Response to Reviewer P8MD

We sincerely thank the reviewer for the time and constructive comments. We find the comments all very helpful. We write in below responses to each of the comment. Each item starts with the original comment by the reviewer and follows with our response.

1 Responses to Comments on Weaknesses

1. **Comment:** From originality perspective, the modeling of $g(\theta) * p(x)$ seems quite resembling to equation (6) in Zhang et al. (2021), which the authors cited in the end of Section 1. Better explanation is needed to stress the difference and similarity to this work.

   Responses: We thank the reviewer for the valuable suggestion. In terms of similarity, both our work and Zhang et al. (2021) consider a surrogate model that integrates the Gaussian process (GP) and the neural network (NN). In the surrogates, NN is employed to capture the observed information, that is, covariates in Zhang et al. (2021) and contextual variables in this work, while GP is used to model the unknown function regarding the user-selected decision variable.

   The difference between the two models is mainly two-fold: 1) the structure of the model and 2) application scenarios. 1) Regarding the structure, Zhang et al. (2021) model the parameters of GP (parameters in the kernel function, trend parameters, etc.) as NNs with respect to the covariates. In comparison, our NN-AGP is an inner product of a vector-valued NN and a multi-output GP. That is, in our work, GP is separated from NN while in Zhang et al. (2021), GP depends on NN. 2) Regarding the application scenarios, Zhang et al. (2021) construct the surrogate with the simulated samples at the offline stage. The points where samples are simulated are decided by the user and the set of selected points is restricted to be a Cartesian product of sets of selected decision variables and covariates; see the first paragraph in Section 2.3.1. In addition, after the covariate is revealed, the optimization is taken on the predictors (condition mean of GP) where the uncertainty of the GP predictor is not taken into account. In comparison with their offline setting, we consider a bandit problem where the data comes in sequential, and the pairs of decision variables and contextual variables do not compose a Cartesian product. This means that the model in Zhang et al. (2021) cannot be employed in our problem. Besides, during the sequential optimization procedures, the prediction uncertainty (conditional variance of GP) is considered in our work.

2. **Comment:** The Gaussianality assumption imposed in the prior (and hence posterior) of the reward function could be unrealistic in applications.

   Responses: We thank the reviewer for the constructive comment. It is worth mentioning that GP is widely used as surrogate models to solve black-box optimization problems such as Bayesian optimization and bandit problems. GP provides an explicit uncertainty quantification for the objective function prediction, which helps in the selection of the next points to sample. A stream of works have modeled the unknown objective function with GP, including Srinivas et al. (2010), Krause and Ong (2011), Frazier (2018), to name a few.

   We also note that GP is also connected with kernelized methods, where the objective function is considered as an element from the reproducing kernel Hilbert space (RKHS) and is deterministic. This kernelized assumption is widely used in machine learning, for example, in supporting vector machines. The difference between modeling the objective function as a GP sample or an element in an RKHS reflects the difference between Bayesianists and frequentists, as discussed in Srinivas et al. (2010). In our work, we adopt a Bayesian view since it helps us better understand the construction of the acquisition function.

   Taking a frequentist view would imply that our objective function is drawn from an RKHS associated with the kernel function $\tilde{K}$. Furthermore, by adjusting the selected β_t in each iteration, theoretical results of our methodology on the regret bound can also be derived, which attains an $\tilde{O}(\sqrt{T\gamma_T})$ upper bound on the cumulative regret as well. We appreciate the opportunity to revise our work, and we plan to include a more detailed discussion of this frequentist setting and provide the theoretical results in the revised version.

3. **Comment:** The assumptions on $p(x)$ being a linear combination of independent scaler GPs appear restrictive.

   Responses: We thank the reviewer for the helpful comment. The assumption on $\mathbf{p}(\mathbf{x})$ is consistent with the prevailing model selection of multi-output GP (Nguyen et al. 2014, Liu et al. 2018). This separate structure supports the analysis of both the regret bound and the information gain. On the other hand, we will consider other structures of $\mathbf{p}(\mathbf{x})$. For example, $\mathbf{p}(\mathbf{x})$ is a convolutional MGP, where the summation of kernels in our manuscript is replaced by an integral of kernels; see Alvarez and Lawrence (2011).

2 Responses to Comments on Questions

1. **Comment:** Does the sublinear regret in Theorem 4.2 achieve any kind of minimax lower bound?

   Responses: We thank the reviewer for the valuable comment. We admit that attaining a minimax lower bound in Gaussian process bandit problems is challenging. To our knowledge, neither our work nor the existing literature has provided the lower bound when the objective function is regarded as a sample from a GP; see also Section A.5 of Cai and Scarlett (2021) as a summary. On the other hand, when taking a frequentist view that the reward function is from an RKHS, existing results focus on two specific commonly-selected kernels, squared exponential (SE)[1] and Matern

$$k_{\mathrm{SE}}\left(\mathbf{x}, \mathbf{x}'\right) = \exp\left(-\frac{\left\|\mathbf{x}-\mathbf{x}'\right\|^2}{2l^2}\right)$$
$$k_{\text{Matérn}}\left(\mathbf{x}, \mathbf{x}'\right) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}\left\|\mathbf{x}-\mathbf{x}'\right\|}{l}\right)^{\nu} J_\nu\left(\frac{\sqrt{2\nu}\left\|\mathbf{x}-\mathbf{x}'\right\|}{l}\right),$$

where $l > 0$ denotes the length-scale, $\nu > 0$ is a smoothness parameter, and J_ν is the modified Bessel function. We note that existing results on the lower bound in Scarlett et al. (2017) can be employed to attain the lower bound of NN-AGP-UCB, since NN-AGP maintains a GP structure regarding the decision variable $\mathbf{x}$.

Specifically, we instead assume that the reward function lives in a reproducing kernel Hilbert space (RKHS). To simplify notation, we follow the model assumption in Section 4.3 that $\mathbf{p}(\mathbf{x}) = au(\mathbf{x})$, where $a \in \mathbb{R}^m$ and $u(\mathbf{x}) \in \mathcal{H}_k$. Here $\mathcal{H}_k$ represents an RKHS associated with the kernel function k. We assume that the RKHS norm of u is bounded, that is $\|u\|_k \leqslant B$. In this way, when $g(\theta)$ and a are exactly known (as assumed in **Theorem 4.2** in our work), we have that

$$\mathbb{E}\left[R_T\right] = \Omega\left(\sqrt{T\sigma_\epsilon^2\left(\log\frac{U^2B^2T}{\sigma_\epsilon^2}\right)^{d/2}}\right)$$

when the kernel function of $u(\mathbf{x})$ is the SE kernel as a representative. Here, R_T is the cumulative regret up to time T, the expectation is taken over the noise $\epsilon_t \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$, d denotes the dimension of the decision variable $\mathbf{x}$, and $U = \sup_{\theta\in\Theta} a^\top g(\theta)$. In summary, when the kernel function k of NN-AGP is assumed as SE,

$$\text{lowerbound} = \Omega\left(\sqrt{T(\log T)^{d/2}}\right)$$
$$\text{upperbound} = \tilde{\mathcal{O}}\left(\sqrt{T(\log T)^{d}}\right).$$

We will include the discussion on the frequentist setting and contain the details of this lower bound in the revised manuscript.

2. **Comment:** The assumptions in the theoretical statements could benefit from providing some concrete examples. For example, the authors mentioned Matern kernel and radial basis function kernel in Definition 4.4. Are these satisfy conditions put forward in Theorem 4.2?

   Responses: We thank the reviewer for the helpful comment. Stationary kernels ($k\left(\mathbf{x}, \mathbf{x}'\right) = k'\left(\mathbf{x}-\mathbf{x}'\right)$ for some function k') that are four times differentiable satisfy the conditions; see Theorem 5 in Ghosal and Roy (2006). Therefore, radial basis function kernels satisfy the condition and Matern kernels with $\nu > 2$ satisfy the condition as well, where ν indicates the smoothness of the associated

[1] Radial basis kernel function and SE kernel function are in fact interchangeable terms used to refer to the same kernel function

GP.

3. **Comment:** The authors mentioned the limitation on the computation cost of the modeling approach. Can the authors provide a comparison in terms of computation cost (e.g. wall-clock time) of the methods listed in, for example, Section 5.1?

   Responses: We thank the reviewer for the constructive comment. In terms of the computational time, we record 1) the training time that constructs the surrogate model based on the historical data and 2) the execution time that selects the decision variable after the contextual variable is revealed. We record the time (seconds) for exactly one round at the 50th, 100th, and 300th rounds. We take the first set of experiments in Section 5.1 as an example and present the results in this online link. The experiment results are based on repeating the experiments 15 times and the recorded data represents training time/ execution time.

   We notice that CGP-UCB is the most efficient in both training time and execution time, since 1) it employs a pre-specified GP model which does not update during iterations and 2) the calculation of the inverse matrix can be accomplished efficiently by existing implementations. On the other hand, all the algorithms that involve neural networks require learning neural networks from data and require a longer training time than CGP-UCB. In terms of the execution time, these neural network-involved algorithms perform in a similar fashion. The reason is that all these algorithms require calculating an inverse matrix and the dimension of the matrix is the data size. We note that NN-UCB and NeuralUCB both require calculating the inverse matrix for the reason that they employ the (approximate) neural tangent kernel (NTK) kernel. In other words, they can be categorized as kernelized methods as well.

   In order to enhance the computational efficiency of NN-AGP-UCB, a sparse version of the NN-AGP model is described in Section C, and we will discuss it in detail for future work.

3 Responses to Comments on Limitations

1. **Comment:** The authors did not seem to mention societal impact.

   Responses: We thank the reviewer for the valuable comment. Our work addresses a challenging problem involving sequential decision-making with an unknown objective function and complex contextual variables, and each evaluation of the objective function is expensive to conduct. Our proposed methodology has potential applications in healthcare, where doctors need to develop therapy plans ($\mathbf{x}$) based on patient information (θ) to achieve optimal treatment effects ($f(\mathbf{x};\theta)$). In some cases, complex and sparse genetic information is also employed, requiring the use of neural networks. Another potential application is the use of Automated Guided Vehicles (AGVs) to enhance workplace safety and reduce carbon emissions, where environmental information (θ) is provided to the AGV, and the AGV takes actions ($\mathbf{x}$) accordingly. We plan to include a discussion on the societal impact of our methodology in the revised version.

Again, we appreciate the valuable comments provided by the reviewer. We will make the necessary revisions to the manuscript based on your suggestions.

References

Alvarez, M. A. and Lawrence, N. D. (2011). Computationally efficient convolved multiple output gaussian processes. *The Journal of Machine Learning Research*, 12:1459–1500.

Cai, X. and Scarlett, J. (2021). On lower bounds for standard and robust gaussian process bandit optimization. In *International Conference on Machine Learning*, pages 1216–1226. PMLR.

Frazier, P. I. (2018). A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*.

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

Krause, A. and Ong, C. (2011). Contextual gaussian process bandit optimization. *Advances in neural information processing systems*, 24.

Liu, H., Cai, J., and Ong, Y.-S. (2018). Remarks on multi-output gaussian process regression. *Knowledge-Based Systems*, 144:102–121.

Nguyen, T. V., Bonilla, E. V., et al. (2014). Collaborative multi-output gaussian processes. In *UAI*, pages 643–652. Citeseer.

Scarlett, J., Bogunovic, I., and Cevher, V. (2017). Lower bounds on regret for noisy gaussian process bandit optimization. In *Conference on Learning Theory*, pages 1723–1742. PMLR.

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. (2010). Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning*, number CONF. Omnipress.

Zhang, H., He, J., Zhan, D., and Zheng, Z. (2021). Neural network-assisted simulation optimization with covariates. In *2021 Winter Simulation Conference (WSC)*, pages 1–12. IEEE.