Gaussian Process Bandits with Aggregated Feedback

Mengyan Zhang, [1,2] **Russell Tsuchida,**[2] **Cheng Soon Ong** [1,2]

[1] The Australian National University [2] Data61, CSIRO
mengyan.zhang@anu.edu.au, russell.tsuchida@data61.csiro.au, chengsoon.ong@anu.edu.au

Abstract

We consider the continuum-armed bandits problem, under a novel setting of recommending the best arms within a fixed budget under aggregated feedback. This is motivated by applications where the precise rewards are impossible or expensive to obtain, while an aggregated reward or feedback, such as the average over a subset, is available. We constrain the set of reward functions by assuming that they are from a Gaussian Process and propose the Gaussian Process Optimistic Optimisation (GPOO) algorithm. We adaptively construct a tree with nodes as subsets of the arm space, where the feedback is the aggregated reward of representatives of a node. We propose a new simple regret notion with respect to aggregated feedback on the recommended arms. We provide theoretical analysis for the proposed algorithm, and recover single point feedback as a special case. We illustrate GPOO and compare it with related algorithms on simulated data.

1 Introduction

In the continuum-armed bandit problem with a fixed budget, an agent adaptively chooses a sequence of N options from a continuous set (*arm space*) in order to minimise some objective given an oracle that provides noisy observations of the objective evaluated at the options (Agrawal 1995; Bubeck, Munos, and Stoltz 2011). The objective may measure the total cost, for example the *cumulative regret*, or may give an indication of the quality of the final choice, for example the *simple regret*. The simple regret setting may be viewed as black-box, zeroth order optimisation of the objective under noisy observations. In practical settings, it is possible that one cannot observe the objective directly. This motivates a more flexible notion of an oracle. In this work we consider an oracle that provides noisy average evaluations of the objective over some grid (defined in a precise sense in (1)).

For the problem of black-box optimisation of a function f under single point stochastic feedback, Munos (2014) proposed a continuum-armed bandit algorithm called Stochastic Optimistic Optimisation (StoOO) with adaptive hierarchical partitioning of arm space, under the *optimism in the face of uncertainty principle*. For bandits with aggregated feedback, Rejwan and Mansour (2020) studied finite-armed case for the combinatorial bandits under full-bandit feedback. Other related settings are discussed in § 6. We consider one important gap in the literature, best arm(s) identification for continuum-armed bandits with average rewards under a fixed budget.

Our goal is to recommend a local area with best average reward feedback. We propose *aggregated regret* (Definition 2) to reflect this objective, devise an algorithm in § 3, and show upper bounds of the aggregated regret under our algorithm in § 4. In § 5, we compare our and with related algorithms in a simulated environment. Our algorithm shows the best empirical performance in terms of aggregated regret.

Our **contributions** are (i) a new continuum-armed bandits setting under the aggregated feedback and corresponding new simple regret notion, (ii) the first fixed budget best arms identification algorithm (GPOO) for continuum-armed bandit with noisy average feedback, (iii) theoretical analysis for the proposed algorithm, and (iv) empirical illustrations of the proposed algorithm.

2 Formulation and Preliminaries

Motivation

Two unique properties of the setting we consider are (i) the reward signal is aggregated, and (ii) the aggregation occurs on hierarchically partitioned continuous space. Gaussian Process Optimistic Optimisation (GPOO) makes use of both of these properties, as illustrated in Figure 1 and described in Algorithm 1.

Aggregated feedback. Quantitative observations of the real-world are often made through smooth rather than instantaneous measurements. Average observations may arise from physical, hardware, privacy constraints. We provide three potential applications to motivate our setting: **(1)** *Radio telescope.* Arm cells are the spatial-frequency (orientation and angular resolution) coordinates of objects in the sky. The average radio wave energy (reward) can be inferred from the radio telescope for the queried area. Only the aggregated reward is observable due to frequency binning in hardware and spatial averaging. The goal is to design a policy so that one can identify the region with the average highest radio energy with a fixed amount of querying. The first radio telescope that was used to detect extra-terrestrial radio sources (Jansky 1933) and was able to determine that the source of the

Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

radiation was from the centre of the Milky Way. **(2)** *Census querying*. Take the age of respondents as an example. Arms are each respondent. The oracle will return the average age (reward) of respondents inside each queried area and each query cost is the same no matter what the query is. Only the aggregated reward is allowed due to privacy concerns. The goal is to design a policy so that one can identify the region with the average highest age with a fixed amount of querying. **(3)** *DNA design*. In synthetic biology, one can modify nucleotides to control protein expression level (reward) (Zhang and Ong 2021a). The arms are all possible DNA sequences. The goal is to find DNA sequences with highest possible protein expression level within a given budget. The experiment is expensive and the search space is too large to enumerate. We can make a mixed culture with similar DNAs in a queried feature space and measure their aggregated reward only. The benefit of hierarchical partition is to efficiently and adaptively search the space.

These smoothing operations present in sensor hardware designs, survey sampling methodologies and privacy-preserving data sharing motivate data analysis techniques that account for smooth or average rather than point or instantaneous measurements (Zhang et al. 2020).

Tree structures for continuous spaces. Function optimisation using bandits may be achieved by simultaneously estimating and maximising some estimated statistic of a black-box objective f. This usually involves an iterative algorithm, whereby at each step a point (or points) is (are) sampled and then the estimated statistic is updated and then maximised. The estimated statistic is called an *acquisition function*, and in continuous spaces, can be computationally expensive to optimise. For example, in GP-UCB, the acquisition function is the upper confidence bound, leading to an overall computational complexity of $\mathcal{O}(N^{2d+3})$ for running the algorithm.

Hierarchically forming arms allows adaptive discretisation over the arm space, which provides a computationally efficient approach for exploring the continuous arm space. Assuming smoothness of the unknown reward function and given a budget, Munos (2014) proposed a Stochastic Optimistic Optimisation (StoOO) algorithm. They adaptively construct a tree which partitions the design space. Each leaf node in the tree represents a subset of the design space and is a candidate to be expanded. The expanded leaf node is chosen based on the *optimisation under uncertainty criteria*. Here the notion of uncertainty captures both stochastic uncertainty due to reward sampling and the inherent function variation. The reward of the leaf node is summarised by the reward obtained by evaluating the noisy objective at a some point in the subset (Munos (2014) calls these *centres*, we call them *representative points*).

Our proposed algorithm, Gaussian Process Optimistic Optimisation (GPOO, Algorithm 1), extends the StoOO algorithm to the case where the f is sampled from an unknown Gaussian Process (GP) and the reward feedback is an average over representatives in a subset. Using a GP allows us to encode smoothness assumptions on the function f through a choice of kernel (see Assumptions 1, 2). It also allows us to exploit the closure of Gaussian vectors under affine maps

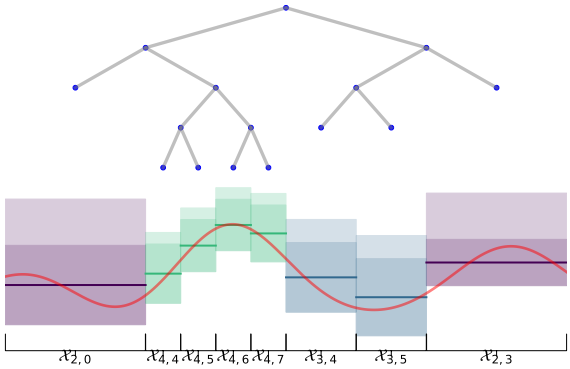

Figure 1: GPOO adaptively constructs a tree where the value associated with each node is an estimate of the aggregated reward over a cell. Red shows the reward function to be optimised. Solid horizontal lines show estimated mean aggregated reward. Dark shaded regions shows probable objective function ranges based on Bayesian uncertainty. Light shaded regions additionally account for potential function variation due to smoothness assumptions.

to update our belief of f under aggregated feedback in a Bayesian framework. In order to build a well-behaved tree-structure, we have to assume a certain regularity of the tree with respect to the function f (see Assumptions 3, 4), mirroring those in the StoOO algorithm. Assumption 3 ensures that as the depth of the tree grows, the allowable function variation around any node decreases. Assumption 4 rules out nodes that represent pathologically shaped subsets of the design space, such as those consisting of sets with measure zero like single points or curves.

Our problem setting recovers the single state reward feedback as a special case. To the best of our knowledge, we are the first work address the continumm-armed bandits function optimisation problem under aggregated feedback.

Problem setting

Let the decision space and the function to be optimised be $\mathcal{X} \subset [0,1]^d$ and $f : \mathcal{X} \to \mathbb{R}$ respectively. We consider a hierarchical partitioning of the space $\mathcal{X}$ through an adaptively-built K-ary tree. Each node (h,i) in the tree is placed at a depth h and an index i. In order to partition the space, each node (h,i) is associated with an attribute $\mathcal{X}_{h,i}$ called a *cell*. At depth h, there are K^h cells. That is, $0 \leq i \leq K^h - 1$. For any fixed h, the cells form a partition of $\mathcal{X}$. Here partition is meant in the formal sense, that is, a partition of $\mathcal{X}$ is a collection of non-empty subsets of $\mathcal{X}$ such that every $\boldsymbol{x} \in \mathcal{X}$ is in exactly one of these subsets.

We may obtain a reward from a given node (h,i) through some abstract reward signal $\mathcal{R}\big((h,i)\big)$, where the mapping $\mathcal{R}$ takes as input the attributes of the node (h,i). These attributes include the cell described above, and may also include other attributes like the representative points, described below. Here we will focus only on a special case of $\mathcal{R}$ and leave other choices of $\mathcal{R}$ for future work.

Each node is associated with S points $\boldsymbol{x}_{h,i^s}$, $1 \leq s \leq S$,

where $\boldsymbol{x}_{h,i^s} \in \mathcal{X}_{h,i}$. We stress that S is a quantity associated with the problem and may *not* be controlled by the agent. We call the collection $\mathcal{C}_{h,i} = \{\boldsymbol{x}_{h,i^s}\}_{1 \le s \le S}$ the *representative points* of $\mathcal{X}_{h,i}$ or (h,i). The reward of each node (h,i) is summarised by the average reward evaluated over the representative points of the cell. More precisely, we denote by $X_{h,i} \in \mathbb{R}^{S \times d}$ the *feature matrix* of cell $\mathcal{X}_{h,i}$ with each row being exactly one element of the representative points $\mathcal{C}_{h,i}$ (where the order of the rows is not important). In the round t, we select a cell $\mathcal{X}_{h_t,i_t}$ to obtain the reward r_t of cell $\mathcal{X}_{h,i}$,

$$r_t = \bar{F}(X_{h_t,i_t}) + \epsilon_t, \quad \bar{F}(X_{h_t,i_t}) := \frac{\sum_{\boldsymbol{x} \in \mathcal{C}_{h_t,i_t}} f(\boldsymbol{x})}{|\mathcal{C}_{h_t,i_t}|}, \tag{1}$$

where $\epsilon_t \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$. $\mathcal{X}_N, X_N$ and $\boldsymbol{x}_N$ will respectively denote the recommended cell, feature matrix and point (if an algorithm returns these objects). A typical goal of best arm identification with fixed budget is to minimise the simple regret (Audibert and Bubeck 2010).

Definition 1 (Simple regret)**.** *We denote an optimal arm* $\boldsymbol{x}^* \in \operatorname{argmax}_{\boldsymbol{x} \in \mathcal{X}} f(\boldsymbol{x})$. *Let* $\boldsymbol{x}_N$ *be our recommended point after* N *rounds. The* simple regret *is defined as*

$$\hat{R}(\boldsymbol{x}_N) = f(\boldsymbol{x}^*) - f(\boldsymbol{x}_N). \tag{2}$$

In our setting, a slightly different surrogate notion of regret will be easier to analyse. Correspondingly, we introduce the *aggregated regret* under our setting in Definition 2. The goal is to minimise the aggregated regret with a fixed budget of N reward evaluations. That is, we aim to recommend a local area with highest possible aggregated reward. This is highly motivated by the applications where the measurement is over a local area instead of precise point querying, for example, the sensor hardware designs and radio telescope we mentioned in the introduction. When $S = 1$, the aggregated regret in Definition 2 is the same as *simple regret*.

Definition 2 (Aggregated regret)**.** *Let* $\mathcal{X}_N$ *be our recommended cell after* N *rounds and* X_N *the corresponding feature matrix. The* aggregated regret *is defined as*

$$R_N = f(\boldsymbol{x}^*) - \bar{F}(X_N). \tag{3}$$

This surrogate may be used to upper bound the simple regret. Note that $\min_{\boldsymbol{x} \in \mathcal{C}_N} \hat{R}_N(\boldsymbol{x}) \le R_N$, where $\mathcal{C}_N$ is the set of representative points in $\mathcal{X}_N$. Given that cell $\mathcal{X}_N$ minimises the aggregated regret, one may apply a (finite) multi-armed bandit algorithm over the set $\mathcal{C}_N$ to solve $\min_{\boldsymbol{x} \in \mathcal{C}_N} \hat{R}_N(\boldsymbol{x})$.

We note that other choices of the abstract reward $\mathcal{R}$ are also natural. For example, a continuous analogue of (1) replaces the discrete sum over $\mathcal{C}_{h_t,i_t}$ with a continuous integral over $\mathcal{X}_{h_t,i_t}$. We sketch in Appendix D how our results might extend to this setting, but leave the details for future work.

Gaussian processes

In order to develop our algorithm and perform our analysis, we will require f to possess a degree of smoothness. We will also need to simultaneously estimate and maximise f.

Assumption 1. *The black-box function* f *is a sample from zero-mean GP with known covariance function* k.

A GP $\{f(\boldsymbol{x})\}_{\boldsymbol{x} \in \mathcal{X}}$ is a collection of random variables indexed by $\boldsymbol{x} \in \mathcal{X}$ such that every finite subset $\{f(\boldsymbol{x}_i)\}_{i=1,\dots m}$ follows a multivariate Gaussian distribution (Rasmussen and Williams 2006). A GP is characterised by its mean and covariance functions, respectively

$$\mu(\boldsymbol{x}) = \mathbb{E}[f(\boldsymbol{x})] \quad \text{and}$$
$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}[(f(\boldsymbol{x}) - \mu(\boldsymbol{x}))(f(\boldsymbol{x}') - \mu(\boldsymbol{x}'))].$$

GPs find widespread use in machine learning as Bayesian functional priors. Some function of interest f is *a priori* believed to be drawn from a Gaussian process with some covariance function k and some mean function μ, where in a typical setting $\mu \equiv 0$. After observing some data, the conditional distribution of f given the data, that is, the posterior of f, is obtained. If the likelihood is Gaussian, the prior is conjugate with the likelihood and the posterior update may be performed in closed form. We describe a precise instantiation of this update tailored to our setting in § 3.

Modelling f as a GP allows us to encode smoothness properties through an appropriate choice of covariance function and also to estimate f in a Bayesian framework. This choice additionally allows us to take advantage of the closure of Gaussian distributions under affine transformations, providing us with a tool to analyse aggregated feedback.

Assumption 2. *The kernel* k *is such that for all* $j = 1, \dots, d$, *some* $a, b > 0$ *and any* $L > 0$,

$$\mathbb{P}\left(\sup_{\boldsymbol{x} \in D} |\partial f / \partial x_j| \ge L\right) \le a \exp\left(-\frac{L^2 b}{2}\right).$$

Assumption 2 implies a tail bound on $|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)|$, and may be shown to hold for a wide class of covariance functions including the squared exponential and Matérn class with smoothness $\nu > 2$. Let ℓ denote the the L_1 distance. By directly applying Theorem 5 of Ghosal and Roy (2006), we show the following in Appendix A.

Proposition 1. *Assumption 2 implies that for some constants* $a, b > 0$ *and any* $L > 0$, $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$,

$$\mathbb{P}\big(|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| \ge L\ell(\boldsymbol{x}_1, \boldsymbol{x}_2)\big) \le a e^{-L^2 b/2}.$$

If $\sup_{x \in \mathcal{X}} \frac{\partial^2}{\partial x_j \partial x_j'} k(x, x')\big|_{x=x'} < \infty$ *and* k *has mixed derivatives of order at least* 4, *then* k *satisfies Assumption 2.*

Assumptions 1 and 2 were also made by (Srinivas et al. 2009), but to the best of our knowledge we are the first to exploit closure of GPs under affine maps in the setting of best arm identification under fixed budget.

3 Algorithm: GPOO

Inspired by StoOO (Munos 2014), we propose Gaussian Process Optimistic Optimisation (GPOO), under the formulations introduced in § 2. In order to describe our algorithm, we first describe how to compute the posterior predictive GP.

Gaussian process posterior update

In round $t \in 1, \ldots, N$, we represent our prior belief over f using a GP. Our prior in round t is the posterior after so-far observed noisy observations of groups up to round t. The posterior update is a minor adjustment to the typical posterior inference step that would be employed if we were to consider only single point (not aggregated) reward feedback, exploiting the fact that multivariate Gaussian vectors are closed under affine transformations.

Define $X_{1:t} \in \mathbb{R}^{tS \times d}$ to be the vertical concatenation of X_{h_j,i_j} for $j = 1, \ldots, t$. Similarly define $Y_{1:t} \in \mathbb{R}^t$ to be a vector with jth row equal to r_j, where $j = 1, \ldots, t$. In round t we observe the feature-reward tuple $\left(X_{h_t,i_t}, r_t\right)$. We may write $r_t = \boldsymbol{a}^\top f(X_{h_t,i_t}) + \epsilon_t$ for corresponding $\boldsymbol{a} \in \mathbb{R}^{S \times 1}$ having all entries equal to $1/S$.

More compactly, all rounds $t \in \{1, \ldots, N\}$ may be represented through a vector equation. Let $A_{1:t} \in \mathbb{R}^{t \times tS}$ with pqth entry equal to $1/K$ if the qth element of $X_{1:t}$ is sampled at round p and zero otherwise. In what follows, we define

$$Z_{1:t} := (X_{1:t}, Y_{1:t}, A_{1:t}),$$

which completely characterises the history of observations up to round t. We may write $Y_{1:t} = A_{1:t} f(X_{1:t}) + \epsilon_{1:t}$, where $\epsilon_{1:t} \in \mathbb{R}^t$ denotes a vector with ith entry ϵ_i.

Let $X_* \in \mathbb{R}^{n_* \times d}$ denote a matrix of test indices, and let $\boldsymbol{a}_* \in \mathbb{R}^{n_* \times 1}$ denote some corresponding weights. The posterior predictive distribution of $\boldsymbol{a}_*^\top f(X_*)$ given all the history $Z_{1:t}$ up to round t is also a Gaussian and satisfies

$$\begin{aligned} &\boldsymbol{a}_*^\top f(X_*) \mid Z_{1:t} \sim \mathcal{N}\left(\boldsymbol{a}_*^\top \mu(X_* \mid Z_{1:t}), \boldsymbol{a}_*^\top \Sigma(X_* \mid Z_{1:t}) \boldsymbol{a}_*\right), \\ &\quad \mu(X_* \mid Z_{1:t}) = M Y_{1:t}, \\ &\quad \Sigma(X_* \mid Z_{1:t}) = k\left(X_*, X_*\right) - M A_{1:t} k(X_t, X_*), \end{aligned} \tag{4}$$

where

$$M = k(X_t, X_*)^T A_{1:t}{}^\top \left(A_{1:t} k(X_t, X_t) A_{1:t}{}^\top + \sigma^2 \boldsymbol{I}\right)^{-1}.$$

With an iterative Cholesky update we may perform all N inference steps in one $\mathcal{O}(N^3)$ sweep. See Appendix B.

Notions of uncertainty and function variation

We introduce the key concept for selecting which node to sample, the *b-value* $b\big(X_* \mid \beta, Z, \boldsymbol{a}_*\big)$, which is the sum of three terms: the posterior mean of the corresponding feature matrix X_*, the confidence interval and function variation,

$$\begin{aligned} &b\big(X_* \mid \beta, Z, \boldsymbol{a}_*\big) := \\ &\qquad \boldsymbol{a}_*^\top \mu(X_* \mid Z) + CI(X_* \mid \beta, Z, \boldsymbol{a}_*) + \delta(h). \end{aligned}$$

The confidence interval is defined in terms of the posterior variance and an exploitation/exploration parameter β,

$$CI\big(X_* \mid \beta, Z, \boldsymbol{a}_*\big) := \beta^{1/2} \sqrt{\boldsymbol{a}_*^\top \Sigma(X \mid Z) \boldsymbol{a}_*}.$$

The last term $\delta(h)$ is the smoothness function depending on the node depth h that satisfies Assumption 3.

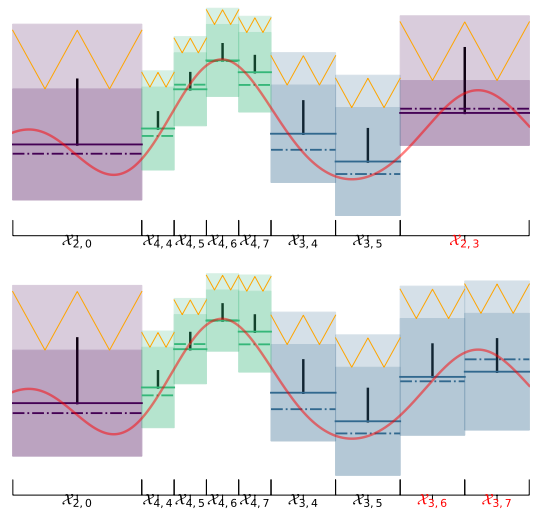

Figure 2: Quantities taken just before line 4 of Algorithm 1 for $t = 9$ (top) and $t = 10$ (bottom). For each cell $\mathcal{X}_{h,i}$, solid and dashed horizontal lines show $\boldsymbol{a}^\top \mu(X_{h,i} \mid Z_t)$ and $\overline{F}(X_{h,i})$ respectively. Dark shaded regions show $CI_t(X_{h,i})$ with colours indicating depth of cell. One-sided b-values are also shown by light shaded regions. Vertical black bars indicate $\delta(h)$. Shown in orange is a probable Lipschitz bound on the function value, due to event ξ and Proposition 1. This bound is in turn bounded by the b-values. When $t = 9$, $\mathcal{X}_{2,3}$ is expanded since it has the highest b-value. When $t = 10$, no cell will be expanded since $\delta(4) < CI_{10}(X_{4,6})$; instead the estimate of the function will be refined by sampling the reward of $\mathcal{X}_{4,6}$.

Assumption 3 (Decreasing diameter)**.** *There exists a decreasing sequence $\delta(h) > 0$ such that for some $L > 0$ (in Assumption 2), any depth h and any cell $\mathcal{X}_{h,i}$, $\sup_{\boldsymbol{x} \in \mathcal{X}_{h,i}} L\ell\left(\boldsymbol{x}_{h,i}, \boldsymbol{x}\right) \leq \delta(h)$ for any representative point $\boldsymbol{x}_{h,i}$.*

Assumption 3 means that at any depth h, the largest possible distance $\sup_{\boldsymbol{x} \in \mathcal{X}_{h,i}} L\ell\left(\boldsymbol{x}_{h,i}, \boldsymbol{x}\right)$ between any point to any representative point within cell $\mathcal{X}_{h,i}$ is decreasing with respect to h. This is intuitive since we have assumed smoothness (Assumption 2) and constructed cells hierarchically. This assumption links the distance in reward space to a $\delta(h)$, which is the core concept in theoretical analysis (see Figure 3). Compared with Munos (2014), we introduce a smoothness parameter L to suit our analysis with GPs.

$\delta(h)$ gives an upper bound of the function deviation of a cell in a given depth. Figure 2 illustrates the above quantities. Define time-dependent b-value and confidence interval

$$b_{h,i}(t) := b\big(X_{h_t,i_t} \mid \beta_t, Z_{1:t-1}, \boldsymbol{a}\big) \quad \text{and} \tag{5}$$

$$CI_t(X_*) := CI\big(X_* \mid \beta_t, Z_{1:t-1}, \boldsymbol{a}\big), \tag{6}$$

where $\beta_t \sim \mathcal{O}(\log t)$ and will be specified in Lemma 1.

We present the GPOO in Algorithm 1. The key idea is that we construct a tree by adaptively discretising over the arm space. The algorithm includes two main parts:

Algorithm 1: GPOO

Input: natural number K (-ary tree), function f to be optimised, smoothness function δ, budget N, the maximum depth nodes can be expanded $h_{\max}$.

Init: tree $\mathcal{T}_0 = \{(0,0)\}$ (corresponds to $\mathcal{X}$), leaves $\mathcal{L}_0 = \mathcal{T}_0$.

1: **for** $t = 1$ to N **do**
2: Select any $(h_t, i_t) \in \operatorname{argmax}_{(h,i)\in\mathcal{L}_{t-1}} b_{h,i}(t)$.
3: Observe reward $r_t = \overline{F}(X_{h_t,i_t}) + \epsilon_t$.
4: Update posteriors $\boldsymbol{a}^\top \mu(\cdot|Z_{1:t}), \boldsymbol{a}^\top \Sigma(\cdot|Z_{1:t})\boldsymbol{a}$
5: Update confidence intervals and b-values for all nodes $(h,i) \in \mathcal{L}_{t-1}$ according to E.q. (6)(5).
6: **if** $\delta(h_t) \geq CI_t(X_{h_t,i_t})$ and $h_t \leq h_{\max}$ **then**
7: Expand node (h_t, i_t) (partition $\mathcal{X}_{h_t,i_t}$ into K subsets) into children nodes $\mathcal{C}_t = \{(h_t+1, i_1), \ldots, (h_t+1, i_K)\}$. $\mathcal{T}_t = \mathcal{T}_{t-1} \cup \mathcal{C}_t$.
8: $\mathcal{L}_{t-1} = \mathcal{L}_{t-1} \backslash \{(h_t, i_t)\}; \mathcal{L}_t = \mathcal{L}_{t-1} \cup \mathcal{C}_t$.
9: **end if**
10: **end for**
11: **Return** The node with index (h', i') such that $h' = \operatorname{argmax}_{h|(h,i)\in\mathcal{T}_N\backslash\mathcal{L}_N} h$. and $i' = \operatorname{argmax}_{i|(h',i)\in\mathcal{T}_N} \mu(X_{h',i}|Z_{1:t})$

Select node and update: In each round $t \in [1, N]$, a node (h_t, i_t) is selected from leaves with the highest b-value (5). The reward is sampled as the average of the function value over S representative points plus Gaussian noise. We then update the posteriors over cells of leaf nodes, allowing the confidence interval and b-values to also be updated.

Expand node: As we increase the number of samples for a given node, the confidence interval of that node continues to decrease. When the confidence interval is smaller than or equal to the function variation, our function estimate is more precise than the current cell range. That is, when the condition on line 6 satisfied, the node can be expanded into K children by partitioning cells into K subsets. When the budget is reached, we return the node with the highest posterior mean prediction among non-leaf nodes. We select recommendations from non-leaf nodes because predictions of non-leaf nodes (satisfying event ξ_2 in Section 4) are more precise than leaves. Figure 2 shows various quantities in the algorithm over two iterations.

Remark 1 (Comparison to StoOO)**.** *(i) StoOO assumes known smoothness of the function but does not utilise correlations between arms and recommend based on predictions. GPOO recommends nodes with the GP predictions. (ii) GPOO can address aggregated feedback, while StoOO can only deal with single state feedback. We also extend StoOO to address aggregated feedback in Appendix C.*

Remark 2 (Computational Complexity)**.** *The cost of performing global maximisation of an acquisition function can be exponential in the design space dimension d. The acquisition function for GP-UCB is the UCB at each point $\boldsymbol{x} \in \mathcal{X}$, leading to a total computational cost of $\mathcal{O}(N^{2d+3})$. The problem of selecting an optimal arm from a finite subset of arms is a case of combinatorial optimisation, and the computational cost can be exponential in the number of arms. For GPOO, for every round t, we consume $\mathcal{O}(t^2)$ for the GP inference procedure. Every time a leaf node is expanded, $K-1$ new leaf nodes are added to the tree, so that less than $(K-1)t$ b-values and confidence intervals must be computed. This leads to a total cost of $\mathcal{O}(N^4(K-1))$.*

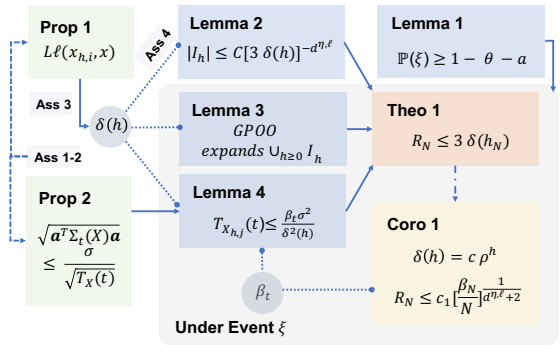

Figure 3: Proof Roadmap

4 Theoretical Analysis for Aggregated Regret

In this section, we provide the theoretical analysis of GPOO. We show our proof roadmap in Figure 3. The full proofs can be found in Appendix A. Our technical contribution is adapting the analysis of the hierarchical partition idea (Munos 2014) to the case reward is sampled from a GP (Srinivas et al. 2009) and is aggregated.

Recall N is the budget, $h_{\max}$ is a parameter representing the deepest allowable depth of tree, $\boldsymbol{x}^* \in \operatorname{argmax}_{\boldsymbol{x}\in\mathcal{X}} f(\boldsymbol{x})$ is an optimal point in input space and $f^* = f(\boldsymbol{x}^*)$. Let $\boldsymbol{x}_{h_t^*, j_t^*}$ be a representative point inside the cell $\mathcal{X}_{h_t^*, j_t^*}$ containing $\boldsymbol{x}^*$ in round t. We first define event ξ, under which we present our aggregated regret upper bound in Theorem 1. Define event ξ as $\xi = \xi_1 \cap \xi_2$, where

$$\xi_1 := \left\{\forall 1 \leq t \leq N \quad f^* - f\left(\boldsymbol{x}_{h_t^*, j_t^*}\right) \leq L\ell(\boldsymbol{x}_{h_t^*, j_t^*}, \boldsymbol{x}^*)\right\},$$

$$\xi_2 := \Big\{\forall 0 \leq h \leq h_{\max}, 0 \leq i < K^h, 1 \leq t \leq N \quad \left|\boldsymbol{a}^\top \mu(X_{h,i} \mid Z_{t-1}) - \overline{F}(X_{h,i})\right| \leq CI_t(X_{h,i})\Big\}.$$

The event ξ provides the probabilities environment for our theoretical analysis, which is the union of two events: event ξ_1 describes the function f local smoothness around its maximum; event ξ_2 captures a concentration property from the estimation to representative summary statistics of reward, which has been shown as an critical part for the regret analysis (Lattimore and Szepesvári 2020; Zhang and Ong 2021b). In the following lemma, b is the constant in Proposition 1. Since θ is positive and less than $1-a$, $\mathbb{P}(\xi) \geqslant 0$.

Lemma 1. *Define $a = h_{\max}\exp(-\frac{L^2 b}{2})$ with L as a constant specified in Assumption 3. Let $\beta_t = 2\log(M\pi_t/\theta)$, where $M = \sum_{h=0}^{h_{\max}} K^h$, $\pi_t = \pi^2 t^2/6$. For K-ary tree and all $\theta \in (0, 1-a)$, we have $\mathbb{P}(\xi) \geq 1 - a - \theta$.*

To obtain the regret bound for our algorithm, we need to upper bound two pieces: the number of expanded nodes in

each depth of the tree (Lemma 2), which depends on the concept of near-optimality dimension in Definition 4 proposed in Munos (2014); and the number of times a node can be sampled before expansion (Lemma 4), which can be inferred from the GP posterior variance upper bound (Proposition 2). We define the set of expanded nodes at depth h as I_h

$$I_h := \{(h,i) | \bar{F}(X_{h_t,j_t}) + 3\delta(h) \geq f^*\}. \tag{7}$$

To upper bound the number of nodes in I_h, we introduce the concept of near-optimality dimension in Definition 4, which relates to function f, ℓ and depends on the constant η.

Definition 3 ($\epsilon-$optimal state)**.** *For any $\epsilon > 0$, define the $\epsilon-$optimal states as $\mathcal{X}_\epsilon := \{\boldsymbol{x} \in \mathcal{X} | f(\boldsymbol{x}) \geq f^* - \epsilon\}$.*

Definition 4 ((η, ℓ)-near-optimality dimension $d^{\eta,\ell}$, (Munos 2014))**.** *For any $\epsilon > 0$, with $\epsilon-$optimal states $\mathcal{X}_\epsilon$ defined in Defition 3, $d^{\eta,\ell}$ is the smallest $d \geq 0$ such that there exists $C > 0$ such that the maximal number of disjoint $\ell-$balls of radius $\eta\epsilon$ with centre in $\mathcal{X}_\epsilon$ is less than $C\epsilon^{-d}$.*

We further introduce the well-shaped cells assumption, which implies that the cells partitioned by our algorithm should contain each representatives points in a $\ell-$ball, e.g. the representatives points should not be on the boundary. This helps us upper bound the number of expanded nodes (Lemma 2). Unlike Munos (2014), we require it to hold for any representative point to suit our aggregated feedback setting. We then show that only the set of nodes in I_h are expanded with GPOO in Lemma 3.

Assumption 4 (Well-shaped cells)**.** *There exists $\nu > 0$ s.t. for any depth $h \geq 0$, any cell $\mathcal{X}_{h,i}$ contains an ℓ-ball of radius $\nu\delta(h)$ centred in each point in the representative set $\boldsymbol{x} \in \mathcal{C}_{h,i}$.*

Lemma 2. *Under Assumption 4, $|I_h| \leq C[3\delta(h)]^{-d^{\eta,\ell}}$.*

Lemma 3. *Under event ξ and Assumption 3 , GPOO only expands nodes in the set $I : \cup_{h \geq 0} I_h$.*

We now move to derive an upper bound of the number of draws of expanded nodes. Following Proposition 3 of Shekhar, Javidi et al. (2018), using mutual information, we upper bound the GP posterior variance in Proposition 2.

Proposition 2. *Let f be a sample from a GP with zero mean and covariance function k. Let $X \in \mathbb{R}^{S \times d}$ and let $\Sigma_t(X)$ denote the posterior predictive GP covariance $\Sigma(X \mid Z_{1:t})$ according to* (4)*. If the history $Z_{1:t}$ contains at least $T_X(t)$ observations following the model $Y = \boldsymbol{a}^\top f(X) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2)$, then we have $\sqrt{\boldsymbol{a}^\top \Sigma_t(X) \boldsymbol{a}} \leq \frac{\sigma}{\sqrt{T_X(t)}}$, where $\boldsymbol{a} \in \mathbb{R}^{S \times 1}$ has all entries equal to $1/S$.*

With Proposition 2, we upper bound the number of draws for any nodes under event ξ in the following lemma.

Lemma 4. *Under event ξ, suppose a node (h, j) (with corresponding feature matrix $X_{h,j}$ as defined in § 2) is sampled at least $T_{X_{h,j}}(t)$ times up to round t. Then we have $T_{X_{h,j}}(t) \leq \frac{\beta_t \sigma^2}{\delta^2(h)}$, where σ^2 is the noise variance, β_t is defined in Lemma 1.*

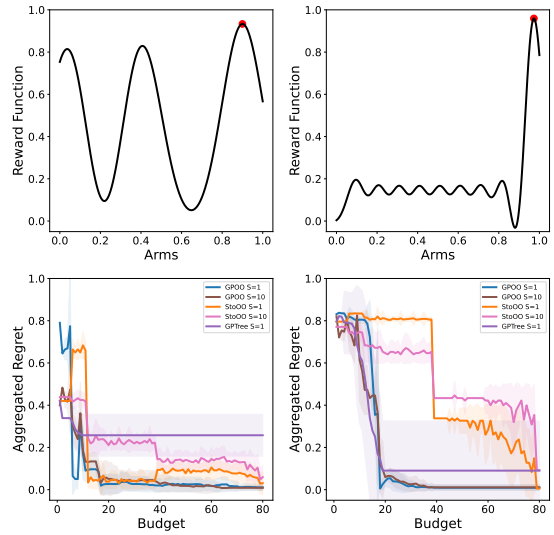

Figure 4: Reward functions f (row 1) and aggregated regrets (row 2), with shaded regions indicate one standard deviation. We perform 30 independent runs with a budget N up to 80.

We show the aggregated regret bound for any δ under our assumptions in Theorem 1, and show a special case of exponential diameter in Corollary 1.

Theorem 1. *Define h_N as the smallest integer h' up to round N such that*

$$\frac{N}{\beta_N} \leq K \sum_{h=0}^{h'} C[3\delta(h)]^{-d^{\eta,\ell}} \frac{\sigma^2}{\delta^2(h)}. \tag{8}$$

For constant a specified in Lemma 1, and $\theta \in (0, 1-a)$, with probability $1-\theta-a$, the simple regret of GPOO satisfies

$$R_N \leq 3\delta(h_N).$$

Corollary 1 (Regret bound for exponential diameters)**.** *Assume $\delta(h) = c\rho^h$ for some constants $c > 0$, $\rho < 1$. For constant a specified in Lemma 1, and $\theta \in (0, 1-a)$, with probability $1-\theta-a$, the simple regret of GPOO satisfies*

$$R_N \leq c_1 \left[\frac{\beta_N}{N}\right]^{\frac{1}{d^{\eta,\ell}+2}},$$

where $c_1 = \frac{3^{-d^{\eta,\ell}} K C \sigma^2}{\rho^{-(d^{\eta,\ell}+2)}-1}$. Recall β_N is in rate $\mathcal{O}(\log N)$.

5 Experiments

We investigate the empirical performance of GPOO on simulated data. We compare the aggregated regret obtained by GPOO with related algorithms, and illustrate how different parameters influence performance.

We show regret curves of different algorithms for two simulated reward functions in Figure 4. Our decision space is chosen to be $\mathcal{X} = [0, 1]$. The reward functions are sampled from a GP posterior conditioned on hand-designed samples (listed in Appendix E), with radial basis function (RBF) kernel having lengthscale 0.05 and variance 0.1. The GP noise

standard variation was set to 0.005. The reward noise is i.i.d. sampled zero-mean Gaussian distribution with standard deviation 0.1.

For our experiment, we consider two cases of feedback: *single point feedback* (S = 1), where the reward is sampled from the centre (representative point) of the selected cell and *average feedback* (S = 10), where the reward is the average of samples from the centre of each sub-cell, which are obtained by splitting the cell into intervals of equal size. Following Corollary 1, we choose $\delta(h) = c2^{-h}$, where c is chosen via cross-validation. The algorithms are evaluated by the aggregated regret in Definition 2 ($S = 1$ for single point feedback, $S = 10$ for average feedback).

The related algorithms we compare with include: (i) StoOO (Munos 2014): the error probability needed for StoOO is chosen to be 0.1. (ii) AVE-StoOO: We extend the StoOO algorithm to the case where the rewards are aggregated feedback in Appendix C. (iii) GPTree (Shekhar, Javidi et al. 2018). We discuss the related algorithms in § 6.

The first reward function (first row) is designed to show the performance of algorithms when there are several relatively similar local maximums. To find the global optimal region, the algorithm needs to predict the function values of high-value cells in high precision quickly. The second reward function is designed to show the case where the reward function is periodic-like. To achieve low regret, the algorithm needs to avoid disruptions by the patterns hidden in rewards, e.g. avoid consistently sampling points with similar function values. GPOO (for both single and average feedback cases) has the best performance for the two reward functions. The aggregated regret convergences quickly and remains stable. GPTree tends to stuck in local maximum (in a finite budget) since their parameter β_N, balancing the posterior mean and standard deviation, only depends on the budget N and does not increase over different rounds, while in our algorithm β_t increases like $\mathcal{O}(\log t)$. As expected, StoOO-based algorithms converge more slowly, since they only use empirical means instead of GP predictions and they are designed with a more broad family of reward functions in mind. We also show an example where aggregated feedback can be beneficial for recommendations in Figure 5, which is deferred to Appendix E due to the page limit.

6 Related Work

We mainly review the related work in two aspects: GP-elated bandits and bandits work considering the aggregated feedback. Refer to Lattimore and Szepesvári (2020) for a comprehensive review of bandits literature.

Gaussian Process Bandits By assuming the unknown reward is sampled from a GP, bandit algorithms can be applied to black-box function optimisation. Srinivas et al. (2009) studied *regret minimisation* under single arm feedback, where arms are recommended sequentially under a Upper Confidence Bound (UCB) policy. They covered both the finite arm space and continuous arm space. GP bandit is also studied and applied in Bayesian optimisation in a simple regret setting (Shahriari et al. 2016).

As far as we know, no current GP bandit works address aggregated feedback. Accabi et al. (2018) studied the *semi-bandit* setting where both the individual labels of arms and the aggregated feedback in the selected subset are observed in each round, while we consider a harder setting where only the aggregated feedback is observed. For non-aggregated feedback, the most related work to ours is Shekhar, Javidi et al. (2018), which extends the StoOO algorithm to the case where the f is sampled from unknown GP with theoretical analysis. They did not provide empirical evaluations on the proposed algorithm and their algorithm highly depends on large amount of theoretical-based parameters. We compared with their algorithm in Section 5.

Aggregated Feedback Related Settings The "aggregated feedback" is first used in Pike-Burke et al. (2018), in which they studied bandits with *Delayed, Aggregated Anonymous Feedback*. At each round, the agent observes a delayed, sum of previously generated rewards. Different from our setting, they considered finite, independent arms setting and cumulative (pseudo-)regret minimisation problem.

There are two types of bandits problems that are related to the aggregated feedback setting. One related setting is *full-Bandits* (Rejwan and Mansour 2020; Du, Kuroki, and Chen 2021) which is studied under *combinatorial bandits* (Chen, Wang, and Yuan 2013) with finite arm space, where only the aggregated feedback over a combinatorial set is observed. No work has addresses GP bandits with full-bandit feedback. Our setting is different from full-bandits since we consider a continuous arm space and the objective is to identify local areas for function optimisation. Another related setting is *slate bandits* (Dimakopoulou, Vlassis, and Jebara 2019), where a slate has fixed number of positions named *slots*. The slate-level reward can be an aggregation over slot-level rewards. However, the slate bandits setting assumes slate-slot two levels rewards and each slate has fixed choices of slots, which is significantly different from our setting.

7 Conclusion and future work

We introduced a novel setting for continuum-armed bandits where only the aggregated feedback can be observed. This is motivated by applications where aggregated reward is the only feedback or precise reward is expensive to access. We proposed Gaussian Process Optimistic Optimisation (GPOO) in Algorithm 1 which adaptively searches a hierarchical partitioning of the space. We provided an upper bound on the aggregated regret in Theorem 1 and empirically evaluated our algorithm on simulated data in § 5.

It may be possible to extend our framework to some other interesting settings. For example, instead of only observing the aggregated function value corrupted by Gaussian noise, one may consider the case where the gradient is additionally observed. This is recently studied in Shekhar and Javidi (2021) on GP bandits with cumulative regret bound. It would be interesting to study the simple regret (pure exploration), under the aggregated feedback setting. Since GPs are closed under linear operators (including integrals, derivatives and Fourier transforms), one can potentially handle the case where some combination of these is observed.

References

Accabi, G. M.; Trovo, F.; Nuara, A.; Gatti, N.; and Restelli, M. 2018. When Gaussian Processes Meet Combinatorial Bandits: GCB. In *14th European Workshop on Reinforcement Learning*, 1–11.

Agrawal, R. 1995. The continuum-armed bandit problem. *SIAM journal on control and optimization*, 33(6): 1926–1951.

Audibert, J.-Y.; and Bubeck, S. 2010. Best Arm Identification in Multi-Armed Bandits. In *Proceedings of the 23rd Annual Conference on Learning Theory (COLT)*.

Bubeck, S.; Munos, R.; and Stoltz, G. 2011. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19): 1832–1852.

Chen, W.; Wang, Y.; and Yuan, Y. 2013. Combinatorial multi-armed bandit: General framework, results and applications. *30th International Conference on Machine Learning, ICML 2013*, 151–159.

Dimakopoulou, M.; Vlassis, N.; and Jebara, T. 2019. Marginal Posterior Sampling for Slate Bandits. In *IJCAI*, 2223–2229.

Du, Y.; Kuroki, Y.; and Chen, W. 2021. Combinatorial pure exploration with full-bandit or partial linear feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 7262–7270.

Ghosal, S.; and Roy, A. 2006. Posterior consistency of Gaussian process prior for nonparametric binary regression. *The Annals of Statistics*, 34(5): 2413–2429.

Golub, G. H.; and van Loan, C. F. 2013. *Matrix Computations*. JHU Press, fourth edition. ISBN 1421407949 9781421407944.

Jansky, K. G. 1933. Electrical disturbances apparently of extraterrestrial origin. *Proceedings of the Institute of Radio Engineers*, 21(10): 1387–1398.

Lattimore, T.; and Szepesvári, C. 2020. *Bandit Algorithms*. Cambridge University Press.

Munos, R. 2014. From bandits to Monte-Carlo Tree Search: The optimistic principle applied to optimization and planning.

Pike-Burke, C.; Agrawal, S.; Szepesvari, C.; and Grunewalder, S. 2018. Bandits with Delayed, Aggregated Anonymous Feedback. *arXiv:1709.06853 [cs, stat]*. ArXiv: 1709.06853.

Rasmussen, C.; and Williams, C. 2006. *Gaussian Process for Machine Learning*. MIT Press.

Rejwan, I.; and Mansour, Y. 2020. Top-k Combinatorial Bandits with Full-Bandit Feedback. In *Algorithmic Learning Theory*, 752–776. PMLR.

Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R. P.; and de Freitas, N. 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE*, 104(1): 148–175. Conference Name: Proceedings of the IEEE.

Shekhar, S.; and Javidi, T. 2021. Significance of Gradient Information in Bayesian Optimization. In *International Conference on Artificial Intelligence and Statistics*, 2836–2844. PMLR.

Shekhar, S.; Javidi, T.; et al. 2018. Gaussian process bandits with adaptive discretization. *Electronic Journal of Statistics*, 12(2): 3829–3874.

Srinivas, N.; Krause, A.; Kakade, S. M.; and Seeger, M. 2009. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*.

Zhang, M.; and Ong, C. S. 2021a. Opportunities and Challenges in Designing Genomic Sequences. *ICML Workshop on Computational Biology*.

Zhang, M.; and Ong, C. S. 2021b. Quantile Bandits for Best Arms Identification. In *International Conference on Machine Learning*, 12513–12523. PMLR.

Zhang, Y.; Charoenphakdee, N.; Wu, Z.; and Sugiyama, M. 2020. Learning from Aggregate Observations. *Advances in Neural Information Processing Systems*, 33.

Supplementary Materials for Gaussian Process Bandits with Aggregated Feedback

We show the full proof of our theoritical analysis in § A. In § B, we illustrate how to update Cholesky decompositions with new rows and column sequentially. We extend StoOO algorithm (Munos 2014) to address aggregated feedback in § C. In § D, we discuss a continuous analogue of the aggregated feedback. Lastly, we show experimental details and additional results in § E.

A Supplementary Theoritical Analysis

In this section, we show proofs of our theoretical statements in the order they are stated in the main paper. For the reader's convenience, we restate our theorems in the main paper whenever needed.

Proposition 1. *Assumption 2 implies that for some constants $a, b > 0$ and any $L > 0$, $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$,*

$$\mathbb{P}\big(|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| \geq L\ell(\boldsymbol{x}_1, \boldsymbol{x}_2)\big) \leq ae^{-L^2 b/2}.$$

If $\sup_{x\in\mathcal{X}} \frac{\partial^2}{\partial x_j \partial x'_j} k(x, x')\big|_{x=x'} < \infty$ and k has mixed derivatives of order at least 4, then k satisfies Assumption 2.

Proof. The mean value theorem says that there exists some $\boldsymbol{x} \in \mathcal{X}$ such that

$$|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| \leq \sup_{x\in\mathcal{X}} |\partial f/\partial x_j| \, \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_1.$$

On the other hand, by Assumption 2 we have that $\sup_{x\in\mathcal{X}} |\partial f/\partial x_j| \leq L$ with probability $1 - ae^{-L^2 b/2}$. Therefore,

$$\mathbb{P}\big(|f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)| \leq L\|\boldsymbol{x}_1 - \boldsymbol{x}_1\|_1\big) \geq 1 - ae^{-L^2 b/2}.$$

Let $\sigma^2_{\max} = \sigma^2_{\max} \sup_{x\in\mathcal{X}} \frac{\partial^2}{\partial x_j \partial x'_j} k(x, x')\big|_{x=x'}$. By Theorem 5 of Ghosal and Roy (2006), there exists some a and c such that for any $L > 0$,

$$\mathbb{P}\left(\sup_{x\in\mathcal{X}} |\partial f/\partial x_j| \geq L\right) \leq ae^{-cL^2/\sigma^2_{\max}}.$$

Setting $b = 2c/\sigma^2_{\max}$, we obtain the desired result. □

Lemma 1. *Define $a = h_{\max} \exp(-\frac{L^2 b}{2})$ with L as a constant specified in Assumption 3. Let $\beta_t = 2\log(M\pi_t/\theta)$, where $M = \sum_{h=0}^{h_{\max}} K^h$, $\pi_t = \pi^2 t^2/6$. For K-ary tree and all $\theta \in (0, 1-a)$, we have $\mathbb{P}(\xi) \geq 1 - a - \theta$.*

Proof. From Proposition 1, $\mathbb{P}\{f^* - f\left(\boldsymbol{x}_{h_t^*, j_t^*}\right) \leq L\ell(\boldsymbol{x}_{h_t^*, j_t^*}, \boldsymbol{x}^*)\} \geq 1 - \exp(-\frac{L^2 b}{2})$. By union bound, we have

$$\begin{aligned}
&\mathbb{P}\{\forall 1 \leq t \leq N, f^* - f\left(\boldsymbol{x}_{h_t^*, j_t^*}\right) \leq L\ell(\boldsymbol{x}_{h_t^*, j_t^*}, \boldsymbol{x}^*)\} \\
=&\mathbb{P}\{\forall 1 \leq h_t^* \leq h_{\max}, f^* - f\left(\boldsymbol{x}_{h_t^*, j_t^*}\right) \leq L\ell(\boldsymbol{x}_{h_t^*, j_t^*}, \boldsymbol{x}^*)\} \\
\geq&1 - h_{\max} \exp(-\frac{L^2 b}{2}).
\end{aligned}$$

We have shown a probability bound for event ξ_1 depending on $h_{\max}$ and L.

Since $\boldsymbol{a}^\top \mu(X_{h,i} \mid Z_{t-1}) - \overline{F}(X_{h,i})$ is Gaussian, we have the tail bound

$$\mathbb{P}\left\{\left|\boldsymbol{a}^\top \mu(X_{h,i} \mid Z_{t-1}) - \overline{F}(X_{h,i})\right| > CI_t(X_{h,i})\right\} \leq e^{-\beta_t/2},$$

Applying the union bound,

$$\left|\boldsymbol{a}^\top \mu(X_{h,i} \mid Z_{t-1}) - \overline{F}(X_{h,i})\right| \leq CI_t(X_{h,i}) \forall h, \forall 0 \leq i < K^h$$

holds with probability $\geq 1 - Me^{-\beta_t/2}$, where $M = \sum_{h=0}^{h_{\max}} K^h$. Choosing $Me^{-\beta_t/2} = \theta/\pi_t$ and using the union bound for $t \in \mathbb{N}$, the statement holds. For example, we can use $\pi_t = \pi^2 t^2/6$.

By the union bound, we have the probability bound for event ξ. □

Lemma 2. *Under Assumption 4, $|I_h| \leq C[3\delta(h)]^{-d^{\eta,\ell}}$.*

Proof. Define $\mathcal{X}_{h,i}^{\geq ave} = \{\boldsymbol{x} \in \mathcal{X}_{h,i} | f(\boldsymbol{x}) \geq \bar{F}(X_{h,i})\}$. At least one point in the representative set $\mathcal{C}_{h,i}$ is in $\mathcal{X}_{h,i}^{\geq ave}$. From Lemma 3, we know I_h is the set of nodes at depth h such that the point with maximum function value in that node is within $3\delta(h)$ of the optimal state. That is,

$$\begin{aligned}
|I_h| &= |\{(h,i) | \bar{F}(X_{h,j}) \geq f^* - 3\delta(h)\}| \\
&\leq |\{(h,i) | f(\boldsymbol{x}) \geq f^* - 3\delta(h), \boldsymbol{x} \in \mathcal{X}_{h,i}^{\geq ave}\}| \\
&= |\{(h,i) | \boldsymbol{x} \in \mathcal{X}_{3\delta(h)} \cap \mathcal{X}_{h,i}^{\geq ave}\}|.
\end{aligned}$$

From Assumption 4, if $|\{(h,i) | \boldsymbol{x} \in \mathcal{X}_{3\delta(h)} \cap \mathcal{X}_{h,i}^{\geq ave}\}| > C[3\delta(h)]^{-d^{\eta,\ell}}$, there exists at least more than $C[3\delta(h)]^{-d^{\eta,\ell}}$ disjoint $\ell-$ball of radius $\nu\delta(h)$, which contradicts the definition of $d^{\eta,\ell}$ (Definition 4, take $\epsilon = 3\delta(h)$). Chaining these together, we have $|I_h| \leq C[3\delta(h)]^{-d^{\eta,\ell}}$. □

Lemma 3. *Under event ξ and Assumption 3 , GPOO only expands nodes in the set $I : \cup_{h \geq 0} I_h$.*

Proof. Let (h_t, j_t) be the node expanded at time t. Define $\mathcal{X}_{h,i}^{\leq ave} = \{\boldsymbol{x} \in \mathcal{X}_{h,i} | f(\boldsymbol{x}) \leq \bar{F}(X_{h,i})\}$. At least one point in the representative set $\mathcal{C}_{h,i}$ is in $\mathcal{X}_{h,i}^{\leq ave}$ and we denote that point as $\boldsymbol{x}_{h,i}^{\leq ave}$, i.e. $\boldsymbol{x}_{h,i}^{\leq ave} \in \mathcal{X}_{h,i}^{\leq ave} \cap \mathcal{C}_{h,i}$. We know from Algorithm 1, the b-value of h_t, j_t is larger than or equal to the b-value of the node (h_t^*, j_t^*) whose corresponding cells contains $\boldsymbol{x}^*$. Define $\boldsymbol{x}_{h_t^*, j_t^*}^{\min} \in \underset{\boldsymbol{x} \in X_{h_t^*, j_t^*}}{\operatorname{argmin}} f(\boldsymbol{x})$ and $\boldsymbol{x}^* \in X_{h_t^*, j_t^*}$. Then under event ξ,

$$\begin{aligned}
&\bar{F}(X_{h_t, j_t}) \\
&\geq \boldsymbol{a}^\top \mu(X_{h_t, j_t} \mid Z_{t-1}) - CI_t(X_{h_t, j_t}) && \text{Lemma 1} \\
&= b_{h_t, j_t}(t) - 2CI_t(X_{h_t, j_t}) - \delta(h_t) && \text{By (5)} \\
&\geq b_{h_t^*, j_t^*}(t) - 3\delta(h_t) && \text{Alg. 1, l7, 3} \\
&\geq \bar{F}\left(X_{h_t^*, j_t^*}\right) + \delta(h_t^*) - 3\delta(h_t) && \text{By (5), event } \xi \\
&\geq f\left(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}\right) + \sup_{\boldsymbol{x} \in X_{h_t^*, j_t^*}} L\ell(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}, \boldsymbol{x}) && \text{Assumption 3} \\
&\quad - 3\delta(h_t) \\
&\geq f\left(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}\right) + L\ell(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}, \boldsymbol{x}^*) - 3\delta(h_t) && \boldsymbol{x}^* \in X_{h_t^*, j_t^*} \\
&\geq f^* - 3\delta(h_t). && \text{Lemma 1}
\end{aligned}$$

□

Proposition 2. *Let f be a sample from a GP with zero mean and covariance function k. Let $X \in \mathbb{R}^{S \times d}$ and let $\Sigma_t(X)$ denote the posterior predictive GP covariance $\Sigma(X \mid Z_{1:t})$ according to* (4)*. If the history $Z_{1:t}$ contains at least $T_X(t)$ observations following the model $Y = \boldsymbol{a}^\top f(X) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2)$, then we have $\sqrt{\boldsymbol{a}^\top \Sigma_t(X) \boldsymbol{a}} \leq \frac{\sigma}{\sqrt{T_X(t)}}$, where $\boldsymbol{a} \in \mathbb{R}^{S \times 1}$ has all entries equal to $1/S$.*

Proof. Partition the entries of $Y_{1:t}$ into two sets and then form the vectors $\boldsymbol{y}_X$, being the vector of reward observations at X, and $\boldsymbol{y}_{X^c}$ being the vector of reward observations at points other than X. Then by the non-negativity of mutual information we have

$$\begin{aligned}
I(\bar{F}(X); \boldsymbol{y}_{X^c} | \boldsymbol{y}_X) &\geq 0 \\
\Rightarrow h(\bar{F}(X) | \boldsymbol{y}_X) - h(\bar{F}(X) | \boldsymbol{y}_X, \boldsymbol{y}_{X^c}) &\geq 0 \qquad (9)
\end{aligned}$$

Recall for univariate Gaussian random variable $Z \sim \mathcal{N}(\mu, \omega^2)$, the differential entropy of Z is $h(Z) = \log(\omega\sqrt{2\pi e})$. As described in (4), we have $\boldsymbol{a}^\top f(X) | \boldsymbol{y}_X, \boldsymbol{y}_{X^c} \sim \mathcal{N}\left(\boldsymbol{a}^\top \mu(X \mid Z_{1:t}), \boldsymbol{a}^\top \Sigma(X | Z_{1:t}) \boldsymbol{a}\right)$, which gives the differential entropy of the second term, i.e. $h\left(f(X) | \boldsymbol{y}_X, \boldsymbol{y}_{X^c}\right) = \log(\sqrt{\boldsymbol{a}^\top \Sigma(X | Z_{1:t}) \boldsymbol{a}} \sqrt{2\pi e})$.

We now evaluate the first term. Let $B \in \mathbb{R}^{t \times t}$ be invertible and let $u, v \in \mathbb{R}^t$. Recall the Sherman-Morrison formula

$$(B + uv^\top)^{-1} = B^{-1} - \frac{B^{-1} u v^\top B^{-1}}{1 + v^\top B^{-1} u}.$$

Let $c = \boldsymbol{a}^\top\ k\left(X, X\right)\boldsymbol{a}$, $z = \frac{c}{\sigma^2}$, and $n = T_X(t)$. For the first term, the posterior variance of $f(\boldsymbol{x})$ given $\boldsymbol{y_x}$ is

$$\begin{aligned}
\Sigma_t(\boldsymbol{x}) &= c - c[1\dots1]\left(c\begin{bmatrix}1 & \dots & 1\\ \dots & \dots & \dots\\ 1 & \dots & 1\end{bmatrix} + \sigma^2 I\right)^{-1} c[1\dots1]^T\\
&= c - [1\dots1]\left(\frac{\sigma^2}{c}I + [1\dots1]^\top[1\dots1]\right)^{-1} c[1\dots1]^T\\
&= c - c[1\dots1]\left(\frac{c}{\sigma^2}I - \frac{\frac{c^2}{\sigma^4}[1\dots1]^\top[1\dots1]}{1+\frac{c}{\sigma^2}[1\dots1][1\dots1]^\top}\right)[1\dots1]^\top\\
&= c - n\frac{c^2}{\sigma^2} + n^2\frac{\frac{c^3}{\sigma^2}}{\sigma^2+nc}\\
&= c\left(1 - \frac{nc}{\sigma^2}(1 - \frac{nc}{\sigma^2+nc})\right)\\
&= c\left(1 - \frac{nc}{\sigma^2}\frac{\sigma^2}{\sigma^2+nc}\right)\\
&= c\frac{\sigma^4}{\sigma^2(\sigma^2+nc)}\\
&= \frac{c\sigma^2}{\sigma^2+nc},
\end{aligned}$$

which gives the differential entropy of the first term as $h(f(X)|\boldsymbol{y}_X) = \log\left(\frac{\sqrt{2\pi e}}{\sqrt{\frac{T_X(t)}{\sigma^2}+\frac{1}{\boldsymbol{a}^\top k(X,X)\boldsymbol{a}}}}\right)$. Then from (9) we have

$$\log\left(\frac{1}{\sqrt{\frac{T_X(t)}{\sigma^2}+\frac{1}{\boldsymbol{a}^\top k(X,X)\boldsymbol{a}}}}\sqrt{2\pi e}\right) - \log(\sqrt{\boldsymbol{a}^\top\Sigma_t(X)\boldsymbol{a}}\sqrt{2\pi e}) \geq 0$$

$$\Rightarrow \frac{\sigma}{\sqrt{T_X(t)}} \geq \sqrt{\boldsymbol{a}^\top\Sigma_t(X)\boldsymbol{a}}.$$

□

Lemma 4. *Under event ξ, suppose a node (h, j) (with corresponding feature matrix $X_{h,j}$ as defined in § 2) is sampled at least $T_{X_{h,j}}(t)$ times up to round t. Then we have $T_{X_{h,j}}(t) \leq \frac{\beta_t\sigma^2}{\delta^2(h)}$, where σ^2 is the noise variance, β_t is defined in Lemma 1.*

Proof. By Algorithm 1 and Proposition 2, we have

$$\delta(h) \leq CI_t(X) \leq \beta_t^{1/2}\frac{\sigma}{\sqrt{T_{X_{h,j}}(t)}} \tag{10}$$

Thus we have

$$T_{X_{h,j}}(t) \leq \frac{\beta_t\sigma^2}{\delta^2(h)}. \tag{11}$$

□

Theorem 1. *Define h_N as the smallest integer h' up to round N such that*

$$\frac{N}{\beta_N} \leq K\sum_{h=0}^{h'} C[3\delta(h)]^{-d^{\eta,\ell}}\frac{\sigma^2}{\delta^2(h)}. \tag{8}$$

For constant a specified in Lemma 1, and $\theta \in (0, 1-a)$, with probability $1-\theta-a$, the simple regret of GPOO satisfies

$$R_N \leq 3\delta(h_N).$$

Proof. Let $T_{X_{h,\cdot}}(N)$ be the number of times a node at depth h has been evaluated up to round N. We consider two types of nodes in terms of whether the nodes have been expanded at round N. According to Lemma 3, for all nodes at level $h \in [0, h_{\max}]$, nodes that have been expanded belong to I_h under event ξ; for those who have not been expanded, we track them according to their parents (which belong to I_{h-1}). Then by Lemma 2 and 4, we have

$$\begin{aligned} N \leq & \sum_{h=0}^{h_{\max}} |I_h| \, T_{X_{h,\cdot}}(N) + (K-1) \sum_{h=1}^{h_{\max}+1} |I_{h-1}| \, T_{X_{h-1,\cdot}}(N) \\ \leq & K \sum_{h=0}^{h_{\max}} C[3\delta(h)]^{-d^{\eta,\ell}} \frac{\beta_{N+1}\sigma^2}{\delta^2(h)}. \end{aligned}$$

From E.q. (8), we know that $h_N \leq h_{\max}$, and from Algorithm 1 and Lemma 3,

$$\bar{F}(X_N) \geq f^* - 3\delta(h_{\max}) \geq f^* - 3\delta(h_N),$$

where X_N is the recommended cell at round N. Thus,

$$R_N = f^* - \bar{F}(X_N) \leq 3\delta(h_N).$$

□

Corollary 1 (Regret bound for exponential diameters)**.** *Assume $\delta(h) = c\rho^h$ for some constants $c > 0$, $\rho < 1$. For constant a specified in Lemma 1, and $\theta \in (0, 1-a)$, with probability $1 - \theta - a$, the simple regret of GPOO satisfies*

$$R_N \leq c_1 \Big[\frac{\beta_N}{N}\Big]^{\frac{1}{d^{\eta,\ell}+2}},$$

where $c_1 = \frac{3^{-d^{\eta,\ell}} KC\sigma^2}{\rho^{-(d^{\eta,\ell}+2)}-1}$. Recall β_N is in rate $\mathcal{O}(\log N)$.

Proof. From Theorem 1, we know

$$\begin{aligned} \frac{N}{\beta_N} &\leq K \sum_{h=0}^{h_n} C[4c\rho^h]^{-d^{\eta,\ell}} \frac{\sigma^2}{\delta^2(h)} \\ &= KC4^{-d^{\eta,\ell}} \sigma^2 c^{-(d^{\eta,\ell}+2)} \frac{\rho^{-(h_n+1)(d^{\eta,\ell}+2)} - 1}{\rho^{-(d^{\eta,\ell}+2)} - 1} \\ &\leq c_1 \delta(h_n)^{-(d^{\eta,\ell}+2)}. \end{aligned}$$

□

B Updating Cholesky decompositions with new rows and columns

Solving a linear system

A Cholesky decomposition may be used to efficiently solve a linear system involving a positive definite matrix. More concretely, suppose $Q \in \mathbb{R}^{t \times t}$ is a positive definite matrix. Then the solution $\boldsymbol{y} \in \mathbb{R}^t$ to

$$Q\boldsymbol{y} = \boldsymbol{b}$$

for $\boldsymbol{b} \in \mathbb{R}^t$ is denoted by $Q \backslash \boldsymbol{b} = Q^{-1}\boldsymbol{b}$. Letting $LL^\top = Q$ denote the Cholesky decomposition of Q (obtained at a computational cost described in the next subsection), we have that $LL^\top \boldsymbol{y} = \boldsymbol{b}$, so that under the substitution $L^\top \boldsymbol{y} = \boldsymbol{z}$, we have

$$\begin{aligned} L\boldsymbol{z} &= \boldsymbol{b} \\ L^\top \boldsymbol{y} &= \boldsymbol{z}. \end{aligned}$$

By exploiting the triangular structure of L, this system may be solved using forward and back substitution at a cost of $\mathcal{O}(t^2)$.

Updating a Cholesky decomposition

Following § 4.2.9 of Golub and van Loan (2013), suppose we have a Cholesky decomposition $Q = LL^\top$ and we wish to find the Cholesky decomposition of

$$\begin{pmatrix} Q & Q_{21}^\top \\ Q_{21} & Q_{22} \end{pmatrix} = \begin{pmatrix} L & \mathbf{0} \\ L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} L & \mathbf{0} \\ L_{21} & L_{22} \end{pmatrix}^\top.$$

Here we are only concerned with the case $Q_{21} \in \mathbb{R}^{1 \times t}$, $Q_{22} \in \mathbb{R}$, $L_{21} \in \mathbb{R}^{1 \times t}$, $L_{22} \in \mathbb{R}$ and $\mathbf{0}$ is a $t \times 1$ vector containing only zeros. We have that

$$\begin{aligned} Q &= LL^\top \\ Q_{21} &= L_{21}L^\top \\ Q_{22} &= L_{21}L_{21}^\top + L_{22}^2, \end{aligned}$$

so that we readily observe that $L_{21}^\top$ is the solution to a linear system that may be obtained by forward and back substitution at a cost of $\mathcal{O}(t^2)$ and subsequently L_{22} is a scalar that may be obtained at an additional cost of $\mathcal{O}(t)$.

C AVE-StoOO

We consider an extension of the StoOO case under aggregated feedback, which we call AVE-StoOO. The setting of (AVE)-StoOO differs slightly to that of GPOO. We make the following alterations to the setting considered in the main paper.

- the reward signal is $r_t = \overline{F}(X_{h_t,i_t}) + \epsilon_t$ where ϵ_t is a random variable satisfying $\mathbb{E}[\epsilon_t | X_{h_t,i_t}] = 0$,
- The reward signal r_t satisfies $r_t \in [0, 1]$.

We define the b-values of the leaves $(h, i) \in \mathcal{L}_t$ as

$$\tilde{b}_{h,i}(t) := \hat{\mu}_{h,i}(t) + \sqrt{\frac{2\log(t^2/\theta)}{T_{X_{h,i}}(t)}} + \delta(h), \tag{12}$$

where $\hat{\mu}_{h,i}(t) := \frac{1}{T_{X_{h,i}}(t)} \sum_{s=1}^{t} r_s \mathbf{1}\{X_s = X_{h,i}\}$ is the empirical average of the rewards received in $X_{h,i}$, and $T_{X_{h,i}}(t) := \sum_{s=1}^{t} \mathbf{1}\{X_s = X_{h,i}\}$ is the number of times (h, i) has been selected up to round t. Recall the shorthand that X_s is the feature matrix corresponding to the cell $\mathcal{X}_s$ selected at round s and r_s.

Algorithm 2: AVE-StoOO

Input: natural number K (-ary tree), function f to be optimised, smoothness function δ, error probability $\theta > 0$, budget N.
Initialisation: tree $\mathcal{T}_0 = \{(0, 0)\}$ (corresponds to $\mathcal{X}$), leaves $\mathcal{L}_0 = \mathcal{T}_0$.
for $t = 1$ to N **do**
  Select any $(h_t, i_t) \in \operatorname{argmax}_{(h,i)\in\mathcal{L}_{t-1}} \tilde{b}_{h,i}(t)$ according to E.q. (12).
  Observe reward $r_t = \overline{F}(X_{h_t,i_t}) + \epsilon_t$.
  if $T_{X_{h,i}}(t) \geq \frac{2\log(t^2/\theta)}{\delta(h)^2}$ **then**
    Expand node (h_t, i_t) (partition $\mathcal{X}_{h_t,i_t}$ into K subsets) and into children nodes $\mathcal{C}_t = \{(h_t + 1, i_1), \ldots, (h_t + 1, i_K)\}$. $\mathcal{T}_t = \mathcal{T}_{t-1} \cup \mathcal{C}_t$. $\mathcal{L}_{t-1} = \mathcal{L}_{t-1} \backslash \{(h_t, i_t)\}$; $\mathcal{L}_t = \mathcal{L}_{t-1} \cup \mathcal{C}_t$.
  end if
end for
Return The any deepest node that has been expanded $\operatorname{argmax}_{(h,i)\in\mathcal{T}_N\backslash\mathcal{L}_N} h$.

Analysis of AVE-StoOO We analyse AVE-StoOO by similar assumptions (Assumption 4, 6, 5) made in Munos (2014).

Assumption 5 (Local smoothness of f)**.** *There exists at least one global optimiser $\boldsymbol{x}^* \in \mathcal{X}$ of f (i.e. $f(\boldsymbol{x}^*) = sup_{\boldsymbol{x}\in\mathcal{X}} f(\boldsymbol{x})$ and for all $\boldsymbol{x} \in \mathcal{X}$, $f(\boldsymbol{x}^*) - f(\boldsymbol{x}) \leq \ell(\boldsymbol{x}, \boldsymbol{x}^*)$.*

Assumption 6 (Decreasing diameters)**.** *There exists a decreasing sequence $\delta(h) > 0$, s.t. for any depth $d \geq 0$ and for any cell $X_{h,i}$ of depth h, we have $\sup_{\boldsymbol{x}\in X_{h,i}} \ell(\boldsymbol{x}_{h,i}, \boldsymbol{x}) \leq \delta(h)$.*

For any $\theta \in (0, 0.5)$, define the following event

$$\tilde{\xi} := \left\{\forall h \geq 0, \forall 0 \leq i \leq K^h - 1, \forall 1 \leq t \leq N \ \left|\hat{\mu}_{h,j}(t) - \bar{F}(X_{h,j})\right| \leq \sqrt{\frac{2\log(t^2/\theta)}{T_{X_{h,j}}(t)}}\right\} \tag{13}$$

Definition 5 (Martingale)**.** *A (discrete-time) martingale (w.r.t $Y_1, Y_2, Y_3, \ldots$) is a discrete-time stochastic process (i.e. a sequence of random variables) $X_1, X_2, X_3, \ldots$ that satisfies for any time n,*

$$\mathbb{E}[|X_n|] < \infty \qquad \mathbb{E}[X_{n+1}|Y_1, \ldots, Y_n] = X_n.$$

Theorem 2 (Azuma's inequality)**.** *Let $\{X_0, X_1, \ldots\}$ be a (super-)martingale w.r.t filtration $\{\mathcal{F}_0, \mathcal{F}_1, \ldots\}$. Assume there are predictable processes $\{A_0, A_1, \ldots\}$ and $B_0, B_1, \ldots$ w.r.t $\{\mathcal{F}_0, \mathcal{F}_1, \ldots\}$, i.e. for all t, A_t, B_t are $\mathcal{F}_{t-1} - measurable$, and constants $0 < c_1, c_2, \cdots < \infty$ s.t. $A_t \leq X_t - X_{t-1} \leq B_t$ and $B_t - A_t \leq c_t$ almost surely. Then for all $\epsilon > 0$,*

$$\mathrm{P}(X_n - X_0 \geq \epsilon) \leq \exp\left(-\frac{2\epsilon^2}{\sum_{t=1}^{n} c_t^2}\right).$$

A submartingale is a supermartingale with signs reversed. If both hold, then by union bound we have a two-sided bound

$$\mathrm{P}(|X_n - X_0| \geq \epsilon) \leq 2\exp\left(-\frac{2\epsilon^2}{\sum_{t=1}^{n} c_t^2}\right).$$

Lemma 5. *For any $\theta \in (0, 0.5)$, for event $\tilde{\xi}$ defined in E.q. 12, we have $\mathbb{P}(\tilde{\xi}) \geq 1 - 2\theta$.*

Proof. The result follows from Azuma's inequality and a union bound.

Let $m \leq N$ be the number of nodes has been expanded up to round t. For $1 \leq i \leq N$, denote t_i as the time step when the ith node is expanded, and (h_{t_i}, j_{t_i}) as the corresponding node. We further denote τ_i^s as time when the node (h_{t_i}, j_{t_i}) has been selected for the sth time, and the reward obtained at that time as $r_{\tau_i^s}$. We re-state event $\tilde{\xi}$ using the above notation,

$$\tilde{\xi} = \left\{ \forall 1 \leq i \leq m, \forall 1 \leq u \leq T_{h_{t_i}, j_{t_i}}(t) \quad \left| \frac{1}{u} \sum_{s=1}^{u} r_{\tau_i^s} - \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right) \right| \leq \sqrt{\frac{2\log\left(t^2/\theta\right)}{u}} \right\} \tag{14}$$

Note $\frac{1}{u}\sum_{s=1}^{u} r_{\tau_i^s} - \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right) = \frac{1}{u}\sum_{s=1}^{u}\left(r_{\tau_i^s} - \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right)\right)$. We have $\mathbb{E}[r_{\tau_i^s} | X_{h_{t_i}, j_{t_i}}] = \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right)$. Let $M_u := \sum_{s=1}^{u}\left(r_{\tau_i^s} - \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right)\right)$ and note that $\{M_u : u = 0, \ldots, T_{h_{t_i}, j_{t_i}}(n)\}$ is a Martingale w.r.t. the filtration generated by the samples collected at $X_{h_{t_i}, j_{t_i}}$, where $M_0 = 0$. Given the assumption that r_t is bounded between 0 and 1, we have $M_u - M_{u-1} = r_{\tau_i^u} - \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right)$ for $u \geq 1$. That is we have $-1 \leq M_u - M_{u-1} \leq 1$, and we make take the difference bound term in Theorem 2 to satisfy $c_u \leq 2$. Then Azuma's inequality implies that

$$\begin{aligned}
&\mathbb{P}\left(\left| \frac{1}{u} \sum_{s=1}^{u} r_{\tau_i^s} - \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right) \right| \geq \sqrt{\frac{2\log\left(t^2/\theta\right)}{u}} \right) \\
=&\mathbb{P}\left(\left| \sum_{s=1}^{u} \left(r_{\tau_i^s} - \bar{F}\left(X_{h_{t_i}, j_{t_i}}\right)\right) \right| \geq u\sqrt{\frac{2\log\left(t^2/\theta\right)}{u}} \right) \\
\leq& 2\exp\left(\frac{-4\left(u\sqrt{\frac{\log(t^2/\theta)}{u}}\right)^2}{4u} \right) \\
=& 2\exp(\log(\theta/t^2)) \\
\leq& 2\theta/t^2.
\end{aligned}$$

Take a union bound over the number of samples $u \leq t$ and the number of expanded nodes $m \leq t$, we deduce the result.

□

Define the set of nodes AVE-StoOO expanded at depth h as

$$\tilde{I}_h := \{(h, i) | \bar{F}\left(X_{h,i}\right) + 3\delta(h) \geq f^*\}. \tag{15}$$

Lemma 6. *On the event $\tilde{\xi}$, AVE-StoOO only expands nodes that belong to the set $\tilde{I} : \cup_{h \geq 0} \tilde{I}_h$.*

Proof. Let (h_t, j_t) be the node expanded at time t. We know from Algorithm 2, the b-value of h_t, j_t is large than or equal to the b-value of the cell h_t^*, j_t^* containing $\boldsymbol{x}^*$. Define $\mathcal{X}_{h,i}^{\leq ave} = \{\boldsymbol{x} \in \mathcal{X}_{h,i} | f(\boldsymbol{x}) \leq \bar{F}\left(X_{h,i}\right)\}$. At least one point in the representative set $\mathcal{C}_{h,i}$ is in $\mathcal{X}_{h,i}^{\leq ave}$ and we denote that point as $\boldsymbol{x}_{h,i}^{\leq ave}$, i.e. $\boldsymbol{x}_{h,i}^{\leq ave} \in \mathcal{X}_{h,i}^{\leq ave} \cap \mathcal{C}_{h,i}$. Then under event $\tilde{\xi}$,

$$\begin{aligned}
\bar{F}\left(X_{h_t, j_t}\right) &\geq \hat{\mu}_{h_t, j_t}(t) - \sqrt{\frac{2\log\left(t^2/\theta\right)}{T_{X_{h,j}}(t)}} && \text{On event } \tilde{\xi} \\
&\geq \hat{\mu}_{h_t, j_t}(t) - \delta\left(h_t\right) && \text{By Algorithm 2} \\
&\geq b_{h_t, j_t}(t) - 3\delta\left(h_t\right) && \text{On event } \tilde{\xi} \\
&\geq b_{h_t^*, j_t^*}(t) - 3\delta\left(h_t\right) && \text{By Algorithm 2} \\
&\geq \bar{F}\left(X_{h_t^*, j_t^*}\right) + \delta\left(h_t^*\right) - 3\delta\left(h_t\right) && \text{On event } \tilde{\xi} \\
&\geq f\left(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}\right) + \delta\left(h_t^*\right) - 3\delta\left(h_t\right) && \text{By defintion of } \boldsymbol{x}_{h,i}^{\leq ave} \\
&\geq f\left(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}\right) + \sup_{\boldsymbol{x} \in X_{h_t^*, j_t^*}} \ell\left(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}, \boldsymbol{x}\right) - 3\delta\left(h_t\right) && \text{By Assumption 6} \\
&\geq f\left(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}\right) + \ell\left(\boldsymbol{x}_{h_t^*, j_t^*}^{\leq ave}, \boldsymbol{x}^*\right) - 3\delta\left(h_t\right) && \boldsymbol{x}^* \in X_{h_t^*, j_t^*} \\
&\geq f^* - 3\delta\left(h_t\right) && \text{By Assumption 5}
\end{aligned}$$

□

Lemma 7. *Recall $d^{\eta,\ell}$ be the $\nu-$near-optimality, and C be the corresponding constant. Then under Assumption 4,*

$$|\tilde{I}_h| \leq C[3\delta(h)]^{-d^{\eta,\ell}}.$$

Proof. Define $\mathcal{X}_{h,i}^{\geq ave} = \{\boldsymbol{x} \in \mathcal{X}_{h,i} | f(\boldsymbol{x}) \geq \bar{F}(X_{h,i})\}$. At least one point in the representative set $\mathcal{C}_{h,i}$ is in $\mathcal{X}_{h,i}^{\geq ave}$. From Lemma 6, we know I_h is the set of nodes at depth h such that the point with maximum function value in that node is within $3\delta(h)$ of the optimal cell. That is,

$$\begin{aligned}
|\tilde{I}_h| &= |\{(h,i) | \bar{F}(X_{h,j}) \geq f^* - 3\delta(h)\}| \\
&\leq |\{(h,i) | f(\boldsymbol{x}) \geq f^* - 3\delta(h), \boldsymbol{x} \in \mathcal{X}_{h,i}^{\geq ave}\}| \\
&= |\{(h,i) | \boldsymbol{x} \in \mathcal{X}_{3\delta(h)} \cap \mathcal{X}_{h,i}^{\geq ave}\}|.
\end{aligned}$$

From Assumption 4, if $|\{(h,i) | \boldsymbol{x} \in \mathcal{X}_{3\delta(h)} \cap \mathcal{X}_{h,i}^{\geq ave}\}| > C[3\delta(h)]^{-d^{\eta,\ell}}$, there exists at least more than $C[3\delta(h)]^{-d^{\eta,\ell}}$ disjoint $\ell-$ball of radius $\nu\delta(h)$, which contradicts the definition of $d^{\eta,\ell}$ (Definition 4, take $\epsilon = 3\delta(h)$). Chaining these together, we have $|\tilde{I}_h| \leq C[3\delta(h)]^{-d^{\eta,\ell}}$. □

Theorem 3. *Let us define h_N be the smallest integer h s.t.*

$$2CK3^{-d^{\eta,\ell}} \sum_{l=0}^{h} \delta(l)^{-(d^{\eta,\ell}+2)} \geq \frac{N}{\log(N^2/\theta)}. \tag{16}$$

Then with probability $1 - 2\theta$, the loss of AVE-StoOO is bounded as $R_N \leq 3\delta(h_N)$.

Proof. Let X_N is the recommended cell at round N. Then following Munos (2014),

$$\begin{aligned}
N &\leq \sum_{h=0}^{h_{\max}} \sum_{j=0}^{K^h-1} T_{X_{h,j}}(N) \mathbb{I}\left\{(h,j) \in \tilde{I}_h\right\} + \sum_{h=1}^{h_{\max}+1} \sum_{j=0}^{K^h-1} T_{X_{h,j}}(N) \mathbb{I}\left\{(h-1,j') \in \tilde{I}_{h-1}\right\} \\
&\leq \sum_{h=0}^{h_{\max}} \left|\tilde{I}_h\right| \frac{2\log\left(N^2/\theta\right)}{\delta(h)^2} + (K-1) \sum_{h=1}^{h_{\max}+1} \left|\tilde{I}_{h-1}\right| \frac{2\log\left(N^2/\theta\right)}{2\delta(h-1)^2} \\
&= K \sum_{h=0}^{h_{\max}} C[3\delta(h)]^{-d^{\eta,\ell}} \frac{2\log\left(N^2/\theta\right)}{\delta(h)^2}
\end{aligned}$$

From E.q. (16), we know that $h_N \leq h_{\max}$, and from Algorithm 2 and Lemma 6,

$$\bar{F}(X_N) \geq f^* - 3\delta(h_{\max}) \geq f^* - 3\delta(h_N),$$

Thus,

$$R_N = f^* - \bar{F}(X_N) \leq 3\delta(h_N).$$

□

D A continuous analogue of aggregated feedback

Here we will informally sketch an extension to a continuous analogue of (1).

Aggregated feedback. Recall that in our construction, each node (h, i) possess two attributes: the cell $\mathcal{X}_{h,i}$ and the S representative points $\mathcal{C}_{h,i}$. We may discard the representative points, and define the reward

$$r_t = \int_{\mathcal{X}_{h_t,i_t}} f(\boldsymbol{x})\, d\boldsymbol{x} + \epsilon_t.$$

Under Assumption 1, such a reward is well-defined whenever the GP f is integrable, i.e. the GP kernel is such that

$$\int_{\mathcal{X}_{h_t,i_t}} \big|f(\boldsymbol{x})\big|\, d\boldsymbol{x} < \infty.$$

Such integrability is guaranteed if $\sup_{\boldsymbol{x} \in \mathcal{X}_{h_t,i_t}} \mathbb{E}|f(\boldsymbol{x})| < \infty$.

GP inference. Since GPs are closed under linear operators (when the action of such operators is well-defined), the reward r_t is Gaussian and the joint distribution of the rewards together with the underlying function f over any set of nodes (h, i) and points $\boldsymbol{x}$ is jointly Gaussian. This allows one to write b-values and confidence intervals for any given node.

E Experiment Details

Detailed Experimental Setting

In this section, we provide detailed experimental settings for the simulated experiments in § 5. Our experiment runs on CPU with Python 3.7. We extended GPy (version 1.10) package for Gaussian Process Regression with aggregated feedback (average).

We considered two reward functions, which are both the posterior mean function from a predefined Gaussian process. We fixed the kernel as radial basis function (RBF) kernel having lengthscale 0.05 and variance 0.1. The GP noise standard variation was set to 0.005. To design the shape of functions to illustrate the performance of algorithms, we predefined a list of points that the GP was conditioned on. For the first function (left in Figure 4), the predefined features are $[0.05, 0.2, 0.4, 0.65, 0.9]$ and the corresponding function values are $[0.85, 0.1, 0.87, 0.05, 0.98]$. For the second function (right in Figure 4), we split the range 0 to 0.9 into 10 equal-size regions, setting the function values of the centers of each region to 0.1, and function values of neighbourhood points (centre plus $2/3$ of the region size) to 0.2. We further added a pair of points $(0.95, 0.9)$ to indicate the optimal point. The shape of the two functions are shown in the first row of Figure 4. To calculate the regret, we find the (empirical optimal point and function) by grid searching the arm space with size 1000. The optimal point is shown in the red point in the first row of Figure 4. We run all algorithms on these two reward functions.

We set the arm space as $[0, 1]$, number of children $K = 2$, budget N ranging from 1 to 80. All experiments are run for 30 independent trials. In the second line of 4, we show the mean of regret curve in solid lines and one standard deviation in shaded area. For GP based algorithms (GPOO, GPTree), we choose to follow our theoretical analysis and use the known GP and do not optimise the hyperparameters. In practice, one can choose to optimise hyperparameters to achieve a potential better performance. Our experimental goal is illustrate the regret rate and do not focus on optimising constant terms. In GPOO, we set $h_{\max} = 10$; $\delta(h) = 14 \times 2^{-h}$. In GPTree, we set all parameters as listed in their theoretical analysis. Similarly, we did not optimise these parameters.

Additional Results

We illustrate an example where aggregated feedback can be beneficial for recommendations for GPOO algorithm in Figure 5. The reward function was generated conditioned the same set of points from the construction of right hand function of 4, plus two pairs of points $(0.94, 0.1), (0.945, 0.2)$. The kernel lengthscale was changed to 0.01. Other settings remained to be the same as above. This toy example shows a case where the objective function has a small amplitude, high frequency component. The aggregated feedback can track the overall pattern of the reward functions quicker than single point feedback The empirical regret curve confirms that for small budget, the average feedback has smaller regret with high probability compared with centre feedback. Note for aggregated feedback case, the recommendation is a cell instead of a single point, that why the converged regret of $S = 10$ case is slightly larger than $S = 1$ case.

We further illustrate how changing parameters (S and K) influences the performance of GPOO algorithm. Based on the same reward function introduced in Figure 5, we set the budget as $N = 25$ and $N = 50$, with 30 independent runs. We run GPOO over all combinations of the split number $S \in \{1, 10, 20, 30, 40, 50\}$, and number of children $K = \{2, 3, 4, 5\}$. The performance in terms of logarithm mean of aggregated regret is shown in Figure 6. $K = 3$ appears to give the best performance in general and larger S tends to provide a better choice.

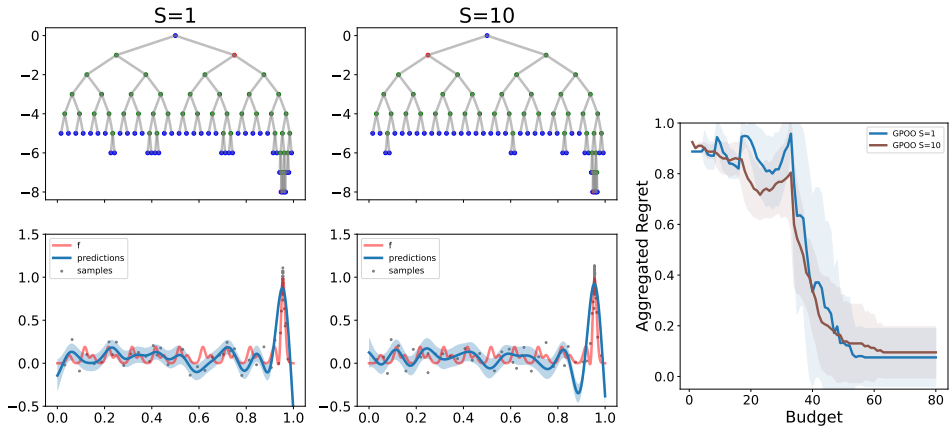

Figure 5: Example showing that aggregated feedback can be beneficial. The first row shows the constructed tree with budget $N = 80$, where node colours indicate the number of times a node has been drawn (blue: 0, green: 1, red: > 1). The second row shows the GP predictions and samples at $N = 80$ (for $S = 10$, the sample points are shown in centre of the selected cells).

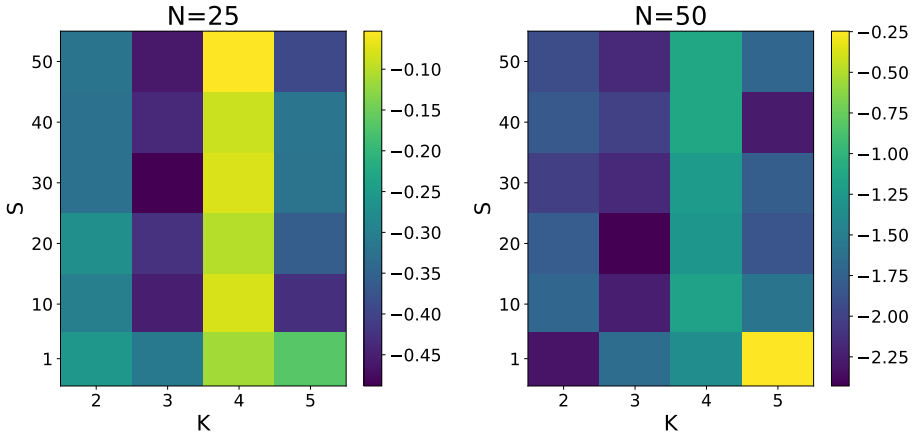

Figure 6: Logarithm Mean of Aggregated Regret for different S and K. The colour shows the logarithm mean of aggregated regret over 30 independent runs. The smaller the colour is, the better the performance is. The left plot is with budget $N = 25$ and the right plot is with budget $N = 50$. Note the colour bar ranges are different for the two plots in order to show the regret changes in one plot clearly.