Recurrent Submodular Welfare and Matroid Blocking Semi-Bandits

Orestis Papadigenopoulos & Constantine Caramanis
The University of Texas at Austin

NeurIPS 2021

Motivating Example: Music Recommendation

- **Application**: Music recommendation platform (Spotify, Soundcloud, Deezer etc.)

Motivating Example: Music Recommendation

- **Application**: Music recommendation platform (Spotify, Soundcloud, Deezer etc.)
- A set of songs is suggested to a user every day.

Motivating Example: Music Recommendation

- **Application**: Music recommendation platform (Spotify, Soundcloud, Deezer etc.)
- A set of songs is suggested to a user every day.
- Every song is associated with a *stochastic reward* (e.g., click probability).

Motivating Example: Music Recommendation

- **Application**: Music recommendation platform (Spotify, Soundcloud, Deezer etc.)
- A set of songs is suggested to a user every day.
- Every song is associated with a *stochastic reward* (e.g., click probability).
- **Goal**: Maximize the total expected reward collected within T days.

Motivating Example: Music Recommendation

- **Application**: Music recommendation platform (Spotify, Soundcloud, Deezer etc.)
- A set of songs is suggested to a user every day.
- Every song is associated with a *stochastic reward* (e.g., click probability).
- **Goal**: Maximize the total expected reward collected within T days.

Nice-to-have features for such a system:

Motivating Example: Music Recommendation

- **Application**: Music recommendation platform (Spotify, Soundcloud, Deezer etc.)
- A set of songs is suggested to a user every day.
- Every song is associated with a *stochastic reward* (e.g., click probability).
- **Goal**: Maximize the total expected reward collected within T days.

Nice-to-have features for such a system:

1. **Diversity within a day** (do <u>not</u> suggest very similar songs)

Motivating Example: Music Recommendation

- **Application**: Music recommendation platform (Spotify, Soundcloud, Deezer etc.)
- A set of songs is suggested to a user every day.
- Every song is associated with a *stochastic reward* (e.g., click probability).
- **Goal**: Maximize the total expected reward collected within T days.

Nice-to-have features for such a system:

1. **Diversity within a day** (do not suggest very similar songs)
2. **Non-repetitiveness** (do not spam the user with the same song again and again)

Motivating Example: Music Recommendation

Diversity within a day (do not suggest very similar songs on the same day)

Motivating Example: Music Recommendation

Diversity within a day (do not suggest very similar songs on the same day)

- d features (e.g., music genres).

Motivating Example: Music Recommendation

Diversity within a day (do not suggest very similar songs on the same day)

- d features (e.g., music genres).
- Associate each song with a vector in $\{0, 1\}^d$.

Motivating Example: Music Recommendation

Diversity within a day (do <u>not</u> suggest very similar songs on the same day)

- d features (e.g., music genres).
- Associate each song with a vector in $\{0,1\}^d$.
- Set the i-th coordinate to 1, if the song belongs to genre i.

Motivating Example: Music Recommendation

Diversity within a day (do not suggest very similar songs on the same day)

- d features (e.g., music genres).
- Associate each song with a vector in $\{0, 1\}^d$.
- Set the i-th coordinate to 1, if the song belongs to genre i.
- **Favor diversity**: The songs suggested to the user in the same day must be linearly independent.

Motivating Example: Music Recommendation

Diversity within a day (do not suggest very similar songs on the same day)

- d features (e.g., music genres).
- Associate each song with a vector in $\{0, 1\}^d$.
- Set the i-th coordinate to 1, if the song belongs to genre i.
- **Favor diversity**: The songs suggested to the user in the same day must be linearly independent.
- Linear independence can be modeled as a matroid!

Motivating Example: Music Recommendation

Diversity within a day (do not suggest very similar songs on the same day)

- d features (e.g., music genres).
- Associate each song with a vector in $\{0,1\}^d$.
- Set the i-th coordinate to 1, if the song belongs to genre i.
- **Favor diversity**: The songs suggested to the user in the same day must be linearly independent.
- Linear independence can be modeled as a matroid!
- **Recall**: A matroid $\mathcal{M}$ over a ground set A of elements is defined as a collection of independent sets $\mathcal{I}$, such that:
  1. If $S \in \mathcal{I}$ and $T \subset S$ then $T \in \mathcal{I}$.
  2. If $S, T \in \mathcal{I}$ and $|T| < |S|$, then $\exists e \in S$ such that $T + e \in \mathcal{I}$.

Motivating Example: Music Recommendation

Non-repetitiveness (do not spam the user with the same song again and again)

Motivating Example: Music Recommendation

Non-repetitiveness (do not spam the user with the same song again and again)

- Each song i is associated with a *delay* d_i.

Motivating Example: Music Recommendation

Non-repetitiveness (do not spam the user with the same song again and again)

- Each song i is associated with a *delay* d_i.
- **Avoid spamming**: After a song i is suggested, it cannot appear again for the next d_i days.

Motivating Example: Music Recommendation

Non-repetitiveness (do not spam the user with the same song again and again)

- Each song i is associated with a *delay* d_i.
- **Avoid spamming**: After a song i is suggested, it cannot appear again for the next d_i days.
- The delay of each song can depend on factors such as popularity, promotion and more.

Problem Definition: Matroid Blocking Semi-Bandits

We consider the following variant of multi-armed bandits (MAB):

- Set A of k arms, each associated with:
  - An unknown nonnegative reward distribution.
  - A fixed and known delay $d_i \in \mathbb{N}_{\geq 1}$.

Problem Definition: Matroid Blocking Semi-Bandits

We consider the following variant of multi-armed bandits (MAB):

- Set A of k arms, each associated with:
  - An unknown nonnegative reward distribution.
  - A fixed and known delay $d_i \in \mathbb{N}_{\geq 1}$.
- Known matroid $\mathcal{M} = (A, \mathcal{I})$ over the ground set of arms (access via independence oracle).

Problem Definition: Matroid Blocking Semi-Bandits

We consider the following variant of multi-armed bandits (MAB):

- Set A of k arms, each associated with:
  - An unknown nonnegative reward distribution.
  - A fixed and known delay $d_i \in \mathbb{N}_{\geq 1}$.
- Known matroid $\mathcal{M} = (A, \mathcal{I})$ over the ground set of arms (access via independence oracle).
- Unknown time horizon of T rounds.

Problem Definition: Matroid Blocking Semi-Bandits

We consider the following variant of multi-armed bandits (MAB):

- Set A of k arms, each associated with:
  - An unknown nonnegative reward distribution.
  - A fixed and known delay $d_i \in \mathbb{N}_{\geq 1}$.
- Known matroid $\mathcal{M} = (A, \mathcal{I})$ over the ground set of arms (access via independence oracle).
- Unknown time horizon of T rounds.

- At each round, we play a subset of arms which is an independent set of $\mathcal{M}$. We observe the realization of the reward of each arm played (semi-bandit feedback), and collect the sum.

Problem Definition: Matroid Blocking Semi-Bandits

We consider the following variant of multi-armed bandits (MAB):

- Set A of k arms, each associated with:
  - An unknown nonnegative reward distribution.
  - A fixed and known delay $d_i \in \mathbb{N}_{\geq 1}$.
- Known matroid $\mathcal{M} = (A, \mathcal{I})$ over the ground set of arms (access via independence oracle).
- Unknown time horizon of T rounds.

- At each round, we play a subset of arms which is an independent set of $\mathcal{M}$. We observe the realization of the reward of each arm played (semi-bandit feedback), and collect the sum.
- Once an arm i is played, it cannot be played again for the subsequent $d_i - 1$ rounds.

Problem Definition: Matroid Blocking Semi-Bandits

We consider the following variant of multi-armed bandits (MAB):

- Set A of k arms, each associated with:
  - An unknown nonnegative reward distribution.
  - A fixed and known delay $d_i \in \mathbb{N}_{\geq 1}$.
- Known matroid $\mathcal{M} = (A, \mathcal{I})$ over the ground set of arms (access via independence oracle).
- Unknown time horizon of T rounds.

- At each round, we play a subset of arms which is an independent set of $\mathcal{M}$. We observe the realization of the reward of each arm played (semi-bandit feedback), and collect the sum.
- Once an arm i is played, it cannot be played again for the subsequent $d_i - 1$ rounds.
- **Goal:** Maximize the expected reward collected within T rounds.

Problem Definition: Matroid Blocking Semi-Bandits

Example: Uniform rank-2 matroid (i.e., play at most 2 arms per round):

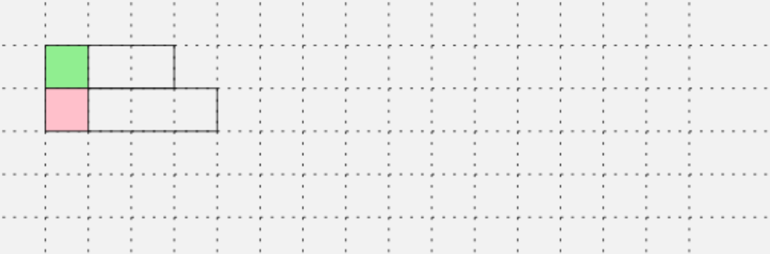

Figure: Round 1

Problem Definition: Matroid Blocking Semi-Bandits

Example: Uniform rank-2 matroid (i.e., play at most 2 arms per round):

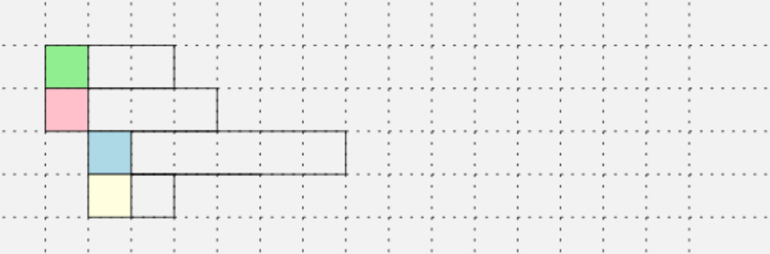

Figure: Round 2

Problem Definition: Matroid Blocking Semi-Bandits

Example: Uniform rank-2 matroid (i.e., play at most 2 arms per round):

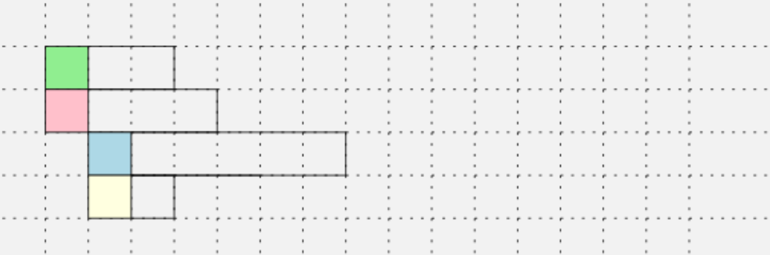

Figure: Round 3 (Idle time)

Problem Definition: Matroid Blocking Semi-Bandits

Example: Uniform rank-2 matroid (i.e., play at most 2 arms per round):

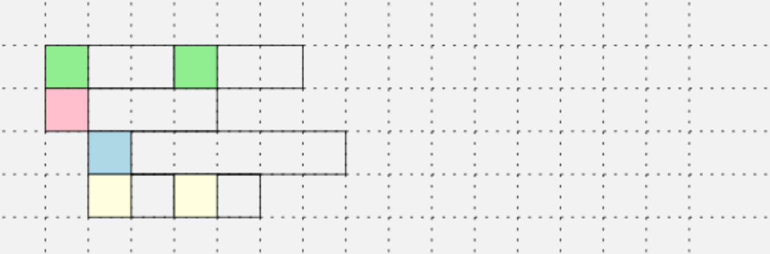

Figure: Round 4

Problem Definition: Matroid Blocking Semi-Bandits

Example: Uniform rank-2 matroid (i.e., play at most 2 arms per round):

Figure: Round ...

Matroid Blocking Semi-Bandits: Related Work

- Rank-1 matroids (i.e., 1 arm per round): $\exists$ $(1 - 1/e)$-approximation for deterministic and known rewards.

  "Blocking Bandits", [Basu, Sen, Sanghavi & Shakkottai, NeurIPS '19].

Matroid Blocking Semi-Bandits: Related Work

- Rank-1 matroids (i.e., 1 arm per round): $\exists\ (1 - 1/e)$-approximation for deterministic and known rewards.

  "Blocking Bandits", [Basu, Sen, Sanghavi & Shakkottai, NeurIPS '19].

- Delays are all 1 (i.e., arms are never blocked):

  "Matroid Bandits: Fast Combinatorial Optimization with Learning", [Kveton, Wen, Ashkan, Eydgahi & Eriksson, AUAI '16].

Matroid Blocking Semi-Bandits: Related Work

- Rank-1 matroids (i.e., 1 arm per round): $\exists\ (1 - 1/e)$-approximation for deterministic and known rewards.

  "Blocking Bandits", [Basu, Sen, Sanghavi & Shakkottai, NeurIPS '19].

- Delays are all 1 (i.e., arms are never blocked):

  "Matroid Bandits: Fast Combinatorial Optimization with Learning", [Kveton, Wen, Ashkan, Eydgahi & Eriksson, AUAI '16].

- Alternative model favoring non-repetitiveness:

  Expected reward of an arm is an increasing concave function of the last time it was played.

  "Recharging Bandits" [Kleinberg & Immorlica, FOCS '18].

Matroid Blocking Semi-Bandits: Full-Information Setting

- Assume for now that the rewards are <u>deterministic</u> and <u>known</u>.

Matroid Blocking Semi-Bandits: Full-Information Setting

- Assume for now that the rewards are deterministic and known.
- The problem is *strongly* NP-hard, even when all the rewards are 1.

Matroid Blocking Semi-Bandits: Full-Information Setting

- Assume for now that the rewards are deterministic and known.
- The problem is *strongly* NP-hard, even when all the rewards are 1.

- **Greedy approach:** At each round, play the maximum reward independent set among the available (i.e., non-blocked) arms.

Matroid Blocking Semi-Bandits: Full-Information Setting

- Assume for now that the rewards are deterministic and known.
- The problem is *strongly* NP-hard, even when all the rewards are 1.
- **Greedy approach:** At each round, play the maximum reward independent set among the available (i.e., non-blocked) arms.
- $(1 - 1/e)$-approx. for one arm per round [Basu et al., NeurIPS '19].

Matroid Blocking Semi-Bandits: Full-Information Setting

- Assume for now that the rewards are deterministic and known.
- The problem is *strongly* NP-hard, even when all the rewards are 1.

- **Greedy approach:** At each round, play the maximum reward independent set among the available (i.e., non-blocked) arms.

- $(1 - 1/e)$-approx. for one arm per round [Basu et al., NeurIPS '19].
- $1/2$-approx. for general independence systems (including matroids) [Atsidakou et al., ICML '21].

Matroid Blocking Semi-Bandits: Full-Information Setting

- Assume for now that the rewards are deterministic and known.
- The problem is *strongly* NP-hard, even when all the rewards are 1.

- **Greedy approach:** At each round, play the maximum reward independent set among the available (i.e., non-blocked) arms.

- $(1 - 1/e)$-approx. for one arm per round [Basu et al., NeurIPS '19].
- $1/2$-approx. for general independence systems (including matroids) [Atsidakou et al., ICML '21].
- But the analysis of $1/2$-approximation is **tight** for general matroids.

Can we do better?

Matroid Blocking Semi-Bandits: Full-Information

Interleaved-Greedy for full-information MBS:

Matroid Blocking Semi-Bandits: Full-Information

Interleaved-Greedy for full-information MBS:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0, 1]$ be a random *offset* drawn uniformly from $[0, 1]$.

Matroid Blocking Semi-Bandits: Full-Information

Interleaved-Greedy for full-information MBS:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0,1]$ be a random *offset* drawn uniformly from $[0,1]$.
2. **Online:** At every round $t = 1, 2, \ldots$:
   2.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.

Matroid Blocking Semi-Bandits: Full-Information

Interleaved-Greedy for full-information MBS:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0,1]$ be a random *offset* drawn uniformly from $[0,1]$.
2. **Online:** At every round $t = 1, 2, \ldots$:
   2.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   2.2 Compute a maximum-reward-independent set, A_t, contained in G_t.

Matroid Blocking Semi-Bandits: Full-Information

Interleaved-Greedy for full-information MBS:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0, 1]$ be a random *offset* drawn uniformly from $[0, 1]$.
2. **Online:** At every round $t = 1, 2, \ldots$:
   2.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t + 1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   2.2 Compute a maximum-reward-independent set, A_t, contained in G_t.
   2.3 Play the arms in A_t and collect the rewards.

Matroid Blocking Semi-Bandits: Full-Information

Theorem

Interleaved-Greedy *collects in expectation at least*

$$\left(1 - \frac{1}{e}\right) \cdot OPT(T) - \mathcal{O}\left(d_{\max} \cdot rk(\mathcal{M})\right),$$

where $OPT(T)$ *is the optimal reward for* T *rounds,* $d_{\max}$ *is the maximum delay of the instance, and* $rk(\mathcal{M})$ *the rank of the matroid.*

Matroid Blocking Semi-Bandits: Full-Information

Theorem

Interleaved-Greedy *collects in expectation at least*

$$\left(1 - \frac{1}{e}\right) \cdot OPT(T) - \mathcal{O}\left(d_{\max} \cdot rk(\mathcal{M})\right),$$

where $OPT(T)$ is the optimal reward for T rounds, $d_{\max}$ is the maximum delay of the instance, and $rk(\mathcal{M})$ the rank of the matroid.

Our proof uses tools from the analysis of submodular functions:

Matroid Blocking Semi-Bandits: Full-Information

Theorem

Interleaved-Greedy *collects in expectation at least*

$$\left(1 - \frac{1}{e}\right) \cdot OPT(T) - \mathcal{O}\left(d_{\max} \cdot rk(\mathcal{M})\right),$$

where $OPT(T)$ *is the optimal reward for* T *rounds,* $d_{\max}$ *is the maximum delay of the instance, and* $rk(\mathcal{M})$ *the rank of the matroid.*

Our proof uses tools from the analysis of submodular functions:

1. Convex relaxation based on the concave closure of submodular functions.

Matroid Blocking Semi-Bandits: Full-Information

Theorem

Interleaved-Greedy *collects in expectation at least*

$$\left(1 - \frac{1}{e}\right) \cdot OPT(T) - \mathcal{O}\left(d_{\max} \cdot rk(\mathcal{M})\right),$$

where $OPT(T)$ is the optimal reward for T rounds, $d_{\max}$ is the maximum delay of the instance, and $rk(\mathcal{M})$ the rank of the matroid.

Our proof uses tools from the analysis of submodular functions:

1. Convex relaxation based on the concave closure of submodular functions.
2. Correlation gap of submodular functions.

Matroid Blocking Semi-Bandits: Bandit Setting

- In the bandit setting, the reward distributions are initially unknown.

Matroid Blocking Semi-Bandits: Bandit Setting

- In the bandit setting, the reward distributions are initially unknown.
- **Goal:** Minimize the $(1 - 1/e)$-approximate regret, defined as:

$$(1 - 1/e) \cdot \mathsf{OPT}(T) - \mathbb{E}\left[\text{Reward of Bandit Policy}\right].$$

- Equivalently, upper bound the difference between the expected reward collected by **Interleaved-Greedy** and the bandit policy.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0, 1]$ be a random *offset* drawn uniformly from $[0, 1]$.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0,1]$ be a random *offset* drawn uniformly from $[0,1]$.
2. For each arm i, maintain a UCB-estimate of its mean reward, based on the observed samples.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0, 1]$ be a random *offset* drawn uniformly from $[0, 1]$.
2. For each arm i, maintain a UCB-estimate of its mean reward, based on the observed samples.
3. **Online:** At every round $t = 1, 2, \ldots$:
   3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t + 1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0,1]$ be a random *offset* drawn uniformly from $[0,1]$.
2. For each arm i, maintain a UCB-estimate of its mean reward, based on the observed samples.
3. **Online:** At every round $t = 1, 2, \ldots$:
   3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   3.2 Compute a maximum-reward-independent set, A_t, contained in G_t, according to the current estimates.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0,1]$ be a random *offset* drawn uniformly from $[0,1]$.
2. For each arm i, maintain a UCB-estimate of its mean reward, based on the observed samples.
3. **Online:** At every round $t = 1, 2, \ldots$:
   - 3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   - 3.2 Compute a maximum-reward-independent set, A_t, contained in G_t, according to the current estimates.
   - 3.3 Play the arms in A_t collect the rewards.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

1. **Offline:** For each arm $i \in A$, let $r_i \sim U[0,1]$ be a random *offset* drawn uniformly from $[0,1]$.
2. For each arm i, maintain a UCB-estimate of its mean reward, based on the observed samples.
3. **Online:** At every round $t = 1, 2, \ldots$:
   3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   3.2 Compute a maximum-reward-independent set, A_t, contained in G_t, according to the current estimates.
   3.3 Play the arms in A_t collect the rewards.
   3.4 Update the estimates.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

3. **Online:** At every round $t = 1, 2, \ldots$:
   3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   3.2 Compute a maximum-reward-independent set, A_t, contained in G_t, according to the current estimates.
   3.3 Play the arms in A_t collect the rewards.
   3.4 Update the estimates.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

3. **Online:** At every round $t = 1, 2, \ldots$:
   - 3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   - 3.2 Compute a maximum-reward-independent set, A_t, contained in G_t, according to the current estimates.
   - 3.3 Play the arms in A_t collect the rewards.
   - 3.4 Update the estimates.

- The sets $\{G_t\}_t$ are independent of the trajectory of the observed rewards.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

3. **Online:** At every round $t = 1, 2, \ldots$:
   3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   3.2 Compute a maximum-reward-independent set, A_t, contained in G_t, according to the current estimates.
   3.3 Play the arms in A_t collect the rewards.
   3.4 Update the estimates.

- The sets $\{G_t\}_t$ are independent of the trajectory of the observed rewards.
- The sequence $\{G_t\}_t$ is identically distributed in **Interleaved-Greedy** and **Interleaved-UCB**.

Matroid Blocking Semi-Bandits: Bandit Setting

Interleaved-UCB for the bandit setting:

3. **Online:** At every round $t = 1, 2, \ldots$:
   - 3.1 Let $G_t \subseteq A$ be the subset of arms such that, for any $i \in G_t$, the interval $[t \cdot \frac{1}{d_i} + r_i, (t+1) \cdot \frac{1}{d_i} + r_i)$ contains an integer.
   - 3.2 Compute a maximum-reward-independent set, A_t, contained in G_t, according to the current estimates.
   - 3.3 Play the arms in A_t collect the rewards.
   - 3.4 Update the estimates.

- The sets $\{G_t\}_t$ are independent of the trajectory of the observed rewards.
- The sequence $\{G_t\}_t$ is identically distributed in **Interleaved-Greedy** and **Interleaved-UCB**.
- **Key-idea**: We can upper-bound the regret "pointwise", assuming that the sequence of sets $\{G_t\}_t$ is the same in both algorithms.

Matroid Blocking Semi-Bandits: Bandit Setting

Combining the above idea with (i) the strong basis exchange property of matroids and (ii) standard UCB arguments, we show the following result:

Theorem

The $(1 - 1/e)$-approximate regret can be upper-bounded as

$$\mathcal{O}\left(k\sqrt{T\ln(T)} + k^2 + d_{\max} \cdot r(\mathcal{M})\right),$$

where k is the number of arms, $r(\mathcal{M})$ is the rank of $\mathcal{M}$, and $d_{\max}$ is the maximum delay of the instance.

Matroid Blocking Semi-Bandits: Bandit Setting

Combining the above idea with (i) the strong basis exchange property of matroids and (ii) standard UCB arguments, we show the following result:

Theorem

The $(1 - 1/e)$-approximate regret can be upper-bounded as

$$\mathcal{O}\left(k\sqrt{T\ln(T)} + k^2 + d_{\max} \cdot r(\mathcal{M})\right),$$

where k is the number of arms, $r(\mathcal{M})$ is the rank of $\mathcal{M}$, and $d_{\max}$ is the maximum delay of the instance.

- Almost matching the regret lower bound for standard (non-blocking) matroid bandits.