A. Pseudocode for Collaborative BO Algorithm

Algorithm 1 Collaborative BO algorithm

```
Require: Objective function $f$, input domain $\mathcal{X}$, number of parties $n$, time budget $T$, initial time exploration budget $T_0$,
    prior mean $m_{\mathbf{x}}$ and covariance $k_{\mathbf{x}\mathbf{x}'}$
for $t = 1, \dots, T$ do
    if $t < T_0$ then
        Select batch $\mathbf{X}_t$ randomly.
    else
        Select batch $\mathbf{X}_t \triangleq \arg\max_{\mathbf{X}_t \in \mathcal{X}^n} \sum_{i=1}^{n} w_i \phi_i((\lambda_t^i + \mu_{\mathbf{x}_t^i | D_{1:t-1}})_{i \in [n]}) + \sqrt{\alpha_t \mathbb{I}(\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1})}$.
    end if
    Query batch $\mathbf{X}_t$ to obtain $\mathbf{y}_t \triangleq (f(\boldsymbol{x}_t^i) + \epsilon)_{i \in [n]}$.
    Perform Gaussian process belief update with $(\mathbf{X}_t, \mathbf{y}_t)$.
end for
```

The code of the algorithm is in `https://github.com/YehongZ/CollaborativeBO`.

B. Proofs

B.1. Proof of Proposition 1

Instead of proving Proposition 1 directly, we prove a more general version for any positive, non-decreasing series of weights.

Lemma 1 (Monotonicity). *Let* $w_1 > w_2 > \cdots > w_n > 0$ *and* $W(\boldsymbol{u}) = \sum_{k=1}^{n} w_k \phi_k(\boldsymbol{u})$ *where* ϕ_k *returns the* k*-th smallest element of* $\boldsymbol{u}$*. For any two utility vectors* $\boldsymbol{a}$ *and* $\boldsymbol{b}$*, if there exists* $i \in [n]$ *such that* $\forall k \in [n] \setminus \{i\}$ $a^k = b^k$ *and* $b^i > a^i$*, then* $W(\boldsymbol{b}) > W(\boldsymbol{a})$*.*

Proof. Let $\boldsymbol{a}_*$ and $\boldsymbol{b}_*$ be the vectors obtained after, respectively, sorting elements of $\boldsymbol{a}$ and $\boldsymbol{b}$ in ascending order and the position of a^i and b^i in $\boldsymbol{a}_*$ and $\boldsymbol{b}_*$ be i_a and i_b, respectively, i.e., $a_*^{i_a} = a^i$ and $b_*^{i_b} = b^i$.

Given $\forall k \in [n] \setminus \{i\}$, $a^k = b^k$ and $b^i > a^i$, we must have $i_a \leq i_b$. Furthermore, (i) for $k \in [0, i_a)$ and $k \in (i_b, n]$, $a_*^k = b_*^k$ and (ii) if $i_b > i_a$, then for $k \in [i_a, i_b)$, $a_*^{k+1} = b_*^k$.

$$
\begin{aligned}
& W(\boldsymbol{b}) - W(\boldsymbol{a}) \\
&= \sum_{k=1}^{n} w_k b_*^k - \sum_{k=1}^{n} w_k a_*^k \\
&= w_{i_b} b_*^{i_b} + \sum_{k=i_a}^{i_b - 1} w_k b_*^k - \sum_{k=i_a+1}^{i_b} w_k a_*^k - w_{i_a} a_*^{i_a} \\
&= w_{i_b} b_*^{i_b} - w_{i_a} a_*^{i_a} + \sum_{k=i_a+1}^{i_b} (w_{k-1} - w_k) a_*^k \\
&\geq w_{i_b} b_*^{i_b} - w_{i_a} a_*^{i_a} + a_*^{i_a} \sum_{k=i_a+1}^{i_b} (w_{k-1} - w_k) \\
&= w_{i_b} b_*^{i_b} - w_{i_a} a_*^{i_a} + a_*^{i_a} (w_{i_a} - w_{i_b}) \\
&= w_{i_b} (b_*^{i_b} - a_*^{i_a}) \\
&= w_{i_b} (b^i - a^i) \\
&> 0 \, .
\end{aligned}
$$

The first equality is from the definition of W. The second equality is because of (i) and separating out $b_*^{i_b}$ and $a_*^{i_a}$. The third equality is due to (ii). The first inequality uses the sorted property of $\boldsymbol{a}_*$: for any $k > i_a$, $a_*^k \geq a_*^{i_a}$ and $w_{k-1} - w_k > 0$. The fourth equality is because of the telescoping series. The last inequality uses the assumption $b^i > a^i$.

□

Lemma 2 (Pigou-Dalton Principle). *Let $w_1 > w_2 > \cdots > w_n > 0$ and $W(\boldsymbol{u}) = \sum_{k=1}^n w_k \phi_k(\boldsymbol{u})$ where ϕ_k returns the k-th smallest element of $\boldsymbol{u}$. For any two utility vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, there exist $i, j \in [n]$ s.t. if (a) $\forall k \in [n] \setminus \{i, j\}$ $a^k = b^k$, (b) $a^i + a^j = b^i + b^j$, and (c) $|a^i - a^j| > |b^i - b^j|$, then $W(\boldsymbol{b}) > W(\boldsymbol{a})$.*

Proof. Let $\boldsymbol{a}_*$ and $\boldsymbol{b}_*$ be the vectors obtained after, respectively, sorting $\boldsymbol{a}$ and $\boldsymbol{b}$ and let l_a be the index of $\min(a^i, a^j)$ in $\boldsymbol{a}_*$ and h_a be the index of $\max(a^i, a^j)$ in $\boldsymbol{a}_*$. l_b and h_b are similarly defined based on $\boldsymbol{b}_*$.

We must have $l_a \leq l_b < h_b \leq h_a$. For the sake of contradiction, consider $l_a > l_b$. Because of (a) and the fact that l_a index a minimum, it would mean $\min(a^i, a^j) > \min(b^i, b^j)$. By (b), we would also have $\max(a^i, a^j) < \max(b^i, b^j)$. Hence, we would have $|a^i - a^j| < |b^i - b^j|$ which contradicts (c).

The generalized Gini utility of $\boldsymbol{a}$ and $\boldsymbol{b}$ can be decomposed as

$$\begin{aligned}
W(\boldsymbol{a}) &= \sum_{k=1}^{l_a-1} w_k a_*^k && + w_{l_a} a_*^{l_a} && + \sum_{k=l_a+1}^{h_a-1} w_k a_*^k && + w_{h_a} a_*^{h_a} && + \sum_{k=h_a+1}^{n} w_k a_*^k \\
W(\boldsymbol{b}) &= \sum_{k=1}^{l_a-1} w_k b_*^k && + \sum_{k=l_a}^{l_b-1} w_k b_*^k + w_{l_b} b_*^{l_b} && + \sum_{k=l_b+1}^{h_b-1} w_k b_*^k + w_{h_b} b_*^{h_b} && + \sum_{k=h_b+1}^{h_a} w_k b_*^k && + \sum_{k=h_a+1}^{n} w_k b_*^k \,.
\end{aligned}$$

Then,

$$\begin{aligned}
& W(\boldsymbol{b}) - W(\boldsymbol{a}) \\
&= \sum_{k=l_a}^{l_b-1} w_k b_*^k + w_{l_b} b_*^{l_b} + w_{h_b} b_*^{h_b} + \sum_{k=h_b+1}^{h_a} w_k b_*^k - \left(w_{l_a} a_*^{l_a} + \sum_{k=l_a+1}^{l_b} w_k a_*^k + \sum_{k=h_b}^{h_a-1} w_k a_*^k + w_{h_a} a_*^{h_a} \right) \\
&= \sum_{k=l_a}^{l_b-1} w_k b_*^k + w_{l_b} b_*^{l_b} + w_{h_b} b_*^{h_b} + \sum_{k=h_b+1}^{h_a} w_k b_*^k - \left(w_{l_a} a_*^{l_a} + \sum_{k=l_a+1}^{l_b} w_k \underline{b_*^{k-1}} + \sum_{k=h_b}^{h_a-1} w_k \underline{b_*^{k+1}} + w_{h_a} a_*^{h_a} \right) \\
&= \underline{\sum_{k=l_a}^{l_b-1} (w_k - w_{k+1}) b_*^k} + (w_{l_b} b_*^{l_b} - w_{l_a} a_*^{l_a}) + (w_{h_b} b_*^{h_b} - w_{h_a} a_*^{h_a}) + \underline{\sum_{k=h_b+1}^{h_a} (w_k - w_{k-1}) b_*^k} \\
&\geq \sum_{k=l_a}^{l_b-1} (w_k - w_{k+1}) \underline{a_*^{l_a}} + (w_{l_b} b_*^{l_b} - w_{l_a} a_*^{l_a}) + (w_{h_b} b_*^{h_b} - w_{h_a} a_*^{h_a}) + \sum_{k=h_b+1}^{h_a} (w_k - w_{k-1}) \underline{a_*^{h_a}} \\
&= \underline{(w_{l_a} - w_{l_b})} a_*^{l_a} + (w_{l_b} b_*^{l_b} - w_{l_a} a_*^{l_a}) + (w_{h_b} b_*^{h_b} - w_{h_a} a_*^{h_a}) + \underline{(w_{h_a} - w_{h_b})} a_*^{h_a} \\
&= -w_{l_b} a_*^{l_a} + w_{l_b} b_*^{l_b} + w_{h_b} b_*^{h_b} - w_{h_b} a_*^{h_a} \\
&= -w_{l_b} a_*^{l_a} + w_{l_b} \underline{(a_*^{l_a} + a_*^{h_a} - b_*^{h_b})} + w_{h_b} b_*^{h_b} - w_{h_b} a_*^{h_a} \\
&= (w_{l_b} - w_{h_b})(a_*^{h_a} - b_*^{h_b}) \\
&> 0
\end{aligned}$$

The first equality is because of (a). As $l_a \leq l_b < h_b \leq h_a$, we have $a_*^k = b_*^k$ for $k \in [1, l_a - 1] \cup [h_a + 1, n] \cup [l_b + 1, h_b - 1]$. Thus,

$$\sum_{k=1}^{l_a-1} w_k a_*^k = \sum_{k=1}^{l_a-1} w_k b_*^k \,, \qquad \sum_{k=h_a+1}^{n} w_k a_*^k = \sum_{k=h_a+1}^{n} w_k b_*^k \,, \qquad \text{and} \quad \sum_{k=l_b+1}^{h_b-1} w_k a_*^k = \sum_{k=l_b+1}^{h_b-1} w_k b_*^k \,.$$

The second equality is because for $k \in (l_a, l_b]$, $a_*^k = b_*^{k-1}$ (ranking decrease) and $k \in [h_b, h_a)$, $a_*^k = b_*^{k+1}$ (ranking increase). The next equality is from regrouping similar terms.

The first inequality is because $w_k - w_{k+1} > 0$ and $b_*^k = a_*^{k+1} \geq a_*^{l_a}$ for $k = l_a, \ldots, l_b - 1$, and $(w_k - w_{k-1})$ is negative and $b_*^k = a_*^{k-1} \leq a_*^{h_a}$ for $k = h_b + 1, \ldots, h_a$.

The fourth equality is due to the telescoping series and can be simplified to give the fifth equality. For the sixth equality, we use property (b) that $a_*^{l_a} + a_*^{h_a} = b_*^{h_b} + b_*^{l_b}$.

The last inequality is because $l_b < h_b$ and $a_*^{h_a} > b_*^{h_b}$, i.e., $\max(a^i, a^j) > \max(b^i, b^j)$. ☐

B.2. Detailed Analysis of (5)

Given any fixed input matrix $\mathbf{X}_t$, we claim that the first term in (5) is maximized when the party with k-th lowest λ_t^i is assigned to evaluate $\mathbf{x}$ with the k-th highest $\mu_{\mathbf{x}|D_{1:t-1}}$ due to the PDP property of the function W (see Appendix B.1). In this section, we will provide a rigorous proof. We start with the following lemma.

Lemma 3. *Let* $\boldsymbol{\lambda} \triangleq (\lambda^1, \ldots, \lambda^n)$ *and* $\mathbf{h} \triangleq (h^1, \ldots, h^i, \ldots, h^j, \ldots, h^n)$ *be two vectors with* $\lambda_1 < \lambda_2 < \ldots < \lambda_n$. $\mathbf{b} = \boldsymbol{\lambda} + \mathbf{h}$ *and* $\mathbf{a} = \boldsymbol{\lambda} + \mathbf{h}'$ *where* $\mathbf{h}' \triangleq (h^1, \ldots, h^j, \ldots, h^i, \ldots, h^n)$ *is achieved by swapping the* i*-th and* j*-th elements of* $\mathbf{h}$ *with* $i < j$*. If* $h^i > h^j$*, then* $W(\mathbf{a}) < W(\mathbf{b})$.

Proof. As W satisfies the PDP property (see Proposition 1), if $\mathbf{a}$ and $\mathbf{b}$ satisfy the PDP's preconditions (a-c), it will imply that $W(\mathbf{a}) < W(\mathbf{b})$. Here, we prove that $\mathbf{a}$ and $\mathbf{b}$ satisfy preconditions (a-c).

It is obvious that the preconditions (a-b) are satisfied since (a) $\forall k \in [n] \setminus \{i, j\}\ a^k = \lambda^k + h^k = b^k$; and (b) $a^i + a^j = \lambda^i + h^i + \lambda^j + h^j = b^i + b^j$.

Also, we have $|b^i - b^j| = |\lambda^i - \lambda^j + h^i - h^j| < |\lambda^i - \lambda^j| + |h^i - h^j| = |\lambda^j - \lambda^i + h^i - h^j| = |a^i - a^j|$. The inequality is due to $\lambda^i - \lambda^j < 0$ and $h^i - h^j > 0$. Therefore, the precondition (c) of PDP is satisfied. ☐

Next, we will use Lemma 3 to support our claim regarding how the first term in (5) is maximized.

Let $\boldsymbol{\lambda} \triangleq (\lambda^1, \ldots, \lambda^n)$ and $\mathbf{h} \triangleq (h^1, \ldots, h^n)$ be two sorted vectors with $\lambda^1 < \lambda^2 < \ldots < \lambda^n$ and $h^1 > h^2 > \ldots > h^n$. We set λ^k to be the k-th lowest λ_t^i and h^k to be the k-th highest $\mu_{\mathbf{x}|D_{1:t-1}}$. This corresponds to the "fairest" assignment where the party with the k-th lowest λ_t^i (λ^k) is assigned to evaluate $\mathbf{x}$ with the k-th highest posterior mean $\mu_{\mathbf{x}|D_{1:t-1}}$ (h^k).

Let $\mathbf{h}'$ be the vector obtained by permuting the elements of $\mathbf{h}$. $\mathbf{h}'$ corresponds to an alternative assignment where the party with the k-th lowest λ_t^i is assigned to evaluate $\mathbf{x}$ whose posterior mean is the k-th element of $\mathbf{h}'$. Additionally, let the function $\text{reindex}_{\mathbf{h}'}(i)$ returns the index of h^i in $\mathbf{h}'$.

We will show that the vector $\mathbf{h}'$ can be achieved by swapping elements in $\mathbf{h}$ such that every swap satisfy Lemma 3. *This implies that swapping the assignments of inputs to parties from the "fairest" assignment will decrease the first term of* (5). We show this by considering a sequence of intermediate vectors: $\mathbf{h} = \mathbf{h}_{(1)} \to \mathbf{h}_{(2)} \to \ldots \to \mathbf{h}_{(p)} = \mathbf{h}'$ constructed using the following steps.

(i) Let $I = \{i \in [n] \mid i = \text{reindex}_{\mathbf{h}'}(i)\}$ be an index set that contains indices where $\mathbf{h}'$ matches $\mathbf{h}$.

(ii) From $\mathbf{h}_{(1)} \to \mathbf{h}_{(2)}$: Let $\mathbf{h}_{(2)}$ be initialized to $\mathbf{h}_{(1)}$.

We identify the largest element in $\mathbf{h}_{(1)}$, located at index i_1, with $\text{reindex}_{\mathbf{h}'}(h^{i_1}) \neq i_1$ (i.e., $i_1 \notin I$). As $\mathbf{h}_{(1)}$ is sorted in descending order, $i_1 < \text{reindex}_{\mathbf{h}'}(i_1)$.

We iterate through $k = i_1 + 1, \ldots, \text{reindex}_{\mathbf{h}'}(i_1)$, if $h^{i_1} > h^k$ and $k \notin I$, swap h^{i_1} with h^k in $\mathbf{h}_{(2)}$. Since $h^{i_1} > h^k$, every swapping step satisfies the conditions of Lemma 3, and thus we have $W(\mathbf{h}_{(1)} + \boldsymbol{\lambda}) > W(\mathbf{h}_{(2)} + \boldsymbol{\lambda})$.

Moreover, the index of h^{i_1} in $\mathbf{h}_{(2)}$ will be the same as its index in $\mathbf{h}'$ after the above swapping steps. We then add new indices to I, including $\text{reindex}_{\mathbf{h}'}(i_1)$, such that it contains all indices where $\mathbf{h}'$ matches $\mathbf{h}_{(2)}$. For any $i \notin I$, the elements h^i are still in descending order.

(iii) We can obtain more intermediate vectors by repeating step (ii) above until $\mathbf{h}'$ is achieved. Note that for any $i \notin I$, the elements h^i are still in descending order. Thus, we have $i_q < \text{reindex}_{\mathbf{h}'}(i_q)$ where i_q is set to the location/index of the largest element in $\mathbf{h}_{(q)}$ with $\text{reindex}_{\mathbf{h}'}(h^{i_q}) \neq i_q$ for $q = 1, \ldots, p - 1$.

All the swapping steps are guaranteed to satisfy the conditions in Lemma 3. Hence, $W(\mathbf{h} + \boldsymbol{\lambda}) = W(\mathbf{h}_{(1)} + \boldsymbol{\lambda}) > W(\mathbf{h}_{(2)} + \boldsymbol{\lambda}) > \ldots > W(\mathbf{h}_{(p)} + \boldsymbol{\lambda}) = W(\mathbf{h}' + \boldsymbol{\lambda})$.

B.3. Proof of Theorem 1

In the following proof, we only consider the case where the domain is discrete as one can generalize the results to a continuous, compact domain via suitable discretizations (Srinivas et al., 2010).

Lemma 4. *Let $\delta \in (0,1)$ be given and $\beta_t = 2\log(|\mathcal{X}|t^2\pi^2/(6\delta))$. Then,*

$$\Pr\left(\forall \mathbf{x} \in \mathcal{X} \;\; \forall t \in \mathbb{N} \;\; \left|f(\mathbf{x}) - \mu_{\mathbf{x}|D_{1:t-1}}\right| \leq \beta_t^{1/2}\Sigma_{\mathbf{x}|D_{1:t-1}}^{1/2}\right) \geq 1-\delta\,.$$

Lemma 1 above corresponds to Lemma 5.1 in (Srinivas et al., 2010).

Lemma 5. *Let $\delta \in (0,1)$ and for all $i \in [n]$, $w_i \in (0,\infty)$ be given. Let $C_0 \triangleq 2/\log(1+\sigma^{-2})$ and $C_1 \triangleq 2C_0$ $\beta_t = 2\log(|\mathcal{X}|t^2\pi^2/(6\delta))$. Let C be a constant with $C \geq \mathbb{I}[f_{\mathbf{x}}; (y_t^{i'})_{i' \in [i]}^\top \mid D_{1:t-1}]$ for all $i \in [n-1]$, $\mathbf{x} \in \mathcal{X}$, and $t = 1,\ldots,T$, and $\alpha_t \triangleq C_1(\sum_{i=1}^n w_i^2)\exp(2C)\log(|\mathcal{X}|t^2\pi^2/(6\delta)) = C_0(\sum_{i=1}^n w_i^2)\exp(2C)\beta_t$. Then,*

$$\Pr\left(\forall \mathbf{X}_t \triangleq (\mathbf{x}_t^i)_{i\in[n]} \in \mathcal{X}^n \;\; \forall t \in \mathbb{N} \;\; \sum_{i=1}^n w_i \left|f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}}\right| \leq \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]}\right) \geq 1-\delta\,. \tag{6}$$

Proof. For all $\mathbf{X}_t \in \mathcal{X}^n$ and $t \in \mathbb{N}$,

$$\sum_{i=1}^n w_i \beta_t^{1/2}\Sigma_{\mathbf{x}_t^i|D_{1:t-1}}^{1/2} \leq \sqrt{\left(\sum_{i=1}^n w_i^2\right)\left(\sum_{i=1}^n \beta_t \Sigma_{\mathbf{x}_t^i|D_{1:t-1}}\right)} \leq \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]}$$

where the first inequality is due to the Cauchy-Schwarz inequality and the second is due to an intermediate step in Lemma 5 in (Daxberger & Low, 2017) which shows that $\sum_{i=1}^n \beta_t \Sigma_{\mathbf{x}_t^i|D_{1:t-1}} \leq C_0 \exp(2C)\beta_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]$. Then, (6) follows from that

$$\begin{aligned}
&\Pr\left(\forall \mathbf{X}_t \in \mathcal{X}^n \;\; \forall t \in \mathbb{N} \;\; \textstyle\sum_{i=1}^n w_i \left|f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}}\right| \leq \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]}\right)\\
&\geq \Pr\left(\forall \mathbf{X}_t \in \mathcal{X}^n \;\; \forall t \in \mathbb{N} \;\; \textstyle\sum_{i=1}^n w_i \left|f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}}\right| \leq \sum_{i=1}^n w_i \beta_t^{1/2}\Sigma_{\mathbf{x}_t^i|D_{1:t-1}}^{1/2}\right)\\
&\geq \Pr\left(\forall \mathbf{x} \in \mathcal{X} \;\; \forall t \in \mathbb{N} \;\; \left|f(\mathbf{x}) - \mu_{\mathbf{x}|D_{1:t-1}}\right| \leq \beta_t^{1/2}\Sigma_{\mathbf{x}|D_{1:t-1}}^{1/2}\right)\\
&\geq 1-\delta
\end{aligned}$$

where the first two inequalities are due to the property that for logical propositions A and B, $[A \Rightarrow B] \Rightarrow [\Pr(A) \leq \Pr(B)]$ and the last inequality is due to Lemma 4.

Remark 6. According to (Daxberger & Low, 2017), one can obtain a constant C *independent* of the batch size n by initializing the algorithm according to Section 4 of (Desautels et al., 2014).

Remark 7. Here, $\{w_i\}_{i\in[n]}$ can be any set of weights and ordering. The ordering need not match the G2SF weights although G2SF weights are used in Lemma 6.

□

Lemma 6. *Let $\delta \in (0,1)$ and for all $i \in [n]$, $w_i \in (0,\infty)$ be given. If*

$$\sum_{i=1}^n w_i \left|f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}}\right| \leq \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]} \tag{7}$$

for all $\mathbf{X}_t \in \mathcal{X}^n$ and $\mathbf{X}_t$ is selected using the acquisition function in (5) then

$$s_t' \triangleq W((f(\mathbf{x}^*) + \lambda_t^i)_{i\in[n]}) - W((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]}) \leq 2\sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]}\,.$$

Proof. Let $\mathbf{X}^*$ be $\mathbf{X}_t$ with $\mathbf{x}_t^i = \mathbf{x}^*$ for all $i \in [n]$ and $\phi_i(\boldsymbol{a})$ denote the i-th element of $\phi(\boldsymbol{a})$. Let $\boldsymbol{u}_*$ be the vector obtained from sorting a vector $\boldsymbol{u}$ in ascending order, $\text{rank}(\cdot)$ be the function take in a vector $\boldsymbol{u}$ and return another vector whose i-th element is the position of $\boldsymbol{u}_i$ in $\boldsymbol{u}^*$, and $[\cdot]$ denote the indexing operator. Let $\phi_{f,i}(\boldsymbol{a})$ denote the i-th element of $\phi_f(\boldsymbol{a}) \triangleq \boldsymbol{a}[\text{rank}((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]})]$ which reordering the elements in $\boldsymbol{a}$ based on the rank of $(f(\mathbf{x}_t^i) + \lambda_t^i)$. Then,

$$\begin{aligned}
s'_t &\triangleq W((f(\mathbf{x}^*) + \lambda_t^i)_{i\in[n]}) - W((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]}) \\
&= \sum_{i=1}^n w_i \phi_i\left((\lambda_t^i)_{i\in[n]}\right) + \sum_{i=1}^n w_i f(\mathbf{x}^*) - \sum_{i=1}^n w_i \phi_i\left((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]}\right) \\
&\leq \sum_{i=1}^n w_i \phi_i\left((\lambda_t^i)_{i\in[n]}\right) + \sum_{i=1}^n w_i \mu_{\mathbf{x}^*|D_{1:t-1}} + \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_{\mathbf{X}^*} \mid D_{1:t-1}]} - \sum_{i=1}^n w_i \phi_i\left((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]}\right) \\
&= \sum_{i=1}^n w_i \phi_i\left((\lambda_t^i + \mu_{\mathbf{x}^*|D_{1:t-1}})_{i\in[n]}\right) + \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_{\mathbf{X}^*} \mid D_{1:t-1}]} - \sum_{i=1}^n w_i \phi_i\left((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]}\right) \\
&\leq \sum_{i=1}^n w_i \phi_i\left((\lambda_t^i + \mu_{\mathbf{x}_t^i|D_{1:t-1}})_{i\in[n]}\right) + \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]} - \sum_{i=1}^n w_i \phi_i\left((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]}\right) \\
&\leq \sum_{i=1}^n w_i \phi_{f,i}\left((\lambda_t^i + \mu_{\mathbf{x}_t^i|D_{1:t-1}})_{i\in[n]}\right) + \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]} - \sum_{i=1}^n w_i \phi_i\left((f(\mathbf{x}_t^i) + \lambda_t^i)_{i\in[n]}\right) \\
&= \sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]} + \sum_{i=1}^n w_i \phi_{f,i}\left(\left(\mu_{\mathbf{x}_t^i|D_{1:t-1}} - f(\mathbf{x}_t^i)\right)_{i\in[n]}\right) \\
&\leq 2\sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]}\ .
\end{aligned}$$

The equalities are due to the regrouping of terms. The first inequality is from applying (7) with $\mathbf{x}_t^i = \mathbf{x}^*$ for all $i \in [n]$ and the triangle inequality: $\left|\sum_{i=1}^n w_i \left(f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}}\right)\right| \leq \sum_{i=1}^n w_i \left|f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}}\right|$. The second inequality is due to the acquisition function: $\mathbf{X}_t \triangleq \arg\max_{\mathbf{X}_t\in\mathcal{X}^n} \sum_{i=1}^n w_i\phi_i((\lambda_t^i + \mu_{\mathbf{x}_t^i|D_{1:t-1}})_{i\in[n]}) + \sqrt{\alpha_t \mathbb{I}(\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1})}$. The third inequality is due to Lemma 1 of (Weymark, 1981). The ordered weighted average utility would not be larger than the weighted average utility under any alternative ordering. The last inequality follows from (7).

□

Proof of Theorem 1 on S'_T

Let $\gamma_T \triangleq \max_{\mathbf{X}_{1:T}} \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_{1:t}]$.

$$\begin{aligned}
S'_T \triangleq \sum_{t=1}^T s'_t &\leq \sum_{t=1}^T 2\sqrt{\alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]} \\
&\leq 2\sqrt{T\sum_{t=1}^T \alpha_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]} \\
&\leq 2\sqrt{T\alpha_T\sum_{t=1}^T \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]} \\
&= 2\sqrt{T\alpha_T \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_{1:t}]} \\
&\leq 2\sqrt{T\alpha_T\gamma_T}\ .
\end{aligned}$$

The first inequality is due to Lemma 6. The second inequality is due to the Cauchy-Schwarz inequality. The third inequality is because α_t is increasing in t. The last equality follows from the chain rule of mutual information.

Convergence to Optimality

By substituting $\mathbf{x}_t^i = \mathbf{x}^*$ for all $i \in [n]$, we have $s'_t = 0$ and can show instantaneity (E2). As $\mathbf{x}^*$ is a maximizer of the objective function f, for any party i, if $\mathbf{x}_t^i \neq \mathbf{x}^*$, we must have $u_t^i \leq f(\mathbf{x}^*)$. Then by the monotonicity property (E1) (i.e., proven in Appendix B.1), s'_t must increase (or stay the same), hence s'_t lowest possible value is 0 and $s'_t = 0$ implies

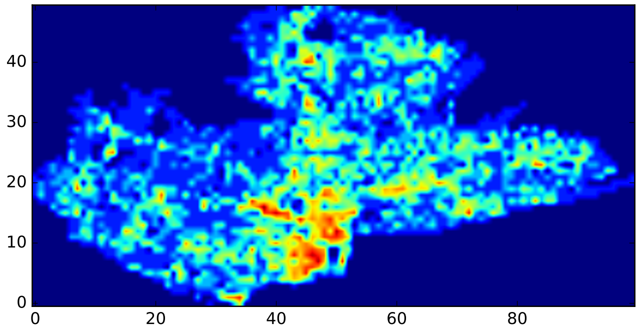


Figure 4. Visualization of the traffic demand distribution.

$\mathbf{x}_t^i = \mathbf{x}^*$ for all $i \in [n]$. Since $S'_T \le \sqrt{T\alpha_T\gamma_T}$, $\lim_{n\to\infty} \min_t s'_t \le \lim_{n\to\infty} S'_T/T = 0$, i.e. we can find an iteration t where $s'_t = 0$ and $\mathbf{x}_t^i = \mathbf{x}^*$ for all i (each party has converged to the maximum).

Remark 8. The acquisition function (1) proposed by Daxberger & Low (2017) can also be used to bound S'_T but the bound may be looser than that in Theorem 1. To be specific, we can replace the inequality (7) in Lemma 6 with:

$$\sum_{i=1}^{n} w_i \left| f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}} \right| \le \left(\max_{i\in[n]} w_i \right) \times \sum_{i=1}^{n} \left| f(\mathbf{x}_t^i) - \mu_{\mathbf{x}_t^i|D_{1:t-1}} \right| \le \sqrt{\alpha'_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]}$$

where $\alpha'_t \triangleq C_1 n(\max_{i\in[n]} w_i)^2 \exp{(2C)} \log{(|\mathcal{X}|t^2\pi^2/(6\delta))} = C_0 n(\max_{i\in[n]} w_i)^2 \exp{(2C)}\beta_t$. The ordering of parties would not matter. Then, we can have $s'_t \le 2\sqrt{\alpha'_t \mathbb{I}[\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1}]}$ and $S'_T \le 2\sqrt{T\alpha'_T\gamma_T}$ following the same steps above. Note that $\alpha'_T > \alpha_T$ as $n(\max_{i\in[n]} w_i)^2 > \sum_{i=1}^{n} w_i^2$.

C. More Experimental Details

C.1. Experimental Setups

For all experiments, we optimize and fix the squared exponential kernel hyperparameters, including the lengthscale for every dimension, in advance. Moreover, we test the performance of our algorithm using multiple c_1 and c_2 and choose c_1 and c_2 such that the CR is minimized, but the algorithm can still find a good estimate of $\mathbf{x}^*$, i.e., overexploration is avoided. More details about each experiment are given below.

Hartmann-$6d$ function. We negate the Hartmann-$6d$ function and add Gaussian noise with $\sigma = 0.1$ to the output of f. We set $c_1 = 0.08$ and $c_2 = 5$.

Hyperparameter tuning of LR with mobile sensor dataset. We set $c_1 = 0.01$ and $c_2 = 10$.

Hyperparameter tuning of CNN with FEMNIST. We set $c_1 = 0.001$ and $c_2 = 1$.

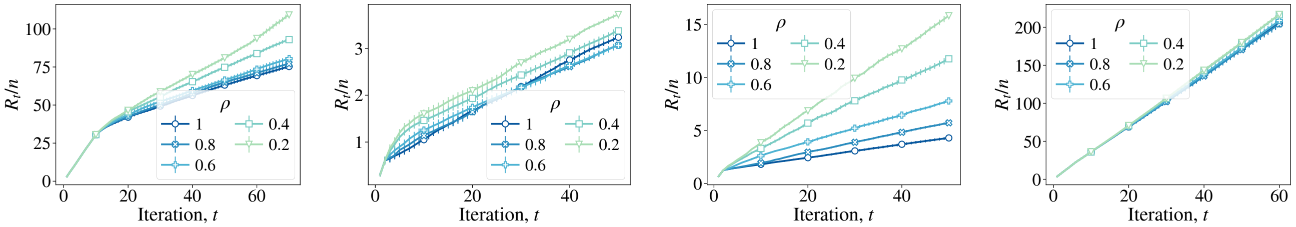
Mobility demand hotspot discovery on traffic dataset. A visualization of the demand distribution is in Fig. 4. As the demand might vary widely, we use $\log(\text{demand} + 1)$ instead. The log-demand is always at least 0.

We consider $n = 8$ parties and set $T_0 = 6$. Each party will then start at its highest non-zero demand region among these 6 explored regions. The mediator will recommend a region that is connected to its current region. If these locations are all visited (or have been evaluated to have 0 demand by others), the mediator will recommend a region that is two steps away from the current region, and so forth. We set $c_1 = 0.1$ and $c_2 = 1$.

C.2. Inefficiency: R_t/n against BO Iteration t

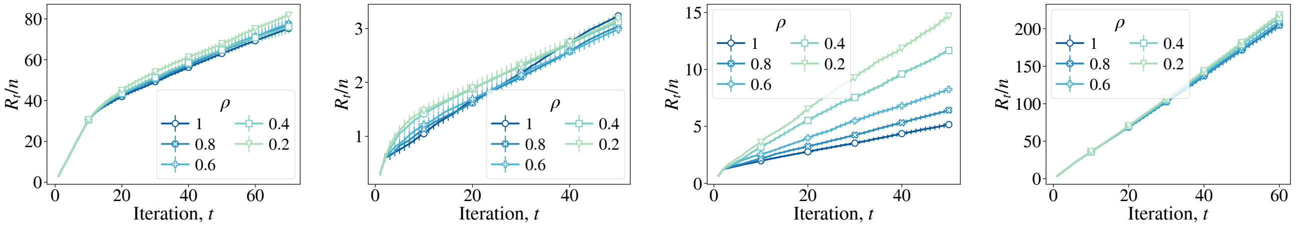
Fig. 5 and Fig. 6 show the t-step CR R_t/n vs. iteration t for fixed c_1 and varying c_1, respectively. It can be observed that for small ρ (e.g., $\rho = 0.2, 0.4$), R_t/n is larger and increases at a faster rate. This is due to the increased exploration as discussed in Sec. 4.1. We can also observe that R_t/n in Figs. 6a-b is lower than that in Figs. 5a-b as we have mitigated the effect of increased exploration relative to exploitation from setting $\rho < 1$ by adjusting c_1. This has been explained in Sec. 5.

For the traffic experiments, R_t/n is high and similar across ρ's due to the constraint that each party can only be assigned a



(a) Hartmann-6d (b) Hyp Tuning, LR (c) Hyp Tuning, CNN (d) Traffic

Figure 5. Cumulative regret R_t/n vs. iteration t with fixed c_1 and different ρ's in (5) for various objective functions.



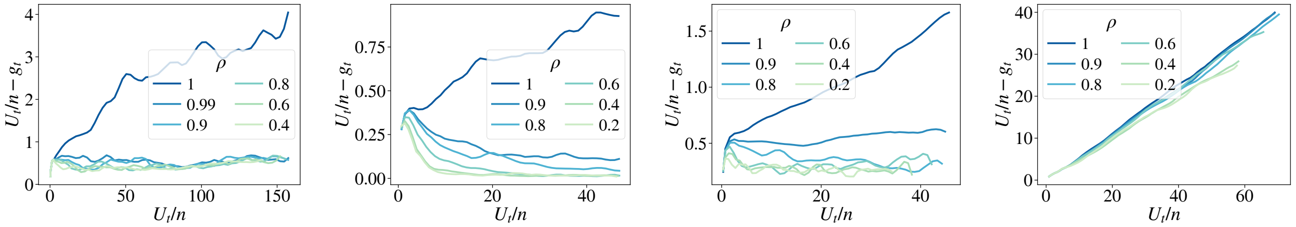
(a) Hartmann-6d (b) Hyp Tuning, LR (c) Hyp Tuning, CNN (d) Traffic

Figure 6. Cumulative regret R_t/n vs. iteration t with varying c_1 and different ρ's in (5) for various objective functions.

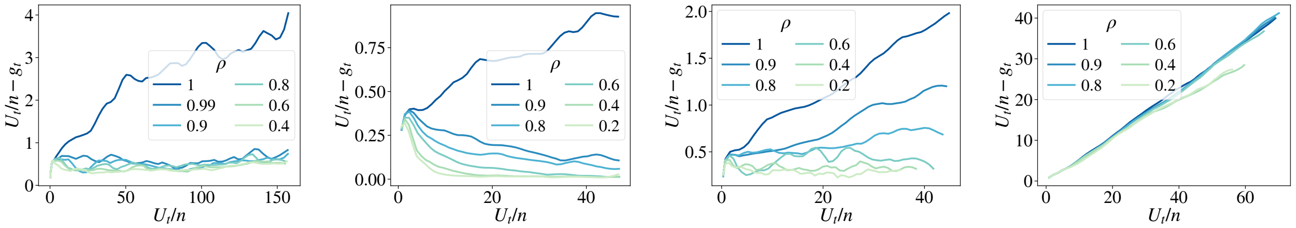
connected input region. The inefficiency can be reduced if a non-myopic approach is used or the constraint is relaxed.

C.3. More Observations about Fairness

We plot $U_t/n - g_t$ vs. U_t/n for fixed c_1 and varying c_1, respectively, in Figs. 7a-d and 7e-h.[11] As ρ decreases, $U_t/n - g_t$ generally decreases in all the graphs of Fig. 7. This trend holds for various choices of fixed c_1 and even for varying c_1 which has a lower R_t/n. For Hartmann-6d and the hyperparameter tuning experiments, we note that the trend is weaker when $\rho < 0.5$. This might be because of more exploration/noise. However, for the traffic experiment, the difference between $U_t/n - g_t$ for different ρ is only significant when $\rho \leq 0.4$. This may be due to the difficulty in ensuring fairness subjected to the constraint that each party can only be assigned to evaluate connected input regions.



(a) Hartmann-6d (b) Hyp Tuning, LR (c) Hyp Tuning, CNN (d) Traffic

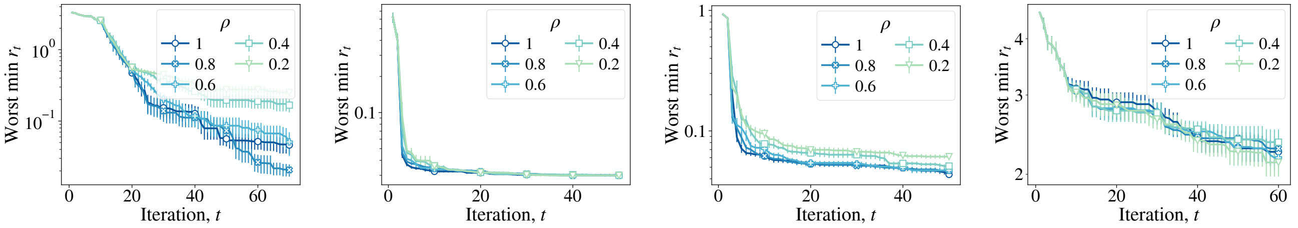


(e) Hartmann-6d (f) Hyp Tuning, LR (g) Hyp Tuning, CNN (h) Traffic

Figure 7. Unfairness $U_t/n - g_t$ vs. U_t/n with (a-d) fixed c_1 and (e-h) varying c_1 but different ρ's in (5) for various objective functions.

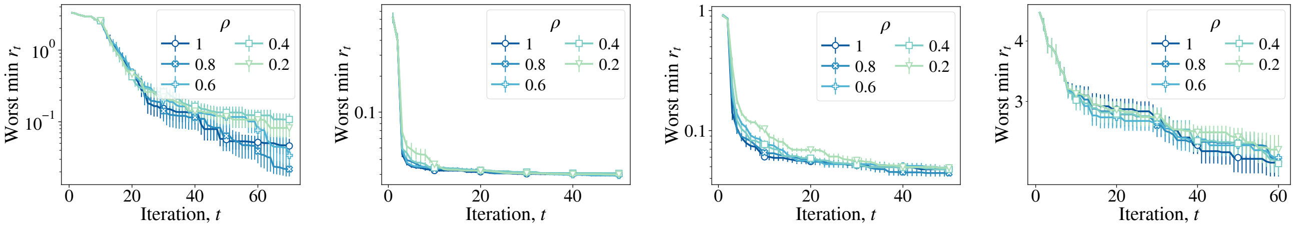
---

[11] We plot against U_t/n instead of t to ensure that larger unfairness $U_t/n - g_t$ is not due to larger U_t/n. Note that in all experiments, each party's t-CU is non-decreasing with increasing t as $f(\mathbf{x})$ is always non-negative.



(a) Hartmann-6d (b) Hyp Tuning, LR (c) Hyp Tuning, CNN (d) Traffic

Figure 8. Worst simple regret across parties with fixed c_1 and different ρ's in (5) for various objective functions.



(a) Hartmann-6d (b) Hyp Tuning, LR (c) Hyp Tuning, CNN (d) Traffic

Figure 9. Worst simple regret across parties with varying c_1 and different ρ in (5) for various objective functions.

C.4. Inefficiency: Simple Regret

In this section, we show that our algorithm with different ρ's would allow each party to find an input $\mathbf{x}$ such that $f(\mathbf{x})$ is close to $f(\mathbf{x}^*)$. We define the *simple regret* (SR) of a party i as $\min_{t'=1,\ldots,t}(f(\mathbf{x}^*) - f(\mathbf{x}^i_{t'}))$. In Figs. 8 and 9, we plot the worst (i.e., maximum) SR across parties, averaged across 10 BO runs. We observe that the worst SR is larger when ρ is smaller. This is expected, as with a smaller ρ, the relative total weight on the exploitation term in (5) is smaller. Thus, the mediator is less likely to assign any party to sample near the likely maximizer in the earlier iterations.

C.5. Individual Rationality

It is *individually rational* for a party to participate in a BO collaboration if it can obtain a better estimate of $\mathbf{x}^*$ for the same CR or alternatively, lower CR for the same estimate of $\mathbf{x}^*$ than performing conventional GP-UCB (Srinivas et al., 2010)) on its own. Note that we should not compare the CR alone since (a) a low CR can be achieved by choosing a small α_t but it would lead to a poor estimate of $\mathbf{x}^*$ and (b) such a comparison is more sensitive to the hyperparameters chosen. The CR or SR of GP-UCB varies largely between runs.

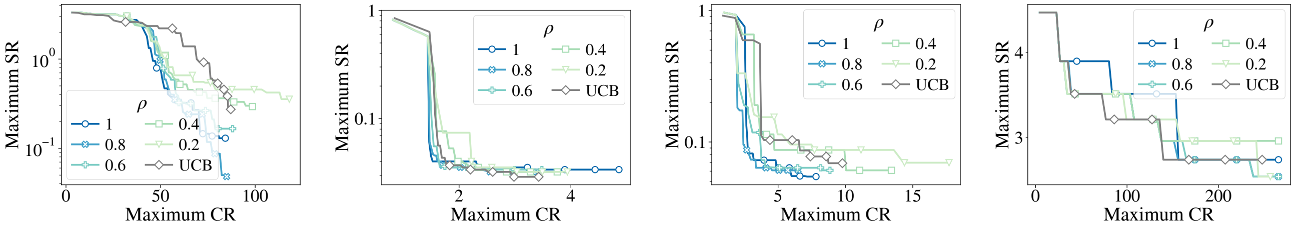
Let $r = 1, \ldots, 10$ denote the BO runs. We compare the performance of our collaborative BO algorithm (for different ρ's) against that of GP-UCB (i) in the worst case and plot $\max_{r\in[10]} \max_{i\in[n]} \min_{t'\in[t]} r^i_{t'}$ (maximum individual t-step SR) against $\max_{r\in[10]} \max_{i\in[n]} \sum_{t'\in[t]} r^i_{t'}$ (maximum individual t-step CR) for various step t; and (ii) in the median case and plot $\text{median}_{r\in[10]} \text{median}_{i\in[n]} \min_{t'\in[t]} r^i_{t'}$ (median individual t-step SR) against $\text{median}_{r\in[10]} \text{median}_{i\in[n]} \sum_{t'\in[t]} r^i_{t'}$ (median individual t-step CR) for various step t.

Fig. 10 shows the results of (i). It can be observed that our collaborative BO algorithm with ρ close to 1 allow all parties, including the worst-off party, to achieve a smaller maximum SR for the same maximum CR than GP-UCB and other settings of ρ. However, due to randomness, GP-UCB might sometimes lead to lower SR than our collaborative BO algorithm.[12]

Next, we consider non-worst-case scenarios and show the results of (ii) in Fig. 11. We chose the median over the mean as it would not be distorted by outliers, e.g., a run where a party has a very large regret. Again, we observe that our algorithm achieves a lower median SR for the same median CR than GP-UCB.

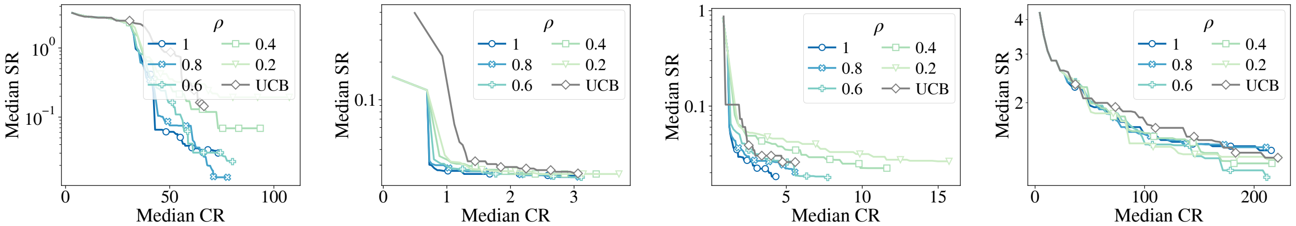
---

[12] Another possible reason is that $\sqrt{\alpha_t \mathbb{I}(\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1})}$ in (5) encourages diversity among inputs selected for all parties and might prevent parties from all evaluating close to $\mathbf{x}^*$ in one BO iteration.



(a) Hartmann-6d (b) Hyp Tuning, LR (c) Hyp Tuning, CNN (d) Traffic

Figure 10. Maximum SR vs. maximum CR with varying c_1 and different ρ for various objective functions.



(a) Hartmann-6d (b) Hyp Tuning, LR (c) Hyp Tuning, CNN (d) Traffic

Figure 11. Median SR vs. median CR with varying c_1 and different ρ for various objective functions.

C.6. Ignoring the Values of λ_t^i

In Sec. 4, we suggest that considering the value of λ_t^i in (5) is essential to achieve fairness in the cumulative sense. In this section, we will analyze the importance of the *values* of $(\lambda_t^i)_{i\in[n]}$ in (5) by comparing our collaborative BO algorithm with an alternative algorithm, *BO with instantaneously fair utility* (BO-IFU), with the following acquisition function:

$$\mathbf{X}_t \triangleq \underset{\mathbf{X}_t \in \mathcal{X}^n}{\arg\max} \sum_{i=1}^{n} w_i \phi_\lambda((\mu_{\mathbf{x}_t^i|D_{1:t-1}})_{i\in[n]}) + \sqrt{\alpha_t \mathbb{I}(\mathbf{f}_{\mathcal{X}}; \mathbf{y}_t \mid D_{1:t-1})}$$

where $\phi_\lambda(\boldsymbol{a}) = \boldsymbol{a}[\text{rank}((\lambda_t^i)_{i=1\ldots n})]$. BO-IFU would assign the selected input with k-th largest posterior mean to the party with the k-th lowest λ_t^i. We also enforce this condition when $\rho = 1$. However, as λ_t^i for all $i \in [n]$ do not appear directly in the objective, it no longer matters if the $(\lambda_t^i)_{i\in[n]}^\top$ values are $(0.1, 0.2, 0.3)$ or $(1, 2, 3)$.

Fig. 12 shows the fairness result of the tested algorithms for the Hartmann-6d function with fixed c_1. It can be observed that a smaller ρ would cause BO-IFU to achieve much larger $U_t/n - g_t$. Even though a smaller ρ makes BO-IFU to select $\mathbf{X}_t$ with more similar/fair $\mu_{\mathbf{x}_t^i|D_{1:t-1}}$ values, this fairness in the individual *instantaneous* utility at each BO iteration does not translate to fairness in the individual t-step *cumulative* utility. BO-IFU is the fairest when its ρ is close to 1. However, this fairest solution still leads to more unfairness in the cumulative sense than our algorithm for $\rho < 1$.

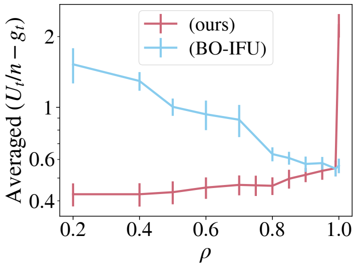
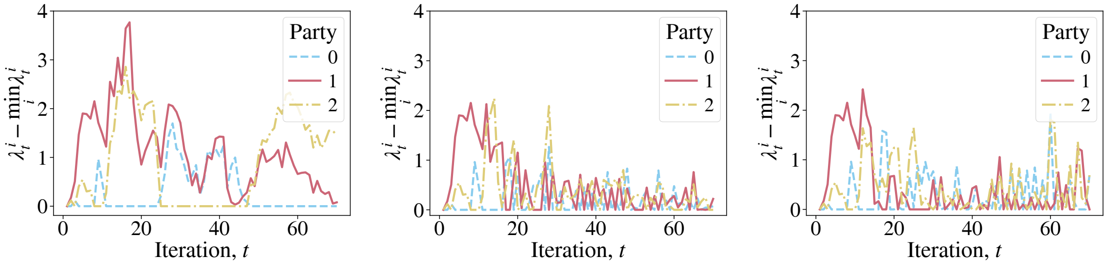


Figure 12. Averaged $(U_t/n - g_t)$ vs. ρ for our algorithm and BO-IFU tested on the Hartmann-6d function with fixed c_1.

C.7. Analysis of λ_t^i across Parties

Recall that the individual $(t-1)$-CU can be measured by its realized/observed variant $\lambda_t^i \triangleq \sum_{t'=1}^{t-1} y_{t'}^i$. In this section, we analyze the behavior of λ_t^i using one BO run of the Hartmann-6d function. Other objective functions and runs would give similar results.

Fig. 13 shows the difference between λ_t^i and $\min_{i\in[n]} \lambda_t^i$ of each party for our algorithm and BO-IFU. A larger difference $\lambda_t^i - \min_{i\in[n]} \lambda_t^i$ (y-value) implies more unfairness. We observe that the line/party with the largest (or zero) $\lambda_t^i - \min_{i\in[n]} \lambda_t^i$ (y-axis) alternates across t. This shows that the 3 parties take turn to have the largest and lowest individual $(t-1)$-CU. Generally, the y-values of BO-IFU (Fig. 13a) are larger than those of our algorithm (Figs. 13b-c). Also, for BO-IFU, the line/party with the largest (or zero) $\lambda_t^i - \min_{i\in[n]} \lambda_t^i$ alternates less frequently, i.e., a party may unfairly have a larger (or lower) utility than others over many iterations. Once again, we can observe that using a smaller ρ in our collaborative BO algorithm will reduce the differences in individual $(t-1)$-CU between parties by comparing the y-values in Fig. 13c and Fig. 13b. Note that the first 10 iterations have larger y-values as they are a result of random exploration ($T_0 = 10$).



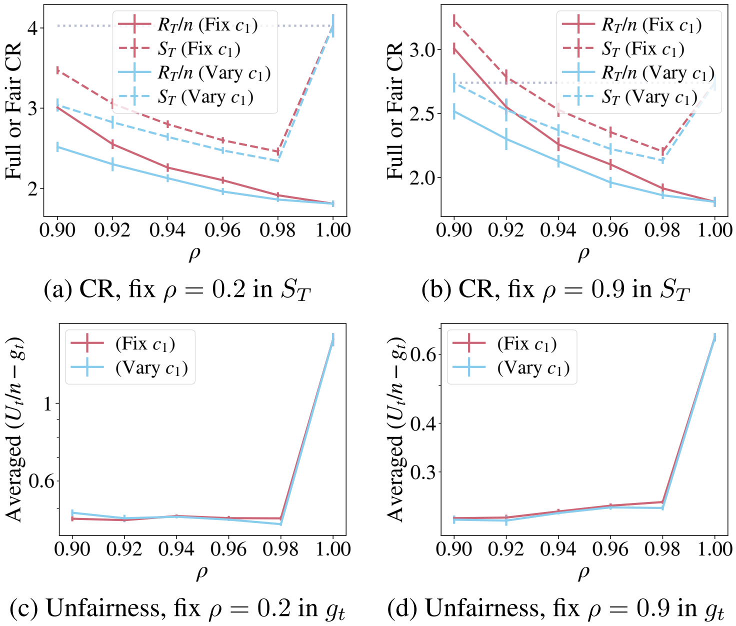
(a) $\rho = 0.5$ (BO-IFU) (b) $\rho = 0.5$ (ours) (c) $\rho = 0.8$ (ours)

Figure 13. The difference in λ_t^i across parties for (a) BO-IFU and (b-c) our collaborative BO algorithm with different ρ's for the Hartmann-6d function with $\sigma = 0.1$.

C.8. Experimental Results with More Parties

As it is harder to jointly optimize the acquisition function over large n, existing batch BO works only considered up to $n = 50$. In future work, we will consider using the Markov approximation of DB-GP-UCB for better scalability in n.

We have extended our CNN hyperparameter tuning experiment to $n = 50$ parties and plot the results in Fig. 14. As $n = 50$ is large and the weights $w_i = \rho^{i-1}$ decrease exponentially, we only consider setting ρ in (5) to values ≥ 0.9. Fig. 14 shows similar trends for R_T/n, S_T and $(U_t/n - g_t)$ with decreasing ρ as described in Sec. 5. The fairness advantage of smaller ρ is more visible when $\rho = 0.9$ in g_t.



(a) CR, fix $\rho = 0.2$ in S_T (b) CR, fix $\rho = 0.9$ in S_T

(c) Unfairness, fix $\rho = 0.2$ in g_t (d) Unfairness, fix $\rho = 0.9$ in g_t

Figure 14. Graphs of (a-b) full CR R_T/n and normalized fair CR S_T averaged over 10 BO runs and (c-d) averaged $U_t/n - g_t$, incurred by the tested algorithms using different ρ's in (5) for hyperparameter tuning of CNN with $n = 50$ parties. We set $c_1 = 0.01$.