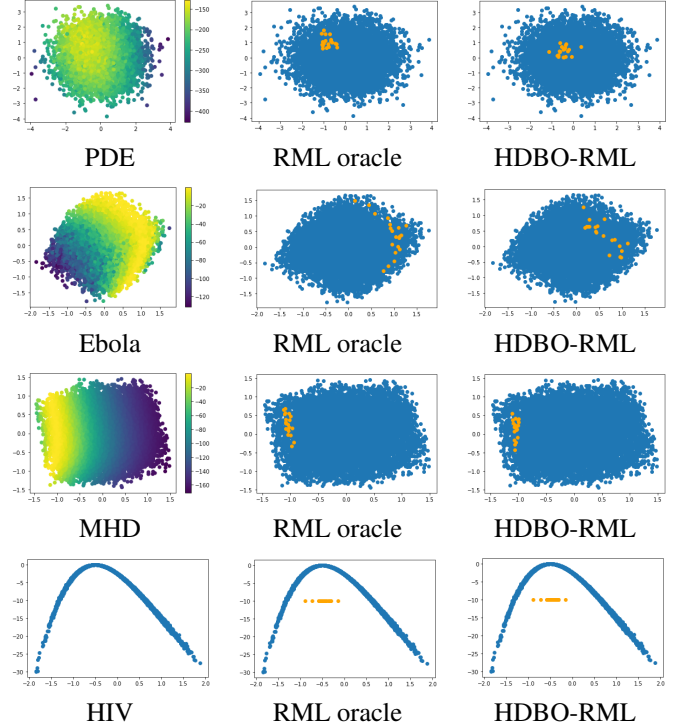
### 3.1. Results

Figure 1 shows the performance of each algorithm for different computational budgets, averaged over 5 trials. Each trial has a budget of  $N = 1000$  simulations in order to find  $n_{RML} = 20$  samples. We can see that HD-BO-RML has the best performance of all the methods for small budgets (i.e., small  $N$ ), and has comparable performance to other methods with larger budgets; in each experiment, HD-BO-RML is the best performing method across the full range of computational budgets for at least one of the 5 trials (and in all the 5 trials for the high-dimensional Elliptic PDE with  $D = 100$ ). With a fixed budget, the performance for HD-BO-RML should remain relatively stable as the number  $n_{RML}$  of RML samples required increases (since exploration/exploitation will still be performed jointly); CMA-ES and BOBYQA will perform significantly worse due to fewer iterations and no data sharing between objectives, while NSGA-II is known to struggle for a large number of objectives [24].



**Fig. 1.** Negative mean returns:  $-1/n_{RML} \sum_{n=1}^{n_{RML}} O_n(x_n^*)$  (lower is better) versus computational budget for each method

As in [9], we can visualize each posterior landscape  $p(x|\mathcal{D})$  in the active subspace. The plots in the left column of



**Fig. 2.** Posterior landscape in the active subspace (left), oracle RML samples (middle), HD-BO-RML samples (right). RML samples are shown in orange, with prior samples in blue.

Figure 2 show 10000 prior samples  $x_i \sim p(x)$  projected into the true two-dimensional (one-dimensional for HIV) active subspace  $A$ , coloured by their unnormalized log-posterior density, i.e.,  $\{A^T x_i, \log p(\mathcal{D}|x_i) + \log p(x_i)\}_{i=1}^{10000}$ . We use an oracle with access to an unlimited amount of simulator evaluations to obtain the ‘true’  $n_{RML} = 20$  RML samples; the resulting samples are represented in orange in the middle column of Figure 2. We see that the RML samples cover the high posterior density regions well in all the experiments. Finally, the right column of Figure 2 shows the analogous samples obtained from using HD-BO-RML (instead of the oracle optimizer), averaged over the 5 trials. We see that the HD-BO-RML samples are close to the oracle samples in the active subspace representation.

## 4. DISCUSSION

Our focus has been on the challenging and commonly occurring problem of posterior sampling using a complex simulator, but with limited computational budget and no access to gradient information. We have introduced an RML approach based on high-dimensional Bayesian optimization that outperforms competing gradient-free optimization methods when there are tight computational budget constraints. Here we assumed the active subspace is unknown, but if it

is known, the same BO approach can be used with the true active subspace instead of the random embeddings.

## 5. REFERENCES

- [1] Philip B. Holden, Neil Robert Edwards, J. S. Hensman, and Richard D. Wilkinson, “Abc for climate: Dealing with expensive simulators,” *Handbook of Approximate Bayesian Computation*, 2018.
- [2] Matthew M. Dunlop and Andrew M. Stuart, “The bayesian formulation of eit: Analysis and algorithms,” *arXiv: Probability*, 2015.
- [3] Marco A. Iglesias, Minh Park, and Michael V. Tretyakov, “Bayesian inversion in resin transfer molding,” *Inverse Problems*, 2018.
- [4] Peter K Kitanidis, “Quasi-linear geostatistical theory for inverting,” *Water resources research*, vol. 31, no. 10, pp. 2411–2419, 1995.
- [5] D. S. Oliver, N. He, and A. C. Reynolds, “Conditioning permeability fields to pressure data,” *European Association of Geoscientists & Engineers*, 1996.
- [6] Dean S Oliver, “On conditional simulation to inaccurate data,” *Mathematical Geology*, vol. 28, no. 6, pp. 811–817, 1996.
- [7] Meng Tang, Yimin Liu, and Louis J. Durlofsky, “A deep-learning-based surrogate model for data assimilation in dynamic subsurface flow problems,” *Journal of Computational Physics*, vol. 413, July 2020.
- [8] Yuming Ba, Jana de Wiljes, Dean S Oliver, and Sebastian Reich, “Randomized maximum likelihood based posterior sampling,” *Computational Geosciences*, pp. 1–23, 2022.
- [9] Paul G. Constantine, Carson Kent, and Tan Bui-Thanh, “Accelerating MCMC with active subspaces,” *arXiv e-prints*, p. arXiv:1510.00024, Sept. 2015.
- [10] Tyson Loudon and Stephen Pankavich, “Mathematical analysis and dynamic active subspaces for a long term model of hiv,” *Mathematical Biosciences and Engineering*, vol. 14, 04 2016.
- [11] Ben Letham, Roberto Calandra, Akshara Rai, and Eytan Bakshy, “Re-examining linear embeddings for high-dimensional bayesian optimization,” *Advances in neural information processing systems*, vol. 33, 2020.
- [12] Carl Edward Rasmussen and Christopher K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*, The MIT Press, 2005.
- [13] Ziyu Wang, Frank Hutter, Masrouf Zoghi, David Matheson, and Nando De Freitas, “Bayesian optimization in a billion dimensions via random embeddings,” *Journal of Artificial Intelligence Research*, vol. 55, 2016.
- [14] Hamidreza Hamdi, Ivo Couckuyt, Mario Costa Sousa, and Tom Dhaene, “Gaussian processes for history-matching: application to an unconventional gas reservoir,” *Computational Geosciences*, vol. 21, 04 2017.
- [15] P. Constantine and R. Howard, “Active subspaces data sets,” <https://github.com/paulcon/as-data-sets>.
- [16] Paul Diaz, Paul Constantine, Kelsey Kalmbach, Eric Jones, and Stephen Pankavich, “A modified seir model for the spread of ebola in western africa and metrics for resource allocation,” *Applied mathematics and computation*, vol. 324, pp. 141–155, 2018.
- [17] Andrew Glaws, Paul G Constantine, John N Shadid, and Timothy M Wildey, “Dimension reduction in magneto-hydrodynamics power generation models: Dimensional analysis and active subspaces,” *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 10, no. 5, pp. 312–325, 2017.
- [18] M. Powell, “A view of algorithms for optimization without derivatives,” *Mathematics TODAY*, vol. 43, 01 2007.
- [19] David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek, “Scalable global optimization via local bayesian optimization,” in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, Eds., 2019, vol. 32.
- [20] Nikolaus Hansen, Youhei Akimoto, and Petr Baudis, “CMA-ES/pycma on Github,” Zenodo, DOI:10.5281/zenodo.2559634, Feb. 2019.
- [21] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: Nsga-ii,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [22] Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy, “Multi-objective bayesian optimization over high-dimensional search spaces,” in *Uncertainty in Artificial Intelligence*. PMLR, 2022.
- [23] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger, “Gaussian process optimization in the bandit setting: No regret and experimental design,” in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, Madison, WI, USA, 2010, ICML’10, p. 1015–1022, Omnipress.

- [24] Mario Köppen and Kaori Yoshida, “Substitute distance assignments in nsga-ii for handling many-objective optimization problems,” in *Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization*, Berlin, Heidelberg, 2007, EMO’07, p. 727–741, Springer-Verlag.

### A. PROOF FOR THE LINEAR SIMULATOR CASE

In case of a linear simulator  $f(x) := Hx$ , Gaussian prior distribution  $x \sim \mathcal{N}(\mu, \Sigma)$ , and Gaussian likelihood  $\mathcal{D}|x \sim \mathcal{N}(Hx, \Sigma_{\text{obs}})$ , the posterior distribution  $p(x|\mathcal{D})$  is also Gaussian:

$$x|\mathcal{D} \sim \mathcal{N}(m, V),$$

where

$$V = (\Sigma^{-1} + H^\top \Sigma_{\text{obs}}^{-1} H)^{-1}$$

and  $m = V(\Sigma^{-1} \mu + H^\top \Sigma_{\text{obs}}^{-1} \mathcal{D})$ .

To see why Randomized Maximum Likelihood (RML) from Algorithm 1 produces exact samples from the posterior distribution in this case, we first recall the RML objective functions from (1):

$$O_n(x) := \log \mathcal{N}_m(Hx|\mathcal{D}_n, \Sigma_{\text{obs}}) + \log \mathcal{N}_D(x|\mu_n, \Sigma),$$

where  $\mu_n := \mu + \epsilon_n$  and  $\mathcal{D}_n := \mathcal{D} + \eta_n$ , with  $\epsilon_n \sim \mathcal{N}(0, \Sigma)$  and  $\eta_n \sim \mathcal{N}(0, \Sigma_{\text{obs}})$  independently.

Differentiating with respect to  $x$  gives

$$\nabla O_n(x) = 2\Sigma^{-1}x - 2\Sigma^{-1}\mu_n + 2H^\top \Sigma_{\text{obs}}^{-1}Hx - 2H^\top \Sigma_{\text{obs}}^{-1}\mathcal{D}_n$$

and setting this equal to zero and rearranging gives

$$\begin{aligned} x &= (\Sigma^{-1} + H^\top \Sigma_{\text{obs}}^{-1} H)^{-1} (\Sigma^{-1} \mu_n + H^\top \Sigma_{\text{obs}}^{-1} \mathcal{D}_n) \\ &= m + V(\Sigma^{-1} \epsilon_n + H^\top \Sigma_{\text{obs}}^{-1} \eta_n). \end{aligned}$$

The distribution of  $x$  can then easily be seen to be

$$\begin{aligned} x &\sim \mathcal{N}(m, V(\Sigma^{-1} \Sigma \Sigma^{-1} + H^\top \Sigma_{\text{obs}}^{-1} \Sigma_{\text{obs}} \Sigma_{\text{obs}}^{-1} H) V^\top) \\ &= \mathcal{N}(m, V), \end{aligned}$$

i.e.,  $x$  is a sample from the true posterior distribution.