A Extensions

A.1 Knapsack constraint

Consider a knapsack constraint $\mathcal{K} = \{S \subseteq [n] : \sum_{e \in S} c_e \leq 1\}$, where $c_e > 0$ for all $e \in [n]$. Our interest is to solve the following robust problem

$$\max_{S \in \mathcal{K}} \min_{i \in [k]} f_i(S) \tag{9}$$

Corollary 1. *For Problem* (9)*, there is a polynomial time algorithm that returns a set* S^{ALG}*, such that for all* $i \in [k]$*, for a given* $0 < \epsilon < 1$*,*

$$f_i(S^{\mathrm{ALG}}) \geq (1-\epsilon) \cdot \max_{S \in \mathcal{K}} \min_{j \in [k]} f_j(S),$$

and $\sum_{e \in S^{\mathrm{ALG}}} c_e \leq \ell$ *for* $\ell = O(\ln \frac{k}{\epsilon})$*. Moreover,* S^{ALG} *can be covered by at most* ℓ *sets in* $\mathcal{K}$*.*

Instead of using the standard greedy for every $\tau = \{1, \ldots, \ell\}$, we design an extended version of the "bang-per-buck" greedy algorithm. We formalize this procedure in Algorithm 3 below. Even though the standard "bang-per-buck" greedy algorithm does not provide any approximation factor, if we relax the knapsack constraint to be $\sum_{e \in S} c_e \leq 2$, then the algorithm gives a $1 - 1/e$ factor. There are other approaches to avoid this relaxation, see e.g. (Sviridenko, 2004).

Algorithm 3 Extended "Bang-per-Buck" Algorithm for Knapsack Constraints

Input: $\ell \geq 1$, monotone submodular function $g : 2^V \to \mathbb{R}_+$, knapsack constraint $\mathcal{K}$.
Output: sets $S_1, \ldots, S_\ell \in \mathcal{K}$.

for $\tau = 1, \ldots, \ell$ **do**
  $S_\tau \leftarrow \emptyset$
  while $V \neq \emptyset$ **do**
    Compute

$$e^* = \mathrm{argmax}_{e \in V} \frac{g(\cup_{j=1}^{\tau} S_j + e) - g(\cup_{j=1}^{\tau} S_j)}{c_e}.$$

    if $\sum_{e \in S^\tau} c_e + c_{e^*} \leq 2$ **then**
      $S_\tau \leftarrow S_\tau + e^*$.
    $V \leftarrow V - e^*$
  Restart ground set V.

Given a monotone submodular function $g : 2^V \to \mathbb{R}_+$, Algorithm 3 produces a set $S^{\mathrm{ALG}} = S_1 \cup \cdots \cup S_\ell$ such that $g(S^{\mathrm{ALG}}) \geq \left(1 - \frac{1}{e^\ell}\right) \cdot \max_{S \in \mathcal{K}} g(S)$. Therefore, Corollary 1 can be easily proved by defining g in the same way as in Theorem 1, and running Algorithm 3 on g with $\ell = O(\ln \frac{k}{\epsilon})$.

A.2 Multiple matroid constraints

Consider a family of r matroids $\mathcal{M}_j = (V, \mathcal{I}_j)$ for $j \in [r]$. Our interest is to solve the following robust problem

$$\max_{S \in \bigcap_{j=1}^r \mathcal{I}_j} \min_{i \in [k]} f_i(S) \tag{10}$$

Corollary 2. *For Problem* (10)*, there is a polynomial time algorithm that returns a set* S^{ALG}*, such that for all* $i \in [k]$*, for a given* $0 < \epsilon < 1$*,*

$$f_i(S^{\mathrm{ALG}}) \geq (1-\epsilon) \cdot \max_{S \in \bigcap_{j=1}^r \mathcal{I}_j} \min_{i \in [k]} f_i(S),$$

where S^{ALG} *is the union of* $O(\log \frac{k}{\epsilon} / \log \frac{r+1}{r})$ *independent sets in* $\mathcal{I}$*.*

Fisher et al. (1978) proved that the standard greedy algorithm gives a $1/(1+r)$ approximation for problem (10) when $k = 1$. Therefore, we can adapt Algorithm 1 to produce a set $S^{\mathrm{ALG}} = S_1 \cup \cdots \cup S_\ell$ such that

$$f(S^{\mathrm{ALG}}) \geq \left(1 - \left(\frac{r}{r+1}\right)^\ell\right) \cdot \max_{S \in \bigcap_{j=1}^r \mathcal{I}_j} f(S).$$

Then, Corollary 2 can be proved similarly to Theorem 1 by choosing $\ell = O(\log \frac{k}{\epsilon} / \log \frac{r+1}{r})$

A.3 Distributionally robust over polyhedral sets

Let $\mathcal{Q} \subseteq \Delta(k)$ be a polyhedral set, where $\Delta(k)$ is the probability simplex on k elements. For $q \in \mathcal{Q}$, denote $f_q := q_1 f_1 + \cdots + q_k f_k$, which is also monotone and submodular. Given a matroid $\mathcal{M} = (V, \mathcal{I})$, our interest is to solve the following distributionally robust problem

$$\max_{S \in \mathcal{I}} \min_{q \in \mathcal{Q}} f_q(S) \tag{11}$$

Denote by $\mathrm{Vert}(\mathcal{Q})$ the set of extreme points of $\mathcal{Q}$, which is finite since $\mathcal{Q}$ is polyhedral. Then, problem (11) is equivalent to $\max_{S \in \mathcal{I}} \min_{q \in \mathrm{Vert}(\mathcal{Q})} f_q(S)$. Then, we can easily derive Corollary 3 (below) by applying Theorem 1 in the equivalent problem. Note that when $\mathcal{Q}$ is the simplex we get the original Theorem 1.

Corollary 3. *For Problem* (11)*, there is a polynomial time algorithm that returns a set* S^{ALG}*, such that for all* $i \in [k]$*, for a given* $0 < \epsilon < 1$*,*

$$f_i(S^{\mathrm{ALG}}) \geq (1-\epsilon) \cdot \max_{S \in \mathcal{I}} \min_{q \in \mathcal{Q}} f_q(S),$$

with $S^{\mathrm{ALG}} = S_1 \cup \cdots \cup S_\ell$ *for* $\ell = O(\log \frac{|Vert(\mathcal{Q})|}{\epsilon})$ *and* $S_1, \ldots, S_\ell \in \mathcal{I}$*.*

B Continuous Offline Algorithm

B.1 Preliminaries of Multilinear Extension

Consider a set function f, and its multilinear extension F as in equation (4). We can state the following facts. For more details see (Calinescu et al., 2011).

Fact 1. *[Multilinear Extensions of Monotone Submodular Functions] Let f be a monotone submodular function and F its multilinear extension.*

1. *By monotonicity of f, we have $\frac{\partial F}{\partial y_e} \geq 0$ for any $e \in V$. This implies that for any $x \leq y$ coordinate-wise, $F(x) \leq F(y)$. On the other hand, by submodularity of f, F is concave in any positive direction, i.e., for any $e_1, e_2 \in V$ we have $\frac{\partial^2 F}{\partial y_{e_1} \partial y_{e_2}} \leq 0$.*

2. *Throughout the paper we will denote by $\nabla_e F(y) := \frac{\partial F(y)}{\partial y_e}$, and $\Delta_e F(y) := \mathbb{E}_{S\sim y}[f_S(e)]$. It is easy to see that $\Delta_e F(y) = (1-y_e)\nabla_e F(y)$. Now, consider two points $x, y \in [0,1]^V$ and two sets sampled independently from these vectors: $S \sim x$ and $U \sim y$. Then, by submodularity*

$$f(S \cup U) \leq f(S) + \sum_{e\in V} \mathbf{1}_U(e) f_S(e). \tag{12}$$

3. *By taking expectation over x and y in (12), we obtain*

$$\begin{aligned} F(x \vee y) &\leq F(x) + \sum_{e\in V} y_e \Delta_e F(x) \\ &\leq F(x) + \sum_{e\in V} y_e \nabla_e F(x). \end{aligned}$$

   Therefore, we get the following important property

$$F(x \vee y) \leq F(x) + y \cdot \nabla F(x). \tag{13}$$

B.2 Algorithm analysis

In this section, we present a continuous algorithm that achieves a tight bi-criteria approximation for the robust submodular optimization problem (1) and prove Theorem 6. This algorithm achieves optimal constants for the approximation, matching the hardness result in (Krause et al., 2008a).

Theorem 6. *Let $(V, \mathcal{I})$ be a matroid and let $f_i : 2^V \to \mathbb{R}_+$ be a monotone submodular function for $i \in [k]$. Then, there is a randomized polynomial time algorithm that with constant probability returns a set S^{ALG}, such that for all $i \in [k]$, for a given $0 < \epsilon < 1$,*

$$f_i(S^{\text{ALG}}) \geq (1-\epsilon) \cdot \max_{S\in\mathcal{I}} \min_{j\in[k]} f_j(S),$$

and $S^{\text{ALG}} = S_1 \cup \cdots \cup S_\ell$ for $\ell = O(\ln \frac{k}{\epsilon})$, and $S_1, \ldots, S_\ell \in \mathcal{I}$.

Our overall approach is to first find a fractional solution with a desirable approximation guarantee and then round it to an integral solution. We use a relaxation of a matroid to its convex hull to accommodate the search for a fractional solution.

For this algorithm, we need an estimate γ of the value of the optimal solution which we denote by OPT. We prove the following lemma which solves an approximate decision version of our optimization problem. The proof of Theorem 6 follows from the lemma and a search over an approximate value for OPT.

Lemma 1. *There is a randomized polynomial time algorithm that given $\gamma \leq \text{OPT}$ and $0 < \epsilon < 1$ returns with constant probability a set S^{ALG} such that for all $i \in [k]$,*

$$f_i(S^{\text{ALG}}) \geq (1-\epsilon) \cdot \gamma,$$

where $S^{\text{ALG}} = \bigcup_{j\in[\ell]} S_j$ with $\ell = O(\ln \frac{k}{\epsilon})$ and $S_j \in \mathcal{I}$ for each $j \in [\ell]$.

First, we finish the proof of Theorem 6 assuming Lemma 1.

Theorem 6. We apply the algorithm from Lemma 1 with approximation loss $\epsilon/2$ and with different values of γ, some of which may be larger than OPT, but at least one of them is guaranteed to satisfy $(1-\epsilon/2)\,\text{OPT} \leq \gamma \leq \text{OPT}$. At the end we return the set S^{ALG} from our runs with the highest value of $\min_{i\in[k]} f_i(S^{\text{ALG}})$.

Before describing the set of candidate values of γ that we try, note that if the algorithm succeeds for the particular value of γ satisfying $(1-\epsilon/2)\,\text{OPT} \leq \gamma \leq \text{OPT}$, then we get

$$\min_{i\in[k]} f_i(S^{\text{ALG}}) \geq (1-\epsilon/2)\cdot\gamma \geq (1-\epsilon)\,\text{OPT},$$

and since we return the set with the highest $\min_{i\in[k]} f_i(S^{\text{ALG}})$, the algorithm's output will have the desired approximation guarantee.

It remains to show that a set of polynomial size of values for γ exists such that one of them satisfies $(1-\epsilon/2)\,\text{OPT} \leq \gamma \leq \text{OPT}$. To this end we simply try $\gamma = n f_i(e)(1-\epsilon/2)^j$ for all $i \in [k]$, $e \in V$, and $j = 0, \ldots, \lceil \ln_{1-\epsilon/2}(1/n) \rceil$. Note that there exists an index $i^* \in [k]$ and a set $S^* \in \mathcal{I}$ such that $\text{OPT} = f_{i^*}(S^*)$. Now let $e^* = \text{argmax}_{e\in S^*} f_{i^*}(e)$. Because of submodularity and monotonicity we have $\frac{1}{|S^*|} f_{i^*}(S^*) \leq f_{i^*}(e^*) \leq f_{i^*}(S^*)$. So, we can conclude that $1 \geq \text{OPT}/n f_{i^*}(e^*) \geq 1/n$, which implies that $j = \lceil \ln_{1-\epsilon/2}(\text{OPT}/n f_{i^*}(e^*)) \rceil$ is in the correct interval, obtaining

$$(1-\epsilon/2)\,\text{OPT} \leq n f_{i^*}(e^*)(1-\epsilon/2)^j \leq \text{OPT}.$$

This finishes the proof. $\square$

We remark that the dependency of the running time on ϵ can be made logarithmic by running a binary search on j as opposed to trying all $j = 0, \ldots, \lceil \ln_{1-\epsilon/2}(1/n) \rceil$. We just need to run the algorithm from Lemma 1 for each γ polynomially many times to make the failure probability exponentially small whenever $\gamma \leq \mathrm{OPT}$.

The rest of this section is devoted to prove Lemma 1. To achieve a strong concentration bound when rounding the fractional solution, we truncate f_i to $\min\{\gamma, f_i\}$. Hereafter, we use f_i^γ to refer to $\min\{\gamma, f_i\}$. Note that submodularity and monotonicity is preserved under this truncation. Also, we denote by F_i^γ the corresponding multilinear extension of f_i^γ.

We describe the continuous process counterpart of the algorithm in this section. The discretization details follow using standard methods (Vondrák, 2008).

Continuous Greedy. We start a continuous gradient step process where $y(\tau)$ represents the point at time τ we are at. We start at $y(0) = 0$ and take continuous gradient steps in direction $\frac{dy}{d\tau} = v_{\mathrm{all}}(y)$, such that $v_{\mathrm{all}}(y)$ satisfies the following conditions:

(a) $v_{\mathrm{all}}(y) \cdot \nabla F_i^\gamma(y) \geq \gamma - F_i^\gamma(y)$ for all $i \in [k]$,

(b) $v_{\mathrm{all}}(y) \in \mathcal{P}(\mathcal{M})$, and

(c) $v_{\mathrm{all}}(y) + y \in [0,1]^V$.

First, we show that such v_{all} always exists. Take x^* to be the indicator vector corresponding to the optimal solution. For any y, $v^* = (x^* - y) \vee 0$ is a positive direction satisfying inequality (13), and for all $i \in [k]$:

$$v^* \cdot \nabla F_i^\gamma(y) \geq F_i^\gamma(y + v^*) - F_i^\gamma(y) = \gamma - F_i^\gamma(y), \quad (14)$$

where the last equality holds since $F_i^\gamma(y) \leq \gamma$ for all y. It is easy to check that v^* satisfies the rest of the constraints (a)-(c), implying that there exists a feasible solution to the above system of linear inequalities. Therefore, we can solve a linear program defined by these inequalities to obtain a solution $v_{\mathrm{all}}(y)$.

The above continuous process goes on until time $\ell = O(\ln \frac{k}{\epsilon})$. We intentionally set $\ell > 1$ to obtain a (fractional) solution with a higher budget, which is useful for achieving a bi-criteria approximation. Next we show the following claim.

Claim 1. *For any $\tau \geq 0$, $y(\tau) \in \tau\mathcal{P}(\mathcal{M}) \cap [0,1]^V$ and for all $i \in [k]$,*

$$F_i^\gamma(y(\tau)) \geq (1 - e^{-\tau})\gamma.$$

Proof. For any $\tau \geq 0$, we have

$$y(\tau) = \int_0^\tau v_{\mathrm{all}}(y(s))\ ds = \int_0^1 \tau \cdot v_{\mathrm{all}}(y(\tau s))\ ds.$$

So, $y(\tau)$ is a convex combination of vectors in $\tau\mathcal{P}(\mathcal{M})$. Moreover, $(v_{\mathrm{all}}(y))_j = 0$ when $y_j = 1$, thus $y(\tau) \in [0,1]^V$ proving the first part of the claim.

For the second part, observe that for all $i \in [k]$ we have

$$\begin{aligned}\frac{dF_i^\gamma(y(\tau))}{d\tau} &= \frac{dy(\tau)}{d\tau} \cdot \nabla F_i^\gamma(y(\tau)) \\ &= v_{\mathrm{all}}(y(\tau)) \cdot \nabla F_i^\gamma(y(\tau)) \geq \gamma - F_i^\gamma(y(\tau)).\end{aligned}$$

Moreover, $F_i^\gamma(0) = 0$. Now we solve the above differential equation to obtain

$$F_i^\gamma(y(\tau)) \geq (1 - e^{-\tau})\gamma$$

for each $i \in [k]$ as claimed. □

Thus, by setting $\ell = \ln \frac{k}{\epsilon} + \ln \frac{1}{c}$, we obtain $F_i^\gamma(y(\ell)) \geq \left(1 - \frac{\epsilon}{k} \cdot c\right) \cdot \gamma$ for all $i \in [k]$ and a desired constant $c < 1$. We next show how to obtain an integral solution.

Rounding. The next lemma summarizes our rounding. We first show that the fractional solution at time ℓ is contained in the matroid polytope of the ℓ-fold union of matroid $\mathcal{M}$. We then do randomized swap rounding, introduced by Chekuri et al. (2010), in this matroid polytope. The truncation of the submodular functions, as well as properties of randomized swap rounding, play a crucial role in the proof.

Lemma 2. *Let $\ell = \lceil \ln \frac{k}{\epsilon} + \ln \frac{1}{c} \rceil$ be an integer and $y(\ell)$ be the output of the continuous greedy algorithm at time ℓ such that $F_i^\gamma(y(\ell)) \geq \left(1 - \frac{\epsilon}{k} \cdot c\right) \cdot \gamma$ for each $i \in [k]$ and some constant $c < 1$. Then, there exists a polynomial time randomized algorithm that outputs a set S such that with probability $\Omega(1)$, for each $i \in [k]$ we have*

$$f_i(S) \geq (1 - \epsilon) \cdot \gamma.$$

Moreover, S is a union of at most ℓ independent sets in $\mathcal{M}$.

Proof. Let $\mathcal{M}_\ell = \bigvee_\ell \mathcal{M}$ be the ℓ-fold union of matroid $\mathcal{M}$, i.e., I is an independent set in $\mathcal{M}_\ell$ if and only if I is a union of ℓ independent sets of $\mathcal{M}$. We denote by $\mathcal{I}_\ell$ the set of independent sets of $\mathcal{M}_\ell$. The rank function of $\mathcal{M}_\ell$ is given by $r_{\mathcal{M}_\ell}(S) = \min_{A \subseteq S} |S \setminus A| + \ell \cdot r_{\mathcal{M}}(A)$ (see (Schrijver, 2003)). We first show that $y = y(\ell)$ is in the convex hull of independent sets of matroid $\mathcal{M}_\ell$, i.e., $\mathcal{P}(\mathcal{M}_\ell)$. This polytope is given by $\mathcal{P}(\mathcal{M}_\ell) = \{x \in \mathbb{R}_+^V \mid x(S) \leq r_{\mathcal{M}_\ell}(S) \quad \forall\ S \subseteq V\}$, where $x(S) = \sum_{e \in S} x_e$. We now prove that $y \in \mathcal{P}(\mathcal{M}_\ell)$. For any $S \subseteq V$ and $A \subseteq S$, we have $y(S) = \sum_{e \in S \setminus A} y_e + y(A) \leq |S \setminus A| + \ell \cdot r_{\mathcal{M}}(A)$, where the last inequality is due to the fact that $y_e \leq 1$ for all e, and $y(A) \leq \ell \cdot r_{\mathcal{M}}(A)$ because $y \in \ell \cdot \mathcal{P}(\mathcal{M})$ by Claim 1. Therefore, $y \in \mathcal{P}(\mathcal{M}_\ell)$.

Next, we apply a randomized swap rounding (see (Chekuri et al., 2010)) in matroid $\mathcal{M}_\ell$ to obtain the solution. A feature of the randomized swap rounding is that it is oblivious to the specific function f_i used, and it is only a randomized function of the matroid space and the fractional solution.

Applying Theorem 5 to fractional solution $y(\ell)$ and matroid $\mathcal{M}_\ell$, we obtain a random set $S \in \mathcal{I}_\ell$ such that

$$\mathbb{E}[f_i^\gamma(S)] \geq F_i^\gamma(y(\ell)) \geq \left(1 - \frac{\epsilon}{k} \cdot c\right) \cdot \gamma$$

for all $i \in [k]$.

Due to the initial truncation, we have that $f_i^\gamma(S) \leq \gamma$ with probability one. Thus, using Markov's inequality for each $i \in [k]$, we obtain that with probability at least $1 - \frac{c}{k}$, we have $f_i^\gamma(S) \geq (1-\epsilon)\gamma$. Therefore, taking a union bound over k functions, we obtain $f_i^\gamma(S) \geq (1-\epsilon)\gamma$ for all $i \in [k]$ with probability at least $1 - c$, and since $f_i(S) \geq f_i^\gamma(S)$ we get an integral solution S with max-min value at least $(1-\epsilon)\gamma$ as claimed. □

B.3 Necessity of monotonicity

In light of the approximation algorithms for non-monotone submodular function maximization under matroid constraints (see, for example, (Lee et al., 2009)), one might hope that an analogous bi-criteria approximation algorithm could exist for robust non-monotone submodular function maximization. However, we show that even without any matroid constraints, getting any approximation in the non-monotone case is NP-hard.

Lemma 3. *Unless $P = NP$, no polynomial time algorithm can output a set $\tilde{S} \subseteq V$ given general submodular functions $f_1, \ldots, f_k$ such that $\min_{i\in[k]} f_i(\tilde{S})$ is within a positive factor of $\max_{S\subseteq V} \min_{i\in[k]} f_i(S)$.*

Proof. We use a reduction from SAT. Suppose that we have a SAT instance with variables $x_1, \ldots, x_n$. Consider $V = \{1, \ldots, n\}$. For every clause in the SAT instance we introduce a nonnegative linear (and therefore submodular) function. For a clause $\bigvee_{i\in A} x_i \vee \bigvee_{i\in B} \overline{x_i}$ define

$$f(S) := |S \cap A| + |B \setminus S|.$$

It is easy to see that f is linear and nonnegative. If we let S be the set of true variables in a truth assignment, then it is easy to see that $f(S) > 0$ if and only if the corresponding clause is satisfied. Consequently, finding a set S such that all functions f corresponding to different clauses are positive is as hard as finding a satisfying assignment for the SAT instance. □

C Preliminaries for the Online Algorithm

C.1 Properties of the Soft-Min function

Consider a set of k twice differentiable, real-valued functions $g_1, \ldots, g_k$. Let g_{min} be the minimum among these functions, i.e., for each point x in the domain, define $g_{min}(x) := \min_{i\in[k]} g_i(x)$. This function can be approximated by using the so-called *soft-min* function H defined as

$$H(x) = -\frac{1}{\alpha} \ln \sum_{i\in[k]} e^{-\alpha g_i(x)},$$

where $\alpha > 0$ is a fixed parameter. We now present some of the key properties of this function in the following lemma.

Lemma 4. *For any set of k twice differentiable, real-valued functions $g_1, \ldots, g_k$, the soft-min function H satisfies the following properties:*

1. *Bounds:*
$$g_{min}(x) - \frac{\ln k}{\alpha} \leq H(x) \leq g_{min}(x). \tag{15}$$
2. *Gradient:*
$$\nabla H(x) = \sum_{i\in[k]} p_i(x) \nabla g_i(x), \tag{16}$$
where $p_i(x) := e^{-\alpha g_i(x)} / \sum_{j\in[k]} e^{-\alpha g_j(x)}$. Clearly, if $\nabla g_i \geq 0$ for all $i \in [k]$, then $\nabla H \geq 0$.
3. *Hessian:*
$$\frac{\partial^2 H(x)}{\partial x_{e_1} \partial x_{e_2}} = \sum_{i\in[k]} p_i(x) \left(-\alpha \frac{\partial g_i(x)}{\partial x_{e_1}} \frac{\partial g_i(x)}{\partial x_{e_2}} + \frac{\partial^2 g_i(x)}{\partial x_{e_1} \partial x_{e_2}} \right) + \alpha \nabla_{e_1} H(x) \cdot \nabla_{e_2} H(x) \tag{17}$$
Moreover, if for all $i \in [k]$ we have $\left|\frac{\partial g_i}{\partial x_{e_1}}\right| \leq L_1$, and $\left|\frac{\partial^2 g_i}{\partial x_{e_1} \partial x_{e_2}}\right| \leq L_2$, then $\left|\frac{\partial^2 H}{\partial x_{e_1} \partial x_{e_2}}\right| \leq 2\alpha L_1^2 + L_2$.
4. *Comparing the average of the g_i functions with H: given $T > 0$ we have*
$$H(x) \leq \sum_{i\in[k]} p_i(x) g_i(x) \leq H(x) + \frac{n + \ln T}{\alpha} + \frac{\ln k}{\alpha} + \frac{k e^{-n}}{T}. \tag{18}$$
So, for $\alpha > 0$ sufficiently large $\sum_{i\in[k]} p_i(x) g_i(x)$ is a good approximation of $H(x)$.

Proof. We will just prove properties 1 and 4, since the rest is an straightforward calculation.

1. First, for all $i \in [k]$ we have $e^{-\alpha g_i(x)} \leq e^{-\alpha g_{min}(x)}$. Thus,

$$H(x) = -\frac{1}{\alpha}\ln\sum_{i\in[k]} e^{-\alpha g_i(x)} \geq -\frac{1}{\alpha}\ln\left(ke^{-\alpha g_{min}(x)}\right) = g_{min}(x) - \frac{\ln k}{\alpha}$$

On the other hand, $\sum_{i\in[k]} e^{-\alpha g_i(x)} \geq e^{-\alpha g_{min}(x)}$. Hence,

$$H(x) \leq -\frac{1}{\alpha}\ln\left(e^{-\alpha g_{min}(x)}\right) = g_{min}(x).$$

4. Let us consider sets $A_1 = \{i \in [k] : \ g_i(x) \leq g_{min}(x) + (n + \ln T)/\alpha\}$ and $A_2 = \{i \in [k] : g_i(x) > g_{min}(x) + (n + \ln T)/\alpha\}$. Our intuitive argument is the following: when α is sufficiently large, those $p_i(x)$'s with $i \in A_2$ are exponentially small, and $p_i(x)$'s with $i \in A_1$ go to a uniform distribution over elements in A_1. First, observe that for each $i \in A_2$ we have

$$p_i(x) = \frac{e^{-\alpha g_i(x)}}{\sum_{i\in[k]} e^{-\alpha g_i(x)}} < \frac{e^{-\alpha[g_{min}(x)+(n+\ln T)/\alpha]}}{e^{-\alpha g_{min}(x)}} = \frac{e^{-n}}{T},$$

so $\sum_{i\in A_2} p_i(x)g_i(x) \leq \frac{ke^{-n}}{T}$. On the other hand, for any $i \in A_1$ we have

$$\sum_{i\in A_1} p_i(x)g_i(x) \leq \left(g_{min}(x) + \frac{n+\ln T}{\alpha}\right)\sum_{i\in A_1} p_i(x) \leq H(x) + \frac{n+\ln T}{\alpha} + \frac{\ln k}{\alpha}$$

where in the last inequality we used the approximation property of the soft-min function. Therefore,

$$\sum_{i\in[k]} p_i(x)g_i(x) \leq H(x) + \frac{n+\ln T}{\alpha} + \frac{\ln k}{\alpha} + \frac{ke^{-n}}{T}.$$

Finally, the other inequality is clear since $\sum_{i\in[k]} p_i(x)g_i(x) \geq g_{min}(x) \geq H(x)$.

□

Now, we prove a lemma which is used to prove Theorem 2. This is done via a simple Taylor approximation.

Lemma 5. *Fix a parameter $\delta > 0$. Consider T collections of k twice-differentiable functions, namely $\{g_i^1\}_{i\in[k]}, \dots, \{g_i^T\}_{i\in[k]}$. Assume $0 \leq g_i^t(x) \leq 1$ for any x in the domain, for all $t \in [T]$ and $i \in [k]$. Define the corresponding sequence of soft-min functions $H^1, \dots, H^T$, with a common parameter $\alpha > 0$. Then, any two sequences of points $\{x^t\}_{t\in[T]}, \{y^t\}_{t\in[T]} \subseteq [0,1]^V$ with $|x^t - y^t| \leq \delta$ satisfy*

$$\sum_{t\in[T]} H^t(y^t) - \sum_{t\in[T]} H^t(x^t) \geq \sum_{e\in V}\sum_{t\in[T]} \nabla_e H^t(x^t)(y_e^t - x_e^t) - O(Tn^2\delta^2\alpha).$$

Proof. For every $t \in [T]$ define a matroid $\mathcal{M}_t = (V \times \{t\}, \mathcal{I} \times \{t\}) = (V_t, \mathcal{I}_t)$. Given this, the union matroid is given by a ground set $V^{[T]} = \bigcup_{t=1}^T V_t$, and independent set family $\mathcal{I}^{[T]} = \{S \subseteq V^{1:T} : \ S \cap V_t \in \mathcal{I}_t\}$. Define $\mathbb{H}(X) := \sum_{t\in[T]} H^t(x^t)$ for any matrix $X \in \mathcal{P}(\mathcal{M})^T$, where x^t denotes the t-th column of X. Clearly, $\nabla_{(e,t)}\mathbb{H}(X) = \nabla_e H^t(x^t)$. Moreover, the Hessian corresponds to

$$\nabla^2_{(e_1,t),(e_2,s)}\mathbb{H}(X) = \begin{cases} 0 & \text{if } t \neq s \\ \nabla^2_{e_1,e_2}\mathbb{H}^t(x^t) & \text{if } t = s \end{cases}$$

Consider any $X, Y \in \mathcal{P}(\mathcal{M})^T$ with $|y_e^t - x_e^t| \leq \delta$. Therefore, a Taylor's expansion of $\mathbb{H}$ gives

$$\mathbb{H}(Y) = \mathbb{H}(X) + \nabla\mathbb{H}(X)\cdot(Y - X) + \frac{1}{2}(Y-X)^\top\nabla^2\mathbb{H}(\xi)\cdot(Y-X)$$

where ξ is on the line between X and Y. If we expand the previous expression we obtain

$$\mathbb{H}(Y) - \mathbb{H}(X) = \sum_{e\in V}\sum_{t\in[T]} \nabla_e H^t(x^t)(y_e^t - x_e^t) + \frac{1}{2}\sum_{e_1,e_2\in V}\sum_{t\in[T]}(y_{e_1}^t - x_{e_1}^t)\nabla^2_{e_1,e_2}H^t(\xi)(y_{e_2}^t - x_{e_2}^t)$$

Finally, by using property 3 in Lemma 4 and by bounding the Hessian (and using the fact that $g_i^t(x) \in [0,1]$) we get

$$\mathbb{H}(Y) - \mathbb{H}(X) \geq \sum_{e\in V}\sum_{t=1}^T \nabla_e H^t(x^t)(y_e^t - x_e^t) - O(Tn^2\delta^2\alpha),$$

which is equivalent to

$$\sum_{t\in[T]} H^t(y^t) - \sum_{t\in[T]} H^t(x^t) \geq \sum_{e\in V}\sum_{t\in[T]} \nabla_e H^t(x^t)(y_e^t - x_e^t) - O(Tn^2\delta^2\alpha).$$

□

C.2 Proof of Theorem 2

In order to get sub-linear regret for the FPL algorithm 4, Kalai and Vempala (2005) assume a couple of conditions on the problem (see Appendix C.3). Similarly, for our online model we need to consider the following for any $t \in [T]$:

1. bounded diameter of $\mathcal{P}(\mathcal{M})$, i.e., for all $y, y' \in \mathcal{P}(\mathcal{M})$, $\|y - y'\|_1 \leq D$;
2. for all $x, y \in \mathcal{P}(\mathcal{M})$, we require $|y \cdot \Delta H^t(x)| \leq L$;
3. for all $y \in \mathcal{P}(\mathcal{M})$, we require $\|\Delta H^t(y)\|_1 \leq A$,

Now, we give a complete proof of Theorem 2 for any given learning parameter $\eta > 0$, but the final result follows with $\eta = \sqrt{D/LAT}$ and assuming $L \leq n$, $A \leq n$ and $D \leq \sqrt{n}$, which gives a $O(n^{5/4})$ dependency on the dimension in the regret.

Proof. Consider the sequence of multilinear extensions $\{F_i^1\}_{i\in[k]}, \ldots, \{F_i^T\}_{i\in[k]}$ derived from the monotone submodular functions f_i^t obtained during the dynamic process. Since f_i^t's have value in $[0,1]$, we have $0 \leq F_i^t(y) \leq 1$ for any $y \in [0,1]^V$ and $i \in [k]$. Consider the corresponding soft-min functions H^t for collection $\{F_i^t\}_{i\in[k]}$ with $\alpha = n^2T^2$ for all $t \in [T]$. Denote $\ell = \lceil \ln \frac{1}{\epsilon} \rceil$ and fix $\tau \in \{\delta, 2\delta, \ldots, \ell\}$ with $\delta = n^{-6}T^{-3}$. According to the update in Algorithm 2, $\{y_\tau^t\}_{t\in[T]}$ and $\{y_{\tau-\delta}^t\}_{t\in[T]}$ satisfy conditions of Lemma 5. Thus, we obtain

$$\sum_{t\in[T]} H^t(y_\tau^t) - H^t(y_{\tau-\delta}^t) \geq \sum_{t\in[T]} \nabla H^t(y_{\tau-\delta}^t) \cdot (y_\tau^t - y_{\tau-\delta}^t) - O(Tn^3\delta^2\alpha).$$

Then, since the update is $y_{\tau,e}^t = y_{\tau-\delta,e}^t + \delta(1 - y_{\tau-\delta,e}^t) z_{\tau,e}^t$, we get

$$\begin{aligned}&\sum_{t\in[T]} H^t(y_\tau^t) - H^t(y_{\tau-\delta}^t)\\ &\geq \delta \sum_{t\in[T]}\sum_{e\in V} \nabla_e H^t(y_{\tau-\delta}^t)(1 - y_{\tau-\delta,e}^t) z_{\tau,e}^t - O(Tn^3\delta^2\alpha)\\ &= \delta \sum_{t\in[T]} \Delta H^t(y_{\tau-\delta}^t) \cdot z_\tau^t - O(Tn^3\delta^2\alpha). \qquad (19)\end{aligned}$$

Observe that an FPL algorithm is implemented for each τ, so we can state a regret bound for each τ by using Theorem 4. Specifically,

$$\mathbb{E}\left[\sum_{t\in[T]} \Delta H^t(y_{\tau-\delta}^t) \cdot z_\tau^t\right] \geq \max_{z\in\mathcal{P}(\mathcal{M})} \mathbb{E}\left[\sum_{t\in[T]} \Delta H^t(y_{\tau-\delta}^t) \cdot z\right] - R_\eta,$$

where $R_\eta = \eta LAT + \frac{D}{\eta}$ is the regret guarantee for a given $\eta > 0$. By taking expectation in (19) and using the regret bound we just mentioned, we obtain

$$\begin{aligned}&\mathbb{E}\left[\sum_{t\in[T]} H^t(y_\tau^t) - H^t(y_{\tau-\delta}^t)\right] \qquad (20)\\ &\geq \delta\left(\max_{z\in\mathcal{P}(\mathcal{M})} \mathbb{E}\left[\sum_{t\in[T]} \Delta H^t(y_{\tau-\delta}^t) \cdot z\right]\right) - \delta R_\eta \qquad (21)\\ &\quad - O(Tn^3\delta^2\alpha)\\ &\geq \delta\mathbb{E}\left(\sum_{t\in[T]}\left[H^t(x^*) - \sum_{i\in[k]} p_i^t(y_{\tau-\delta}^t) F_i^t(y_{\tau-\delta}^t)\right]\right)\\ &\quad - \delta R_\eta - O(Tn^3\delta^2\alpha), \qquad (22)\end{aligned}$$

where x^* is the true optimum for $\max_{x\in\mathcal{P}(\mathcal{M})} \sum_{t\in[T]} \min_{i\in[k]} F_i^t(x)$. Observe that (20) follows from monotonicity and submodularity of each f_i^t, specifically we know that

$$\begin{aligned}\Delta H^t(y) \cdot z &= \sum_{i\in[k]} p_i^t(y) \Delta F_i^t(y) \cdot z\\ &\geq \sum_{i\in[k]} p_i^t(y) F_i^t(x^*) - \sum_{i\in[k]} p_i^t(y) F_i^t(y) \qquad \text{(eq. (13))}\\ &\geq F_{min}^t(x^*) - \sum_{i\in[k]} p_i^t(y) F_i^t(y)\\ &\geq H^t(x^*) - \sum_{i\in[k]} p_i^t(y) F_i^t(y).\end{aligned}$$

By applying property (18) of the soft-min in expression (20) we get

$$\begin{aligned}&\mathbb{E}\left[\sum_{t\in[T]} H^t(y_\tau^t) - H^t(y_{\tau-\delta}^t)\right] \geq\\ &\delta\mathbb{E}\left(\sum_{t\in[T]} H^t(x^*) - H^t(y_{\tau-\delta}^t)\right) - \delta R_\eta - O(Tn^3\delta^2\alpha)\\ &- \delta T\left(\frac{n + \ln T}{\alpha} - \frac{\ln k}{\alpha} - \frac{ke^{-n}}{T}\right), \qquad (23)\end{aligned}$$

Given the choice of α and δ, the last two terms in the right-hand side of inequality (23) are small compared to

R_η, so by re-arranging terms we can state the following

$$\sum_{t\in[T]} H^t(x^*) - \mathbb{E}\left[\sum_{t\in[T]} H^t(y^t_\tau)\right] \leq (1-\delta)\left(\sum_{t\in[T]} H^t(x^*) - \mathbb{E}\left[\sum_{t\in[T]} H^t(y^t_{\tau-\delta})\right]\right) + 2\delta R_\eta$$

By iterating $\frac{\ell}{\delta}$ times in τ, we get

$$\begin{aligned}\sum_{t\in[T]} H^t(x^*) - \mathbb{E}\left[\sum_{t\in[T]} H^t(y^t_\ell)\right] &\leq (1-\delta)^{\frac{\ell}{\delta}}\left(\sum_{t\in[T]} H^t(x^*) - \sum_{t\in[T]} H^t(y^t_0)\right) + O\left(R_\eta \ln\frac{1}{\epsilon}\right)\\ &\leq \epsilon\left[\sum_{t\in[T]} H^t(x^*) + \frac{\ln k}{n^2 T}\right] + O\left(R_\eta \ln\frac{1}{\epsilon}\right),\end{aligned}$$

where in the last inequality we used $(1-\delta) \leq e^{-\delta}$. Given that the term $\epsilon\frac{\ln(k)}{n^2T}$ is small (for T and n sufficiently large) we can bound it by $O(R_\eta \ln\frac{1}{\epsilon})$. Since α is sufficiently large, we can apply the approximation property of soft-min function to obtain the following regret bound

$$(1-\epsilon)\cdot \sum_{t\in[T]} \min_{i\in[k]} F^t_i(x^*) - \mathbb{E}\left[\sum_{t\in[T]} \min_{i\in[k]} F^t_i\left(y^t_\ell\right)\right] \leq O\left(R_\eta \ln\frac{1}{\epsilon}\right).$$

Since we are doing randomized swap rounding on each y^t_ℓ, Theorem 5 shows that there is a random set S^t that is independent in $\mathcal{M}_\ell$ (i.e., S^t is the union of at most ℓ independent sets in $\mathcal{I}$) such that $\mathbb{E}\left[f^t_i(S^t)\right] \geq F^t_i\left(y^t_\ell\right)$ for all $t \in [T]$ and $i \in [k]$. Thus, we finally obtain

$$(1-\epsilon)\cdot \max_{S\in\mathcal{I}} \sum_{t\in[T]} \min_{i\in[k]} f^t_i(S) - \sum_{t\in[T]} \min_{i\in[k]} \mathbb{E}\left[f^t_i(S^t)\right] \leq O\left(R_\eta \ln\frac{1}{\epsilon}\right).$$

□

Observation 1. *Theorem 2 could be easily extended to an adaptive adversary by sampling in each stage $t \in [T]$ a different perturbation $q_t \sim [0, 1/\eta]^V$ as shown in (Kalai and Vempala, 2005).*

C.3 Follow-the-Perturbed-Leader algorithm

In this section, we briefly recall the well-known Follow-the-Perturbed-Leader (FPL) algorithm introduced in (Kalai and Vempala, 2005) and used in many online optimization problems (see e.g., (Rakhlin, 2009)). The classical online learning framework is as follows: Consider a dynamic process over T time steps. In each stage $t \in [T]$, a decision-maker has to choose a point $d_t \in \mathcal{D}$ from a fixed (possibly infinite) set of actions $\mathcal{D} \subseteq \mathbb{R}^n$, then an adversary chooses a vector s_t from a set $\mathcal{S}$. Finally, the player observes vector s_t and receives reward $s_t \cdot d_t$, and the process continues. The goal of the player is to maximize the total reward $\sum_{t\in[T]} s_t \cdot d_t$, and we compare her performance with respect to the best single action picked in hindsight, i.e., $\max_{d\in\mathcal{D}} \sum_{t=1}^T s_t \cdot d$. This performance with respect to the best single action in hindsight is called (expected) *regret*, formally:

$$\mathbf{Regret}(T) = \max_{d\in\mathcal{D}} \sum_{t\in[T]} s_t \cdot d - \mathbb{E}\left[\sum_{t\in[T]} s_t \cdot d_t\right].$$

Kalai and Vempala (2005) showed that even if one has only access to a linear programming oracle for $\mathcal{D}$, i.e., we can efficiently solve $\max_{d\in\mathcal{D}} s \cdot d$ for any $s \in \mathcal{S}$, then the FPL algorithm 4 achieves sub-linear regret, specifically $O(\sqrt{T})$.

In order to state the main result in (Kalai and Vempala, 2005), we need the following. We assume that the decision set $\mathcal{D}$ has diameter at most D, i.e., for all $d, d' \in \mathcal{D}$, $\|d - d'\|_1 \leq D$. Further, for all $d \in \mathcal{D}$ and $s \in \mathcal{S}$ we assume that the absolute reward is bounded by L, i.e., $|d \cdot s| \leq L$ and that the ℓ_1-norm of the reward vectors is bounded by A, i.e., for all $s \in \mathcal{S}$, $\|s\|_1 \leq A$.

Theorem 7 ((Kalai and Vempala, 2005))**.** *Let $s_1, \ldots, s_T \in \mathcal{S}$ be a sequence of rewards. Running the FPL algorithm 4 with parameter $\eta \leq 1$ ensures regret*

$$\mathbf{Regret}(T) \leq \eta LAT + \frac{D}{\eta}.$$

Moreover, if we choose $\eta = \sqrt{D/LAT}$, then $\mathbf{Regret}(T) \leq 2\sqrt{DLAT} = O(\sqrt{T})$.

Algorithm 4 Follow-the-Perturbed-Leader (FPL), (Kalai and Vempala, 2005)

Input: Parameter $\eta > 0$
Output: Sequence of decisions $d_1, \ldots, d_T$
1: Sample $q \sim [0, 1/\eta]^n$.
2: **for** $t = 1$ to T **do**
3: **Play** $d_t = \mathrm{argmax}_{d\in\mathcal{D}} \left(\sum_{j=1}^{t-1} s_j + q\right)^\top d$.