Scalable Trust Region Bayesian Optimization with Product of Experts

Saulius Tautvaišas Julius Žilinskas

Institute of Data Science and Digital Technologies, Vilnius University

Bayesian Optimization

- Bayesian optimization (BO) is effective and popular approach for global optimization of black-box functions [2].
- Using BO we want to find an input $\mathrm{x} \in \mathcal{X}$ that maximizes real-valued black-box function $f\colon \mathcal{X} \to \mathbb{R}$ defined on a compact domain $\mathcal{X} \subseteq \mathbb{R}^D$

$$x^* = \mathrm{argmax}_{\mathrm{x}\in\mathcal{X}} f(x)$$

given noisy observations $y \sim \mathcal{N}(f(x), \sigma_\epsilon^2)$ with noise variance σ_ϵ^2.

- Build *probabilistic surrogate model* based on observations $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n$.
- Find the next candidate point x_{n+1} which maximizes the *acquisition function* $\alpha\colon \mathcal{X} \to \mathbb{R}$

$$\mathrm{x}_{n+1} = \mathrm{argmax}_{\mathrm{x}\in\mathcal{X}}\, \alpha(\mathrm{x}|\mathcal{D}_n)$$

.

Probabilistic Surrogate Models

Gaussian process

- A Gaussian process $GP(\mu, \kappa)$ is fully specified by a *mean function* $\mu(\cdot)$ and a *covariance function* $k(\cdot,\cdot)$ [3].
- The objective is to infer the latent function f from a training set $(\boldsymbol{X}, \boldsymbol{y})$ where $\boldsymbol{X} = \{x_i\}_{i=1}^n, \boldsymbol{y} = \{y_i\}_{i=1}^n$.
- GP posterior predictive distribution at a test point $p(f_*|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\theta}, x_*) = \mathcal{N}(\mu_*, \sigma_*^2)$ is Gaussian with the mean and variance given by

$$\mu_* = \boldsymbol{k}_{*n}\left(\boldsymbol{K}_{nn} + \sigma_\epsilon^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y}, \quad (1)$$

$$\sigma_*^2 = \boldsymbol{k}_{**} - \boldsymbol{k}_{*n}\left(\boldsymbol{K}_{nn} + \sigma_\epsilon^2 \boldsymbol{I}\right)^{-1} {\boldsymbol{k}_{*n}}^T, \quad (2)$$

where $\boldsymbol{k}_{*n} = k(x_*, \boldsymbol{X})$ and $\boldsymbol{k}_{**} = k(x_*, x_*)$.

> The main challenge of GP is that training requires the inversion and the determinant of $\boldsymbol{K}_{nn} + \sigma_\epsilon^2 \boldsymbol{I}$, which is frequently realised via the Cholesky decomposition with computational cost of $O(n^3)$. For this reason, training GP on large datasets is computationally intractable.

Generalized Product Of Experts

- Partitions the data into M subsets $\mathcal{D}^{(i)} = \left\{\boldsymbol{X}^{(i)}, \boldsymbol{y}^{(i)}\right\}$, where $1 \le i \le$ M, and train GP on $\mathcal{D}^{(i)}$ as an expert GP model [1].
- Predictive distribution of GP expert i conditioned on the related subset of the data $\mathcal{D}^{(i)}$ and test input $x_* \in \mathbb{R}^D$ is Gaussian $p_i\left(y_*|\mathcal{D}^{(i)}, x_*\right) \sim \mathcal{N}(\mu_i(x_*), \sigma_i^2(x_*))$ with mean and covariance

$$\mu_i(x_*) = \boldsymbol{k}_{*i}\left(\boldsymbol{K}_i + \sigma_{\epsilon,i}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y}_i, \quad (3)$$

$$\sigma_i^2(x_*) = \boldsymbol{k}_{**} - \boldsymbol{k}_{*i}\left(\boldsymbol{K}_i + \sigma_{\epsilon,i}^2 \boldsymbol{I}\right)^{-1} {\boldsymbol{k}_{*i}}^T + \sigma_{\epsilon,i}^2. \quad (4)$$

- The Generalized Product Of Expert (gPoE) model combines each individual GP expert prediction into the final aggregate model

$$p_{\mathcal{A}}(y_*|x_*, \mathcal{D}) = \prod_{i=1}^{M} p_i^{\alpha_i(x_*)}\left(y_*|x_*, \mathcal{D}^{(i)}\right), \quad (5)$$

which is again Gaussian $\mathcal{N}(\mu_{\mathcal{A}}(x_*), \sigma_{\mathcal{A}}^2(x_*))$ with mean and covariance given by

$$\mu_{\mathcal{A}} = \sigma_{\mathcal{A}}^2(x_*) \sum_{i=1}^{M} \alpha_i(x_*)\, \sigma_i^{-2}(x_*)\, \mu_i(x_*), \quad (6)$$

$$\sigma_{\mathcal{A}}^{-2}(x_*) = \sum_{i=1}^{M} \alpha_i(x_*)\, \sigma_i^{-2}(x_*). \quad (7)$$

The weight $\alpha_i(x_*)$ is a measure of reliability and controls the contribution of each expert i at test point x_*, where $\alpha_i(x_*) > 0$ and $\sum_{i=1}^{M} \alpha_i(x_*) = 1$.

- The factorization of the log-marginal likelihood degenerates the full covariance matrix $\mathbf{K}_{nn} = k(\boldsymbol{X}, \boldsymbol{X})$ into block-diagonal matrix:

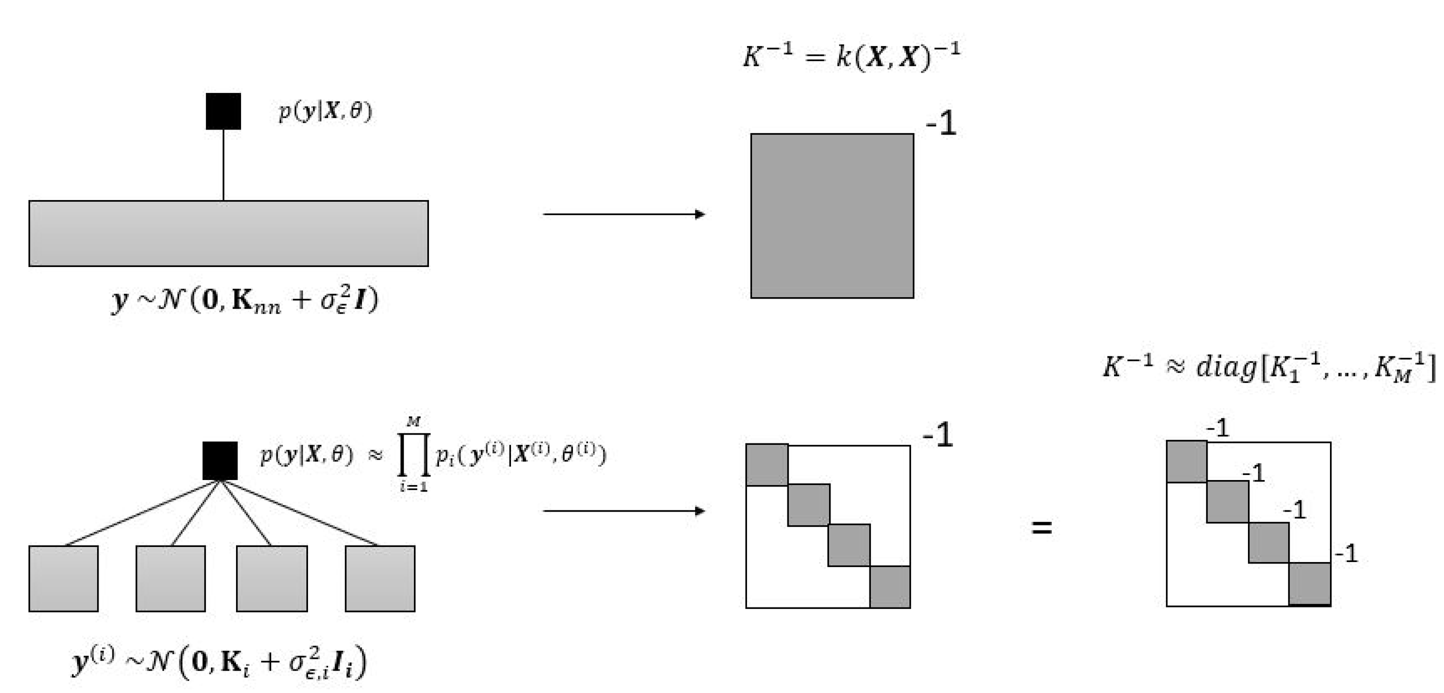

Figure 1. Block-diagonal covariance matrix.

> The gPoE reduces the training complexity time to $O(M{n_i}^3)$, where M is the number of experts and n_i is the number of training points assigned to the i GP expert. If we train GP experts in parallel with M compute nodes the training time complexity can be reduced to $O(n_i^3)$.

Trust region Bayesian optimization with Generalized PoE

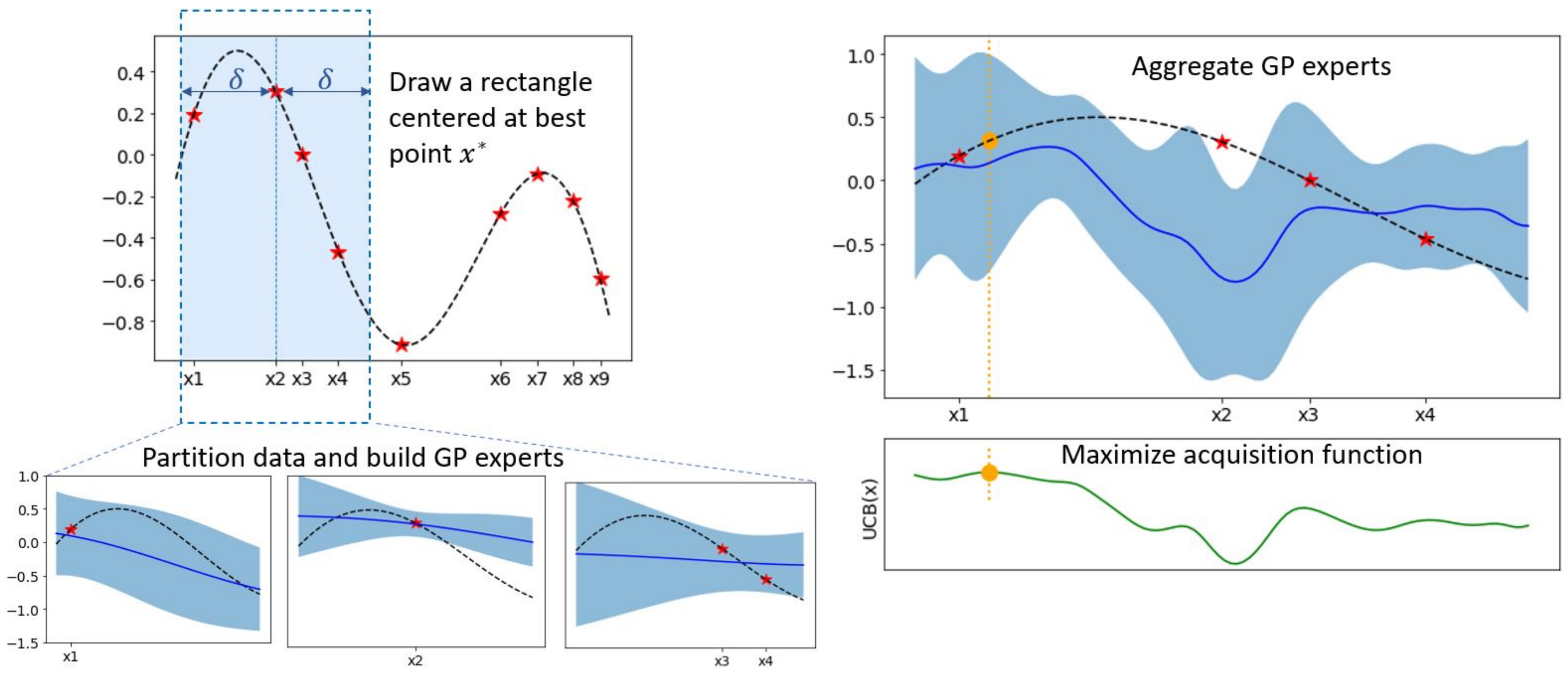

Figure 2. The workflow of **gPoETRBO** algorithm.

Algorithm 1 Generalized PoE based Trust Region Bayesian Optimization (gPoETRBO)

Input: Number of initializing points N, iterations T, points per expert n_i, initial TR parameters.
Output: The best recommendation x_T^*.
1: Randomly select and evaluate N points in the search space $\mathcal{D}_0 = \{(x_i, f(x_i))\}_{i=1}^N$.
2: **for** $t = 1$ to T **do**
3: Randomly partition $\mathcal{D}_{t-1}$ into $M = |D_{t-1}|/n_i$ subsets.
4: Train M local GP experts on $\{\mathcal{D}_{t-1}^i\}_{i=1}^M$ subsets.
5: Construct TR of length δ around the best point $x_t^* = \max_{1\le i\le|\mathcal{D}_{t-1}|} f(x_i)$.
6: Generate q candidate points $\mathrm{X}^c = \{x_1^c, \dots, x_q^c\}$ from $TR(x_t^*)$.
7: Evaluate i local GP expert posterior mean μ_t^i and variance σ_t^i on X^c points.
8: Aggregate $\mu_t^{\mathcal{A}}$ and $\sigma_t^{\mathcal{A}}$ using (6) and (7).
9: Maximize UCB acquisition function $\hat{x} = \mathrm{argmax}_{x\in\mathrm{X}^c}\, \mu_t^{\mathcal{A}}(x) + \sqrt{\beta}\sigma_t^{\mathcal{A}}(x)$
10: Evaluate the objective function $\hat{y} = f(\hat{x})$.
11: Add a new data point to the dataset $\mathcal{D}_t = \mathcal{D}_{t-1} \cup \{\hat{x}, \hat{y}\}$
12: Update the TR parameters and check whether to restart.

Numerical experiments

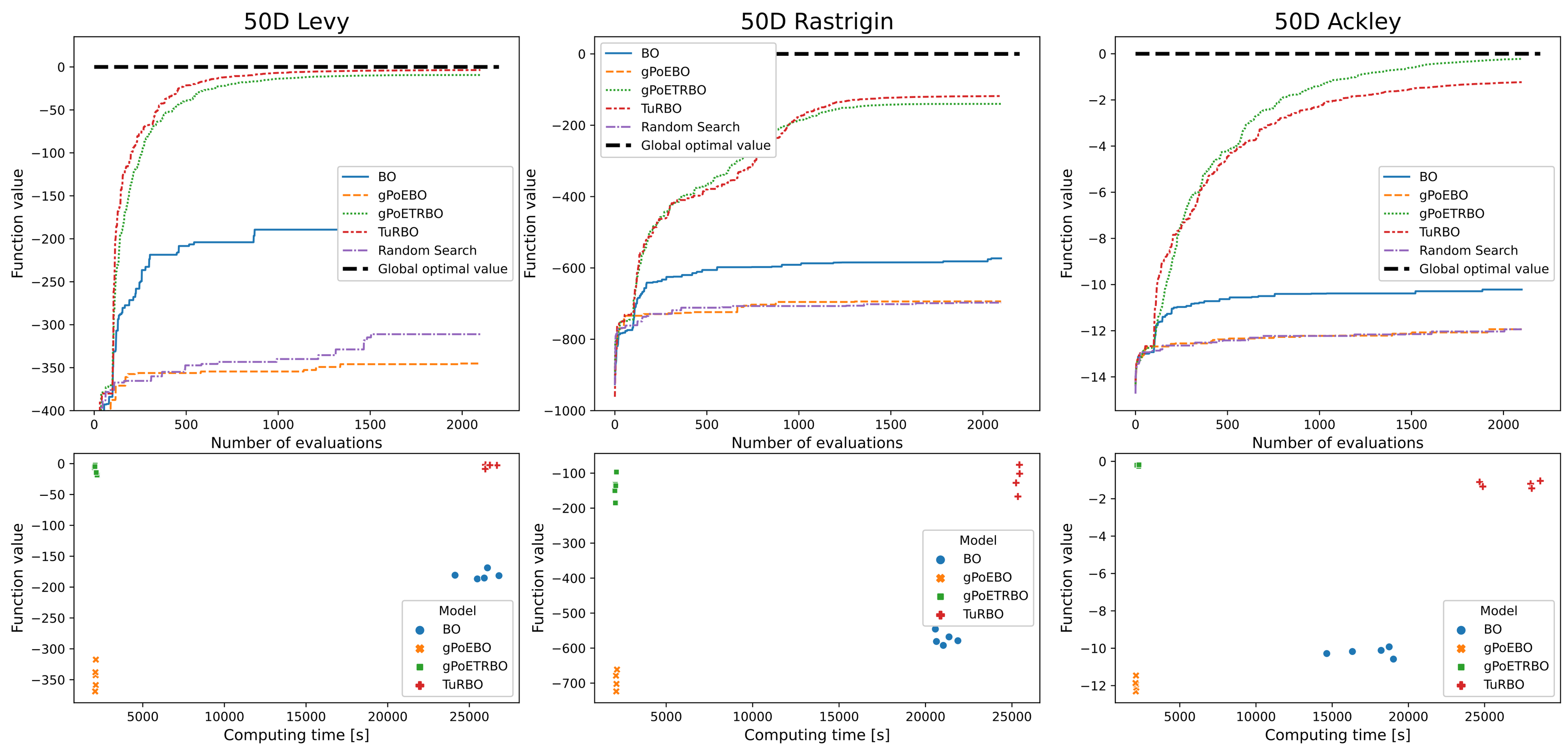

Figure 3. Optimization performance and running times on 50D benchmark functions.

Ablation study

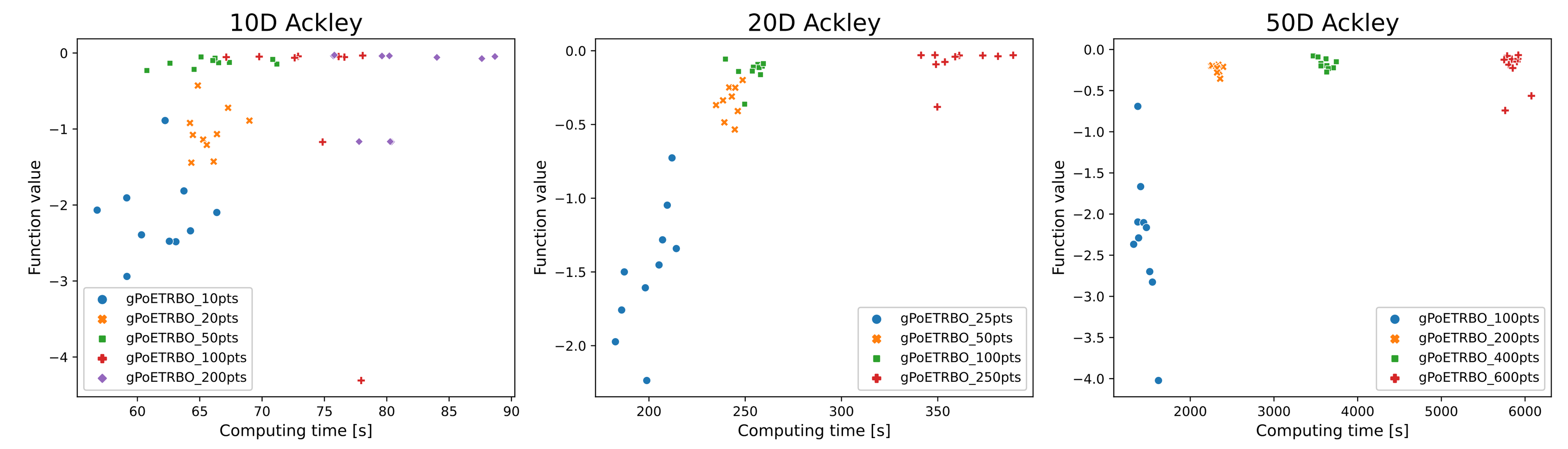

Figure 4. The effect of number of data points per expert on accuracy and computing time.

References

[1] Yanshuai Cao and David J Fleet. Generalized product of experts for automatic and principled fusion of Gaussian process predictions. *Modern Nonparametrics 3: Automating the Learning Pipeline workshop at NIPS. arXiv:1410.7827*, 2014.

[2] Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of Bayesian methods for seeking the extremum. In *Towards Global Optimisation*, volume 2, pages 117–129, 1978.

[3] Christopher K Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, USA, 2006.