Multi-Objective Neural Bandits with Random Scalarization

Ji Cheng[1,2], **Bo Xue**[1,2], **Chengyu Lu**[1,2], **Ziqiang Cui**[1] and **Qingfu Zhang**[1,2]

[1]Department of Computer Science, City University of Hong Kong, Hong Kong
[2]The City University of Hong Kong Shenzhen Research Institute, Shenzhen, China
{J.Cheng, boxue4-c, chengyulu3-c, ziqiang.cui}@my.cityu.edu.hk, qingfu.zhang@cityu.edu.hk

Abstract

Multi-objective multi-armed bandit (MOMAB) problems are crucial for complex decision-making scenarios where multiple conflicting objectives must be simultaneously optimized. However, most existing works are based on the linear assumption of the feedback rewards, which significantly constrains their applicability and efficacy in capturing the intricate dynamics of real-world environments. This paper explores a multi-objective neural bandit (MONB) framework, which integrates neural networks with the classical MOMABs. We adopt random scalarization to accommodate the special needs of a practitioner by setting an appropriate distribution on the regions of interest. Using the trade-off capabilities of upper confidence bound (UCB) and Thompson sampling (TS) strategies, we design two novel algorithms, MONeural-UCB and MONeural-TS. Theoretical and empirical analysis demonstrate the superiority of our methods in multi-objective or multi-task bandit problems, which makes great improvement over the classical linear MOMABs. Code is available through https://github.com/jicheng9617/MONB.

1 Introduction

Multi-armed bandits (MABs), a fundamental concept in decision theory and reinforcement learning, offer a rich framework to study decision making under uncertainty [Bubeck *et al.*, 2012; Lattimore and Szepesvári, 2020]. Traditional MAB problems involve a single agent that interacts with multiple options, each with unknown reward distributions. The agent's objective is to maximize its cumulative reward over time through sequential interactions, balancing the exploration of lesser-known arms against exploiting those known to yield high rewards. This exploration-exploitation trade-off is central to many real-world applications, ranging from clinical trials [Villar *et al.*, 2015] to personalized recommendations [Li *et al.*, 2010; Li *et al.*, 2011].

Expanding on this framework, multi-objective multi-armed bandits (MOMABs) introduce complexity by having multiple, often conflicting objectives that need to be optimized simultaneously [Drugan and Nowe, 2013]. In these problems, the challenge is not only to balance exploration and exploitation, but also to navigate the trade-offs between competing objectives [Tekin and Turgay, 2018]. This extension is crucial in scenarios where decisions must be made under multiple criteria, for example, diversity and novelty in the recommendation system [Rodriguez *et al.*, 2012]. To measure the performance of multiple rewards, the Pareto order is generally used to fit the scenario where there is no preference between objectives [Lu *et al.*, 2019]. Another way is to transform multi-objective metrics into a single value with scalarization functions [Drugan and Nowe, 2013], where linear and Chebyshev scalarization are widely used. Based on different methods, the corresponding optimal arms can be determined, which are used to gauge the performance of an MOMAB algorithm.

However, a significant limitation in the existing literature on MOMABs is the predominance of models that assume linear reward functions. Although linear models offer simplicity and analytical tractability, they often fail to capture the complexity and nonlinearity inherent in many practical scenarios [Srinivas *et al.*, 2010]. This limitation restricts the applicability and effectiveness of the derived solutions, especially in environments where the relationships between actions and objectives are inherently nonlinear or where interactions between different objectives are complex.

To address these challenges, this work develops provably efficient multi-objective neural bandits (MONBs), which harness deep neural networks (DNNs) [LeCun *et al.*, 2015] as universal approximators to model the reward functions in MOMABs. Random scalarization is considered to cater for the user's preference between multiple metrics. To trade off the abilities between exploration and exploitation, the *upper confidence bound* (UCB) and *Thompson sampling* (TS) are considered to minimize regret. The main contributions are summarized as follows.

- We propose a flexible framework for MONBs using random scalarizations that can flexibly cater to the preference of users by specifying the region of interests. Leveraging DNNs, we alleviate the limitation of linear assumption on the feedback rewards and propose two algorithms: multi-objective neural UCB (MONeural-UCB) and multi-objective neural TS (MONeural-TS). To the best of our knowledge, this is the first work to analyze DNNs-based MOMAB algorithms with regret

Paper	**Objective**	**Model**	**Regret**
[Zhou *et al.*, 2020]	Single	Neural Bandits	$\widetilde{O}(\tilde{d}\sqrt{T})$
[Zhang *et al.*, 2021]	Single	Neural Bandits	$\widetilde{O}(\tilde{d}\sqrt{T})$
[Hwang *et al.*, 2023]	Single	Neural Bandits	$\widetilde{O}(\tilde{d}\sqrt{T})$ or $\widetilde{O}(\sqrt{\tilde{d}TK})$
[Turgay *et al.*, 2018]	Multiple	Lipschitz Bandits	$\widetilde{O}(T^{(1+d_p)/(2+d_p)})$
[Lu *et al.*, 2019]	Multiple	Generalized Linear Bandits	$\mathcal{O}(d\sqrt{T})$
[Cheng *et al.*, 2024]	Multiple	Linear Bandits	$\widetilde{O}((dT)^{2/3})$
MONeural-UCB **(This work)**	Multiple	Neural Bandits	$\widetilde{O}(m\tilde{d}\sqrt{T})$
MONeural-TS **(This work)**	Multiple	Neural Bandits	$\widetilde{O}(\tilde{d}\sqrt{Tm}/\tilde{p}^m)$

Table 1: **Comparison of Related Works on the number of objective, model type, and the regret guarantee.** Our methods reach the optimal regret compared with the multi-objective methods, while alleviate the linear assumption by the use of neural bandit model.

guarantees.

- The proposed MONeural-UCB is statistically efficient and can run with at most $\widetilde{O}(m\tilde{d}\sqrt{T})$ regret, where $\tilde{d}$ is the effective dimension of the neural tangent kernel matrix, and m is the number of objectives. In addition, the MONeural-TS method is efficient in achieving $\widetilde{O}\left(\tilde{d}\sqrt{Tm}/\tilde{p}^m\right)$ regret. The results matches the corresponding regret bounds in linear bandits.
- We conduct experiments on synthetic, multi-objective optimization, and real-world cases, where the evaluations demonstrate the superior performance of our methods. We observe that the benchmark methods perform extremely poorly when the modeling assumptions are broken, while in contrast our methods work well.

2 Related Work

2.1 Neural Bandits

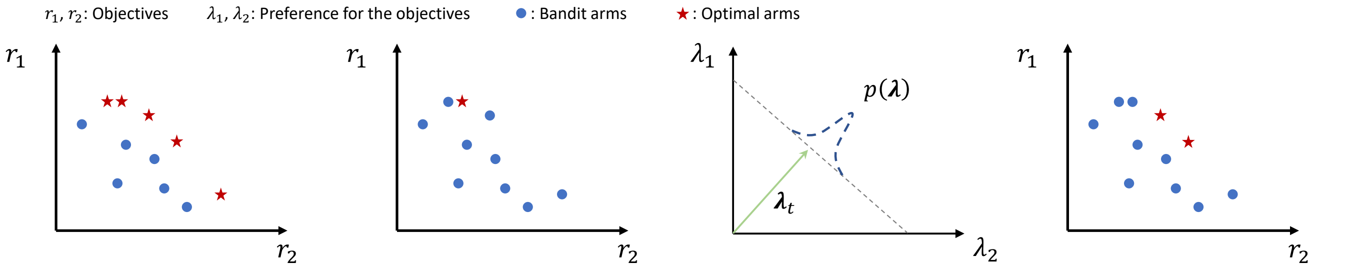
In the realm of stochastic bandits, linear models have been extensively studied and are commonly employed due to their simplicity and analytical tractability. The foundational work on linear stochastic bandits can be traced back to the seminal paper by [Auer, 2002], which introduced the use of confidence bounds to efficiently manage the exploration-exploitation trade-off in environments with linear reward dependencies. Building on these concepts, Dani et al. [2008] explored stochastic linear optimization under bandit feedback, which emphasized the efficiency of algorithms in scenarios where only partial feedback is available, marking a significant step in understanding the dynamics of linear stochastic optimization. Further advancements were made by [Abbasi-yadkori *et al.*, 2011], who contributed to this field by proposing improved algorithms for linear stochastic bandits, which enhance the regret bound by a logarithmic factor. Besides, the linear stochastic bandits have been thoroughly explored over decades [Chu *et al.*, 2011; He *et al.*, 2022; Hu *et al.*, 2021; Alieva *et al.*, 2021; Xue *et al.*, 2023].

Although successful in both theory and practice, the linear assumption highly restricts or even fails to apply to real-world problems, which strongly motivates the study of nonlinear or nonparametric bandits. Filippi et al. [2010] explored the non-linear bandit setting using the Generalized Linear Model (GLM) framework, however, fairly restrictive assumptions are still required on the feedback functions. Furthermore, Valko et al. [2013] extended the linear assumption by the kernel tricks, which requires that the expected reward be an arbitrary linear function of the contexts' images in the related reproducing kernel Hilbert space (RKHS). To further alleviate the limitation of the assumption, researchers attempted to use the universal approximation ability of DNNs as a surrogate of the reward feedback. For example, the work [Riquelme *et al.*, 2018; Zahavy and Mannor, 2019] achieved great success in empirical experiments by taking all but the last layers of a DNN as feature mapping, thanks to the strong representation ability of DNNs. However, no theoretical guarantee for the performance was provided due to the non-linearity of DNNs.

With the development of generalization and optimization theorem of DNNs [Jacot *et al.*, 2018; Cao and Gu, 2019], neural contextual bandits have attracted tremendous attention [Zhou *et al.*, 2020; Zhang *et al.*, 2021; Kassraie and Krause, 2022]. Zhou et al. [2020] first proved the upper bound of UCB-typed neural bandits, whose regret is guaranteed by $\widetilde{O}(\tilde{d}\sqrt{T})$. Later, Zhang et al. [2021] developed the neural bandit algorithm with the traditional TS strategy and proved its advanced performance in the real world dataset. Recently, Salgia [2023] extended the analysis by ReLU neural networks to a general set of smooth activation functions, and non-asymptotic error bounds between the over-parameterized net and the NTK were analyzed to link the smoothness of the activation functions and the kernels. In addition, DNNs were employed in combinatorial bandits [Hwang *et al.*, 2023] to improve the performance of select super arm sets, and in federated setting [Dai *et al.*, 2023] when multiple agents are involved. Based on the development of the theorem, applications have been made in the fields of recommendation and large language models [Ban *et al.*, 2024; Chen *et al.*, 2024].

2.2 Multi-Objective Bandits

To handle bandit problems with multiple objectives, Drugan and Nowe [2013] first introduced the MOMAB with Pareto order and developed two algorithms that involve the upper bound $O(K \log T)$ of the Pareto regret. Turgay et al. [2018]



(a) MOMAB with **Pareto Order** (b) MOMAB with **Lexicographic Order** (c) Prior Distribution of the Preference (d) MOMAB with **Random Scalarization**

Figure 1: A demonstration for MOMABs with different preference. (a) MOMAB with Pareto order treats the multiple objectives equally, and the arms with the objectives in Pareto front are optimal. (b). MOMAB with lexicographic order has the highest priority for the first objective followed by the later ones. (c) A prior $p(\boldsymbol{\lambda})$, the dotted line, imposes a distribution on the preference vector. $\boldsymbol{\lambda}_t$ is a sampled vector. Setting preferred distribution, the users can cater their preference on specific scenarios. (d) Based on the specialized distribution, the MOMAB can identify the specific optimal arms with the preference.

then studied the bandit model with contextual information, where the expected reward satisfied the Lipschitz condition. Later, Lu et al. [2019] considered GLM based bands with an online learning framework. Xu et al. [2023] presented new algorithms and analyses for adversarial MOMAB, providing insights into the formulation of Pareto regrets and their applications. Related works focused on the identification of Pareto optimal arms considering limited budget are also interested in MOMABs [Van Moffaert *et al.*, 2014; Auer *et al.*, 2016; Kone *et al.*, 2023; Kim *et al.*, 2024].

Besides the Pareto relationship among multiple objectives, the dominance and hierarchical relations are also considered in MOMABs community. For example, an algorithm for MOMAB problems where one objective dominates the other was proposed in [Tekin and Turgay, 2018] and achieves $\widetilde{O}(T^{(2\alpha+d)/(3\alpha+d)})$ on both their developed 2D regret and Pareto regret. [Hüyük and Tekin, 2021] first analyzed MOMAB under lexicographic ordering and developed a priority-based regret to assess the bandit algorithm. Their developed algorithm obtained a suboptimal upper bound $\widetilde{O}\left(K^{\frac{2}{3}}T^{\frac{2}{3}}\right)$ for the priority-based regret. Based on this concept, [Xue *et al.*, 2024; Xue *et al.*, 2025] introduced a new parameter to depict the difficulty of lexicographical relations, and improved the algorithm with a multi-stage decision-making strategy. [Cheng *et al.*, 2024] considered a more general relationship, namely mixed lexicographic-Pareto orders, between involved multiple metrics and developed the corresponding stochastic linear algorithms. Most existing work in MOMABs still requires the linear assumption, leaving a huge gap between the general reward functions and the MOMABs community.

3 Problem Setting

In this section, we provide the details of the model and formulate the multi-objective regret minimization problem with personalized scalarization.

Model: We consider a stochastic contextual bandit problem with K arms and specific T rounds in this work. At each round, the learner first observes the contextual information of the K arms $\mathcal{X}_t = \{\mathbf{x}_{t,a} \in \mathbb{R}^d \mid a \in [K]\}$. For brevity, $\{\mathbf{x}_i\}_{i=1}^{TK}$ denotes the collection of $\{\mathbf{x}_{1,1}, \mathbf{x}_{1,2}, \ldots, \mathbf{x}_{T,K}\}$.

Rewards: Once the agent selects an action a_t, it receives a stochastic reward vector consisting of m components $\boldsymbol{r}_{t,a_t} = [r^1_{t,a_t}, r^2_{t,a_t}, \ldots, r^m_{t,a_t}]$. In this work, we assume that each reward comes from an unknown function, which can be formulated as,

$$r^i_{t,a_t} = h^i(\mathbf{x}_{t,a_t}) + \xi_t, \tag{1}$$

where h^i is the unknown function for i-th objective which satisfies $0 \leq h^i(\mathbf{x}) \leq 1$ for any $\mathbf{x} \in \{\mathbf{x}_i\}_{i=1}^{TK}$, and ξ^i_t is v-sub-Gaussian noise satisfying $\mathbb{E}[\xi^i_t \mid \mathbf{x}_{1,a_1}, \ldots, \mathbf{x}_{t-1,a_{t-1}}] = 0$.

Performance Metric: The performance of MOMABs cannot be directly assessed due to the presence of multiple objectives, leading to the categorization of MOMABs based on how preferences over these objectives are defined. As illustrated in Figure 1, the Pareto order is a commonly used metric for evaluating performance, where the Pareto suboptimal gap serves as a measure of deviation from optimality. However, this gap can be minimized by focusing exclusively on a single objective, which may result in biased and unfair arm selection. Practitioners can incorporate their preferences by employing a lexicographic order among the objectives. Nevertheless, the lexicographic approach assumes that one objective has absolute priority over the others, which limits its applicability in many real-world scenarios. In contrast to these preference settings, random scalarization offers a more flexible solution by introducing a distribution over the preference weights. This allows the algorithm to focus on specific regions of interest, as demonstrated in Figure 1 (d), thereby better accommodating diverse user needs.

To judge the performance with m objectives, we consider a set of scalarization functions $\mathcal{S}_{\boldsymbol{\lambda}}$ parameterized by the weight vector $\boldsymbol{\lambda} = [\lambda^1, \ldots, \lambda^m] \in \Lambda$. Without loss of generality, we make the following assumption about the scalarization function.

Assumption 1. *For all $\boldsymbol{\lambda} \in \Lambda$, $\mathcal{S}_{\boldsymbol{\lambda}}$ is $L_{\boldsymbol{\lambda}}$-Lipschitz and monotonically increasing in all the coordinates. Formally,*

$$\begin{aligned} &\mathcal{S}_{\boldsymbol{\lambda}}(\boldsymbol{r}_1) - \mathcal{S}_{\boldsymbol{\lambda}}(\boldsymbol{r}_2) \leq L_{\boldsymbol{\lambda}}\|\boldsymbol{r}_1 - \boldsymbol{r}_2\|, \forall \boldsymbol{\lambda} \in \Lambda, \boldsymbol{r}_1, \boldsymbol{r}_2 \in \mathbb{R}^m, \\ &\mathcal{S}_{\boldsymbol{\lambda}}(\boldsymbol{r}_1) < \mathcal{S}_{\boldsymbol{\lambda}}(\boldsymbol{r}_2) \text{ whenever } r^i_1 < r^i_2. \end{aligned} \tag{2}$$

Algorithm 1 Multi-objective Neural Upper Confidence Bound with Scalarization (MONeural-UCB)

```
Input: Weight distribution $p(\boldsymbol{\lambda})$, Time horizons $T$, regularization parameter $\kappa$, network width $m$.
Initialization: Initialize $m$ neural networks $\boldsymbol{\theta}^i_0$, and set $\mathbf{U}^i_0 = \kappa\mathbf{I}$ for all $i \in [m]$.
for $t = 1, \dots, T$ do
    Sample $\boldsymbol{\lambda}_t$ from $p(\boldsymbol{\lambda})$
    Observe arms' contexts $\{\mathbf{x}_{t,k}\}_{k\in[K]}$
    for $k = 1, \dots, K$ do
        for $i = 1, \dots, m$ do
            Evaluate upper confidence bound $u^i_{t,k} = f^i(\mathbf{x}_{t,k}; \boldsymbol{\theta}^i_{t-1}) + \gamma_{t-1}||\boldsymbol{g}(\mathbf{x}_{t,k}; \boldsymbol{\theta}_{t-1})/\sqrt{m}||_{(\mathbf{U}^i_{t-1})^{-1}}$
        end for
    end for
    Pull the arm $a_t$ and observe the rewards, where $a_t = \max_{a\in[K]} \mathcal{S}_{\boldsymbol{\lambda}_t}(\boldsymbol{u}_{t,k})$ and $\boldsymbol{u}_{t,k} = [u^1_{t,k}, \dots, u^m_{t,k}]^\top$
    Optimize $\boldsymbol{\theta}^i_t$ by gradient descent solving Eq. (5) for $J$ steps
    Update $\mathbf{U}^i_t = \mathbf{U}^i_{t-1} + \boldsymbol{g}(\mathbf{x}_{t,a_t}; \boldsymbol{\theta}^i_{t-1})\boldsymbol{g}(\mathbf{x}_{t,a_t}; \boldsymbol{\theta}^i_{t-1})^\top/m$ for all $i \in [m]$
end for
```

Leveraging the Theorem 1, the best arm at each round can be determined as the one with the maximum scalarization values with the specific weight vector.

Theorem 1 ([Miettinen, 1999]). *Let z^* be the optimal solution of an strongly decreasing scalarization function $\mathcal{S}_{\boldsymbol{\lambda}} : \mathbb{R}^m \to \mathbb{R}$. Then z^* is (weakly) Pareto optimal.*

Assume that the weight vector comes from a prior distribution $p(\boldsymbol{\lambda})$ with support Λ. In this scenario, users can define their personalized distribution to get desirable performance. Based on the scalarization, the performance of an algorithm can be gauged by *pseudo regret* (or *regret* for short) as,

$$\mathcal{R}_T = \mathbb{E}_{\boldsymbol{\lambda}\sim p(\boldsymbol{\lambda})}\left[\mathbb{E}\left[\sum_{t=1}^{T} \mathcal{S}_{\boldsymbol{\lambda}}(\boldsymbol{r}_{t,a^*_t}) - \mathcal{S}_{\boldsymbol{\lambda}}(\boldsymbol{r}_{t,a_t})\right]\right], \quad (3)$$

where $a^*_t = \arg\max_{a\in[K]} \mathbb{E}\left[\mathcal{S}_{\boldsymbol{\lambda}}(\boldsymbol{r}_{t,a})\right]$ is the optimal action at round t that maximizes the expectation of the scalarized rewards.

Reward Approximation: To predict the reward values at each round, we employ neural network to learn the reward h^i in Eq. (1), formally,

$$f^i(\mathbf{x}; \boldsymbol{\theta}) = \sqrt{M}\mathbf{W}_L\sigma\left(\mathbf{W}_{L-1}\sigma\left(\cdots\sigma(\mathbf{W}_1\mathbf{x})\right)\right), \quad (4)$$

where $\sigma(x) = max\{x, 0\}$ is the ReLU activation function, $\mathbf{W}_1 \in \mathbb{R}^{M\times d}$, $\mathbf{W}_l \in \mathbb{R}^{M\times M}$ for $2 \leq l \leq L-1$, $\mathbf{W}_L \in \mathbb{R}^{1\times M}$, and $\boldsymbol{\theta} = [\text{vec}(\mathbf{W}_1)^\top, \dots, \text{vec}(\mathbf{W}_L)^\top]^\top \in \mathbb{R}^p$ with $p = M + Md + M^2(L-2)$. The gradient of the network is denoted by $\boldsymbol{g}^i(\mathbf{x}; \boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} f^i(\mathbf{x}; \boldsymbol{\theta})$.

4 Multi-objective Neural UCB (MONeural-UCB)

4.1 Learning Algorithm

In this section, we first present the algorithm MONeural-UCB, which is described in Algorithm 1. To approximate multiple feedback, m neural networks are maintained through the algorithm for m objectives independently, moreover, the networks are initialized by random generating each entry of $\boldsymbol{\theta}_0$ from an appropriate Gaussian distribution: for $l \in [L-1]$, $\mathbf{W}_l = (\mathbf{W}, \mathbf{0}; \mathbf{0}, \mathbf{W})$, where each entry of $\mathbf{W}$ is generated independently from $\mathcal{N}(0, 4/m)$; $\mathbf{W}_L$ is set as $(\mathbf{w}^\top, -\mathbf{w}^\top)$ with entry sampling from $\mathcal{N}(0, 2/m)$. To balance between exploration and exploitation, MONeural-UCB employs the optimism in the face of uncertainty (OFU) principle during the process. In round t, the learner first evaluates the upper confidence bounds $u^i_{t,k}$ for each arm with respect to m objectives. Then the possibly optimal arm is determined through the scalarization function and sampled personal preference. After observing the reward from the oracle, the algorithm updates the parameters of the neural network $\{\boldsymbol{\theta}^i_t\}^m_{i=1}$ by minimizing the following loss function using gradient descent with step size η for J times.

$$\mathcal{L}^i(\boldsymbol{\theta}) = \frac{1}{2}\sum_{k=1}^{t}\left(f^i(\mathbf{x}_{k,a_k}; \boldsymbol{\theta}) - r^i_{k,a_k}\right)^2 + \frac{M\kappa}{2}||\boldsymbol{\theta} - \boldsymbol{\theta}_0||^2, \quad (5)$$

where, the hyperparameter κ controls the level of ℓ_2-regularization with respect to the initialization of the networks.

4.2 Theoretical Analysis

The analysis is based on the neural tangent kernel, whose main component is introduced by the following expression.

Definition 1 ([Jacot *et al.*, 2018]). *Define*

$$\widetilde{\mathbf{H}}^{(1)}_{i,j} = \mathbf{\Sigma}^{(1)}_{i,j} = \left\langle \mathbf{x}^i, \mathbf{x}^j \right\rangle, \mathbf{A}^{(\ell)}_{i,j} = \begin{pmatrix} \mathbf{\Sigma}^{(\ell)}_{i,i} & \mathbf{\Sigma}^{(\ell)}_{i,j} \\ \mathbf{\Sigma}^{(\ell)}_{j,i} & \mathbf{\Sigma}^{(\ell)}_{j,j} \end{pmatrix},$$

$$\mathbf{\Sigma}^{(\ell+1)}_{i,j} = 2\mathbb{E}_{(y,z)\sim\mathcal{N}\left(\mathbf{0}, \mathbf{A}^{(\ell)}_{i,j}\right)}[\sigma(y)\sigma(z)],$$

$$\widetilde{\mathbf{H}}^{(\ell+1)}_{i,j} = 2\widetilde{\mathbf{H}}^{(\ell)}_{i,j}\mathbb{E}_{(y,z)\sim\mathcal{N}\left(\mathbf{0}, \mathbf{A}^{(\ell)}_{i,j}\right)}\left[\sigma'(y)\sigma'(z)\right] + \mathbf{\Sigma}^{(\ell+1)}_{i,j}.$$

Then, $\mathbf{H} = (\widetilde{\mathbf{H}}^{(L)} + \mathbf{\Sigma}^{(L)})/2$ is called the *neural tangent kernel (NTK)* matrix on the contexts $\{\mathbf{x}_i\}^{TK}_{i=1}$.

Based on the kernel methods, the *effective dimension* of the NTK matrix $\mathbf{H}$ can be defined as follows.

Definition 2. *The effective dimension $\tilde{d}$ of the NTK matrix is defined as*

$$\tilde{d} = \frac{\log\det(\mathbf{I} + \mathbf{H}/\lambda)}{\log(1 + TK/\lambda)} \quad (6)$$

Algorithm 2 Multi-objective Neural Thompson Sampling with Scalarization (MONeural-TS)

```
Input: Weight distribution $p(\boldsymbol{\lambda})$, Time horizons $T$, regularization parameter $\kappa$, network width $m$, exploration variance $\rho$.
Initialization: Initialize $m$ neural networks $\boldsymbol{\theta}_0^i$, and set $\mathbf{U}_0^i = \kappa\mathbf{I}$ for all $i \in [m]$.
for $t = 1, \ldots, T$ do
    Sample $\boldsymbol{\lambda}_t$ from $p(\boldsymbol{\lambda})$
    Observe arms' contexts $\{\mathbf{x}_{t,k}\}_{k\in[K]}$
    for $k = 1, \ldots, K$ do
        for $i = 1, \ldots, m$ do
            $(\sigma_{t,k}^i)^2 = \kappa \boldsymbol{g}^i(\mathbf{x}_{t,k}; \boldsymbol{\theta}_{t-1}^i)^\top (\mathbf{U}_{t-1}^i)^{-1} \boldsymbol{g}^i(\mathbf{x}_{t,k}; \boldsymbol{\theta}_{t-1}^i)/m$
            Sample reward $\tilde{r}_{t,k}^i \sim \mathcal{N}\left(f^i(\mathbf{x}_{t,k}; \boldsymbol{\theta}_{t-1}^i), (\rho\sigma_{t,k}^i)^2\right)$
        end for
    end for
    Pull the arm $a_t$ and observe the rewards, where $a_t = \max_{a\in[K]} \mathcal{S}_{\boldsymbol{\lambda}_t}(\tilde{\boldsymbol{r}}_{t,k})$ and $\tilde{\boldsymbol{r}}_{t,k} = [\tilde{r}_{t,k}^1, \ldots, \tilde{r}_{t,k}^m]^\top$
    Optimize $\boldsymbol{\theta}_t^i$ by gradient descent solving Eq. (5)
    $\mathbf{U}_t^i = \mathbf{U}_{t-1}^i + \boldsymbol{g}^i(\mathbf{x}_{t,a_t}; \boldsymbol{\theta}_t^i) \boldsymbol{g}^i(\mathbf{x}_{t,a_t}; \boldsymbol{\theta}_t^i)^\top / m$ for all $i \in [m]$
end for
```

The effective dimension can be thought of the actual dimension of the Reproducing Kernel Hilbert Space (RKHS) restricted by the given contexts, and it measures how quickly the eigenvalues diminishes by logarithm of T. Without loss of generality, we suppose the following conditions on the contexts:

Assumption 2. $\mathbf{H} \succeq \lambda_0\mathbf{I}$ *for some* $\lambda_0 > 0$. *Moreover, for any* $i \in [TK]$, $\|\mathbf{x}_i\|_2 = 1$ *and* $[\mathbf{x}_i]_j = [\mathbf{x}_i]_{j+\frac{d}{2}}$ *for* $1 \leq j \leq \frac{d}{2}$.

This mild assumption can be simply reached by setting $\mathbf{x}' = [\mathbf{x}^\top, \mathbf{x}^\top]^\top$ followed by normalization. Based on the assumptions and definitions above, we can upper-bound the regret of the proposed algorithm as the following theorem states. The proof of the theorem can be found in Appendix.

Theorem 2. *Assume the number of arms to be finite, i.e.* $K < \infty$. *Let* $\tilde{d}$ *be the effective dimension for the network, and* $\mathbf{h}^i = [h^i(\mathbf{x}_j)]_{j=1}^{TK} \in \mathbb{R}^{TK}$. *If MONeural-UCB runs with*

$$
\begin{aligned}
& m \geq \textit{poly}(T, L, K, \kappa, \lambda_0^{-1}, S^{-1}, \log(1/\delta)), \\
& \eta = C_1(TL + M\kappa)^{-1}, \delta \in (0,1), \\
& \kappa \geq C_2LK, S = \max_{i\in[m]} \sqrt{2\mathbf{h}^{i\top}\mathbf{H}^{-1}\mathbf{h}^i}, \\
& J = 2\log\frac{\kappa S}{\sqrt{T}(\kappa + C_3TL)}\frac{TL}{\kappa} = \tilde{\mathcal{O}}(TL/\kappa),
\end{aligned}
$$

for some positive constants C_1, C_2, C_3, *then with probability at least* $1-\delta$, *the cumulative regret can be upper-bounded by*

$$
\mathcal{R}_T = \widetilde{O}\left(\sqrt{m\tilde{d}T}\sqrt{\max\left\{m\tilde{d}, S\right\}}\right). \tag{7}
$$

Remark 1. *Theorem 2 establishes the upper bound of regret by* $\mathcal{O}(m\tilde{d}\sqrt{T})$. *The result matches that of the start-of-the-art multi-objective contextual bandits. If the feedback function* h *belongs to the RKHS* $\mathcal{H}$ *induced by the NTK w.r.t. each objective, then the RKHS norm* $\|h^i\|_{\mathcal{H}} \geq \mathbf{h}^{i\top}\mathbf{H}^{-1}\mathbf{h}^i$, *and the regret bound can be further denoted as*

$$
\mathcal{R}_T = \widetilde{O}\left(\sqrt{m\tilde{d}T}\sqrt{\max\left\{m\tilde{d}, \max_{i\in[m]}\|h^i\|_{\mathcal{H}}\right\}}\right). \tag{8}
$$

5 Multi-objective Neural TS (MONeural-TS)

5.1 Learning Algorithm

In this section, we develop the TS-based method considering neural networks, MONeural-TS. Instead of focusing on the posterior estimate of the model parameters, the method maintains a Gaussian distribution for each feedback reward, where the mean values are the output of the networks and the variance is evaluated from the corresponding feature map. The TS method samples the rewards from the distribution and then uses the sampled rewards with scalarization function to determine the sub-optimal arms.

5.2 Theoretical Analysis

Under the assumption stated above, the performance of the proposed MONeural-TS method can be measured using the following theorem.

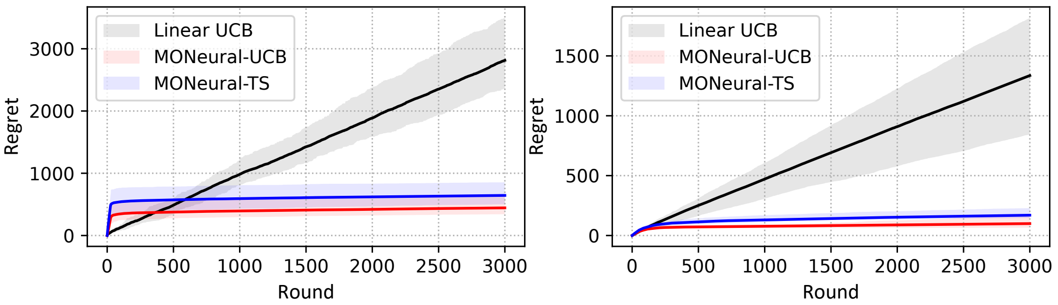
Theorem 3. *Assume that the width of the neural networks satisfies the condition in Theorem 2. If MONeural-TS runs with*

$$
\begin{aligned}
& \eta = C_1(TML + M\kappa)^{-1}, \kappa = \max\{1 + 1/T, C_2LK\}, \\
& J = (1 + TL/\kappa)(C_3 + \log(T^3L\kappa^{-1}\log(1/\delta)))/C_1, \\
& \rho = B + \nu\sqrt{\tilde{d}\log(1 + TK/\kappa) + 2 - 2\log\delta}, \\
& B = \max\left\{1/(22e\sqrt{\pi}), \sqrt{\max_{i\in[m]} 2\mathbf{h}^{i\top}\mathbf{H}^{-1}\mathbf{h}^i}\right\},
\end{aligned}
$$

for positive constants $C_1, C_2, C_3, \tilde{p} = 1/(4e\sqrt{\pi})$, *then with probability at least* $1-\delta$, *the regret can be bounded as*

$$
\mathcal{R}_T = \widetilde{O}\left(\tilde{d}\sqrt{Tm}/\tilde{p}^m\right). \tag{9}
$$

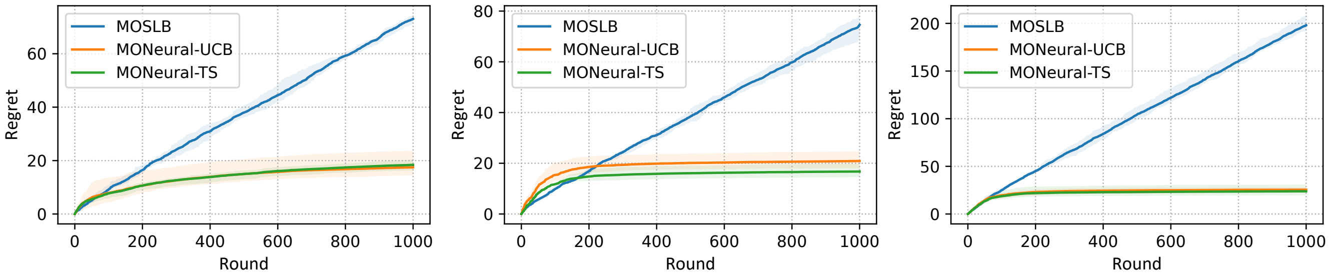
Remark 2. *The regret bound involves an exponential factor of* m, *which may be computationally prohibitive in many-objective problems. To mitigate this issue, we can draw multiple independent samples and select the most optimistic one, who has the highest scalarization function metric, following*



(a) Quadratic

(b) Consine

Figure 2: Cumulative regret of MONeural-UCB and MONeural-TS on the synthetic cases. Results are averaged over 10 different randomly sampled parameters with the standard deviation shown as shaded areas.



(a) F1

(b) F2

(c) DTLZ2

Figure 3: Iterative history of the cumulative regrets on the MOO cases.

a technique adapted from [Hwang et al.*, 2023]. This results in a regret bound of*

$$\mathcal{R}_T = \widetilde{O}(\tilde{d}\sqrt{Tm}). \tag{10}$$

Remark 3. *Theorem 2 and Theorem 3 all depend on the condition that the width of the networks must be sufficiently large, while empirical experiments find that networks with much small size work efficiently. This phenomenon is due to the huge gap between the practical application and the NTK theorem, which has also been indicated in the single-objective version of neural bandits [Zhou* et al.*, 2020; Zhang* et al.*, 2021].*

Remark 4. *When dealing with a scenario where the time horizon is unknown, the doubling trick [Cesa-Bianchi and Lugosi, 2006] can be employed to adapt the algorithm. The key implementation is to restart the algorithm at each non-overlapping interval, and a similar regret $\widetilde{O}(\tilde{d}\sqrt{Tm})$ can be verified.*

Remark 5. *Updating m neural networks can be costly when m is large. In this way, we can conduct the algorithm in batched setting, i.e., update the networks after several rounds. The rarely switching strategy [Abbasi-yadkori* et al.*, 2011] can be used to save computation. Besides, a large number of the parameters of the neural netowrks may result in enormous computational cost in the inverse of the matrix. In practical use, we can only maintain the diagonal of the matrix U_t^i to reduce the cost, as indicated in [Zhou* et al.*, 2020].*

6 Numerical Experiments

In this section, we test numerical experiments on synthetically generated and real-world data sets to verify the performance of our theoretical findings. We compare our methods with a multi-objective stochastic linear bandits (MOSLB) method, which is adapted from the scalarized MOMAB method with UCB exploration [Drugan and Nowe, 2013]. All implementations are performed on a dedicated system configured with an Intel Core i7-9700K CPU and an NVIDIA GeForce RTX 2080 Ti GPU.

6.1 Synthetic Cases

We first validate our proposed methods on two synthetic data generated as follows.

- **Quadratic Reward.** The reward for each objective is given by $r_{t,k}^i = \mathbf{x}_{t,k}^\top \mathbf{A}_i^\top \mathbf{A}_i \mathbf{x}_{t,k} + \xi_t^i, \forall i \in [m]$, where the contexts $\mathbf{x}_{t,k}, \forall k \in [K]$ are generated by uniform sampling in $[0,1]^d$ independently, and each entry of the matrix $\mathbf{A}_i \in \mathbb{R}^{d\times d}$ is sampled by $\mathcal{N}(0,1)$. The noise for each objective is independently drawn from the Gaussian distribution $\mathcal{N}(0, 0.25)$ for all arms w.r.t. each objective. We choose $m = 2$, $d = 10$, and $K = 10$ at this time.
- **Cosine Reward.** The cosine function $\cos(3\mathbf{x}_{t,k}^\top \boldsymbol{\theta}_i^*)$ is adapted as the unknown function h^i in this case, the contexts are generated based on $U[0,1]^d$ independently, and the parameters $\boldsymbol{\theta}_i^*$ are first generated according to

Algorithm	Metric	Varying K			Varying d			Varying m		
		$K = 10$	$K = 20$	$K = 30$	$d = 10$	$d = 20$	$d = 30$	$m = 2$	$m = 3$	$m = 5$
MOSLB	Regret	83.47	147.8	219.6	83.47	1269.4	1481.0	83.47	215.9	225.0
	Time (s)	3.90	4.31	4.93	3.90	4.10	4.53	3.90	5.13	5.55
MONeural (UCB)	Regret	**14.60**	**20.79**	**23.91**	**14.60**	94.79	307.6	**14.60**	**15.56**	**15.24**
	Time (s)	73.02	75.71	76.45	73.02	79.21	76.83	73.02	82.40	161.27
MONeural (TS)	Regret	29.97	33.64	39.14	29.97	**69.56**	**301.8**	29.97	23.47	22.23
	Time (s)	54.61	66.78	62.35	54.61	52.50	54.42	54.61	79.79	164.96

Table 2: Performance Comparison of the cosine reward function under Different Parameter Settings. We report the average cumulative regrets for $T = 3000$ over three repetitive runs. We use $K = 10, d = 10, m = 2$ as the default setting, and only change the varying parameter for each experiment.

$U[0,1]^d$ and then normalized to satisfy $\|\boldsymbol{\theta}_i^*\| = 1$, and the noise is sampled as the same. We choose $m = 3$, $d = 20$, and $K = 20$ at this time.

We conducted the experiments for $T = 3000$, and each experiment was repeated 10 times. Each objective was estimated by a neural network with one hidden layer containing 100 neurons, and furthermore, we trained the networks with Adam optimizer by the learning rate $\eta = 0.005$, and with the step $J = 1$ each round. For the exploration factor, γ in MONeural-UCB and ρ in MONeural-TS, we chose 0.1 in these experiments. The scalarization vectors are sampled uniformly from the simplex. Detailed discussion on the hyperparameters can be found in the Appendix D. The iterative graphs of the cumulative regret are shown in Figure 2, from which we can observe that the proposed methods converge after hundreds of rounds, while the linear method was performed with linear regret.

Effects of the Parameters. We further tested the performance of our algorithms with cosine reward functions under varying parameters. As shown in Table 2, it demonstrates that changes in K and m have minimal impact on regret values, while variations in feature dimension d significantly influence the algorithm performance. This dimensional sensitivity strongly aligns with our theoretical analysis, which predicts a linear relationship between regret and feature dimensionality. The running time of our algorithms is closely related to the number of objectives m, yet even with five objectives, action selection takes only 50ms.

6.2 MOO Cases

We repeat the experiments using three classical multi-objective optimization (MOO) problems [Zhang *et al.*, 2009; Lin *et al.*, 2022]. At each round, ten arms with the context randomly sampled from the decision space are evaluated by the Linear and our proposed methods. Based on hyperparameter tuning, we train two-layer neural networks with hidden layers $M = 200$. For neural bandits, we choose $\gamma = 0.01$ and $\rho = 0.05$ for the UCB and TS methods, respectively. The per-instance regret results are shown in Figure 3. We can observe that the proposed methods work pretty well on the tested three cases, which demonstrates the ability to handle the non-linear feedback function under the regret of squared time horizon.

6.3 Real-world Dataset

We further empirically evaluate our methods in two real-world public datasets in the multitask learning community, i.e. *multiMNIST* [Sabour *et al.*, 2017] and *multiFashionMNIST* [Lin *et al.*, 2019]. To fit the classification tasks to the MOMAB problems, we pair each input feature with the output labels to form the contextual feature vector for each arm. In addition to the correctly labeled arm, the other $K - 1$ arms are randomly selected with the same input features and random labels. The algorithm received a 1 reward if the correct arm was selected. The rewards for each objective are shown in the Appendix D. Consistent with the previous finding, using the universal approximator, performance is improved compared to that using linear model.

7 Conclusions

In this paper, we studied the general case of contextual multi-objective multi-armed bandit problems, where the unknown rewards of each arm are modeled by neural networks. Based on two strategies to balance between exploration and exploitation, we proposed two algorithms: MONeural-UCB and MONeural-TS. Given the recent advances in generalization and optimization theorem of deep neural networks, we theoretically prove that the MONeural-UCB method can run with regret less than $\widetilde{O}\left(\sqrt{m\tilde{d}T}\sqrt{\max\left\{m\tilde{d}, \max_{i\in[m]}\|h^i\|_{\mathcal{H}}\right\}}\right)$, while MONeural-TS can be run with the regret less than $\widetilde{O}\left(\tilde{d}\sqrt{Tm}/\tilde{p}^m\right)$. Empirical results in synthetic and real-world problems demonstrate the promising performance of the proposed methods.

Limitation and future work. Modeling each objective with a neural network is computationally expensive when there is a large amount of objectives. In future work, we may try to take advantage of the thoughts of Pareto set learning (PSL) [Lin *et al.*, 2022] and combine it with the MONBs smoothly.

Acknowledgments

The work described in this paper was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China [GRF Project No. CityU 11212524].

References

[Abbasi-yadkori *et al.*, 2011] Yasin Abbasi-yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

[Alieva *et al.*, 2021] Ayya Alieva, Ashok Cutkosky, and Abhimanyu Das. Robust pure exploration in linear bandits with limited budget. In *International Conference on Machine Learning*, pages 187–195, 2021.

[Allen-Zhu *et al.*, 2019] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 242–252, 2019.

[Auer *et al.*, 2016] Peter Auer, Chao-Kai Chiang, Ronald Ortner, and Madalina Drugan. Pareto front identification from stochastic bandit feedback. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 939–947, 2016.

[Auer, 2002] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

[Ban *et al.*, 2024] Yikun Ban, Yunzhe Qi, and Jingrui He. Neural contextual bandits for personalized recommendation. In *Companion Proceedings of the ACM on Web Conference 2024*, WWW '24, page 1246–1249, 2024.

[Bubeck *et al.*, 2012] Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

[Cao and Gu, 2019] Yuan Cao and Quanquan Gu. Generalization bounds of stochastic gradient descent for wide and deep neural networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[Cesa-Bianchi and Lugosi, 2006] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[Chen *et al.*, 2024] Zekai Chen, Weeden Daniel, Po yu Chen, and Francois Buet-Golfouse. Online personalizing white-box llms generation with neural bandits, 2024.

[Cheng *et al.*, 2024] Ji Cheng, Bo Xue, Jiaxiang Yi, and Qingfu Zhang. Hierarchize pareto dominance in multi-objective stochastic linear bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11489–11497, 2024.

[Chu *et al.*, 2011] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.

[Dai *et al.*, 2023] Zhongxiang Dai, Yao Shu, Arun Verma, Flint Xiaofeng Fan, Bryan Kian Hsiang Low, and Patrick Jaillet. Federated neural bandits. In *The Eleventh International Conference on Learning Representations*, 2023.

[Dani *et al.*, 2008] Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. In *Proceedings of the 21st Annual Conference on Learning*, pages 355–366, 2008.

[Drugan and Nowe, 2013] Madalina M Drugan and Ann Nowe. Designing multi-objective multi-armed bandits algorithms: A study. In *The 2013 international joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2013.

[Filippi *et al.*, 2010] Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, volume 23, 2010.

[He *et al.*, 2022] Jiafan He, Dongruo Zhou, Tong Zhang, and Quanquan Gu. Nearly optimal algorithms for linear contextual bandits with adversarial corruptions. In *Advances in Neural Information Processing Systems*, volume 35, pages 34614–34625, 2022.

[Hu *et al.*, 2021] Jiachen Hu, Xiaoyu Chen, Chi Jin, Lihong Li, and Liwei Wang. Near-optimal representation learning for linear bandits and linear rl. In *International Conference on Machine Learning*, pages 4349–4358, 2021.

[Hüyük and Tekin, 2021] Alihan Hüyük and Cem Tekin. Multi-objective multi-armed bandit with lexicographically ordered and satisficing objectives. *Machine Learning*, 110(6):1233–1266, 2021.

[Hwang *et al.*, 2023] Taehyun Hwang, Kyuwook Chai, and Min-hwan Oh. Combinatorial neural bandits. In *International Conference on Machine Learning*, pages 14203–14236. PMLR, 2023.

[Jacot *et al.*, 2018] Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[Kassraie and Krause, 2022] Parnian Kassraie and Andreas Krause. Neural contextual bandits without regret. In *International Conference on Artificial Intelligence and Statistics*, pages 240–278. PMLR, 2022.

[Kim *et al.*, 2024] Wonyoung Kim, Garud Iyengar, and Assaf Zeevi. Learning the pareto front using bootstrapped observation samples, 2024.

[Kone *et al.*, 2023] Cyrille Kone, Emilie Kaufmann, and Laura Richert. Adaptive algorithms for relaxed pareto set identification, 2023.

[Lattimore and Szepesvári, 2020] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[LeCun *et al.*, 2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[Li *et al.*, 2010] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

[Li *et al.*, 2011] Lihong Li, Wei Chu, John Langford, and Xuanhui Wang. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 297–306, 2011.

[Lin *et al.*, 2019] Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qing-Fu Zhang, and Sam Kwong. Pareto multi-task learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[Lin *et al.*, 2022] Xi Lin, Zhiyuan Yang, Xiaoyuan Zhang, and Qingfu Zhang. Pareto set learning for expensive multi-objective optimization. In *Advances in Neural Information Processing Systems*, volume 35, pages 19231–19247, 2022.

[Lu *et al.*, 2019] Shiyin Lu, Guanghui Wang, Yao Hu, and Lijun Zhang. Multi-objective generalized linear bandits. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, page 3080–3086, 2019.

[Miettinen, 1999] Kaisa Miettinen. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media, 1999.

[Riquelme *et al.*, 2018] Carlos Riquelme, George Tucker, and Jasper Snoek. Deep bayesian bandits showdown: An empirical comparison of bayesian deep networks for thompson sampling. In *International Conference on Learning Representations*, 2018.

[Rodriguez *et al.*, 2012] Mario Rodriguez, Christian Posse, and Ethan Zhang. Multiple objective optimization in recommender systems. In *Proceedings of the Sixth ACM Conference on Recommender Systems*, page 11–18, 2012.

[Sabour *et al.*, 2017] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[Salgia, 2023] Sudeep Salgia. Provably and practically efficient neural contextual bandits. In *International Conference on Machine Learning*, pages 29800–29844. PMLR, 2023.

[Srinivas *et al.*, 2010] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: no regret and experimental design. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, page 1015–1022, 2010.

[Tekin and Turgay, 2018] Cem Tekin and Eralp Turgay. Multi-objective contextual multi-armed bandit with a dominant objective. *IEEE Transactions on Signal Processing*, 66(14):3799–3813, 2018.

[Turgay *et al.*, 2018] Eralp Turgay, Doruk Oner, and Cem Tekin. Multi-objective contextual bandit problem with similarity information. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, pages 1673–1681, 2018.

[Valko *et al.*, 2013] Michal Valko, Nathan Korda, Rémi Munos, Ilias Flaounas, and Nello Cristianini. Finite-Time Analysis of Kernelised Contextual Bandits. In *Uncertainty in Artificial Intelligence*, 2013.

[Van Moffaert *et al.*, 2014] Kristof Van Moffaert, Kevin Van Vaerenbergh, Peter Vrancx, and Ann Nowé. Multi-objective χ-armed bandits. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 2331–2338, 2014.

[Villar *et al.*, 2015] Sofía S Villar, Jack Bowden, and James Wason. Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 30(2):199, 2015.

[Xu and Klabjan, 2023] Mengfan Xu and Diego Klabjan. Pareto regret analyses in multi-objective multi-armed bandit. In *International Conference on Machine Learning*, pages 38499–38517, 2023.

[Xue *et al.*, 2023] Bo Xue, Yimu Wang, Yuanyu Wan, Jinfeng Yi, and Lijun Zhang. Efficient algorithms for generalized linear bandits with heavy-tailed rewards. *Advances in Neural Information Processing Systems*, 36:70880–70891, 2023.

[Xue *et al.*, 2024] Bo Xue, Ji Cheng, Fei Liu, Yimu Wang, and Qingfu Zhang. Multiobjective lipschitz bandits under lexicographic ordering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16238–16246, 2024.

[Xue *et al.*, 2025] Bo Xue, Xi Lin, Xiaoyuan Zhang, and Qingfu Zhang. Multiple trade-offs: An improved approach for lexicographic linear bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 21850–21858, 2025.

[Zahavy and Mannor, 2019] Tom Zahavy and Shie Mannor. Deep neural linear bandits: Overcoming catastrophic forgetting through likelihood matching, 2019.

[Zhang *et al.*, 2009] Qingfu Zhang, Wudong Liu, Edward Tsang, and Botond Virginas. Expensive multiobjective optimization by moea/d with gaussian process model. *IEEE Transactions on Evolutionary Computation*, 14(3):456–474, 2009.

[Zhang *et al.*, 2021] Weitong Zhang, Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural thompson sampling. In *International Conference on Learning Representations*, 2021.

[Zhou *et al.*, 2020] Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with UCB-based exploration. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 11492–11502, 2020.