Two-stage Kernel Bayesian Optimization in High Dimensions (Supplementary Material)

Jian Tan [*,1] **Niv Nayman** [*,2]

[*]Equal contribution
[1]Alibaba Group , Sunnyvale, California, USA
[2]Technion - Israel Institute of Technology , Haifa, Israel

1 COMPARISON TO REMBO AND ALEBO

REMBO Wang et al. [2016] and ALEBO Letham et al. [2020] are designed for high-dimensional (large D) problems with low intrinsic dimensions (small d), which essentially assumes that the function does not change along certain directions. They do not necessarily perform well for problems without redundant dimensions, as shown by the following experiments with $D = d$.

First, we compare REMBO and CobBO using Ackley 200D with 4000 iterations and 50 initial points. Even though $D = d = 200$ in this case, we treat REMBO as if the effective dimension were $d = 20$, similar to CobBO's subspaces with an average size about 15. REMBO and CobBO reach the mean best values of 15.1 and 3.8, respectively, running for 31.2 and 3.4 hours, respectively. This shows that CobBO could outperform REMBO by a large margin for problems without redundant dimensions. In addition, CobBO requires about 10% of the computation time of REMBO for this experiment, which demonstrates the advantage of the two-stage kernels in reducing the computation time.

We further validate that CobBO is superior to SAASBO Eriksson and Jankowiak [2021] for the above example. We tested SAASBO by running its official code with the official default settings: it takes more than 32 hours for SAASBO to complete 250 iterations and it achieves a best value of 11.17. It is far slower than CobBO and although the found result is already better than REMBO's (15.1), it is much worse than CobBO's (3.8).

Next, we compare with ALEBO, which has demonstrated great performance for problems with large D but small d in Letham et al. [2020]. Through extensive experiments we find that ALEBO works only when the underlying effective dimension satisfies $d \leq 20$. Otherwise, the algorithm suffers from the same curse of dimensionality as vanilla BO algorithms do, since the subproblem in the embedding space of d dimensions is also challenging for large d.

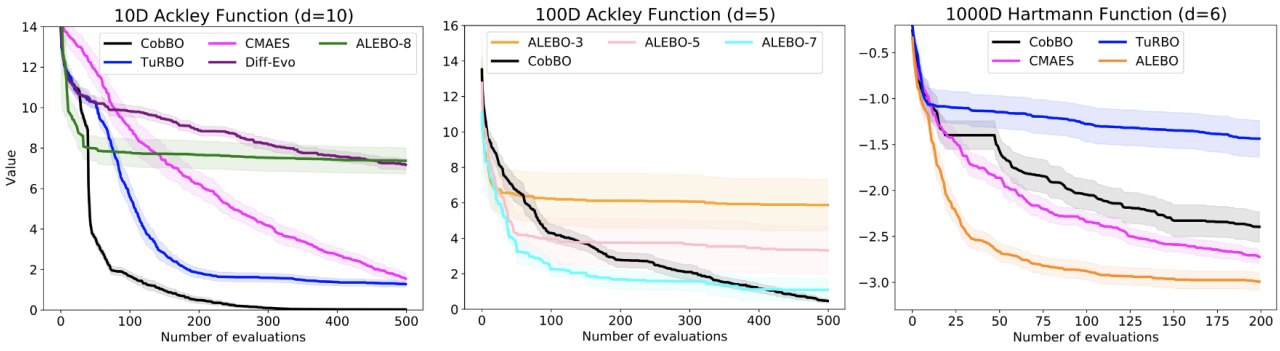

Figure 1: Experiments with $D = 10, 100, 1000$ spaces of small effective dimensions $d = 10, 5, 6$, respectively

To this end, we design three different experiments. First, we study the general problems for $D = d$. Since ALEBO has performance issues for large d, we test Ackley ($D = d = 10$). As ALEBO requires $d < D$, we treat it as if $d = 8$

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

(ALEBO-8). In this case, ALEBO does not show good performance and is outperformed by CobBO, TurBO and CMAES, as shown in Fig. 1 (left). Second, we test Ackley ($D = 100, d = 5$). In reality, we do not know the effective dimension d. Therefore, we teat it as if $d = 3, 5, 7$ to obtain ALEBO-3, ALEBO-5 and ALEBO-7, respectively. Although this problem indeed has a very small $d = 5$, CobBO can still perform well compared to ALEBO, as shown in Fig. 1 (middle). The third experiment is using exactly the same setting as in Letham et al. [2020] for Hartmann6 with $D = 1000$ and $d = 6$. As shown in Fig. 1 (right), ALEBO outperforms CobBO, since CobBO is not designed for a function with a very high dimension $D = 1000$ and a very low effective dimension $d = 6$. The reason is because CobBO relies on selection of subspaces of an average dimension 15, which cannot easily cover the optima in a high dimensional space $D \geq 1000$. In this case, after projecting the original function into a low d dimensional embedding space, CobBO can be applied to solve the subproblem when d is still considered to be too large, e.g., $d > 20$.

2 COMPARISON TO LINEBO

Although sharing some common basic ideas, LineBO Kirschner et al. [2019] reduces the acquisition maximization cost by restricting on a line, but it does not address the computational issue of the GP regression in the full space by using a single kernel, i.e., the first stage of CobBO. In addition, it is difficult to find a good direction to form the line space at each iteration, since searching for the optima in a high dimensional space on a random line is not computationally efficient. Fig. 2 shows that LineBO is significantly outperformed by CobBO using a typical example, e.g., Ackley, with $D = 10, 30$. For

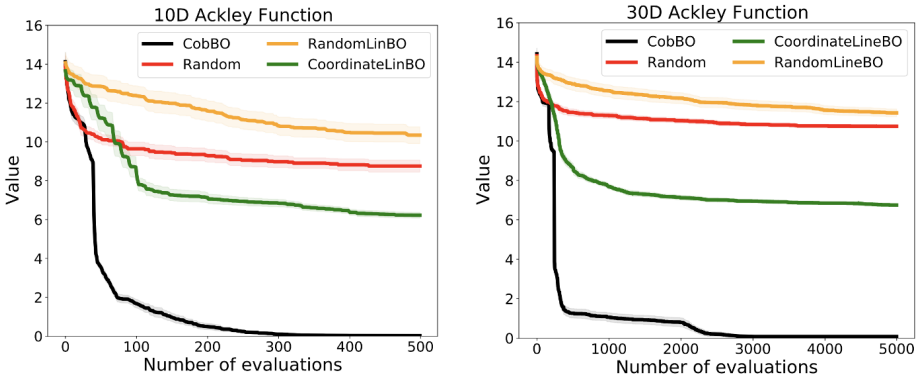

Figure 2: CobBO outperforming different variants of LineBO

$D = 10$ with a query budget of 500, CobBO almost reaches the optimal solution 0.0 while LineBO (CoordinateLineBO) only obtains 6.2. For $D = 30$ with a query budget of 5000, CobBO reaches 0.12 and LineBO (CoordinateLineBO) only obtains 7.6. In both cases, RandomLineBO performs even worse than random search.

3 PROOFS

In this section we provide proofs for the theorems in Section 3.1.1.To make non-negative temporal losses, we modify the losses in Eq. (3) to be non-negative by adding the same constant $\log(\tilde{\alpha})$,

$$\tilde{\ell}_{t,i} = \begin{cases} 0 & \text{if } i \in C_t \text{ and } y_t > M_{t-1} \\ \log(\tilde{\alpha}\tilde{\beta}) & \text{if } i \in C_t \text{ and } y_t \leq M_{t-1} \\ \log(\tilde{\alpha}) & \text{if } i \notin C_t. \end{cases}$$

This modification does not change the resulted distribution π_t induced over the coordinates as it is invariant to shifts of the losses, $\pi_{t,i} = w_{t,i}/W_t$,

$$\pi_{t,i} = \frac{e^{-\eta \sum_{\tau=1}^{t} \tilde{\ell}_{\tau,i}}}{\sum_{j=1}^{D} e^{-\eta \sum_{\tau=1}^{t} \tilde{\ell}_{\tau,j}}} = \frac{e^{-\eta \sum_{\tau=1}^{t} (\ell_{\tau,i} + \log(\tilde{\alpha}))}}{\sum_{j=1}^{D} e^{-\eta \sum_{\tau=1}^{t} (\ell_{\tau,j} + \log(\tilde{\alpha}))}}$$
$$= \frac{e^{-\eta t \log(\tilde{\alpha})} e^{-\eta \sum_{\tau=1}^{t} \ell_{\tau,i}}}{e^{-\eta t \log(\tilde{\alpha})} \sum_{j=1}^{D} e^{-\eta \sum_{\tau=1}^{t} \ell_{\tau,j}}} = \frac{e^{-\eta \sum_{\tau=1}^{t} \ell_{\tau,i}}}{\sum_{j=1}^{D} e^{-\eta \sum_{\tau=1}^{t} \ell_{\tau,j}}}.$$

Thus, $\tilde{\pi}_{t,i}$ and $\hat{\pi}_{t,i}$ introduced in Sections 3.1 and 3.2 remain unchanged as well. For simplicity we refer to $\tilde{\ell}$ as ℓ throughout this section.

3.1 REGRET ANALYSIS FOR SAMPLING FROM THE COMBINATORIAL SPACE OF COORDINATE BLOCKS

The probability $\tilde{\pi}_{t,\mathcal{I}_t}$ of selecting a certain coordinate block $\mathcal{I}_t \subset \mathcal{I} = \{1, \cdots, D\}$ of size $|\mathcal{I}_t| = c \in \mathcal{C}$ follows sampling according to π_t such that

$$\tilde{w}_{t,\mathcal{I}_t} = \prod_{i \in \mathcal{I}_t} w_{t,i}^{\frac{1}{|\mathcal{I}_t|}}, \quad \tilde{W}_t = \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{w}_{t,\mathcal{I}_t}, \quad \tilde{\pi}_{t,\mathcal{I}_t} = \frac{\tilde{w}_{t,\mathcal{I}_t}}{\tilde{W}_t} \quad \forall \mathcal{I}_t \in \bigcup_{c \in \mathcal{C}} \mathcal{S}_c \tag{1}$$

with

$$\sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{\pi}_{t,\mathcal{I}_t} = 1. \tag{2}$$

Lemma 3.1 *For $\eta > 0$ and non-negative losses $\ell_{t,i} \geq 0$ the update rule in (3) satisfies for any block of coordinates $\mathcal{I}^*$:*

$$\sum_{t=1}^{T} \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{\pi}_{t,\mathcal{I}_t} \cdot \frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} - \sum_{t=1}^{T} \frac{1}{|\mathcal{I}^*|} \sum_{i \in \mathcal{I}^*} \ell_{t,i} \leq \eta \sum_{t=1}^{T} \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{\pi}_{t,\mathcal{I}_t} \cdot \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 + \frac{D \log(D)}{\eta}. \tag{3}$$

Proof: Set

$$\tilde{w}_{0,\mathcal{I}_t} = 1 \quad \forall \mathcal{I}_t \in \bigcup_{c \in \mathcal{C}} \mathcal{S}_c \tag{4}$$

Thus,

$$\begin{aligned}
\tilde{W}_{t+1} &= \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{w}_{t+1,\mathcal{I}_t} = \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} w_{t+1,i}^{\frac{1}{|\mathcal{I}_t|}} \\
&= \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} w_{t,i}^{\frac{1}{|\mathcal{I}_t|}} e^{-\frac{\eta}{|\mathcal{I}_t|} \ell_{t,i}} = \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} w_{t,i}^{\frac{1}{|\mathcal{I}_t|}} \cdot e^{-\frac{\eta}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i}} \\
&= \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{w}_{t,\mathcal{I}_t} \cdot e^{-\frac{\eta}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i}} \\
&= \tilde{W}_t \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{\pi}_{t,\mathcal{I}_t} \cdot e^{-\frac{\eta}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i}} && (5) \\
&\leq \tilde{W}_t \sum_{c \in \mathcal{C}} \sum_{\mathcal{I}_t \in \mathcal{S}_c} \tilde{\pi}_{t,\mathcal{I}_t} \left(1 - \frac{\eta}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} + \eta^2 \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 \right) && (6) \\
&\leq \tilde{W}_t \left(1 + \sum_{c \in \mathcal{C}} \left(\sum_{\mathcal{I}_t \in \mathcal{S}_c} \eta^2 \tilde{\pi}_{t,\mathcal{I}_t} \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 - \frac{\eta}{|\mathcal{I}_t|} \tilde{\pi}_{t,\mathcal{I}_t} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right) \right) && (7) \\
&\leq \tilde{W}_t e^{\sum_{c \in \mathcal{C}} \cdot \left(\sum_{\mathcal{I}_t \in \mathcal{S}_c} \eta^2 \tilde{\pi}_{t,\mathcal{I}_t} \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 - \frac{\eta}{|\mathcal{I}_t|} \tilde{\pi}_{t,\mathcal{I}_t} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)}, && (8)
\end{aligned}$$

where (5) follows from (1), (6) holds since $e^{-x} \leq 1 - x + x^2$ for $x \geq 0$, (7) holds due to Eq. (2) and (8) holds since $1 + x \leq e^x$.

Due to Eq. (4), we have,

$$\tilde{w}_{t,\mathcal{I}_t} = \prod_{i \in \mathcal{I}_t} w_{t,i}^{\frac{1}{|\mathcal{I}_t|}} = \prod_{i \in \mathcal{I}_t} w_{0,i}^{\frac{1}{|\mathcal{I}_t|}} e^{-\frac{\eta}{|\mathcal{I}_t|} \sum_{t=1}^{T} \ell_{t,i}} = e^{-\frac{\eta}{\mathcal{I}_t} \sum_{t=1}^{T} \sum_{i \in \mathcal{I}_t} \ell_{t,i}}. \tag{9}$$

And,

$$W_0 = \sum_{c\in\mathcal{C}} \sum_{\mathcal{I}_t\in\mathcal{S}_c} \tilde{w}_{0,\mathcal{I}_t} = \sum_{c\in\mathcal{C}} \sum_{\mathcal{I}_t\in\mathcal{S}_c} 1 = \sum_{c\in\mathcal{C}} |\mathcal{S}_c| = \sum_{c\in\mathcal{C}} \binom{D}{c} \leq (D!)^{|\mathcal{C}|}. \tag{10}$$

Given that the weight of a certain coordinate block $\mathcal{I}^*$ is less than the total sum of all weights, together with Eq. (8), (4) and (10) we have

$$\begin{aligned} e^{-\frac{\eta}{|\mathcal{I}^*|}\sum_{t=1}^{T}\sum_{i\in\mathcal{I}^*}\ell_{t,i}} &= \tilde{w}_{t,\mathcal{I}^*} \leq \tilde{W}_T \\ &\leq (D!)^{|\mathcal{C}|} e^{\sum_{t=1}^{T}\sum_{c\in\mathcal{C}}\cdot\left(\sum_{\mathcal{I}_t\in\mathcal{S}_c}\eta^2\tilde{\pi}_{t,\mathcal{I}_t}\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2 - \frac{\eta}{|\mathcal{I}_t|}\tilde{\pi}_{t,\mathcal{I}_t}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)}. \end{aligned}$$

Taking the $\log$ of both sides, we have

$$-\eta\sum_{t=1}^{T}\frac{1}{|\mathcal{I}^*|}\sum_{i\in\mathcal{I}^*}\ell_{t,i} \leq \sum_{t=1}^{T}\sum_{c\in\mathcal{C}}\cdot\left(\sum_{\mathcal{I}_t\in\mathcal{S}_c}\eta^2\tilde{\pi}_{t,\mathcal{I}_t}\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2 - \frac{\eta}{|\mathcal{I}_t|}\tilde{\pi}_{t,\mathcal{I}_t}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right) + |\mathcal{C}|\log(D!),$$

which, using $D! \leq D^D$, finishes the proof.

Proof of Theorem 1:Since $\ell_{t,i} \leq \log(\tilde{\alpha}\tilde{\beta})$, then

$$\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2 \leq \left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\log(\tilde{\alpha}\tilde{\beta})\right)^2 \leq \log(\tilde{\alpha}\tilde{\beta})^2.$$

Thus, due to Eq. (2), one has

$$\sum_{c\in\mathcal{C}}\sum_{\mathcal{I}_t\in\mathcal{S}_c}\tilde{\pi}_{t,\mathcal{I}_t}\cdot\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2 \leq \sum_{c\in\mathcal{C}}\sum_{\mathcal{I}_t\in\mathcal{S}_c}\tilde{\pi}_{t,\mathcal{I}_t}\log(\tilde{\alpha}\tilde{\beta})^2 = \log(\tilde{\alpha}\tilde{\beta})^2.$$

Setting $\eta = \frac{1}{\log(\tilde{\alpha}\tilde{\beta})}\sqrt{\frac{|\mathcal{C}|D\log(D)}{T}}$ in Eq. (3) yields

$$Regret_t \leq \eta T\log(\tilde{\alpha}\tilde{\beta})^2 + \frac{|\mathcal{C}|D\log(D)}{\eta} = 2\log(\tilde{\alpha}\tilde{\beta})\sqrt{T|\mathcal{C}|D\log(D)}. \tag{11}$$

3.2 REGRET ANALYSIS FOR SAMPLING COORDINATES WITHOUT REPLACEMENT

Denote by p_c the probability of choosing a certain block size $c \in \mathcal{C}$, such that $p_c > 0$ and $\sum_{c\in\mathcal{C}} p_c = 1$, e.g., for a uniform sampling of the block size $p_c = 1/|\mathcal{C}|$ for all $c \in \mathcal{C}$.

The probability $\hat{\pi}_{t,\mathcal{I}_t}$ of selecting a certain coordinate block $\mathcal{I}_t \subset \mathcal{I} = \{1, \cdots, D\}$ of size $|\mathcal{I}_t| = c \in \mathcal{C}$ follows sampling according to π_t (Eq. (2) without replacement, such that,

$$\begin{aligned} \hat{\pi}_{t,\mathcal{I}_t} &= \sum_{p\in perm(\mathcal{I}_t)}\prod_{k\in p}\frac{\pi_{t,k}}{1-\sum_{j\in p_{1:k}}\pi_{t,j}} \\ &= \left(\prod_{i\in\mathcal{I}_t}\pi_{t,i}\right)\cdot\left(\sum_{p\in perm(\mathcal{I}_t)}\prod_{k\in p}\left(1-\sum_{j\in p_{1:k}}\pi_{t,j}\right)^{-1}\right) = \mathcal{P}(\mathcal{I}_t)\cdot\mathcal{R}(\mathcal{I}_t) \end{aligned} \tag{12}$$

where $perm(\mathcal{I}_t)$ are all the permutations of the set $\mathcal{I}_t$ and $p_{1:k}$ are the first k coordinates in the permutation p. Eq. (12) holds due to the common numerator of all permutations where the left term $\mathcal{P}(\mathcal{I}_t)$ corresponds to the probability of sampling a subset of coordinates with replacement, and the right term $\mathcal{R}(\mathcal{I}_t)$ is associated with sampling without replacement. Of course, summing over all the possible blocks of size c results $\sum_{\mathcal{I}_t\in\mathcal{S}_c}\hat{\pi}_{t,\mathcal{I}_t} = 1$ for all $c \in \mathcal{C}$.

Thus $\tilde{\pi}_{t,\mathcal{I}_t} = p_c \cdot \hat{\pi}_{t,\mathcal{I}_t}$ and the probability of sampling every block of coordinates of any size sum up to 1 as well:

$$\sum_{c\in\mathcal{C}}\sum_{\mathcal{I}_t\in\mathcal{S}_c}\tilde{\pi}_{t,\mathcal{I}_t} = \sum_{c\in\mathcal{C}}p_c\sum_{\mathcal{I}_t\in\mathcal{S}_c}\hat{\pi}_{t,\mathcal{I}_t} = \sum_{c\in\mathcal{C}}p_c = 1 \tag{13}$$

Lemma 3.2 *Sample a block size $c \in \mathcal{C}$ with probability $p_c > 0$ and c coordinates without replacement according to π_t. Assume $\mathcal{C} \supset \{1\}$, $\eta > 0$ and non-negative losses $\ell_{t,i} \geq 0$. Then the update rule in (3) satisfies for any block of coordinates $\mathcal{I}^*$:*

$$\sum_{t=1}^{T} \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \hat{\pi}_{t,\mathcal{I}_t} \cdot \frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} - \sum_{t=1}^{T} \frac{1}{|\mathcal{I}^*|} \sum_{i \in \mathcal{I}^*} \ell_{t,i} \leq \eta \sum_{t=1}^{T} \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \hat{\pi}_{t,\mathcal{I}_t} \cdot \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 + \frac{\log(D)}{\eta} - \frac{T \log(p_1)}{\eta} \tag{14}$$

Proof: Starting with a uniform distribution over the coordinates $w_{0,i} \equiv \frac{1}{D}$ such that $W_0 = 1$ and we have:

$$\begin{aligned}
p_1 \cdot W_{t+1} &= p_1 \cdot \sum_{i \in \mathcal{I}} w_{t+1,i} \\
&\leq \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} w_{t+1,i} && (15) \\
&= W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} W_t^{-1} \prod_{i \in \mathcal{I}_t} w_{t,i} e^{-\eta \ell_{t,i}} \\
&\leq W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} W_t^{-|\mathcal{I}_t|} \prod_{i \in \mathcal{I}_t} w_{t,i} e^{-\eta \ell_{t,i}} \cdot |perm(\mathcal{I}_t)| && (16) \\
&= W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} \frac{w_{t,i}}{W_t} e^{-\eta \ell_{t,i}} \cdot \sum_{p \in perm(\mathcal{I}_t)} 1 \\
&= W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} \pi_{t,i} e^{-\eta \ell_{t,i}} \cdot \sum_{p \in perm(\mathcal{I}_t)} \prod_{k \in p} 1 \\
&\leq W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} e^{-\eta \sum_{i \in \mathcal{I}_t} \ell_{t,i}} \prod_{i \in \mathcal{I}_t} \pi_{t,i} \cdot \sum_{p \in perm(\mathcal{I}_t)} \prod_{k \in p} \left(1 - \sum_{j \in p_{1:k}} \pi_{t,j} \right)^{-1} \\
&= W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \hat{\pi}_{t,\mathcal{I}_t} e^{-\eta \sum_{i \in \mathcal{I}_t} \ell_{t,i}} && (17) \\
&\leq W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \hat{\pi}_{t,\mathcal{I}_t} e^{-\frac{\eta}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i}} \\
&\leq W_t \sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \hat{\pi}_{t,\mathcal{I}_t} \left(1 - \frac{\eta}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} + \eta^2 \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 \right) && (18) \\
&\leq W_t \left(1 + \sum_{c \in \mathcal{C}} p_c \cdot \left(\sum_{\mathcal{I}_t \in \mathcal{S}_c} \eta^2 \hat{\pi}_{t,\mathcal{I}_t} \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 - \frac{\eta}{|\mathcal{I}_t|} \hat{\pi}_{t,\mathcal{I}_t} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right) \right) && (19) \\
&\leq W_t e^{\sum_{c \in \mathcal{C}} p_c \cdot \left(\sum_{\mathcal{I}_t \in \mathcal{S}_c} \eta^2 \hat{\pi}_{t,\mathcal{I}_t} \left(\frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)^2 - \frac{\eta}{|\mathcal{I}_t|} \hat{\pi}_{t,\mathcal{I}_t} \sum_{i \in \mathcal{I}_t} \ell_{t,i} \right)} && (20)
\end{aligned}$$

where

- (15) holds since $\mathcal{C} \supset \{1\}$ always contains a block size of 1 and thus

$$\begin{aligned}
\sum_{c \in \mathcal{C}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} w_{t+1,i} &= p_1 \sum_{\mathcal{I}_t \in \mathcal{S}_1} \prod_{i \in \mathcal{I}_t} w_{t+1,i} + \sum_{c \in \mathcal{C} \setminus \{1\}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} w_{t+1,i} \\
&= p_1 \sum_{i \in \mathcal{I}} w_{t+1,i} + \sum_{c \in \mathcal{C} \setminus \{1\}} p_c \sum_{\mathcal{I}_t \in \mathcal{S}_c} \prod_{i \in \mathcal{I}_t} w_{t+1,i} \geq p_1 \sum_{i \in \mathcal{I}} w_{t+1,i}
\end{aligned}$$

- (16) holds since $W_0 = 1$ and W_t is monotonically non-increasing following the update rule (3) with non-negative losses, thus $w_t \leq 1$ for all t; (17) follows from (12); (18) holds since $e^{-x} \leq 1 - x + x^2$ for $x \geq 0$; (19) holds due to Eq. 13; (20) holds since $1 + x \leq e^x$.

Given that the sum of weights of a certain coordinate block $\mathcal{I}^*$ is less than the total sum of weights, together with Eq. 20, $w_{0,i} \equiv \frac{1}{D}$ and $W_0 = 1$ we have

$$\begin{aligned}\frac{1}{D}\sum_{i\in\mathcal{I}^*} e^{-\eta\sum_{t=1}^{T}\ell_{t,i}} &= \sum_{i\in\mathcal{I}^*} w_{t,i} \leq W_T \\ &\leq p_1^{-T} e^{\sum_{t=1}^{T}\sum_{c\in\mathcal{C}} p_c\cdot\left(\sum_{\mathcal{I}_t\in\mathcal{S}_c}\eta^2\hat{\pi}_{t,\mathcal{I}_t}\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2-\frac{\eta}{|\mathcal{I}_t|}\hat{\pi}_{t,\mathcal{I}_t}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)},\end{aligned}$$

Taking the $\log$ of both sides, we have

$$\begin{aligned}&\log\left(\sum_{i\in\mathcal{I}^*} e^{-\eta\sum_{t=1}^{T}\ell_{t,i}}\right) - \log(D) \\ &\leq \sum_{t=1}^{T}\sum_{c\in\mathcal{C}} p_c\cdot\left(\sum_{\mathcal{I}_t\in\mathcal{S}_c}\eta^2\hat{\pi}_{t,\mathcal{I}_t}\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2-\frac{\eta}{|\mathcal{I}_t|}\hat{\pi}_{t,\mathcal{I}_t}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right) - T\log(p_1)\end{aligned} \tag{21}$$

Following the same certain block, all the participating coordinates suffer the same loss ℓ_t^* at every time step as follows from Eq. (3), hence

$$\begin{aligned}\log\left(\sum_{i\in\mathcal{I}^*} e^{-\eta\sum_{t=1}^{T}\ell_{t,i}}\right) = \log\left(\sum_{i\in\mathcal{I}^*} e^{-\eta\sum_{t=1}^{T}\ell_t^*}\right) &= \log\left(|\mathcal{I}^*|e^{-\eta\sum_{t=1}^{T}\ell_t^*}\right) \\ &= \log(|\mathcal{I}^*|) - \eta\sum_{t=1}^{T}\ell_t^* \geq -\eta\sum_{t=1}^{T}\ell_t^*,\end{aligned}$$

which, together with Eq. (21), yields

$$\begin{aligned}&-\eta\sum_{t=1}^{T}\ell_t^* - \log(D) \\ &\qquad\leq \sum_{t=1}^{T}\sum_{c\in\mathcal{C}} p_c\cdot\left(\sum_{\mathcal{I}_t\in\mathcal{S}_c}\eta^2\hat{\pi}_{t,\mathcal{I}_t}\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2-\frac{\eta}{|\mathcal{I}_t|}\hat{\pi}_{t,\mathcal{I}_t}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right) - T\log(p_1),\end{aligned}$$

which finishes the proof.

Proof of Theorem 2: Since $\ell_{t,i} \leq \log(\tilde{\alpha}\tilde{\beta})$ then

$$\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2 \leq \left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\log(\tilde{\alpha}\tilde{\beta})\right)^2 \leq \log(\tilde{\alpha}\tilde{\beta})^2.$$

Thus, due to Eq. (13), we have

$$\sum_{c\in\mathcal{C}} p_c\sum_{\mathcal{I}_t\in\mathcal{S}_c}\hat{\pi}_{t,\mathcal{I}_t}\cdot\left(\frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\ell_{t,i}\right)^2 \leq \sum_{c\in\mathcal{C}} p_c\sum_{\mathcal{I}_t\in\mathcal{S}_c}\hat{\pi}_{t,\mathcal{I}_t}\log(\tilde{\alpha}\tilde{\beta})^2 = \log(\tilde{\alpha}\tilde{\beta})^2.$$

Eq. (14) reads

$$Regret_t \leq \eta T\log(\tilde{\alpha}\tilde{\beta})^2 + \frac{\log(D)}{\eta} - \frac{T\log(p_1)}{\eta}. \tag{22}$$

Choosing $\eta \geq 1$, we have

$$Regret_t \leq \eta T\log(\tilde{\alpha}\tilde{\beta})^2 + \frac{\log(D)}{\eta} - \eta T\log(p_1) = \eta T(\log(\tilde{\alpha}\tilde{\beta})^2 - \log(p_1)) + \frac{\log(D)}{\eta}.$$

Thus setting $\eta = \sqrt{\frac{\log(D)}{T(\log(\tilde{\alpha}\tilde{\beta})^2-\log(p_1))}} \geq 1$ finally we have

$$Regret_t \leq \mathcal{O}\left(\sqrt{(\log(\tilde{\alpha}\tilde{\beta})^2 - \log(p_1))\cdot T\log(D)}\right).$$

Remark: Note that the condition $\eta \geq 1$ can be replaced by setting an appropriate $p_1 = \sqrt[T]{\epsilon}$ for $0 < \epsilon \leq 1$. Thus Eq. (22) reads

$$Regret_t \leq \eta T \log(\tilde{\alpha}\tilde{\beta})^2 + \frac{\log(D) - \log(\epsilon)}{\eta}.$$

Thus, setting $\eta = \frac{1}{\log(\tilde{\alpha}\tilde{\beta})}\sqrt{\frac{\log(D)-\log(\epsilon)}{T}}$ yields $Regret_t \leq \mathcal{O}\left(\log(\tilde{\alpha}\tilde{\beta})^{-1}\sqrt{T(\log(D) - \log(\epsilon))}\right)$.

3.3 REGRET ANALYSIS FOR CONSISTENT QUERIES

The regret analyses presented in Sections 3.1 and 3.2 hold when incorporating the consistent queries mentioned in section 3.2 for an adapted settings.

Consider the update rule of Eq. (3) at each time step $t = 1, \ldots, T$ where the sampling of next coordinate blocks happens for $K \leq T$ time steps at $0 = t_0 < t_1 < \cdots < t_k < \cdots < t_{K-1} < t_k = T$. Both K and $\{t_k\}_{k=0}^{K-1}$ are unknown in advance and are revealed to the decision maker along the process. At each time t_k a coordinate block is selected and fixed for the next $t_{k+1} - t_k$ steps. The effective losses incurred to the coordinates are the aggregation of all the temporal losses in this time interval $t \in [t_k, t_{k+1} - 1]$, and thus $\bar{\ell}_{k,i} = \sum_{t=t_k}^{t_{k+1}-1} \ell_{t,i}$ where $\bar{\ell}_{k,i} \geq 0$ due to $\ell_{t,i} \geq 0$.

Since the update rule in Eq. (3) is applied in every time step $t = 1, \ldots, T$, we effectively have

$$w_{k+1,i} = w_{k,i} \prod_{t=t_k}^{t_{k+1}-1} e^{-\eta \ell_{t,i}} = w_{k,i} e^{-\eta \sum_{t=t_k}^{t_{k+1}-1} \ell_{t,i}} = w_{k,i} e^{-\eta \bar{\ell}_{k,i}}.$$

Define the stopping rule mentioned in section 3.23.23.2 such that the number of consistent queries in a subspace does not cross $\tau \in [1, 2, \ldots, T]$, such that $t_{k+1} - t_k \leq \tau$ for all $k = 0, \ldots, K-1$ and thus $\bar{\ell}_{k,i} \leq \tau \log(\tilde{\alpha}\tilde{\beta})$ since $\ell_{t,i} \leq \log(\tilde{\alpha}\tilde{\beta})$.

Hence, all the results hold by replacing T with K and $\log(\tilde{\alpha}\tilde{\beta})$ with $\tau \log(\tilde{\alpha}\tilde{\beta})$.

4 BACKOFF STOPPING RULE HYPERPARAMETERS

The values of the hyperparameters ξ and τ of the stopping rule, described in Section 3.2, depend on the query budget T and the problem dimension D, such that,

$$\tau = \frac{T}{1000} + \begin{cases} 1 & D < 20 \\ 2 & 20 \leq D < 70 \\ 3 & 70 \leq D < 100 \\ 4 & 100 \leq D < 200 \\ 5 & 200 \leq D \end{cases} \qquad \xi = \begin{cases} 4 & \Delta_t < 0.05 \\ 2 & 0.05 \leq \Delta_t \leq 0.1 \\ 0 & \Delta_t > 0.1 \end{cases}$$

This heuristic stopping rule is designed to take into account several considerations:

1. A maximal query budget (τ) in each subspace grows with the total query budget (T) and dimension (D).
2. A sufficient progress (Δ_t) needs to be made in the subspace to avoid only harvesting marginal improvements due to local fluctuation. The more significant progress the more consecutive improvements (ξ) are allowed in this subspace.

This heuristic stopping rule is robust to all the problems presented in this work and to many other that we have tested.

5 THE UPPER BOUND OF THE BLOCK SIZES

At each iteration, the block size $|C_t|$ of CobBO is uniformly sampled from a set formed through capping the elements from $\{1, 4, 6, 8, 12, 14, 16, 22, 24, 26, 30\}$ by the dimension D of the problem. Hence the average block size is about 15, the lower bound is 1 and the upper bound is 30. This set is chosen to prefer relatively lower dimensions and works well for the problems we experimented with. In Fig. 3 we present an ablation study focusing on the selection of the upper bound of this set, which plots the means and variances of the best searched function values for Rastrigin on $[-3, 4]^{50}$. Considering

that the differences of the mean values of the best obtained minimization solutions are small compared to the standard deviations, we conclude that the algorithm is not very sensitive to the choice of the upper bound, while higher values are slightly favourable, as expected, yet require more computation.

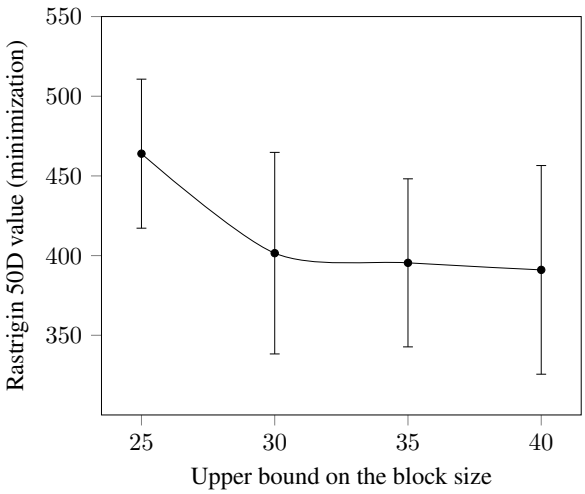

Figure 3: Impact of the block size upper bound on the best function values for Rastrigin on $[-3, 4]^{50}$

6 MORE ON IMPLEMENTATION AND ADDITIONAL EXPERIMENTS

The proposed CobBO algorithm is implemented in Python 3. The source code and the original log files of all the experiments are attached for review. The code has been utilized for various complex real-world applications and handles many corner cases (hence the error fallbacks). For example, a parameter "smooth" of Scipy RBF (kernel=multiquadric, default=0.0) is increased by 0.02 upon "try catch" numerical issues of ill conditioning.

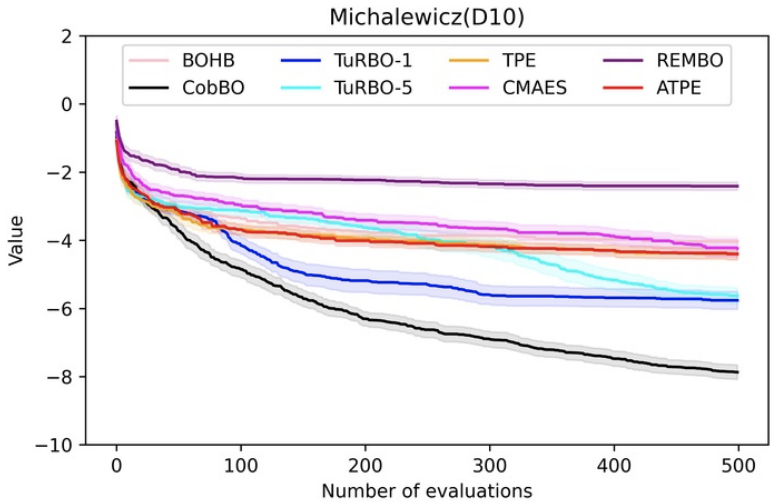

Figure 4: Performance over the low dimensional Michalewicz function with symmetrical and asymmetrical subspaces

In Fig. 4 we show that CobBO also optimizes well the Michalewicz function on 10 dimensions, although it has symmetric bumps, where certain subspaces pass through a point in a symmetrical manner and others break it. Other real applications include parameter tuning for recommendation systems, database online performance tuning, and simulation based parameter optimization. However, due to deviating from the main study of this paper, we refrain from presenting these results that require elaborated description on the application backgrounds.

References

David Eriksson and Martin Jankowiak. High-dimensional bayesian optimization with sparse axis-aligned subspaces. In *Uncertainty in Artificial Intelligence*, pages 493–503. PMLR, 2021.

Johannes Kirschner, Mojmir Mutny, Nicole Hiller, Rasmus Ischebeck, and Andreas Krause. Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3429–3438, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

Ben Letham, Roberto Calandra, Akshara Rai, and Eytan Bakshy. Re-examining linear embeddings for high-dimensional bayesian optimization. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1546–1558. Curran Associates, Inc., 2020.

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando De Freitas. Bayesian optimization in a billion dimensions via random embeddings. *J. Artif. Int. Res.*, 55(1):361–387, January 2016. ISSN 1076-9757.