BATCHED CASCADING BANDITS

Anonymous authors
Paper under double-blind review

ABSTRACT

Cascading bandits model ordered recommendation under stochastic prefix feedback: in each round, a learner selects and orders K out of L items but observes outcomes only up to the first click. Standard cascading bandit algorithms update after every round, which can be impractical in real-world scenarios where instant feedback is not always available. We study batched cascading bandits, where the learner commits all recommendation lists in a batch using the feedback available at its start and updates the policy only at its batch endpoint. To address different batch constraints, we propose two algorithms: Batched Cascade Successive Accepts and Rejects (BC-SAR) and Batched Cascade Upper Confidence Bound (BC-UCB). Both algorithms combine first-position rotation with mechanisms tailored to their respective batch constraints. On homogeneous-gap instances with common attraction-probability gap Δ and constant algorithm parameters, BC-SAR has expected regret $\mathcal{O}(K(L-K)\log(LT)/\Delta)$ using $\mathcal{O}(\log T)$ batches. BC-UCB has expected regret $\mathcal{O}((L-K)\log(LT)/\Delta)$ using $\mathcal{O}(L\log(1+KT/L))$ batches. Numerical experiments evaluate our algorithms and examine their empirical performance under batched feedback.

1 INTRODUCTION

Cascading bandits (Kveton et al., 2015a) study online learning for ranked recommendations under the cascade click model (Craswell et al., 2008; Bendada et al., 2020), which captures user interactions in applications such as web search, news feeds, online retail, and carousel interfaces. In this model, a user examines a recommended list from the first position onward and clicks the first attractive item, if any. The learner observes outcomes up to and including the first click, while outcomes of later items remain unobserved. Over a horizon of T rounds, the learner selects and orders K out of L items in each round and receives reward 1 if at least one recommended item is attractive and 0 otherwise. The goal is to minimize cumulative regret relative to the list containing the K most attractive items.

Standard cascading-bandit algorithms receive feedback after each round and update the policy before the next recommendation. Such round-by-round updates may be unavailable under delayed feedback (Wang et al., 2023), periodic policy deployment, or advance planning of several recommendation lists. We therefore study batched cascading bandits to tackle these practical constraints.

At the beginning of a batch, the learner uses only previous feedback to determine all recommendation lists in that batch. The learner receives feedback from that batch only when it ends and can then update the policy. Throughout the batch, a suboptimal recommended set can remain in use, and a fixed ordering can prevent later items from being observed. The learner must consider both the recommended sets that generate reward and the list orderings that determine which outcomes are observed, using only the feedback available at the start of the batch.

Following the distinction in batched bandits (Gao et al., 2019), we consider two batch constraints. Under a static grid, the batch partition is specified before learning, representing an externally imposed update constraint. The algorithm must plan how unresolved items will be observed before each update. Under an adaptive grid, the learner chooses each new batch length at its start using feedback from earlier batches.

For static grids, we propose Batched Cascade Successive Accepts and Rejects (BC-SAR), which uses successive acceptance and rejection to construct recommendation lists. At each batch endpoint, BC-SAR classifies items as accepted, active, or rejected. In the next batch, accepted items form

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: Batch counts and leading expected-regret bounds on homogeneous-gap instances.

Algorithm	Number of Batches	Leading expected regret
CascadeUCB1 (Kveton et al., 2015a)	T	$\mathcal{O}((L-K)\log T/\Delta)$
CTS (Zhong et al., 2021)	T	$\mathcal{O}(L\log T/\Delta)$
BaSE (Gao et al., 2019)	M	$\mathcal{O}(LT^{1/M}\log L\log(LT)/\Delta)$
BC-SAR	$\mathcal{O}(\log T)$	$\mathcal{O}(K(L-K)\log(LT)/\Delta)$
BC-UCB	$\mathcal{O}((1+\lambda^{-1})L\log(1+KT/L))$	$\mathcal{O}((L-K)(1+\lambda)\log(LT)/\Delta)$

a fixed suffix of every recommendation list, while active items fill the remaining positions. The algorithm rotates active items through the first position to guarantee observations and uses a middle-position quota to control their exposure. This construction allocates observations to unresolved items while retaining accepted items. For adaptive grids, we propose Batched Cascade Upper Confidence Bound (BC-UCB). At the beginning of each batch, BC-UCB selects the K items with the largest UCB values and sets the batch length from the smallest observation count among them. It rotates the first position within the batch to balance guaranteed observations across the items.

For the comparison below, M denotes the total number of online batches, Δ is the common attraction-probability gap, and $\lambda > 0$ is the BC-UCB batch-length parameter. The grid-growth and quota parameters of BC-SAR are treated as fixed constants. Table 1 summarizes the theoretical upper bounds for our methods and representative cascading-bandit and batched-bandit algorithms, comparing their update frequency and the leading term in expected regret on homogeneous-gap instances.

Our main contributions are as follows:

- For static-grid batched cascading bandits, we propose BC-SAR, which achieves expected regret $\mathcal{O}(K(L-K)\log(LT)/\Delta)$ with $\mathcal{O}(\log T)$ batches.
- For adaptive-grid batched cascading bandits, we propose BC-UCB, which achieves expected regret $\mathcal{O}((L-K)(1+\lambda)\log(LT)/\Delta)$ with $\mathcal{O}((1+\lambda^{-1})L\log(1+KT/L))$ batches.
- For the problem class, we show that any logarithmic-regret guarantee uniform over list lengths requires $\Omega(\log(1+T/L)/\log\log T)$ batches.
- We also conduct numerical experiments to validate the effectiveness of our algorithms and support our theoretical analysis.

2 Related Work

Cascading bandits model sequential recommendation under first-click prefix feedback (Kveton et al., 2015a). Subsequent regret-minimization work considers combinatorial and linear models, Thompson sampling, and variance-aware confidence bounds (Kveton et al., 2015b; Zong et al., 2016; Zhong et al., 2021; Vial et al., 2022). Other click models use different examination assumptions, including multiple-click, position-based, and stochastic-ranking models (Katariya et al., 2016; Lagrée et al., 2016; Lattimore et al., 2018). CascadeBAI uses successive acceptance and rejection for fixed-confidence top-K identification under the same feedback model (Zhong et al., 2020). Combinatorial semi-bandits observe every selected item (Gai et al., 2012; Chen et al., 2013; Kveton et al., 2015c), whereas cascading feedback reveals only the prefix up to the first click. BatchRank also uses staged updates for stochastic click models, but its stages partition items and positions rather than delay temporal feedback (Zoghi et al., 2017).

Batched bandits divide the horizon into a small number of batches and update policies only between them: delayed-feedback analyses (Joulani et al., 2013) and two-armed static grids (Perchet et al., 2016) preceded BaSE, which establishes rate-optimal regret guarantees for static and adaptive grids (Gao et al., 2019). Related analyses cover instance-dependent elimination (Esfandiari et al., 2021), asymptotically optimal infinite grids (Lai & Robbins, 1985; Jin et al., 2021), and Thompson sampling with data-dependent batch endpoints (Kalkanlı & Özgür, 2021). Our static grid follows the hard M-batch design of BaSE, whereas BC-UCB sets each new batch length from the observation counts available at the preceding batch endpoint. Fixed-confidence analyses characterize the sample

complexity of best-arm identification (Kaufmann et al., 2016; Garivier & Kaufmann, 2016), while batched pure-exploration work asks how many sampling rounds identification requires (Jun et al., 2016; Tuynman & Degenne, 2025).

3 PRELIMINARIES

3.1 PROBLEM SETUP

We assume a batched cascading bandit with L items, a recommend list with $K < L$ positions for items, and known horizon T. Item i has an unknown attraction probability $\mu_i \in (0, 1)$. In round t, the user receives an ordered recommended list $A_t = (A_t(1), \dots, A_t(K))$ and examines it from the first position onward. Let $w_t = (w_t(1), \dots, w_t(L)) \in \{0, 1\}^L$ be the attraction vector in round t, where $w_t(i) = 1$ means that item i is attractive to the user. The user stops at the first item $A_t(k)$ with $w_t(A_t(k)) = 1$. The round reward is 1 if a click occurs and 0 otherwise. The cascade feedback contains all item outcomes up to and including the clicked item, or all K outcomes if no click occurs.

Let $S_t = \mathrm{set}(A_t)$ be the recommended set. The round reward is 1 exactly when at least one item in S_t is attractive. Under the product Bernoulli model, the probability that no item in a set S is attractive is $\prod_{i \in S}(1 - \mu_i)$. Hence, the expected reward of S is

$$f(S) = 1 - \prod_{i \in S}(1 - \mu_i).$$

So permuting a recommended list does not change its expected reward, while the order of A_t still determines which item outcomes are observed. Let $S^\star$ denote the unique set of K items with the largest attraction probabilities. The cumulative regret is $\mathrm{Reg}(T) = \sum_{t=1}^{T} \big(f(S^\star) - f(S_t)\big)$.

Before the horizon, each item receives one observation. The rounds of horizon are partitioned into M batches by batch endpoints $0 = t_0 < t_1 < \cdots < t_M = T$, where batch m contains rounds $t_{m-1} + 1, \dots, t_m$. At the beginning of batch m, the learner has to commit all recommended lists in that batch, and the cascade feedback of this batch is available only at batch endpoint t_m.

We consider two ways to determine the batch partition, static-grid setting and adaptive-grid setting. In the *static-grid setting*, M and all batch endpoints $t_1, \dots, t_M$ are specified before learning and do not depend on observed feedback (Perchet et al., 2016; Gao et al., 2019). In the *adaptive-grid setting*, at the beginning of batch m, the learner chooses its length $t_m - t_{m-1}$ using feedback available through t_{m-1}. Consequently, later batch endpoints and the realized batch count M can depend on observed feedback.

3.2 ASSUMPTIONS

We assume that attraction vectors are independent across rounds and follow the product Bernoulli model $w_t \sim \bigotimes_{i=1}^{L} \mathrm{Bernoulli}(\mu_i)$ with mean vector μ. Without loss of generality, we index the items by decreasing attraction probability, $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_K > \mu_{K+1} \geq \cdots \geq \mu_L$, so that $S^\star = \{1, \dots, K\}$. We define the pairwise and boundary gaps by

$$\Delta_{i,j} := \mu_j - \mu_i \ (1 \leq j < i \leq L), \qquad \Delta_i := \Delta_{i,K} \ (i > K).$$

Whenever a ranking or selection is not unique, we break ties by item index.

4 BC-SAR FOR STATIC GRIDS

The pseudocode of BC-SAR is given in Algorithm 1. We write $\mathrm{ord}(S)$ for the items of S in insertion order and $\mathrm{concat}(X, Y)$ for the concatenation of ordered lists X and Y.

BC-SAR maintains an accepted set $\mathcal{P}$, an active set $\mathcal{A}$ and a rejected set $\mathcal{R} := [L] \setminus (\mathcal{P} \cup \mathcal{A})$. Accepted items form the final $|\mathcal{P}|$ positions of every recommendation list, and active items fill the remaining $r := K - |\mathcal{P}|$ positions. Figure 1 shows this list structure.

At each non-final batch endpoint, BC-SAR classifies active items through confidence comparisons. The update combines confidence-based elimination (Even-Dar et al., 2006) with accept–reject rules

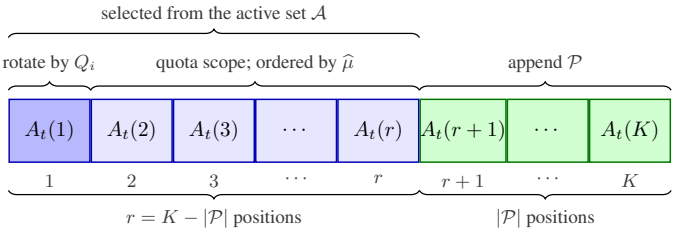

Figure 1: BC-SAR recommendation-list construction.

for top-K and combinatorial identification (Kalyanakrishnan et al., 2012; Bubeck et al., 2013; Chen et al., 2014). Let N_i and $\widehat{\mu}_i$ denote the observation count and empirical mean of item i, including its initial observation, and define $\mathrm{rad}(n) := \sqrt{\log(4LT^2)/(2n)}$. For every $i \in \mathcal{A}$, BC-SAR uses the lower confidence bound $L_i := \widehat{\mu}_i - \mathrm{rad}(N_i)$ and upper confidence bound $U_i := \widehat{\mu}_i + \mathrm{rad}(N_i)$. It rejects each item with $U_i < \theta_r$, where θ_r is the r-th largest L_i among active items. Then it accepts the largest prefix whose L_i values all exceed the U_i values outside the prefix. Algorithm 2 repeats these rejection and acceptance steps until the sets no longer change.

Within a batch, first-position rotation guarantees observations for active items. For each $i \in \mathcal{A}$, the counter Q_i records how often item i has occupied the first position. In every round, BC-SAR places an active item with the smallest Q_i first. This position is always observed.

An $\mathcal{A}$-epoch is a maximal interval over which $\mathcal{A}$ remains unchanged and $r > 0$. Within an $\mathcal{A}$-epoch, the counters q_i and d_i record the first-position and middle-position assignments of active item i. An item remains eligible for a middle position while

$$d_i < \lceil \rho(r-1)(q_i+1) \rceil, \qquad \rho > 2. \tag{1}$$

We call this middle-position quota, which controls item exposure in positions $2, \ldots, r$. The algorithm fills these positions with the eligible items having the largest empirical means. Active items occupy the earlier positions to gain more observations, while accepted items, identified as optimal with high confidence after sufficient observations, occupy the final suffix. When $\mathcal{A}$ changes, the algorithm resets q_i, d_i for the remaining active items so that allocations from the previous epoch do not affect the new quota.

4.1 Regret Analysis of BC-SAR

For the regret analysis, we use the geometric grid of BaSE (Gao et al., 2019): fix $a > 1$ and take the distinct values of $\min\{\lceil a^m \rceil, T\}$, $m = 1, 2, \ldots$, as successive batch endpoints until reaching T. Denote the resulting grid by $0 = t_0 < t_1 < \cdots < t_M = T$. For fixed $a > 1$, this grid uses $M = \mathcal{O}(\log T)$ batches.

Let $X_{i,n}$ be the n-th observed Bernoulli sample of item i, including initialization, and define the uniform confidence event

$$\mathcal{E} := \bigcap_{i=1}^{L} \bigcap_{n=1}^{T+1} \left\{ \left| \frac{1}{n} \sum_{u=1}^{n} X_{i,u} - \mu_i \right| \le \mathrm{rad}(n) \right\}. \tag{2}$$

Theorem 1 (BC-SAR instance-dependent regret)**.** *Assume Bernoulli cascade feedback, $\mu_1 \ge \cdots \ge \mu_K > \mu_{K+1} \ge \cdots \ge \mu_L$, and fixed constants $\rho > 2$ and $a > 1$. With $\mathrm{rad}(n) = \sqrt{\log(4LT^2)/(2n)}$, Algorithm 1 satisfies*

$$\mathbb{E}[\mathrm{Reg}(T)] = \mathcal{O}\left(LK \log \frac{L}{K} \log(LT) \left[\frac{1}{\Delta_{K+1,K}} + \sum_{k=1}^{K-1} \frac{\Delta_{k+1,k}}{\max\{\Delta_{K+1,K}, \Delta_{k+1,k}\}^2} \right] \right). \tag{3}$$

Proof sketch. On $\mathcal{E}$, accepted items are optimal and rejected items are suboptimal, and regret is confined to positions before the accepted suffix and bounded by the gaps $\Delta_i = \Delta_{i,K}$ and $\Delta_{k+1,k}$, where $i > K$ and $|\mathcal{P}| < k < K$ (Lemma 3). Epoch-wise recommendation accounting contributes a quota factor of at most ρK (Lemma 4). After bounding the first batch directly, the gaps control

Algorithm 1 BC-SAR

Input: L, K, T, static grid $0 = t_0 < \cdots < t_M = T$, and quota ratio $\rho > 2$.
Initialize: Sample observation $\widehat{\mu}_i$ for each $i \in [L]$; $N_i \leftarrow 1$, $Q_i, q_i, d_i \leftarrow 0$; $\mathcal{P} \leftarrow \varnothing, \mathcal{A} \leftarrow [L]$.
for $m = 1, \ldots, M-1$ **do**
  $r \leftarrow K - |\mathcal{P}|$.
  for $t = t_{m-1}+1, \ldots, t_m$ **do**
    if $r = 0$ **then**
      $A_t \leftarrow \mathrm{ord}(\mathcal{P})$.
    else
      $A_t(1) \leftarrow \arg\min_{i \in \mathcal{A}} Q_i$.
      $\mathcal{A}_t^{\mathrm{mid}} \leftarrow \{i \in \mathcal{A} \setminus \{A_t(1)\} : d_i < \lceil \rho(r-1)(q_i+1) \rceil\}$.
      $\big(A_t(2), \ldots, A_t(r)\big) \leftarrow$ the $r-1$ items in $\mathcal{A}_t^{\mathrm{mid}}$ with the largest $\widehat{\mu}_i(t_{m-1})$.
      $\big(A_t(r+1), \ldots, A_t(K)\big) \leftarrow \mathrm{ord}(\mathcal{P})$.
      $Q_{A_t(1)} \leftarrow Q_{A_t(1)} + 1$, $q_{A_t(1)} \leftarrow q_{A_t(1)} + 1$.
      $d_i \leftarrow d_i + 1$ for every $i \in \{A_t(2), \ldots, A_t(r)\}$.
  Execute the batch plan and receive its feedback at t_m.
  Update $N_i, \widehat{\mu}_i$.
  Set $L_i \leftarrow \widehat{\mu}_i - \mathrm{rad}(N_i)$ and $U_i \leftarrow \widehat{\mu}_i + \mathrm{rad}(N_i)$ for every $i \in \mathcal{A}$.
  $(\mathcal{P}^+, \mathcal{A}^+) \leftarrow$ SAR-UPDATE$(\mathcal{P}, \mathcal{A}, \{(\widehat{\mu}_i, L_i, U_i)\}_{i \in \mathcal{A}})$.
  if $\mathcal{A}^+ \neq \mathcal{A}$ or $\mathcal{P}^+ \neq \mathcal{P}$ **then**
    Reset $q_i, d_i \leftarrow 0$ for every $i \in \mathcal{A}^+$.
  $(\mathcal{P}, \mathcal{A}) \leftarrow (\mathcal{P}^+, \mathcal{A}^+)$.
$r \leftarrow K - |\mathcal{P}|$.
if $r = 0$ **then**
  For $t = t_{M-1}+1, \ldots, T$, $A_t \leftarrow \mathrm{ord}(\mathcal{P})$.
else
  $S_{\mathrm{final}} \leftarrow \mathrm{TopK}_r(\mathcal{A}; \widehat{\mu}(t_{M-1}))$.
  For $t = t_{M-1}+1, \ldots, T$, $A_t \leftarrow \mathrm{concat}(\mathrm{ord}(S_{\mathrm{final}}), \mathrm{ord}(\mathcal{P}))$.

Algorithm 2 SAR-UPDATE: successive accept/reject closure

Input: $\mathcal{P}, \mathcal{A}$, and $\{(\widehat{\mu}_i, L_i, U_i)\}_{i \in \mathcal{A}}$.
while $|\mathcal{P}| < K$ **do**
  $r \leftarrow K - |\mathcal{P}|$.
  $\theta_r \leftarrow$ the r-th largest L_i in $\mathcal{A}$.
  $D \leftarrow \{i \in \mathcal{A} : U_i < \theta_r\}$, $\mathcal{A} \leftarrow \mathcal{A} \setminus D$.
  Order $\mathcal{A} = (i_1, \ldots, i_s)$ by decreasing $\widehat{\mu}_i$.
  if $|\mathcal{A}| = r$ **then**
    $h \leftarrow r$.
  else
    $h \leftarrow \max\big(\{0\} \cup \{k : 1 \le k < r,\ \min_{u \le k} L_{i_u} > \max_{k < v \le s} U_{i_v}\}\big)$.
  $H \leftarrow \{i_1, \ldots, i_h\}$, with $H = \varnothing$ when $h = 0$.
  if $D = \varnothing$ and $H = \varnothing$ **then**
    Return $(\mathcal{P}, \mathcal{A})$.
  $(\mathcal{P}, \mathcal{A}) \leftarrow (\mathcal{P} \cup H, \mathcal{A} \setminus H)$.
Return $(\mathcal{P}, \mathcal{A})$.

each regret term's duration (Lemma 5). Grouping by stopping endpoint gives the factor $\log(L/K)$ (Lemma 6), and at most $\min\{L, M\}$ epochs add overhead. Combining Propositions 1 and 2, using $T\mathbb{P}(\mathcal{E}^c) \le 1$, and retaining the leading logarithmic term in T yields (3). Appendix A.3 gives the complete proof. □

Remark 1 (General static-grid extension)**.** *The analysis can be extended to any static grid satisfying $t_m \le \alpha t_{m-1} + b$ for $2 \le m \le M$, with $\alpha \ge 1$ and $b \ge 0$. These parameters control how far each update can extend beyond the preceding batch endpoint. The regret bound in (16) continues to hold with corresponding adjustments to the grid-dependent terms.*

Corollary 1 (Homogeneous-gap BC-SAR bound)**.** *Suppose the attraction probabilities form a homogeneous-gap instance with $0 < \Delta < \mu < 1$, where $\mu_1 = \cdots = \mu_K = \mu$ and $\mu_{K+1} = \cdots = \mu_L = \mu - \Delta$. For fixed ρ and a,*

$$\mathbb{E}[\mathrm{Reg}(T)] = \mathcal{O}(K(L-K)\log(LT)/\Delta)\,.$$

Algorithm 3 BC-UCB

Input: L, K, T, and batch parameter $\lambda > 0$.
Initialize: Sample observation $\widehat{\mu}_i$ for each $i \in [L]$; $N_i \leftarrow 1$; $t \leftarrow 0, m \leftarrow 1$.
while $t < T$ **do**
  $U_i \leftarrow \widehat{\mu}_i + \mathrm{rad}(N_i)$ for every $i \in [L]$.
  $S_m \leftarrow \mathrm{TopK}\{U_i : i \in [L]\}$.
  $n_m \leftarrow \min_{i \in S_m} N_i$, $h_m \leftarrow \min\{\lceil \lambda n_m \rceil, T - t\}$.
  $N_i^{\mathrm{start}} \leftarrow N_i$, $q_i \leftarrow 0$ for every $i \in S_m$.
  for $s = t+1, \ldots, t+h_m$ **do**
    $a_1^s \leftarrow \arg\min_{i \in S_m} (N_i^{\mathrm{start}} + q_i)$.
    $A_s \leftarrow (a_1^s, S_m \setminus \{a_1^s\}$ in decreasing U_i order$)$.
    $q_{a_1^s} \leftarrow q_{a_1^s} + 1$.
  Execute the batch plan, receive feedback at $t + h_m$.
  Update $N_i, \widehat{\mu}_i$.
  $t \leftarrow t + h_m$, $m \leftarrow m + 1$.

5 BC-UCB FOR ADAPTIVE GRIDS

BC-UCB adapts the recommendation rule of CASCADEUCB1 to batched feedback on an adaptive grid (Auer et al., 2002; Kveton et al., 2015a). It selects batch lengths adaptively and rotates items in each selected set through the first position. Algorithm 3 gives the pseudocode of BC-UCB.

At the beginning of batch m, the algorithm sets S_m to the K items with the largest UCBs:

$$U_i^m := \widehat{\mu}_i^m + \mathrm{rad}(N_i^m), \qquad S_m := \mathrm{TopK}\{U_i^m : i \in [L]\}.$$

Here, N_i^m and $\widehat{\mu}_i^m$ denote the observation count and empirical mean of item i at the beginning of batch m. The confidence radius $\mathrm{rad}(n) = \sqrt{\log(4LT^2)/(2n)}$ is the same as in BC-SAR. After selecting S_m, define $n_m := \min_{i \in S_m} N_i^m$ as the minimum selected-item observation count. This count acts as an information bottleneck: a small n_m leads to a short batch and an early opportunity to revise S_m. A larger n_m permits a longer commitment because more observations reduce the selected items' confidence radii. The algorithm sets the batch length to $h_m := \min\{\lceil \lambda n_m \rceil, T - t_{m-1}\}$, where $\lambda > 0$ controls the update frequency.

Within this batch, the recommended set S_m remains fixed, but first-position rotation varies its ordering to distribute guaranteed observations. The rotation uses the batch-start observation count $N_i^{\mathrm{start}} := N_i^m$ and the batch-local counter $q_i := 0$, which records the first-position assignments to item i within this batch. In each round s, the algorithm places the item first that satisfies $a_1^s = \arg\min_{i \in S_m} \left(N_i^{\mathrm{start}} + q_i\right)$, and increments $q_{a_1^s}$. The other recommended items follow in decreasing UCB order. Because the first position is always observed, these assignments balance guaranteed observations across the selected items before the next update. This rotation also lead to a steady increase in the counts used to set later batch lengths.

5.1 REGRET AND BATCH ANALYSIS OF BC-UCB

The BC-UCB analysis uses the uniform confidence event $\mathcal{E}$ defined in Section 4.1. We bound its expected cumulative regret and the number of batches it uses.

Theorem 2 (BC-UCB regret)**.** *For $\lambda > 0$, Algorithm 3 satisfies*

$$\mathbb{E}[\mathrm{Reg}(T)] \leq C_\lambda \log(4LT^2) \sum_{i=K+1}^{L} \frac{1}{\Delta_i} + 1, \quad C_\lambda := 2(\sqrt{\lambda + 2} + 1)\sqrt{\lambda + 2}.$$

Proof sketch. On $\mathcal{E}$, we charge each batch's loss to observations of its selected suboptimal items, weighted by their batch-start confidence radii (Lemma 7). Each charged observation increases N_i and shrinks the confidence radius that controls later batch losses. For a fixed suboptimal item i, the adaptive batch rule gives $h_m \leq \lambda N_i^m + 1$. The UCB comparison limits the contributing batches to those with $N_i^m \leq 2\log(4LT^2)/\Delta_i^2$, giving a total of at most $C_\lambda \log(4LT^2)/\Delta_i$. Summing over $i > K$ and adding the confidence-failure contribution $T\mathbb{P}(\mathcal{E}^c) \leq 1$ proves the theorem. Appendix A.4 gives the complete proof. □

Theorem 3 (BC-UCB batch-count upper bound)**.** *The number M of adaptive batches generated by Algorithm 3 satisfies*

$$M \le (4 + 6/\lambda)\, L \log\left(1 + KT/L\right) + 1.$$

The complete proof is given in Appendix A.5.

Remark 2 (Same-order comparison of BC-SAR and BC-UCB bounds)**.** *For fixed L, K, and ρ, take a constant $a > 1$ for* BC-SAR *and $\lambda = \Theta(1)$ for* BC-UCB*. Then both batch counts are $\mathcal{O}(\log T)$, and the expected regret bounds are*

$$\mathbb{E}[\mathrm{Reg}_{\text{BC-SAR}}(T)] = \mathcal{O}\left(\frac{K(L-K)\log(LT)}{\Delta}\right), \mathbb{E}[\mathrm{Reg}_{\text{BC-UCB}}(T)] = \mathcal{O}\left(\frac{(L-K)\log(LT)}{\Delta}\right).$$

5.2 Problem-Class Batch-Count Lower Bound

To identify a necessary batch budget for the problem class, we use the $K = 1$ subclass, for which the cascade model reduces exactly to an L-item Bernoulli bandit. Let $\Pi_1(L, M, T)$ be the policies with one initial observation per item and at most M adaptive batches, and let $\mathcal{B}_\Delta^L$ contain the Bernoulli instances whose minimum nonzero gap is at least Δ. Define $\mathfrak{R}_1^{\mathrm{mm}}(L, M, T) := \inf_{\pi \in \Pi_1(L,M,T)} \sup_{\Delta > 0} \Delta \sup_{\nu \in \mathcal{B}_\Delta^L} \mathbb{E}_\nu[\mathrm{Reg}(T; \pi)]$.

Theorem 4 (Regret and batch-count lower bound for problem class)**.** *Suppose $L \ge 2$, $T \ge L$, and $1 \le M \le \log_2(1 + T/L)$. There is a universal constant $c > 0$ such that $\mathfrak{R}_1^{\mathrm{mm}}(L, M, T) \ge cM^{-3}L[(1+T/L)^{1/M} - 1]$. Consequently, a guarantee uniform over list lengths with $\mathfrak{R}_1^{\mathrm{mm}}(L, M, T) \le C(L-1)\log T$ at $K = 1$ for a constant $C > 0$ requires*

$$M = \Omega(\log(1 + T/L)/\log\log T)\,.$$

The complete proof is given in Appendix A.6.

6 Experiments

We adapt CTS, TS-Cascade (Zhong et al., 2021), CascadeUCB1 and CascadeKL-UCB (Kveton et al., 2015a) to batching by delaying feedback updates until each batch endpoint. CascadeUCB1 and CascadeKL-UCB use UCB (Auer et al., 2002) and KL-UCB (Garivier & Cappé, 2011) confidence indices. Their batched versions repeat the list ranked by the batch-start indices throughout the batch. Within each batch, CTS and TS-Cascade resample each round. All lists are planned before the batch, and all statistical updates are deferred to its batch endpoint. Each experiment is evaluated over 20 runs. Appendix B specifies the shared protocol and numerical companions.

6.1 Synthetic Comparison

To evaluate our algorithms, we compare them with the cascading bandit baselines described above under batched cascade feedback. We set $L = 256$, $T = 200{,}000$, and $K \in \{4, 8\}$, with attraction probabilities decreasing uniformly from 0.35 to 0.15. For the static-grid comparison, every method uses the same geometric grid with $a \in \{2, 1.5\}$, giving 18 and 31 learning batches respectively. For the adaptive one, each baseline uses the batch endpoints generated by BC-UCB with $\lambda \in \{4, 1\}$.

Figure 2 shows that BC-SAR and BC-UCB attain lower mean final regret than all four baselines across the two static-grid and two adaptive-grid settings, respectively. Upper triangles indicate curves continuing beyond the displayed range. Table 4 in Appendix B.2 gives the full results.

To control the potential disadvantage of replaying BC-UCB-generated batches in the adaptive comparison, we rerun each baseline on a geometric grid with the same realized batch count. Geometric scheduling indeed reduces baseline regret, but all remain above the corresponding BC-UCB result (Appendix B.6).

6.2 Parameter Scaling

We vary L, K, and Δ to study parameter scaling. The experiments use homogeneous-gap instances with $T = 200{,}000$ and $\mu = 0.20$.

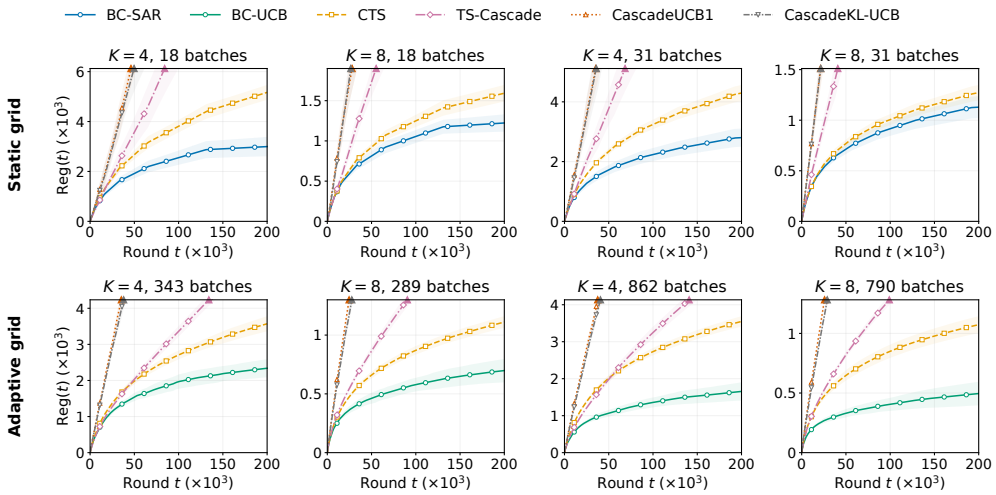

Figure 2: Synthetic comparison under static and adaptive grids. Shading shows ± 1 SD.

Table 2: Regret and batch count under varying problem parameters at $T = 200{,}000$.

Instance			BC-SAR		BC-UCB	
L	K	Δ	Regret	Batches	Regret	Batches
64	2	0.05	4726 ± 1256	18	1444 ± 1462	353
128	2	0.05	9168 ± 1499	18	1987 ± 359	658
256	2	0.05	11045 ± 1005	18	4131 ± 744	1285.4
64	4	0.05	3678 ± 547	18	794 ± 298	260.7
64	8	0.05	2364 ± 502	18	787 ± 873	242.1
64	2	0.10	2412 ± 645	18	505 ± 121	261
64	2	0.15	1781 ± 323	18	302 ± 54	208.5

Increasing L raises regret for both methods and yields near-linear growth in BC-UCB's batch count, consistent with the $L \log(1 + KT/L)$ bound in Theorem 3. Widening the gap Δ accelerates item discrimination, substantially lowering regret for both methods. Increasing K consistently reduces regret, as larger recommendation lists offer more click opportunities.

6.3 MovieLens Evaluation under Correlated Feedback

To assess the practical utility of our algorithms, we test them on MovieLens 1M (Harper & Konstan, 2015) with $K = 4$ and nested item pools of $L \in \{256, 2048\}$ items selected by rating count. Each round samples one of the 6,040 user profiles and draws all binary outcomes from that profile, which induces intra-round correlation across items and places the feedback outside the independent cascade model. Following the MovieLens protocol for cascading bandits (Zong et al., 2016), we measure cumulative regret against a greedy maximum-coverage reference, which gives a $(1 - 1/e)$-approximation for this monotone submodular objective (Nemhauser et al., 1978). Figure 3 shows this regret over $T = 100{,}000$ rounds. Terminal results are in Appendix B.2.

Both BC-SAR and BC-UCB have lower mean regret than Batched CTS under their respective matched batch schedules, especially on the larger item pool ($L = 2048$). In this experiment setting, intra-round correlation violates the independent cascade model, which suggests that our algorithms are robust to such deviations in practical applications.

6.4 Regret–Batch Tradeoff

We examine how the two methods trade regret for policy-update frequency. The experiment uses a homogeneous-gap instance with $L = 128$, $K = 4$, $T = 200{,}000$, $\mu = 0.20$, and $\Delta = 0.075$. We vary the batch count M for BC-SAR and λ for BC-UCB, with smaller λ inducing more updates.

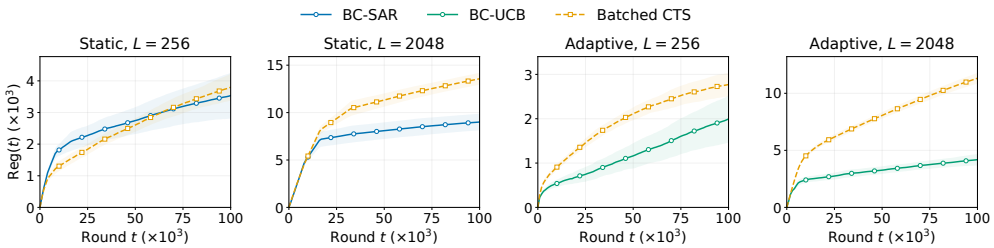

Figure 3: MovieLens cumulative regret for $K = 4$, with ± 1 SD bands.

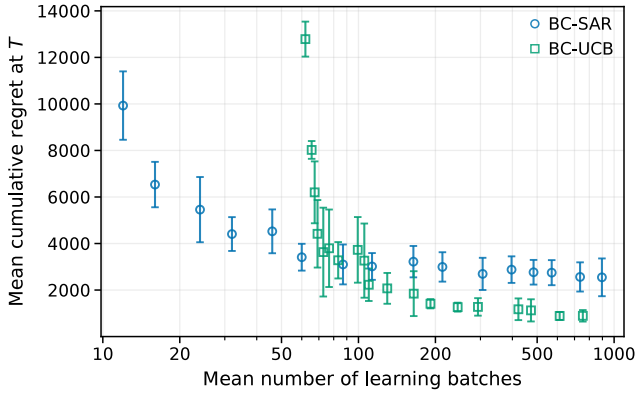

Figure 4: Regret–batch tradeoff. Error bars show ± 1 SD.

Figure 4 shows that BC-SAR performs better with fewer static-grid updates, while BC-UCB performs better with more frequent adaptive updates. The crossover occurs around 100 realized batches. Regret fluctuates with batch count because batch partitions can delay the elimination of sufficiently observed suboptimal items.

We also conduct ablation and sensitivity experiments on the acceptance, rejection, and middle-position quota in BC-SAR and on the first-position rotation in BC-UCB (Appendices B.3–B.5). The ablations show that acceptance and rejection reduce BC-SAR's regret and that first-position rotation reduces BC-UCB's regret. The middle-position quota controls recommendation counts in the regret analysis, although it does not consistently reduce empirical regret.

7 Conclusion

We developed BC-SAR and BC-UCB for static and adaptive batch control in cascading bandits, with instance-dependent regret guarantees. BC-SAR provides regret control under a prescribed static-grid budget, while BC-UCB achieves logarithmic regret with logarithmically many adaptive batches. Future work can determine whether the additional K factor in the BC-SAR regret bound and the L dependence of the BC-UCB batch bound are necessary, and whether variance-aware or KL-based confidence bounds sharpen these guarantees. Extensions of the batched cascading bandit setting could consider finite user patience or early abandonment, as well as robustness to adversarial feedback corruption (Hu et al., 2026).

Reproducibility Statement

The appendices contain complete proofs, the experimental protocol, detailed numerical results, and additional ablation and sensitivity analyses.

The accompanying anonymous supplementary package contains the experiment code, configurations, per-seed results, and instructions and scripts for reproducing the experiments and regenerating the figures and tables.

Statement on the Use of AI

Generative AI tools were employed for language editing, stylistic refinement, and grammatical corrections during this work. All core intellectual contributions, including algorithmic design, mathematical formulations, theoretical derivations, and experimental analyses, were conducted independently by the authors without automated assistance. The authors have critically reviewed and verified all AI-assisted revisions and assume full responsibility for the integrity, accuracy, and final content of this work.

References

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256, 2002. doi: 10.1023/A:1013689704352.

Walid Bendada, Guillaume Salha, and Théo Bontempelli. Carousel personalization in music streaming apps with contextual bandits. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pp. 420–425, 2020. doi: 10.1145/3383313.3412217.

Sébastien Bubeck, Tengyao Wang, and Nitin Viswanathan. Multiple identifications in multi-armed bandits. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 258–265. PMLR, 2013.

Shouyuan Chen, Tian Lin, Irwin King, Michael R. Lyu, and Wei Chen. Combinatorial pure exploration of multi-armed bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 27, 2014.

Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, volume 28 of *Proceedings of Machine Learning Research*, pp. 151–159, 2013.

Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pp. 87–94, 2008. doi: 10.1145/1341531.1341545.

Hossein Esfandiari, Amin Karbasi, Abbas Mehrabian, and Vahab Mirrokni. Regret bounds for batched bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7340–7348, 2021. doi: 10.1609/aaai.v35i8.16901.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7(39):1079–1105, 2006.

Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Transactions on Networking*, 20(5):1466–1478, 2012. doi: 10.1109/TNET.2011.2181864.

Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. Batched multi-armed bandits problem. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Aurélien Garivier and Olivier Cappé. The KL-UCB algorithm for bounded stochastic bandits and beyond. In *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pp. 359–376. PMLR, 2011.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Proceedings of the 29th Conference on Learning Theory (COLT)*, volume 49 of *Proceedings of Machine Learning Research*, pp. 998–1027, 2016.

F. Maxwell Harper and Joseph A. Konstan. The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4):19:1–19:19, 2015. doi: 10.1145/2827872.

Zicheng Hu, Yuchen Wang, and Cheng Chen. Robust decentralized multi-armed bandits: From corruption-resilience to byzantine-resilience. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21912–21920, 2026. doi: 10.1609/aaai.v40i26.39344.

Tianyuan Jin, Jing Tang, Pan Xu, Keke Huang, Xiaokui Xiao, and Quanquan Gu. Almost optimal anytime algorithm for batched multi-armed bandits. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5065–5073. PMLR, 2021.

Pooria Joulani, András György, and Csaba Szepesvári. Online learning under delayed feedback. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1453–1461, 2013.

Kwang-Sung Jun, Kevin Jamieson, Robert Nowak, and Xiaojin Zhu. Top arm identification in multi-armed bandits with batch arm pulls. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 139–148, 2016.

Cem Kalkanlı and Ayfer Özgür. Batched thompson sampling. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. PAC subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, pp. 655–662, 2012.

Sumeet Katariya, Branislav Kveton, Csaba Szepesvári, and Zheng Wen. DCM bandits: Learning to rank with multiple clicks. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1215–1224, 2016.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17(1):1–42, 2016.

Branislav Kveton, Csaba Szepesvári, Zheng Wen, and Azin Ashkan. Cascading bandits: Learning to rank in the cascade model. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 767–776, 2015a.

Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvári. Combinatorial cascading bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, 2015b.

Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvári. Tight regret bounds for stochastic combinatorial semi-bandits. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 38 of *Proceedings of Machine Learning Research*, pp. 535–543, 2015c.

Paul Lagrée, Claire Vernade, and Olivier Cappé. Multiple-play bandits in the position-based model. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985. doi: 10.1016/0196-8858(85)90002-8.

Tor Lattimore, Branislav Kveton, Shuai Li, and Csaba Szepesvári. TopRank: A practical algorithm for online stochastic ranking. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14:265–294, 1978. doi: 10.1007/BF01588971.

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, and Erik Snowberg. Batched bandit problems. *The Annals of Statistics*, 44(2):660–681, 2016. doi: 10.1214/15-AOS1381.

Adrienne Tuynman and Rémy Degenne. The batch complexity of bandit pure exploration. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 60442–60468, 2025.

Daniel Vial, Sujay Sanghavi, Sanjay Shakkottai, and R. Srikant. Minimax regret for cascading bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Dairui Wang, Junyu Cao, Yan Zhang, and Wei Qi. Cascading bandits: Optimizing recommendation frequency in delayed feedback environments. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Zixin Zhong, Wang Chi Cheung, and Vincent Tan. Best arm identification for cascading bandits in the fixed confidence setting. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11481–11491. PMLR, 2020.

Zixin Zhong, Wang Chi Cheung, and Vincent Y. F. Tan. Thompson sampling algorithms for cascading bandits. *Journal of Machine Learning Research*, 22(218):1–66, 2021.

Masrour Zoghi, Tomas Tunys, Mohammad Ghavamzadeh, Branislav Kveton, Csaba Szepesvári, and Zheng Wen. Online learning to rank in stochastic click models. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 4199–4208, 2017.

Shi Zong, Hao Ni, Kenny Sung, Nan Rosemary Ke, Zheng Wen, and Branislav Kveton. Cascading bandits for large-scale recommendation problems. In *Proceedings of the 32nd Conference on Uncertainty in Artificial Intelligence (UAI)*, 2016.

A Technical Analysis

This appendix provides the complete proofs of the four main theoretical results: the regret bound for BC-SAR, the regret and batch-complexity bounds for BC-UCB, and the problem-class lower bound.

A.1 Notation

Table 3: Notation used in the proofs.

Symbol	Meaning
Problem and feedback	
L, K, T	Number of items, list size, and learning horizon.
A_t, S_t	Ordered list at round t and its unordered item set.
$S^\star$	Unique optimal top-K item set.
B_t, O_t, σ_t	Selected suboptimal items, omitted optimal items, and their Kveton matching.
μ_i	Attraction probability of item i, under the canonical ordering.
Δ_i	Boundary gap $\mu_K - \mu_i$ for $i > K$.
$\Delta_{i,j}$	Pairwise gap $\mu_j - \mu_i$ for $1 \le j < i \le L$.
$N_i(t)$	Observation count through round t, including initialization.
$\widehat{\mu}_i(t)$	Empirical mean based on observations available through t.
$\mathrm{rad}(n)$	Confidence radius $\sqrt{\log(4LT^2)/(2n)}$.
β	Proof-only shorthand $\beta := \log(4LT^2)$.
L_i, U_i	Lower and upper confidence bounds $\widehat{\mu}_i \mp \mathrm{rad}(N_i)$.
$f(S)$	Expected cascade reward of item set S.
$\mathrm{Reg}(T)$	Cumulative pseudo-regret.
BC-SAR	
$\mathcal{P}, \mathcal{A}, \mathcal{R}$	Accepted, active, and rejected item sets.
r	Remaining list positions $K - \lvert\mathcal{P}\rvert$.
$Q_i(t)$	Persistent first-position count through round t.
q_i, d_i	Within-epoch first-position and middle-position counts.
ρ	Middle-position quota ratio, with $\rho > 2$.
$\bar{M}$	Nominal geometric-grid power depth $\lceil \log_a T \rceil$, used only in the proofs.
$C_{\rho,r}$	Quota factor $1 + \rho(r-1)$.
$\mathrm{Reg}_{\mathrm{pre}}$	Regret accumulated before the final batch.
BC-UCB	
λ	Adaptive batch parameter.
S_m	UCB-selected item set in batch m.
$N_i^m, \widehat{\mu}_i^m$	Observation count and empirical mean of item i at the beginning of batch m.
rad_i^m, U_i^m	Batch-start radius $\mathrm{rad}(N_i^m)$ and upper bound $\widehat{\mu}_i^m + \mathrm{rad}_i^m$.
n_m	Minimum observation count among items in S_m.
h_m	Planned batch length $\lceil \lambda n_m \rceil$, truncated at T.
N_i^{start}	Frozen observation count of item i at the beginning of the current batch.
q_i	Number of first-position assignments to item i already planned in the current batch.
Ψ_m	Logarithmic observation-count potential.

A.2 Cascade Loss Decomposition

The following result bounds one-round regret by pairing recommended suboptimal items with omitted optimal items and weighting their mean gaps by observation probabilities. It supplies the common loss decomposition for the BC-SAR and BC-UCB analyses.

Theorem 5 (Cascade loss decomposition). *Let $B_t = S_t \setminus S^\star$ and $O_t = S^\star \setminus S_t$. For every ordered list A_t, there is a bijection $\sigma_t : B_t \to O_t$. For $i \in B_t$ and $j \in O_t$, define*

$$G_{i,j,t} := \{\exists k \in [K] \text{ such that } A_t(k) = i, \sigma_t(i) = j, w_t(A_t(\ell)) = 0 \text{ for all } \ell < k\}.$$

Thus, $G_{i,j,t}$ is the event that suboptimal item i replaces optimal item j under the matching and is observed in the cascade feedback. The one-round regret satisfies

$$f(S^\star) - f(S_t) \le \sum_{i \in B_t} \Delta_{i,\sigma_t(i)} \mathbb{P}_t\big(G_{i,\sigma_t(i),t}\big).$$

Here $\mathbb{P}_t$ conditions on the history and A_t (Kveton et al., 2015a, Theorem 1).

A.3 Proof of Theorem 1: BC-SAR Regret Upper Bound

The proof first analyzes the regret on the uniform confidence event $\mathcal{E}$ defined in Section 4.1, Equation (2). For the geometric grid in Section 4, let $\bar{M} := \lceil \log_a T \rceil$ denote its nominal power depth. The capped power sequence reaches T no later than index $\bar{M}$, so the batch indices below refer to $M \le \bar{M}$ distinct batch endpoints. For $M \ge 2$, merging repeated batch endpoints preserves $t_1 = \lceil a \rceil$ and $t_m \le a t_{m-1} + 1$ for $2 \le m \le M$. Indeed, if j is the first power index attaining t_m, then $\min\{T, \lceil a^{j-1} \rceil\} = t_{m-1}$, so $t_m \le a\lceil a^{j-1} \rceil + 1$. Thus all stopping-duration and final-batch bounds below retain the same constants, and the epoch count remains at most $\min\{L, M\} \le \min\{L, \bar{M}\}$. The pre-final arguments apply when $M \ge 2$; a single batch has no pre-final regret.

Lemma 1 (Uniform confidence). *For $T \ge 1$, $\mathbb{P}(\mathcal{E}^c) \le 1/T$.*

Proof of Lemma 1. Fix an item $i \in [L]$ and a sample size $n \in \{1, \dots, T+1\}$, and define its observation-indexed empirical mean by

$$\widehat{\mu}_{i,n} := \frac{1}{n} \sum_{u=1}^{n} X_{i,u}.$$

In any round, whether item i is observed is determined by the history, the recommended order, and the attraction outcomes of preceding items, all of which are independent of item i's current attraction outcome. Thus $X_{i,1}, \dots, X_{i,n}$ are independent Bernoulli samples with mean μ_i. Hoeffding's inequality and $\mathrm{rad}(n)^2 = \beta/(2n)$ therefore give

$$\mathbb{P}(|\widehat{\mu}_{i,n} - \mu_i| > \mathrm{rad}(n)) \le 2\exp\big(-2n\,\mathrm{rad}(n)^2\big) = 2e^{-\beta}.$$

Initialization contributes one observation per item, and each of the T online rounds contributes at most one additional observation to a fixed item. Hence every attainable observation count lies in $\{1, \dots, T+1\}$, and $\mathcal{E}$ contains $L(T+1)$ item–sample-size conditions. A union bound over these conditions yields

$$\mathbb{P}(\mathcal{E}^c) \le \sum_{i=1}^{L} \sum_{n=1}^{T+1} \mathbb{P}(|\widehat{\mu}_{i,n} - \mu_i| > \mathrm{rad}(n)) \le 2L(T+1)e^{-\beta} = \frac{T+1}{2T^2} \le \frac{1}{T}.$$

The equality uses $\beta = \log(4LT^2)$, and the last inequality follows from $T + 1 \le 2T$ for $T \ge 1$. □

At every batch endpoint, the confidence bounds on $\mathcal{E}$ satisfy

$$L_i \le \mu_i \le U_i, \qquad L_i \ge \mu_i - 2\,\mathrm{rad}(N_i), \qquad U_i \le \mu_i + 2\,\mathrm{rad}(N_i). \tag{4}$$

The pre-final regret analysis uses safety, the accepted-prefix loss decomposition, and three bounds on recommendation counts and stopping durations. The proofs of these five lemmas are deferred to Appendix A.7.

Lemma 2 (Safety of SAR-Update). *On $\mathcal{E}$, every endpoint classification preserves*

$$\mathcal{P} \subseteq S^\star, \qquad S^\star \setminus \mathcal{P} \subseteq \mathcal{A}, \qquad \mathcal{R} \cap S^\star = \varnothing, \qquad |S^\star \setminus \mathcal{P}| = r, \qquad |\mathcal{A}| \ge r, \tag{5}$$

where $r := K - |\mathcal{P}|$. The accepted set remains a true top prefix of $S^\star$, and strict confidence separation cannot separate items with equal attraction probabilities within $S^\star$.

Lemma 3 (BC-SAR accepted-prefix loss decomposition)**.** *For $m = 1, \dots, M$, let $(\mathcal{A}_m, \mathcal{P}_m, \mathcal{R}_m)$ denote the state used to construct the lists in batch m, and set $r_m := K - |\mathcal{P}_m|$. On $\mathcal{E}$, every round $t \in \{t_{m-1}+1, \dots, t_m\}$, including a round in the final batch, satisfies the following accepted-prefix loss decomposition:*

$$f(S^\star) - f(S_t) \le \sum_{i \in B_t} \left(\Delta_i + \sum_{k=1}^{K-1} \Delta_{k+1,k} \mathbf{1}\{|\mathcal{P}_m| < k\} \right). \tag{6}$$

Lemma 4 ($\mathcal{A}$-epoch recommendation count)**.** *For $1 \le r \le K$, define $C_{\rho,r} := 1 + \rho(r-1)$. Fix a $\mathcal{A}$-epoch e whose first round is t_e^{start} and whose length is ℓ_e, and write $s_e := |\mathcal{A}_e|$ and $r_e := K - |\mathcal{P}_e|$. If $r_e \ge 1$, then the following bound holds for every $i \in \mathcal{A}_e$ and $h \in \{1, \dots, \ell_e\}$:*

$$\sum_{t=t_e^{\mathrm{start}}}^{t_e^{\mathrm{start}}+h-1} \mathbf{1}\{i \in S_t\} \le C_{\rho,r_e} \left(\frac{h}{s_e} + 3 \right). \tag{7}$$

For item i, let $N_i^{\mathrm{rec}}(t) := \sum_{u=1}^{t} \mathbf{1}\{i \in S_u\}$ denote its recommendation count through round t. For the same i and h as in Lemma 4, this definition gives

$$\sum_{t=t_e^{\mathrm{start}}}^{t_e^{\mathrm{start}}+h-1} \mathbf{1}\{i \in S_t\} = N_i^{\mathrm{rec}}(t_e^{\mathrm{start}} + h - 1) - N_i^{\mathrm{rec}}(t_e^{\mathrm{start}} - 1).$$

For $i > K$, define the stopping endpoint of its boundary-gap term by

$$\tau_{i,0} := \inf\{t_m : i \notin \mathcal{A}_{m+1}\}, \tag{8}$$

and define the stopping endpoint of its k-th adjacent-gap term by

$$\tau_{i,k} := \inf\{t_m : i \notin \mathcal{A}_{m+1} \text{ or } |\mathcal{P}_{m+1}| \ge k\}, \qquad 1 \le k < K, \tag{9}$$

where the infima range over $m = 1, \dots, M-1$. If a term does not stop before batch endpoint t_{M-1}, set $\tau_{i,k} := t_{M-1}$.

Lemma 5 (Boundary-gap and adjacent-gap stopping duration)**.** *On $\mathcal{E}$, the durations excluding the first batch satisfy*

$$\tau_{i,0} - t_1 \le \frac{9aL\beta}{\Delta_i^2}, \tag{10}$$

$$\tau_{i,k} - t_1 \le \frac{9aL\beta}{\max\{\Delta_i, \Delta_{k+1,k}\}^2}, \qquad 1 \le k < K. \tag{11}$$

Lemma 6 (Stopping-endpoint aggregation excluding the first batch)**.** *On $\mathcal{E}$, index the epochs in the pre-final phase by e. For each epoch e, let $n_e := |\mathcal{A}_e \setminus S^\star|$, and let ℓ_e^+ be the number of its rounds excluding those in the first batch. Then*

$$\begin{aligned} &\sum_e \frac{C_{\rho,K}\ell_e^+}{K + n_e} \left[\sum_{i \in \mathcal{A}_e \setminus S^\star} \Delta_i + n_e \sum_{k=1}^{K-1} \Delta_{k+1,k} \mathbf{1}\{|\mathcal{P}_e| < k\} \right] \\ &\quad \le 9aL\beta C_{\rho,K} \log \frac{L}{K} \left[\frac{1}{\Delta_{K+1,K}} + \sum_{k=1}^{K-1} \frac{\Delta_{k+1,k}}{\max\{\Delta_{K+1,K}, \Delta_{k+1,k}\}^2} \right]. \end{aligned} \tag{12}$$

Proposition 1 (BC-SAR pre-final regret)**.** *Let $\mathrm{Reg}_{\mathrm{pre}}$ denote the regret incurred in the pre-final phase. Then*

$$\begin{aligned} \mathbb{E}[\mathrm{Reg}_{\mathrm{pre}} \mathbf{1}\{\mathcal{E}\}] &\le 9aL\beta C_{\rho,K} \log \frac{L}{K} \left[\frac{1}{\Delta_{K+1,K}} + \sum_{k=1}^{K-1} \frac{\Delta_{k+1,k}}{\max\{\Delta_{K+1,K}, \Delta_{k+1,k}\}^2} \right] \\ &\quad + C_{\rho,K} \left(\frac{\lceil a \rceil}{L} + 3\min\{L, M\} \right) \sum_{i=K+1}^{L} \Delta_{i,1}. \end{aligned} \tag{13}$$

Proposition 2 (BC-SAR final-batch regret)**.** *Let* $\mathrm{Reg}_{\mathrm{final}}$ *denote the regret incurred in the final batch. Then*

$$\mathbb{E}[\mathrm{Reg}_{\mathrm{final}}\,\mathbf{1}\{\mathcal{E}\}] \le 4K + \frac{8(a-1)LK\big(1+\log(2L)\big)}{\Delta_{K+1,K}}. \tag{14}$$

The proof of Proposition 2 is deferred to Appendix A.7.

Proof of Proposition 1. Lemmas 12 and 14 ensure that every endpoint classification and every recommended list during a $\mathcal{A}$-epoch is always well defined.

We work on $\mathcal{E}$. Apply Lemma 3 in every pre-final round to bound the loss by a sum over the selected suboptimal items. A round with $r = 0$, or with no active suboptimal item, has zero regret.

Fix a $\mathcal{A}$-epoch e, let ℓ_e be its length, and write $\mathcal{A}_e, \mathcal{P}_e$ for its active and accepted sets. Let $n_e := |\mathcal{A}_e \setminus S^\star|$. Let Reg_e denote the regret incurred in epoch e. Lemma 2 gives

$$|\mathcal{A}_e| = K - |\mathcal{P}_e| + n_e.$$

Applying Lemma 4 to the suboptimal items in the epoch and using the accepted-prefix loss decomposition gives

$$\begin{aligned}\mathrm{Reg}_e \le \frac{C_{\rho,K-|\mathcal{P}_e|}\ell_e}{K-|\mathcal{P}_e|+n_e}\left[\sum_{i\in\mathcal{A}_e\setminus S^\star}\Delta_i + n_e\sum_{k=1}^{K-1}\Delta_{k+1,k}\mathbf{1}\{|\mathcal{P}_e|<k\}\right] \\ + 3C_{\rho,K-|\mathcal{P}_e|}\sum_{i\in\mathcal{A}_e\setminus S^\star}\left[\Delta_i + \sum_{k=1}^{K-1}\Delta_{k+1,k}\mathbf{1}\{|\mathcal{P}_e|<k\}\right].\end{aligned} \tag{15}$$

For $p \in \{1,\dots,K-1\}$ and $n_e \ge 1$,

$$\frac{C_{\rho,p+1}}{p+1+n_e} - \frac{C_{\rho,p}}{p+n_e} = \frac{\rho n_e + \rho - 1}{(p+n_e)(p+n_e+1)} > 0.$$

Therefore the first term of (15) is at most

$$\frac{C_{\rho,K}\ell_e}{K+n_e}\left[\sum_{i\in\mathcal{A}_e\setminus S^\star}\Delta_i + n_e\sum_{k=1}^{K-1}\Delta_{k+1,k}\mathbf{1}\{|\mathcal{P}_e|<k\}\right].$$

Throughout the first batch, $\mathcal{A} = [L]$, $\mathcal{P} = \varnothing$, and $n_e = L - K$. Applying Lemma 4 with $h = t_1$ and using the accepted-prefix loss decomposition gives the upper bound

$$\frac{C_{\rho,K}t_1}{L}\sum_{i=K+1}^{L}\Delta_{i,1}.$$

If the initial $\mathcal{A}$-epoch extends beyond the first batch, then $\ell_e = t_1 + \ell_e^+$. The first term of (15) is linear in ℓ_e, whereas its second term is independent of ℓ_e. Lemma 6 bounds the sum of the ℓ_e^+-parts of the first term over all $\mathcal{A}$-epochs.

For the second term of (15), every suboptimal item $i \in \mathcal{A}_e \setminus S^\star$ satisfies

$$\Delta_i + \sum_{k=1}^{K-1}\Delta_{k+1,k}\mathbf{1}\{|\mathcal{P}_e|<k\} = \Delta_{i,|\mathcal{P}_e|+1} \le \Delta_{i,1}.$$

Therefore, since $C_{\rho,K-|\mathcal{P}_e|} \le C_{\rho,K}$, the second term of (15) is at most $3C_{\rho,K}\sum_{i\in\mathcal{A}_e\setminus S^\star}\Delta_{i,1}$. Since each $\mathcal{A}$-epoch is determined by its active set, which changes only at batch endpoints, the number of $\mathcal{A}$-epochs is at most $\min\{L, M\}$. Summing the second term of (15) over all $\mathcal{A}$-epochs therefore yields the upper bound

$$3C_{\rho,K}\min\{L,M\}\sum_{i=K+1}^{L}\Delta_{i,1}.$$

Combining the stopping-endpoint bound from Lemma 6 with the bounds for the first and second terms above, and using $t_1 = \lceil a \rceil$, proves (13). □

The propositions above yield the following explicit finite-time form of Theorem 1:

$$\begin{aligned}\mathbb{E}[\mathrm{Reg}(T)] \le 9aL\log(4LT^2)\rho K \log\frac{L}{K}\left[\frac{1}{\Delta_{K+1,K}} + \sum_{k=1}^{K-1}\frac{\Delta_{k+1,k}}{\max\{\Delta_{K+1,K},\Delta_{k+1,k}\}^2}\right] \\ + \rho K\left(\frac{\lceil a\rceil}{L} + 3\min\{L,M\}\right)\sum_{i=K+1}^{L}\Delta_{i,1} \\ + \frac{8(a-1)LK\big(1+\log(2L)\big)}{\Delta_{K+1,K}} + 4K + 1.\end{aligned} \tag{16}$$

Proof of Theorem 1. If $M = 1$, Proposition 2 and $T\mathbb{P}(\mathcal{E}^c) \le 1$ imply the claimed bound directly. Assume henceforth that $M \ge 2$. Propositions 1 and 2 control the pre-final and final-batch contributions on $\mathcal{E}$, respectively. Every round has regret at most one, so Lemma 1 implies

$$\mathbb{E}[\mathrm{Reg}(T)\mathbf{1}\{\mathcal{E}^c\}] \le T\mathbb{P}(\mathcal{E}^c) \le 1.$$

Applying $C_{\rho,K} \le \rho K$ in Proposition 1 and adding Proposition 2 and the confidence-failure bound proves (16). For fixed a and ρ, the remaining terms are lower order in T, while $\log(4LT^2) = \Theta(\log(LT))$, which proves (3). □

Proof of Remark 1. For a static grid satisfying $t_m \le \alpha t_{m-1} + b$, the stopping-duration argument gives $t_m \le 8\alpha L\beta/g^2 + b \le (8\alpha + b)L\beta/g^2$, where g is the relevant gap and $L\beta/g^2 \ge 1$. Thus the pre-final bound uses $8\alpha + b$ in place of $9a$, and its first-batch term uses the actual first batch endpoint t_1. For the final batch, $T - t_{M-1} \le (\alpha-1)t_{M-1} + b$. For $B_0, c > 0$ and $z \ge 0$,

$$\big((\alpha-1)z + b\big)\min\{1, B_0 e^{-cz}\} \le b + \frac{\alpha-1}{c}\big(1 + \log_+ B_0\big).$$

Applying the same gap grouping as in the proof of Proposition 2 therefore replaces $4K$ by $4bK$ and $a - 1$ by $\alpha - 1$. The epoch-count and confidence-failure terms are unchanged. □

Proof of Corollary 1. If $M = 1$, then $T \le \lceil a\rceil$, so for fixed a the hidden constant absorbs the bound $\mathrm{Reg}(T) \le T$. Assume henceforth that $M \ge 2$. For a homogeneous-gap instance, $\Delta_{k+1,k} = 0$ for every $k < K$, so the adjacent-gap sum in (15) vanishes. In an epoch with n active suboptimal items, the first term of (15) is at most

$$C_{\rho,K}\ell_e\frac{n\Delta}{K+n} \le C_{\rho,K}\ell_e\frac{(L-K)\Delta}{L}.$$

Setting $\ell_e = t_1$ gives the upper bound

$$\frac{C_{\rho,K}t_1(L-K)\Delta}{L}.$$

After the first batch, if any suboptimal item remains active, the active set can only shrink and the remaining active suboptimal items form a nonempty terminal interval of rounds. Choose a suboptimal item that remains active through the last round of this interval. Its active duration equals the length of the interval, so Lemma 5 bounds this length by $9aL\beta/\Delta^2$. Thus the first terms after the first batch sum to at most

$$\frac{9a\beta C_{\rho,K}(L-K)}{\Delta}.$$

The sum of the second term of (15) over all $\mathcal{A}$-epochs is at most

$$3C_{\rho,K}\min\{L,M\}(L-K)\Delta.$$

Applying $C_{\rho,K} \le \rho K$ to these three bounds and adding Proposition 2 and $\mathbb{E}[\mathrm{Reg}(T)\mathbf{1}\{\mathcal{E}^c\}] \le 1$ gives an $\mathcal{O}(K(L-K)\log(LT)/\Delta)$ leading term for fixed a and ρ, proving Corollary 1. □

A.4 Proof of Theorem 2: BC-UCB Regret Upper Bound

The theorem combines the batchwise UCB comparison with an observation-weighted sum of confidence radii.

For batch m, let $\mathcal{H}_m$ be the history at its beginning, and define

$$\mathrm{rad}_i^m := \mathrm{rad}(N_i^m), \qquad \mathcal{E}_m := \bigcap_{i=1}^{L} \{|\widehat{\mu}_i^m - \mu_i| \le \mathrm{rad}_i^m\}.$$

Since $\widehat{\mu}_i^m$ is based on N_i^m observations, $\mathcal{E} \subseteq \mathcal{E}_m$ for every batch m.

Lemma 7 (BC-UCB batchwise loss decomposition)**.** *Let $B_m := S_m \setminus S^\star$, fix a bijection $\sigma_m : B_m \to S^\star \setminus S_m$, and set $\mathcal{T}_m := \{t_{m-1}+1, \dots, t_m\}$. Define the batch regret by*

$$\mathrm{Reg}_m := \sum_{t \in \mathcal{T}_m} \big(f(S^\star) - f(S_t)\big).$$

On $\mathcal{E}_m$, every $i \in B_m$ satisfies

$$\Delta_{i,\sigma_m(i)} \le 2\,\mathrm{rad}_i^m\,.$$

The regret in batch m satisfies the following batchwise loss decomposition:

$$\mathrm{Reg}_m \le \sum_{i \in B_m} 2\,\mathrm{rad}_i^m\ \mathbb{E}\left[\sum_{t \in \mathcal{T}_m} \mathbf{1}\{G_{i,\sigma_m(i),t}\} \,\middle|\, \mathcal{H}_m\right],$$

where $G_{i,\sigma_m(i),t}$ is defined as in Theorem 5.

Proof of Lemma 7. Write $U_i^m := \widehat{\mu}_i^m + \mathrm{rad}_i^m$. Fix $i \in B_m$. Since i is selected while $\sigma_m(i)$ is omitted, the UCB selection rule gives $U_i^m \ge U_{\sigma_m(i)}^m$. On $\mathcal{E}_m$,

$$\Delta_{i,\sigma_m(i)} = \mu_{\sigma_m(i)} - \mu_i \le 2\,\mathrm{rad}_i^m\,.$$

Applying Theorem 5 in every round of $\mathcal{T}_m$, using the same pairing σ_m, and conditioning on $\mathcal{H}_m$ yields

$$\mathrm{Reg}_m \le \sum_{i \in B_m} \Delta_{i,\sigma_m(i)} \mathbb{E}\left[\sum_{t \in \mathcal{T}_m} \mathbf{1}\{G_{i,\sigma_m(i),t}\} \,\middle|\, \mathcal{H}_m\right] \le \sum_{i \in B_m} 2\,\mathrm{rad}_i^m\, \mathbb{E}\left[\sum_{t \in \mathcal{T}_m} \mathbf{1}\{G_{i,\sigma_m(i),t}\} \,\middle|\, \mathcal{H}_m\right].$$

□

Proof of Theorem 2. For each item i, let $X_{i,n}$ denote its n-th observed Bernoulli sample and define

$$\mathcal{E} := \bigcap_{i=1}^{L} \bigcap_{n=1}^{T+1} \left\{ \left| \frac{1}{n} \sum_{u=1}^{n} X_{i,u} - \mu_i \right| \le \mathrm{rad}(n) \right\}.$$

This is the same uniform confidence event used in the BC-SAR analysis. Under the same Bernoulli feedback, initialization, and confidence radius, Lemma 1 gives $\mathbb{P}(\mathcal{E}^c) \le 1/T$. Because $\widehat{\mu}_i^m$ is the empirical mean after N_i^m observations, the global event implies every batch-start event: $\mathcal{E} \subseteq \mathcal{E}_m$.

If $i \in B_m$ and

$$\mathbf{1}\{\mathcal{E}_m\} \sum_{t \in \mathcal{T}_m} \mathbf{1}\{G_{i,\sigma_m(i),t}\} > 0,$$

the UCB argument in Lemma 7 gives

$$\Delta_i \le \Delta_{i,\sigma_m(i)} \le 2\,\mathrm{rad}_i^m, \qquad N_i^m \le \frac{2\beta}{\Delta_i^2}.$$

Moreover,

$$\mathbf{1}\{\mathcal{E}_m\} \sum_{t \in \mathcal{T}_m} \mathbf{1}\{G_{i,\sigma_m(i),t}\} \le h_m \le \lambda N_i^m + 1,$$

since each round contributes at most one paired observation and $h_m = \lceil \lambda n_m \rceil$ with $n_m \le N_i^m$.

Fix a suboptimal item $i > K$ and list the batches satisfying the preceding positivity condition as $m_1 < \cdots < m_s$. For $\ell = 1, \ldots, s$, write

$$N_\ell := N_i^{m_\ell}, \qquad x_\ell := \mathbf{1}\{\mathcal{E}_{m_\ell}\} \sum_{t \in \mathcal{T}_{m_\ell}} \mathbf{1}\Big\{G_{i,\sigma_{m_\ell}(i),t}\Big\}.$$

Then

$$x_\ell \le (\lambda+1)N_\ell, \qquad N_{\ell+1} \ge N_\ell + x_\ell.$$

For $n > 0$ and $0 \le x \le (\lambda+1)n$,

$$\frac{x}{\sqrt{n}} \le (\sqrt{\lambda+2}+1)(\sqrt{n+x}-\sqrt{n}).$$

The square-root increments therefore telescope, and the final count satisfies

$$N_s + x_s \le (\lambda+2)N_s \le (\lambda+2)\frac{2\beta}{\Delta_i^2}.$$

Consequently,

$$\sum_{\ell=1}^{s} 2\,\mathrm{rad}_i^{m_\ell}\, x_\ell \le 2(\sqrt{\lambda+2}+1)\sqrt{\lambda+2}\,\frac{\beta}{\Delta_i} = C_\lambda \frac{\beta}{\Delta_i}.$$

Lemma 7 and iterated expectations give

$$\sum_m \mathbb{E}[\mathrm{Reg}_m\, \mathbf{1}\{\mathcal{E}_m\}] \le \sum_{i=K+1}^{L} \mathbb{E}\left[\sum_{m: i \in B_m} 2\,\mathrm{rad}_i^m\, \mathbf{1}\{\mathcal{E}_m\} \sum_{t \in \mathcal{T}_m} \mathbf{1}\big\{G_{i,\sigma_m(i),t}\big\}\right].$$

For all online batches, $\mathcal{E} \subseteq \mathcal{E}_m$ and the per-round regret is at most one, so

$$\mathbb{E}\left[\sum_m \mathrm{Reg}_m\right] \le \sum_m \mathbb{E}[\mathrm{Reg}_m\, \mathbf{1}\{\mathcal{E}_m\}] + T\mathbb{P}(\mathcal{E}^c).$$

Substituting the per-item radius bound above and using $T\mathbb{P}(\mathcal{E}^c) \le 1$ proves Theorem 2. □

A.5 Proof of Theorem 3: BC-UCB Batch-Complexity Upper Bound

The proof counts complete adaptive batches through the logarithmic observation-count potential

$$\Psi_m := \sum_{i=1}^{L} \log N_i^m.$$

The first-position rotation gives the following uniform increase in this potential. Its proof is deferred to Appendix A.7.

Lemma 8 (Log-potential growth)**.** *Every complete adaptive batch m satisfies*

$$\Psi_{m+1} - \Psi_m \ge \frac{\lambda}{4\lambda+6}.$$

Proof of Theorem 3. Let M_{comp} be the number of complete adaptive batches. The algorithm has at most one additional batch whose planned length is truncated by the horizon, so

$$M \le M_{\mathrm{comp}} + 1.$$

Let Z_T be the total number of item observations received during the T online rounds, and let N_i^{end} be the terminal observation count of item i. Initialization gives

$$\sum_{i=1}^{L} N_i^{\mathrm{end}} = L + Z_T.$$

Since $N_i^1 = 1$ for every item, $\Psi_1 = 0$. Since observation counts are nondecreasing, Jensen's inequality yields

$$\Psi_m \le \sum_{i=1}^{L} \log N_i^{\mathrm{end}} \le L \log\left(1 + \frac{Z_T}{L}\right)$$

at every batch endpoint. Each online round reveals at most K items, so $Z_T \le KT$. Summing Lemma 8 over all complete adaptive batches therefore gives

$$M_{\mathrm{comp}} \frac{\lambda}{4\lambda + 6} \le L \log\left(1 + \frac{KT}{L}\right).$$

Combining this inequality with $M \le M_{\mathrm{comp}} + 1$ proves Theorem 3. □

A.6 Proof of Theorem 4: Regret and Batch-Complexity Lower Bound

At $K = 1$, the learner recommends one item and always observes it, so the cascade model coincides with an L-item Bernoulli bandit. The proof adapts the finite-horizon adaptive-grid construction of Gao et al. Gao et al. (2019) to the initialization convention used here. For two instances that differ only in item i, the following lemma bounds the KL divergence between their interaction-history distributions in terms of the expected number of observations of i. The three auxiliary lemmas used below are proved in Appendix A.7.

Lemma 9 (Observation information)**.** *Consider Bernoulli bandit instances ν and ν' that differ only in item i, whose means are p and $p + \delta$, with $p, p + \delta \in [1/4, 1/2]$. Let $\mathbb{P}_\nu^s$ and $\mathbb{P}_{\nu'}^s$ denote their interaction-history distributions through online round s, and let $N_i(s)$ include the initial observation of item i. Then*

$$\mathrm{KL}(\mathbb{P}_\nu^s, \mathbb{P}_{\nu'}^s) = \mathbb{E}_\nu[N_i(s)]\,\mathrm{kl}(\nu_i, \nu_i') \le 6\delta^2 \mathbb{E}_\nu[N_i(s)].$$

Fix $\pi \in \Pi_1(L, M, T)$, and let $\pi_t \in [L]$ be its recommended item in online round t. At $K = 1$, its regret is

$$\sum_{t=1}^{T} (\mu_{i^\star} - \mu_{\pi_t}), \qquad i^\star \in \arg\max_i \mu_i.$$

The initial observations are excluded from the online horizon, regret, and batch count. Since the recommended item is observed in every online round,

$$\sum_{i=1}^{L} N_i(s) = L + s.$$

Pad its batch-endpoint sequence with empty batches when π uses fewer than M adaptive batches, and write

$$0 = t_0 \le t_1 \le \cdots \le t_M = T$$

without changing its observations, regret, or number of actual batches. Set

$$g := \left(1 + \frac{T}{L}\right)^{1/M}, \qquad s_j := \lfloor L g^j \rfloor - L, \qquad j = 0, \ldots, M.$$

Then $s_0 = 0$, $s_M = T$, and the assumed range of M gives $g \ge 2$. Let

$$J := \min\{j \in [M] : t_j \ge s_j\}, \qquad \mathcal{G}_j := \{J = j\}.$$

Since $t_M = s_M = T$, the events $\mathcal{G}_1, \ldots, \mathcal{G}_M$ form a partition. For $j \ge 2$, the event $\mathcal{G}_j$ implies $t_{j-1} < s_{j-1}$; the first batch endpoint is determined from the initial history when $j = 1$. Hence $\mathcal{G}_j$ is measurable from the initial history, the interaction history through online round s_{j-1}, and the policy's external randomness. On $\mathcal{G}_j$, the actions through round s_j were fixed before the feedback collected after round s_{j-1}.

Write $d = L - 1$, let $b = 1/4$, fix $c_0 = 1/128$, and define

$$\delta_j := \frac{c_0}{M} \sqrt{\frac{d}{L + s_{j-1}}}, \qquad j \in [M]. \tag{17}$$

These gaps are nonincreasing in j, and $\delta_j \le c_0$. Define a reference world P_M with means

$$\mu_L = b + \delta_M, \qquad \mu_i = b \quad (i < L).$$

For $j < M$ and $k \in [d]$, define $P_{j,k}$ by additionally raising item k to mean $b + \delta_j + \delta_M$. Finally set

$$p_j := \frac{1}{d}\sum_{k=1}^{d} P_{j,k}(\mathcal{G}_j) \quad (j < M), \qquad p_M := P_M(\mathcal{G}_M). \tag{18}$$

All means used below lie in $[1/4, 1/2]$.

The batch-endpoint partition satisfies the following two bounds.

Lemma 10 (Crossing-scale probability)**.** *The probabilities in* (18) *satisfy*

$$\sum_{j=1}^{M} p_j \ge \frac{31}{32}.$$

In particular, some $j \in [M]$ satisfies $p_j \ge 1/(2M)$.

Lemma 11 (Crossing-scale regret)**.** *If $p_j \ge 1/(2M)$, then*

$$\sup_{\nu \in \mathcal{B}^L_{\delta_j}} \mathbb{E}_\nu[\mathrm{Reg}(T;\pi)] \ge \frac{5}{32M}\,\delta_j s_j.$$

Proof of Theorem 4. Lemma 10 gives an index $j \in [M]$ with $p_j \ge 1/(2M)$. Lemma 11 then yields

$$\sup_{\nu \in \mathcal{B}^L_{\delta_j}} \mathbb{E}_\nu[\mathrm{Reg}(T;\pi)] \ge \frac{5}{32M}\,\delta_j s_j.$$

Since $g \ge 2$, $L \ge 2$, and $L + s_{j-1} = \lfloor L g^{j-1} \rfloor$,

$$\frac{s_j}{L + s_{j-1}} \ge \frac{g-1}{2}.$$

Multiplying the crossing-scale regret bound by δ_j, then using (17) and $d = L - 1 \ge L/2$, gives

$$\sup_{\Delta > 0} \Delta \sup_{\nu \in \mathcal{B}^L_\Delta} \mathbb{E}_\nu[\mathrm{Reg}(T;\pi)] \ge \frac{5c_0^2}{128}\, M^{-3} L(g-1).$$

This proves the fixed-policy testing bound with $c = 5c_0^2/128$.

Taking the infimum in this fixed-policy bound over $\pi \in \Pi_1(L, M, T)$ gives

$$\mathfrak{R}_1^{\mathrm{mm}}(L, M, T) \ge c M^{-3} L \left[\left(1 + \frac{T}{L}\right)^{1/M} - 1 \right].$$

For the batch-complexity statement, suppose an algorithmic guarantee holds uniformly over recommendation lengths K and satisfies $\mathfrak{R}_1^{\mathrm{mm}}(L, M, T) \le C(L-1)\log T$ at $K = 1$. The bound above requires

$$M^{-3} \left[\left(1 + \frac{T}{L}\right)^{1/M} - 1 \right] = O(\log T).$$

Write $H := \log(1 + T/L)$. Suppose $M \le aH/\log\log T$ for a sufficiently small constant $a > 0$. For all sufficiently large T, this assumption also gives $M \le H/\log 2 = \log_2(1 + T/L)$, so the required range condition holds. Then

$$\left(1 + \frac{T}{L}\right)^{1/M} \ge \exp\left(\frac{1}{a}\log\log T\right) = (\log T)^{1/a}.$$

Taking $a \le 1/5$ makes the bracketed term at least $\frac{1}{2}(\log T)^5$ for all sufficiently large T. Moreover, $H \le \log(1+T)$, so

$$M^3 \le \frac{a^3 H^3}{(\log\log T)^3} \le \frac{a^3(\log(1+T))^3}{(\log\log T)^3}.$$

Consequently,

$$\frac{M^{-3}\left[(1+T/L)^{1/M}-1\right]}{\log T} \to \infty,$$

contradicting the assumed $\mathcal{O}((L-1)\log T)$ upper bound on $\mathfrak{R}_1^{\mathrm{mm}}(L,M,T)$. Therefore

$$M = \Omega\left(\frac{\log(1+T/L)}{\log\log T}\right).$$

If L is fixed, or more generally $L = T^{o(1)}$, then $\log(1+T/L) = \Theta(\log T)$, which gives $M = \Omega(\log T/\log\log T)$. □

A.7 Proofs of Auxiliary Results

Auxiliary Results for Theorem 1

Lemma 12 (Pathwise executability of SAR-Update)**.** *Suppose an endpoint classification iteration starts with $r = K - |\mathcal{P}| \ge 1$ and $|\mathcal{A}| \ge r$. Let $\mathcal{R}_{\mathrm{new}} = \{i \in \mathcal{A} : U_i < \theta_r\}$ be the items rejected in this iteration, and let $\mathcal{P}_{\mathrm{new}}$ be the items accepted after the rejection step. Then θ_r is defined, $|\mathcal{A} \setminus \mathcal{R}_{\mathrm{new}}| \ge r$, and*

$$|\mathcal{A} \setminus (\mathcal{R}_{\mathrm{new}} \cup \mathcal{P}_{\mathrm{new}})| \ge K - |\mathcal{P} \cup \mathcal{P}_{\mathrm{new}}|.$$

Consequently, every successive iteration of SAR-Update has enough active items to fill its remaining positions.

Proof. Because θ_r is the r-th largest active LCB, at least r active items satisfy $L_i \ge \theta_r$. Since $U_i \ge L_i$, none of these items belongs to $\mathcal{R}_{\mathrm{new}}$, so $|\mathcal{A} \setminus \mathcal{R}_{\mathrm{new}}| \ge r$. The newly accepted items $\mathcal{P}_{\mathrm{new}}$ form a subset of $\mathcal{A} \setminus \mathcal{R}_{\mathrm{new}}$ with $|\mathcal{P}_{\mathrm{new}}| \le r$, including the case $\mathcal{P}_{\mathrm{new}} = \mathcal{A} \setminus \mathcal{R}_{\mathrm{new}}$ when $|\mathcal{A} \setminus \mathcal{R}_{\mathrm{new}}| = r$. Therefore

$$|\mathcal{A} \setminus (\mathcal{R}_{\mathrm{new}} \cup \mathcal{P}_{\mathrm{new}})| = |\mathcal{A} \setminus \mathcal{R}_{\mathrm{new}}| - |\mathcal{P}_{\mathrm{new}}| \ge r - |\mathcal{P}_{\mathrm{new}}| = K - |\mathcal{P} \cup \mathcal{P}_{\mathrm{new}}|.$$

□

Lemma 13 (Persistent first-position balance)**.** *During the pre-final phase, every item that remains active through round t satisfies*

$$Q_i(t) \ge \left\lfloor \frac{t}{L} \right\rfloor, \qquad N_i(t) \ge 1 + Q_i(t).$$

Moreover, in a $\mathcal{A}$-epoch of length ℓ and active-set size s, the local first-position counter satisfies

$$q_i \le \frac{\ell}{s} + 2.$$

Proof. All persistent first-position counters are zero initially. Each online round increments an active item attaining the minimum counter value, which keeps the range at most one. Acceptance or rejection at a batch endpoint replaces the active set by a subset, which cannot enlarge the range.

The first position is always observed, so $N_i(t) \ge 1 + Q_i(t)$. If fewer than K items have been accepted by round t, this also holds at every earlier online round because $\mathcal{P}$ only grows. We prove by induction that after round qL, every item that remains active has $Q_i(qL) \ge q$. The claim is immediate for $q = 0$. Suppose item i remains active through round $(q+1)L$. If $Q_i(qL) \ge q+1$, there is nothing to prove. Otherwise, $Q_i(qL) = q$, and at most $L-1$ active items can precede i at this minimum level, so the rule that selects the smallest Q-counter places i first within the next L rounds. This proves

$$Q_i(t) \ge \left\lfloor \frac{t}{L} \right\rfloor, \qquad N_i(t) \ge 1 + Q_i(t).$$

For the local first-position counter bound, consider an interval with fixed active set. Its initial counter values differ by at most one. An initial leveling phase contributes at most one extra placement to any item, and over any subsequent s placements, each item is placed at most once. Hence

$$q_i \le \left\lceil \frac{\ell}{s} \right\rceil + 1 \le \frac{\ell}{s} + 2.$$

□

Proof of Lemma 2. The initialization $\mathcal{P} = \varnothing, \mathcal{A} = [L], \mathcal{R} = \varnothing$ satisfies (5). Assume the invariant before an endpoint closure iteration. Lemma 12 shows that the rejection step and the subsequent acceptance step are well defined and leave enough active items to fill the remaining positions.

The active set contains every optimal item not yet accepted. Because $\mathcal{P}$ is a true prefix, these r items are $\{|\mathcal{P}| + 1, \dots, K\}$ and include item K. On $\mathcal{E}$, $L_K \le \mu_K$. For every $u > K$, $L_u \le \mu_u < \mu_K$. Thus at most $r - 1$ active LCBs are strictly above μ_K, which gives

$$\theta_r \le \mu_K.$$

Every $j \in S^\star \setminus \mathcal{P}$ satisfies

$$U_j \ge \mu_j \ge \mu_K \ge \theta_r,$$

so the rejection rule cannot reject it.

After rejection, suppose the empirical prefix $\{i_1, \dots, i_k\}$ satisfies the acceptance condition in Section 4. For every $u \le k < v$,

$$\mu_{i_u} \ge L_{i_u} > U_{i_v} \ge \mu_{i_v}.$$

This prefix is therefore the true top-k subset of the current active set. The true top-r active items are exactly $S^\star \setminus \mathcal{P}$, so any empirical prefix selected for acceptance with $k < r$ lies in $S^\star$ and follows the existing accepted prefix. If $|\mathcal{A}| = r$, then $\mathcal{A} = S^\star \setminus \mathcal{P}$ and can be accepted. After acceptance, the invariant remains valid by Lemma 12. The closure repeats these two steps with the same batch-endpoint confidence intervals. If two adjacent items have equal attraction probabilities, their LCB and UCB cannot satisfy a strict confidence separation on $\mathcal{E}$, so no empirical prefix selected for acceptance separates them. □

Lemma 14 (Middle-position quota feasibility)**.** *Fix a $\mathcal{A}$-epoch with $s := |\mathcal{A}|$ and $r := K - |\mathcal{P}|$. If $\rho > 2$, every round with $r \ge 2$ has at least $r - 1$ eligible active items for the middle positions besides the first-position item. Hence Algorithm 1 can always fill its middle positions.*

Proof. The case $r = 1$ has no middle positions. Assume $r \ge 2$ and consider the first round at which fewer than $r-1$ non-first-position items have unexhausted quota. At least $s-r+1$ non-first-position items have exhausted their quota. Choose a subset W of this size. Quota exhaustion gives

$$\sum_{i \in W} d_i \ge \rho(r-1) \sum_{i \in W} (q_i + 1). \tag{19}$$

At the beginning of the epoch and at the current time, the persistent first-position counters Q_i have range at most one. Since q_i is their difference, $q_i \le q_j + 2$ for all $i, j \in \mathcal{A}$. Also $|\mathcal{A} \setminus W| = r - 1$, so

$$\sum_{i \in \mathcal{A} \setminus W} q_i \le \frac{r-1}{|W|} \sum_{i \in W} q_i + 2(r-1). \tag{20}$$

When the first-position item lies outside W, set W receives at most $\min\{|W|, r-1\}$ middle-position placements. When it lies in W, the first item cannot also occupy a middle position, giving at most $\min\{|W| - 1, r - 1\}$. Therefore

$$\begin{aligned} \sum_{i \in W} d_i \le \min\{|W|, r-1\} \sum_{i \in \mathcal{A} \setminus W} q_i \\ + \min\{|W| - 1, r - 1\} \sum_{i \in W} q_i. \end{aligned} \tag{21}$$

If $|W| \le r-1$, substitute (20) into (21) and use $|W| \le r-1$. If $|W| \ge r$, both coefficients in (21) are at most $r-1$, and $(r-1)/|W| \le 1$. Both cases give

$$\sum_{i \in W} d_i \le 2(r-1) \sum_{i \in W} (q_i + 1).$$

Combining this inequality with (19) implies $\rho \le 2$, a contradiction. □

Proof of Lemma 4. Fix $h \in \{1, \dots, \ell_e\}$, and let q_i and d_i be the local counters after the first h rounds of epoch e. Before every middle-position placement, Algorithm 1 checks the quota in (1). After that placement, d_i increases by one, while the quota is nondecreasing in q_i. At the end of this prefix,

$$d_i \le \lceil \rho(r_e - 1)(q_i + 1) \rceil \le \rho(r_e - 1)(q_i + 1) + 1.$$

Thus

$$\sum_{t=t_e^{\mathrm{start}}}^{t_e^{\mathrm{start}}+h-1} \mathbf{1}\{i \in S_t\} = q_i + d_i \le C_{\rho, r_e}(q_i + 1).$$

Lemma 13 applied to this prefix gives $q_i \le h/s_e + 2$, which proves (7). □

Proof of Lemma 3. Fix a round t in batch m. The cascade loss decomposition in Theorem 5 gives

$$f(S^\star) - f(S_t) \le \sum_{i \in B_t} \Delta_{i,\sigma_t(i)} \mathbb{P}_t\big(G_{i,\sigma_t(i),t}\big) \le \sum_{i \in B_t} \Delta_{i,\sigma_t(i)}.$$

On $\mathcal{E}$, Lemma 2 gives $\mathcal{P}_m = \{1, \dots, |\mathcal{P}_m|\}$. Every missing matched optimal item $j = \sigma_t(i)$ therefore satisfies $|\mathcal{P}_m| < j \le K$. Hence

$$\Delta_{i,j} = \Delta_i + \sum_{k=j}^{K-1} \Delta_{k+1,k} \le \Delta_i + \sum_{k=|\mathcal{P}_m|+1}^{K-1} \Delta_{k+1,k}.$$

Summing this inequality over $i \in B_t$ proves (6). □

Proof of Lemma 5. Fix $m \in \{2, \dots, M-1\}$ such that item i remains in $\mathcal{A}_m$ and $r_m \ge 1$. Write

$$\varepsilon_m = \max_{j \in \mathcal{A}_m} \mathrm{rad}(N_j(t_{m-1})).$$

The active set contains all r_m items in $S^\star \setminus \mathcal{P}_m$. Thus its r_m-th largest LCB threshold is at least $\mu_K - 2\varepsilon_m$. Item i remains active after the rejection step, so

$$\mu_i + 2\varepsilon_m \ge U_i \ge \theta_{r_m} \ge \mu_K - 2\varepsilon_m.$$

Therefore

$$\Delta_i \le 4\varepsilon_m. \tag{22}$$

Suppose also that $|\mathcal{P}_m| < k < K$. If $\Delta_{k+1,k} > 4\varepsilon_m$, every item in the true active top-$(k - |\mathcal{P}_m|)$ prefix has LCB larger than the UCB of every item below that prefix. The acceptance rule would then accept through item k, which contradicts $|\mathcal{P}_m| < k$. Hence

$$\Delta_{k+1,k} \le 4\varepsilon_m. \tag{23}$$

Lemma 13 gives, for every $j \in \mathcal{A}_m$,

$$N_j(t_{m-1}) \ge 1 + \left\lfloor \frac{t_{m-1}}{L} \right\rfloor.$$

Consequently,

$$\varepsilon_m \le \sqrt{\frac{\beta}{2(1 + \lfloor t_{m-1}/L \rfloor)}}.$$

Equation (22) implies

$$t_{m-1} < \frac{8L\beta}{\Delta_i^2}.$$

The rounded geometric grid satisfies $t_m \le at_{m-1} + 1$ for $2 \le m \le M$, so

$$t_m \le at_{m-1} + 1 \le \frac{9aL\beta}{\Delta_i^2},$$

using $a, L, \beta \ge 1$ and $\Delta_i \le 1$.

When $|\mathcal{P}_m| < k < K$, equations (22) and (23) give

$$t_{m-1} < \frac{8L\beta}{\max\{\Delta_i, \Delta_{k+1,k}\}^2}.$$

The same overshoot argument yields

$$t_m \le \frac{9aL\beta}{\max\{\Delta_i, \Delta_{k+1,k}\}^2}.$$

If $\tau_{i,0} = t_1$, the first claim is immediate. Otherwise, $\tau_{i,0} = t_m$ for a batch satisfying the first set of conditions above, and $\tau_{i,0} - t_1 \le t_m$ proves (10). Similarly, if $\tau_{i,k} > t_1$, the batch ending at stopping endpoint $\tau_{i,k}$ satisfies both conditions above, so $\tau_{i,k} - t_1 \le t_m$ proves (11); the case $\tau_{i,k} = t_1$ is immediate. □

Proof of Lemma 6. For each $k \in \{0, \dots, K-1\}$, let $\tau_{1,k} < \cdots < \tau_{J_k,k}$ be the distinct stopping endpoints. Define

$$G_{q,k} := \{i > K : \tau_{i,k} = \tau_{q,k}\}, \qquad \mathcal{U}_{q,k} := \{i > K : \tau_{i,k} \ge \tau_{q,k}\}, \qquad n_{q,k} := |\mathcal{U}_{q,k}|.$$

Set $\mathcal{U}_{J_k+1,k} := \varnothing$, so that $n_{J_k+1,k} = 0$ and

$$n_{q+1,k} = n_{q,k} - |G_{q,k}|.$$

If $i \in G_{q,k}$, then every epoch contributing to its corresponding gap term has at least $n_{q,k}$ active suboptimal items. Lemma 5 therefore bounds the left-hand side of (12) by

$$9aL\beta C_{\rho,K}\left[\sum_q \frac{1}{K+n_{q,0}} \sum_{i \in G_{q,0}} \frac{1}{\Delta_i} + \sum_{k=1}^{K-1} \sum_q \frac{1}{K+n_{q,k}} \sum_{i \in G_{q,k}} \frac{\Delta_{k+1,k}}{\max\{\Delta_i, \Delta_{k+1,k}\}^2}\right].$$

Since $\Delta_i \ge \Delta_{K+1,K}$, each inverse-gap expression in this display is at most its value at $i = K+1$. Moreover,

$$\sum_q \frac{|G_{q,k}|}{K+n_{q,k}} \le \sum_q \log \frac{K+n_{q,k}}{K+n_{q+1,k}} = \log \frac{K+n_{1,k}}{K} \le \log \frac{L}{K}.$$

Substitution proves (12). □

Proof of Proposition 2. If $M = 1$, then $\lceil a \rceil \ge T$, and the right-hand side of (14) dominates $T \ge \mathrm{Reg}_{\mathrm{final}}$. Assume henceforth that $M \ge 2$. On every sample path, let B_{final} be the selected suboptimal items in the final recommended set and let O_{final} be the omitted optimal items. Pair them by a fixed deterministic rule to obtain a bijection $\sigma_{\mathrm{final}} : B_{\mathrm{final}} \to O_{\mathrm{final}}$. Then $|B_{\mathrm{final}}| = |O_{\mathrm{final}}| \le K$. Lemma 3 bounds the final-batch one-round regret by

$$\sum_{i \in B_{\mathrm{final}}} \Delta_{i,\sigma_{\mathrm{final}}(i)}.$$

On $\mathcal{E}$, every accepted item is optimal, every optimal item not accepted remains active, and every suboptimal item remains active until it is rejected. Therefore, if $i \in B_{\mathrm{final}}$ and $j = \sigma_{\mathrm{final}}(i)$, then i and j are both active at the beginning of the final batch and $\widehat{\mu}_i(t_{M-1}) \ge \widehat{\mu}_j(t_{M-1})$. By Lemma 13 and initialization, both items have at least

$$n_\star := 1 + \left\lfloor \frac{t_{M-1}}{L} \right\rfloor \ge \frac{t_{M-1}}{L}$$

observations.

For a fixed item v, let $Y_u = X_{v,u} - \mu_v$. Hoeffding's lemma implies that, for every $\lambda \in \mathbb{R}$,

$$\exp\left(\lambda \sum_{u=1}^{n} Y_u - \frac{n\lambda^2}{8}\right)$$

is a nonnegative supermartingale in the observation index n. Ville's inequality with $\lambda = 4\varepsilon$ therefore gives, for every $n_0 \geq 1$ and $\varepsilon > 0$,

$$\mathbb{P}\left(\exists n \geq n_0 : \frac{1}{n}\sum_{u=1}^{n} Y_u \geq \varepsilon\right) \leq \exp(-2n_0\varepsilon^2).$$

For $i \in B_{\text{final}}$ and $j = \sigma_{\text{final}}(i)$, the empirical inversion and $\Delta_{i,j} = \mu_j - \mu_i$ imply that either item i has an upper-tail deviation of at least $\Delta_{i,j}/2$, or item j has a lower-tail deviation of at least $\Delta_{i,j}/2$. Applying the time-uniform bound to both tails gives

$$\mathbb{P}\big(i \in B_{\text{final}},\ \sigma_{\text{final}}(i) = j,\ \mathcal{E}\big) \leq 2\exp\left(-\frac{t_{M-1}\Delta_{i,j}^2}{2L}\right). \tag{24}$$

Consequently,

$$\mathbb{E}[\mathrm{Reg}_{\text{final}}\,\mathbf{1}\{\mathcal{E}\}] \leq 2(T - t_{M-1}) \sum_{i>K}\sum_{j\leq K} \Delta_{i,j} \exp\left(-\frac{t_{M-1}\Delta_{i,j}^2}{2L}\right).$$

Group possible pair gaps according to

$$2^h\Delta_{K+1,K} \leq \Delta_{i,j} < 2^{h+1}\Delta_{K+1,K}, \qquad h = 0, 1, \ldots,$$

and retain only nonempty groups with $2^h\Delta_{K+1,K} \leq 1$. Let n_h^{pair} be the number of possible pairs in group h, and define

$$Z_h := \sum_{\substack{i\in B_{\text{final}}:\\ 2^h\Delta_{K+1,K} \leq \Delta_{i,\sigma_{\text{final}}(i)} < 2^{h+1}\Delta_{K+1,K}}} \Delta_{i,\sigma_{\text{final}}(i)}.$$

The matching capacity and (24) give

$$Z_h \leq K\,2^{h+1}\Delta_{K+1,K}, \qquad \mathbb{E}[Z_h\mathbf{1}\{\mathcal{E}\}] \leq 2n_h^{\text{pair}}\,2^{h+1}\Delta_{K+1,K}\exp\left(-\frac{t_{M-1}4^h\Delta_{K+1,K}^2}{2L}\right).$$

Thus

$$\mathbb{E}[Z_h\mathbf{1}\{\mathcal{E}\}] \leq K\,2^{h+1}\Delta_{K+1,K}\min\left\{1, \frac{2n_h^{\text{pair}}}{K}\exp\left(-\frac{t_{M-1}4^h\Delta_{K+1,K}^2}{2L}\right)\right\}.$$

For $B_0, c > 0$ and $z \geq 0$,

$$\big((a-1)z+1\big)\min\{1, B_0e^{-cz}\} \leq 1 + \frac{a-1}{c}\big(1 + \log_+ B_0\big), \qquad \log_+ B_0 := \max\{0, \log B_0\},$$

as follows by considering $cz \leq \log_+ B_0$ and its complement. The overshoot relation $t_M \leq at_{M-1} + 1$ gives $T - t_{M-1} \leq (a-1)t_{M-1} + 1$. Taking $B_0 = 2n_h^{\text{pair}}/K$, $c = 4^h\Delta_{K+1,K}^2/(2L)$, and using $n_h^{\text{pair}} \leq K(L-K) \leq KL$, the contribution of group h is at most

$$K\,2^{h+1}\Delta_{K+1,K}\left[1 + \frac{2(a-1)L}{4^h\Delta_{K+1,K}^2}\big(1 + \log(2L)\big)\right].$$

The dyadic sums satisfy

$$\sum_h 2^{h+1}\Delta_{K+1,K} \leq 4, \qquad \sum_h \frac{2^{h+1}\Delta_{K+1,K}}{4^h\Delta_{K+1,K}^2} \leq \frac{4}{\Delta_{K+1,K}}.$$

Summing over all nonempty groups proves (14). □

Auxiliary Lemmas for the BC-UCB Batch-Complexity Upper Bound

Proof of Lemma 8. For a complete adaptive batch m, let

$$n_m := \min_{i \in S_m} N_i^m, \qquad h_m := \lceil \lambda n_m \rceil.$$

In internal round $\tau = 0, \dots, h_m - 1$, let e_τ be the first-position item and let $q_i(\tau)$ count the earlier first-position assignments to item i in this batch. Let $Y_i(\tau)$ count its earlier non-first-position observations in the same batch. Choose $j \in S_m$ with $N_j^m = n_m$. Before internal round τ,

$$N_j^m + q_j(\tau) \le n_m + \tau.$$

The first-position rule minimizes $N_i^m + q_i(\tau)$ over $i \in S_m$, so

$$N_{e_\tau}^m + q_{e_\tau}(\tau) \le n_m + \tau.$$

Since $Y_{e_\tau}(\tau) \le \tau$, the observation count of e_τ immediately before its first-position observation is at most

$$N_{e_\tau}^m + q_{e_\tau}(\tau) + Y_{e_\tau}(\tau) \le n_m + 2\tau \le n_m + 2h_m.$$

This observation increases the log potential by at least

$$\log\left(1 + \frac{1}{n_m + 2h_m}\right).$$

The other observations can only increase the potential further, and hence

$$\Psi_{m+1} - \Psi_m \ge h_m \log\left(1 + \frac{1}{n_m + 2h_m}\right).$$

The batch length satisfies $h_m \ge \lambda n_m$ and $h_m \le \lambda n_m + 1$. Initialization gives $n_m \ge 1$, so

$$n_m + 2h_m \le (2\lambda + 3) n_m.$$

Using $\log(1 + 1/x) \ge 1/(2x)$ for $x \ge 1$, we obtain

$$\Psi_{m+1} - \Psi_m \ge \lambda n_m \frac{1}{2(2\lambda + 3) n_m} = \frac{\lambda}{4\lambda + 6}.$$

□

Auxiliary Lemmas for the Problem-Class Lower Bound

Proof of Lemma 9. For $p, q \in [1/4, 1/2]$,

$$\mathrm{kl}(\mathrm{Ber}(p), \mathrm{Ber}(q)) \le \frac{(p-q)^2}{q(1-q)} \le 6(p-q)^2.$$

If two instances differ only on item i, the chain rule for their interaction histories through online round s gives

$$\mathrm{KL}(\mathbb{P}_\nu^s, \mathbb{P}_{\nu'}^s) = \mathbb{E}_\nu[N_i(s)]\ \mathrm{kl}(\nu_i, \nu_i').$$

The count $N_i(s)$ includes the initial observation. Actions contribute no KL term because the policy kernel is identical under the two instances. Combining the identity with the Bernoulli KL bound proves the lemma. □

Proof of Lemma 10. For $j < M$, the instances P_M and $P_{j,k}$ differ only on item k, by $\delta_j + \delta_M \le 2\delta_j$. Lemma 9, Pinsker's inequality, and concavity of the square root give

$$\begin{aligned}\frac{1}{d}\sum_{k=1}^{d} \mathrm{TV}\left(P_M^{s_{j-1}}, P_{j,k}^{s_{j-1}}\right) &\le \sqrt{\frac{12\delta_j^2}{d}\sum_{k=1}^{d}\mathbb{E}_{P_M}[N_k(s_{j-1})]} \\ &\le \frac{\sqrt{12}\, c_0}{M} \le \frac{1}{32M},\end{aligned}$$

where $\sum_{k=1}^{d} N_k(s_{j-1}) \le L + s_{j-1}$ includes the initial observations. The event $\mathcal{G}_j$ is measurable from this history, so

$$|P_M(\mathcal{G}_j) - p_j| \le \frac{1}{32M} \qquad (j < M).$$

Since $\mathcal{G}_1, \dots, \mathcal{G}_M$ form a partition and $p_M = P_M(\mathcal{G}_M)$,

$$\sum_{j=1}^{M} p_j \ge \sum_{j=1}^{M} P_M(\mathcal{G}_j) - \frac{M-1}{32M} \ge \frac{31}{32}.$$

At least one index therefore satisfies $p_j \ge 31/(32M) > 1/(2M)$. □

Proof of Lemma 11. Fix $j \in [M]$ with $p_j \ge 1/(2M)$. If $j < M$, fix $k \in [d]$ and use $P_{j,k}$ as the center world. If $j = M$, use P_M as the center and set $k = L$. Denote the center by Q_k. For each $\ell \ne k$, form Q_ℓ by increasing item ℓ's center mean by $3\delta_j$. Item ℓ is uniquely optimal under Q_ℓ, every wrong recommendation incurs regret of at least δ_j, and each Q_ℓ belongs to $\mathcal{B}^L_{\delta_j}$.

The center and Q_ℓ differ only on item ℓ. Applying Lemma 9, Pinsker's inequality, and concavity of the square root gives

$$\frac{1}{d}\sum_{\ell \ne k} \mathrm{TV}\big(Q_k^{s_{j-1}}, Q_\ell^{s_{j-1}}\big) \le \frac{\sqrt{27}\, c_0}{M} \le \frac{1}{16M}.$$

For any two measures P, Q and event A,

$$\int_A \min\{dP, dQ\} \ge P(A) - \frac{3}{2}\,\mathrm{TV}(P, Q).$$

Indeed,

$$\int_A \min\{dP, dQ\} = \frac{P(A) + Q(A) - \int_A |dP - dQ|}{2},$$

while $Q(A) \ge P(A) - \mathrm{TV}(P, Q)$ and $\int_A |dP - dQ| \le 2\,\mathrm{TV}(P, Q)$.

At each online round t, the pointwise star inequality centered at k gives

$$\frac{1}{L}\sum_{\ell=1}^{L} Q_\ell(\pi_t \ne \ell) \ge \frac{1}{L}\sum_{\ell \ne k} \int \min\{dQ_k^t, dQ_\ell^t\}.$$

If the decision is k, each overlap is bounded by the corresponding Q_ℓ-density. If the decision is $a \ne k$, the overlap on edge a is bounded by the Q_k-density and every other overlap by its own Q_ℓ-density. Every wrong recommendation under Q_ℓ costs at least δ_j, and the maximum regret over the L worlds is at least their average regret. Restricting the overlap integrals to $\mathcal{G}_j$ can only decrease the resulting lower bound. On $\mathcal{G}_j$, the policy fixes its decisions through round s_j using the history available at the start of the crossing batch, before round s_{j-1}, together with its external randomness. Data processing and the overlap inequality therefore yield

$$\begin{aligned}\sup_{\nu \in \mathcal{B}^L_{\delta_j}} \mathbb{E}_\nu[\mathrm{Reg}(T;\pi)] &\ge \delta_j s_j \frac{1}{L}\sum_{\ell \ne k} \int_{\mathcal{G}_j} \min\{dQ_k^{s_{j-1}}, dQ_\ell^{s_{j-1}}\} \\ &\ge \delta_j s_j \left[\frac{1}{2} Q_k(\mathcal{G}_j) - \frac{3}{32M}\right].\end{aligned}$$

For $j < M$, average this inequality over $k \in [d]$. For $j = M$, $Q_k(\mathcal{G}_j) = p_M$. Since $p_j \ge 1/(2M)$, both cases give

$$\sup_{\nu \in \mathcal{B}^L_{\delta_j}} \mathbb{E}_\nu[\mathrm{Reg}(T;\pi)] \ge \frac{5}{32M}\,\delta_j s_j.$$

□

B Experimental Protocol and Supporting Results

B.1 Unified Experimental Protocol

Every reported configuration uses 20 independent runs, and paired comparisons use matched seeds and item orderings. The main synthetic comparisons and the baseline batch-endpoint placement diagnostic use the same shared random item permutations. Each item receives one independent initial observation before the evaluated horizon. The initialization phase contributes to the empirical observation counts but is excluded from T, cumulative regret, click-rate evaluation, and the reported online batch count. All runs are retained when computing means and standard deviations. The main synthetic schedule results use population standard deviations; the layered mechanism, sensitivity, and MovieLens results use sample standard deviations. When confidence intervals are reported, unpaired pointwise intervals use the normal approximation, while paired differences use two-sided Student-t intervals.

Algorithms plan complete recommended lists at the beginning of each batch and update their statistics only at batch endpoints. The reported batch count is the number of online batches and excludes initialization. Cumulative regret is evaluated against the optimal recommended set under the synthetic attraction probabilities.

For MovieLens 1M, we retain all 6,040 user profiles and select the L movies with the most observed ratings, breaking ties by increasing movie ID. The item pools with $L \in \{256, 2048\}$ are nested prefixes of this ordering. We set $W_{u,i} = 1$ when user u rated movie i above three, and set $W_{u,i} = 0$ for ratings of three or below and for unrated pairs.

Within each run, we apply one random permutation to the item-pool columns and share it across the compared algorithms. For initialization, we independently sample one user uniformly for each item and reveal that item's binary outcome. For each online round, we sample a user independently and uniformly with replacement and reveal the corresponding row along the recommended list through the first click, or the entire list if no click occurs. Paired methods share the initialization draws and the online user sequence. The population reward of a recommended set S is its row coverage

$$F_L(S) = \frac{1}{6040} \sum_{u=1}^{6040} \mathbf{1}\{\exists i \in S : W_{u,i} = 1\}.$$

We construct the greedy maximum-coverage set S_L^{g} by adding the movie that covers the largest number of previously uncovered users at each of K steps, breaking ties by its position in the original item-pool ordering. We compute this reference before the random column permutation and map its items to the permuted indices for each run. For $K = 4$, its coverage is $F_L(S_L^{\mathrm{g}}) = 4830/6040$ for both item-pool sizes. Let $Y_s \in \{0, 1\}$ denote the click reward in online round s. For each item-pool size L, we report realized cumulative regret against the greedy reference:

$$\mathrm{Reg}(t) = tF_L(S_L^{\mathrm{g}}) - \sum_{s=1}^{t} Y_s.$$

Its expectation equals the cumulative population-reward gap to the greedy reference.

The supplementary code package provides the configurations, run commands, and per-seed values needed to reproduce the reported results.

B.2 Numerical Companions to Main Results

Table 4 reports the complete static- and adaptive-grid results shown in Figure 2. It gives mean cumulative regret at T with population standard deviations and the online batch count, prescribed on static-grid rows and realized on adaptive-grid rows. The BC-SAR/UCB column reports BC-SAR on static-grid rows and BC-UCB on adaptive-grid rows.

Table 2 reports every parameter-scaling configuration. Table 5 gives cumulative regret at T and online batch count for the two item-pool sizes shown in Figure 3. The regret column reports $\mathrm{Reg}(T)$ as mean $\pm$ sample standard deviation over 20 paired runs. Online batch counts are prescribed on static-grid rows and realized on adaptive-grid rows.

The results underlying Figure 4 are stored as a machine-readable table in the supplementary artifacts.

Table 4: Complete synthetic results under static and adaptive grids.

Grid	K	BC-SAR/UCB	CTS	TS-Cascade	CascadeUCB1	CascadeKL-UCB	Batches
$a = 2$	4	2996 ± 388	5169 ± 299	15432 ± 2414	27787 ± 5366	25387 ± 4631	18.0
$a = 2$	8	1223 ± 108	1593 ± 103	7100 ± 1001	14596 ± 2529	14054 ± 2163	18.0
$a = 1.5$	4	2802 ± 306	4296 ± 272	14352 ± 2277	28220 ± 3376	29006 ± 4155	31.0
$a = 1.5$	8	1129 ± 109	1273 ± 77	7175 ± 950	14473 ± 1495	13617 ± 1392	31.0
$\lambda = 4$	4	2339 ± 252	3571 ± 210	5743 ± 282	22351 ± 2777	19507 ± 2587	342.9
$\lambda = 4$	8	698 ± 100	1111 ± 50	2249 ± 114	9109 ± 766	7752 ± 661	288.6
$\lambda = 1$	4	1656 ± 248	3548 ± 142	5306 ± 306	22506 ± 2046	19219 ± 2167	862.1
$\lambda = 1$	8	496 ± 99	1073 ± 71	2051 ± 73	8357 ± 766	6774 ± 845	790.4

Table 5: MovieLens cumulative regret against the greedy maximum-coverage reference at $T = 100{,}000$ and $K = 4$. Values are means $\pm$ sample SD over 20 runs.

L	Grid	Algorithm	Regret	Online batches
256	static	BC-SAR	3533.7 ± 722.0	17.0 ± 0.0
256	static	Batched CTS	3797.5 ± 424.3	17.0 ± 0.0
256	adaptive	BC-UCB	1994.5 ± 537.6	631.2 ± 34.4
256	adaptive	Batched CTS	2771.5 ± 265.3	631.2 ± 34.4
2048	static	BC-SAR	9006.4 ± 878.9	17.0 ± 0.0
2048	static	Batched CTS	13571.7 ± 627.9	17.0 ± 0.0
2048	adaptive	BC-UCB	4185.7 ± 504.7	2867.7 ± 215.5
2048	adaptive	Batched CTS	11321.3 ± 479.5	2867.7 ± 215.5

B.3 BC-SAR Mechanism Ablations

We test how first-position balancing, middle-position quotas, acceptance, and rejection affect BC-SAR on layered instances.

The BC-SAR ablations use $L = 64$, $K = 4$, $T = 50{,}000$, and the single-strong and low-attraction profiles in Table 6; the table also lists the high-attraction profile used in the BC-UCB ablations. The profiles are specified before the shared random item permutation, and each table entry gives an attraction probability and its number of items. For each profile, all configurations share 20 seeds, item permutations, initialization samples, and round-by-item Bernoulli outcomes. These seeds are independent of the preliminary runs used to choose the profiles. The default controls are $a = 2$ and $\rho = 7/3$.

NoQuota removes the middle-position quota; RandomFirst chooses the first active item uniformly at random; RandomFirst-NoQuota applies both changes. NoAccept retains rejection but disables acceptance and the accepted suffix. NoReject retains confidence-based acceptance but disables rejection. All six configurations use the same static grid and parameter settings. Table 7 reports cumulative regret at T as mean $\pm$ sample standard deviation over 20 paired runs on the single-strong and low-attraction profiles. The intervals are paired pointwise 95% confidence intervals for each variant minus Full, and both profiles use the same static grid.

Disabling either acceptance or rejection increases mean regret on both profiles. The full method records at least one confidence-based acceptance in all 20 runs on each profile. RandomFirst raises mean regret on the single-strong profile, while its paired interval includes zero on the low-attraction profile. Removing the quota lowers mean regret in both cases, with or without first-position balancing. The quota bounds repeated recommendations in the analysis (Lemma 4); here, its restriction on greedy middle-position choices carries an exploration cost.

Table 8 shows the resulting allocation of suboptimal-item recommendations and observations over 20 paired runs. Within each run, the counts aggregate over the 60 suboptimal items; the table reports their means in thousands, with observations counted after initialization.

On the single-strong profile, NoQuota lowers regret while increasing both counts, illustrating that aggregate exposure alone does not determine regret on a layered instance. On the low-attraction profile, it lowers regret and both exposure counts. One RandomFirst run on the single-strong profile rejects an optimal item; the remaining mechanism runs have no incorrect acceptance or rejection. All configurations complete every run without quota overrides.

Table 6: Layered attraction profiles.

Profile	Layer 1	Layer 2	Layer 3	Layer 4
Single strong	0.65×1	0.25×3	0.20×12	0.06×48
Low attraction	0.25×2	0.16×2	0.12×12	0.04×48
High attraction	0.80×2	0.40×2	0.35×12	0.20×48

Table 7: BC-SAR mechanism ablations on layered profiles.

Variant	Single strong: regret	95% CI	Low: regret	95% CI
Full	519 ± 90	–	1838 ± 287	–
NoQuota	406 ± 76	$[-167, -59]$	1139 ± 159	$[-855, -542]$
RandomFirst	629 ± 130	$[33, 187]$	1792 ± 232	$[-242, 151]$
RandomFirst-NoQuota	388 ± 91	$[-188, -74]$	1188 ± 283	$[-820, -479]$
NoAccept	1679 ± 448	$[942, 1378]$	2361 ± 508	$[340, 706]$
NoReject	2907 ± 160	$[2303, 2473]$	4252 ± 596	$[2084, 2744]$

B.4 BC-UCB Mechanism Ablations

We test the roles of batch-start observation counts in first-position rotation and of the minimum selected-item observation count in the batch-length rule.

The experiments use the single-strong and high-attraction profiles in Table 6, with $L = 64$, $K = 4$, $T = 50{,}000$, and $\lambda = 1$ for the full method. Within each profile, all variants share 20 seeds, item permutations, initialization samples, and round-by-item Bernoulli outcomes. NoRotate repeats the UCB-sorted list throughout each batch. LocalQ balances first-position assignments using only the counts accumulated within the current batch, omitting batch-start observation counts from the rotation priority. MeanCount retains the full rotation rule and replaces the minimum selected-item observation count in the batch-length rule with the arithmetic mean of the selected items' observation counts.

The first four rows of Table 9 use each variant's own adaptive batch endpoints. The final two rows replay the full method's batch endpoints for each seed, comparing list construction under the same sequence of batch endpoints. The table reports cumulative regret at T as mean $\pm$ sample standard deviation over 20 paired runs and gives M as the mean realized online batch count.

The full method has the lowest mean regret among these variants on both profiles. On the high-attraction profile, removing rotation increases the mean realized online batch count from 280.1 to 3016.8, and using only within-batch counts increases it to 1155.2. MeanCount reduces the number of updates but increases regret on both profiles. Under the full method's replayed batch endpoints, NoRotate and LocalQ incur substantially higher regret, supporting the use of batch-start observation counts in list construction after controlling for the batch endpoints.

B.5 Parameter Sensitivity

We measure how the quota ratio and the two batch-control parameters affect regret, observation allocation, and update frequency. Each experiment varies one control while retaining the default values of all other parameters. Within each experimental setting, all parameter values share the same 20 seeds.

BC-SAR Quota Ratio

The middle-position quota controls repeated recommendations of active items in the regret analysis (Lemma 4).

We vary ρ to examine its effect on regret and observation allocation, using a homogeneous-gap instance with $L = 128$, $K = 4$, $T = 200{,}000$, $\mu = 0.20$, $\Delta = 0.075$, and static-grid growth factor $a = 2$. All six settings use the same 20 seeds and the initialization protocol in Appendix B.1. Table 10 reports cumulative regret at T as mean $\pm$ standard deviation over 20 runs, together with

Table 8: BC-SAR allocation diagnostics.

Variant	Single strong: recs.	Obs.	Low: recs.	Obs.
Full	26.7	22.5	41.1	37.3
NoQuota	28.3	25.8	34.7	32.9
RandomFirst	28.6	24.3	39.6	35.7
RandomFirst-NoQuota	26.1	23.8	34.6	32.3
NoAccept	26.3	19.1	39.9	33.0
NoReject	88.8	78.3	73.2	68.4

Table 9: BC-UCB mechanism ablations on layered profiles.

Variant	Single strong: regret	M	High: regret	M
Full	164 ± 128	225.2	37 ± 29	280.1
NoRotate	359 ± 175	444.3	154 ± 35	3016.8
LocalQ	213 ± 109	288.6	66 ± 28	1155.2
MeanCount	1766 ± 390	57.8	390 ± 53	50.0
NoRotate, replay	2295 ± 1046	225.2	386 ± 82	280.1
LocalQ, replay	1432 ± 772	225.2	221 ± 107	280.1

suboptimal-item recommendation and post-initialization observation counts. The two allocation measures aggregate over the 124 suboptimal items, and every setting uses 18 online batches.

Regret and both allocation measures vary nonmonotonically with ρ. Among the tested values, $\rho = 12$ gives the lowest mean regret and the fewest mean suboptimal-item recommendations and observations. The quota changes exposure and rejection in these runs. Confidence-based acceptance does not activate.

BATCH-CONTROL PARAMETERS

We test how a and λ change regret and update frequency under the static and adaptive grids.

Both sweeps use $L = 64$, $K = 4$, $T = 50{,}000$, and the initialization protocol in Appendix B.1. The BC-SAR sweep uses the single-strong and low-attraction profiles, while the BC-UCB sweep uses the single-strong and high-attraction profiles. Table 11 varies the static-grid growth factor a for BC-SAR, with M denoting the prescribed online batch count. Table 12 varies the adaptive batch parameter λ for BC-UCB, with M denoting the mean realized online batch count. Both tables report cumulative regret at T as mean $\pm$ sample standard deviation over 20 paired runs.

For BC-SAR, increasing a from 1.5 to 4 reduces the prescribed online batch count from 27 to 8 and raises mean regret on both profiles. For BC-UCB, increasing λ reduces the mean realized online batch count throughout the tested grid, while mean regret varies nonmonotonically. On the high-attraction profile, $\lambda = 4$ yields mean regret 36 with 117.3 realized online batches, compared with 37 and 280.1 at $\lambda = 1$.

Section 6.4 complements these sweeps by varying the static-grid batch count M for BC-SAR and λ for BC-UCB on the main synthetic settings.

B.6 BASELINE BATCH-ENDPOINT PLACEMENT DIAGNOSTIC

We examine how batch-endpoint placement affects baseline regret at a matched realized online batch count.

Each baseline is evaluated under replayed BC-UCB batch endpoints and a geometric grid with the same realized online batch count. We set $L = 256$, $T = 200{,}000$, and $K \in \{4, 8\}$, with attraction probabilities decreasing uniformly from 0.35 to 0.15 and reference parameters $\lambda \in \{4, 1\}$. For each seed, let M be the realized online batch count of the reference BC-UCB run. Starting from $\lfloor T^{m/M} \rfloor$, we adjust the integer batch endpoints to be strictly increasing, reserve at least one round for each remaining batch, and set the final batch endpoint to T. All four baselines share this grid and retain the main synthetic comparison's item permutations, initialization samples, and algorithm random seeds. The geometric grid uses the realized online batch count of the completed adaptive run, giving a retrospective comparison of batch-endpoint placement.

Table 10: BC-SAR quota-ratio sensitivity.

ρ	Regret at T	Subopt. recs. (10^3)	Subopt. obs. (10^3)
3	6929 ± 1504	160.7 ± 33.8	133.6 ± 26.8
4	5827 ± 1347	135.3 ± 31.2	113.1 ± 25.6
6	5868 ± 1929	137.0 ± 43.6	115.2 ± 35.6
8	6003 ± 1667	141.0 ± 37.8	119.6 ± 30.9
12	5207 ± 1538	123.9 ± 34.3	106.5 ± 27.5
16	5524 ± 2692	132.3 ± 62.5	113.4 ± 51.4

Table 11: BC-SAR static-grid growth-factor sensitivity.

a	Single strong: regret	M	Low: regret	M
1.5	374 ± 60	27.0	1430 ± 267	27.0
2	519 ± 90	16.0	1838 ± 287	16.0
4	828 ± 103	8.0	2929 ± 697	8.0

Table 13 reports the replayed and geometric baseline results over 20 paired runs using the main synthetic comparison's item permutations. The regret columns show means, and BC-UCB is the adaptive reference for both sequences of batch endpoints.

The geometric grid lowers mean regret in 15 of the 16 comparisons, showing that batch-endpoint placement can affect baseline performance even when the online batch count is unchanged. The paired 95% confidence interval for geometric minus replayed regret lies below zero in 10 comparisons and includes zero in the other six. Every geometric-grid baseline has higher mean regret than the corresponding BC-UCB reference.

Reproducibility artifacts. The supplementary package includes the simulation code, per-seed records, aggregate tables, checkpoint and batch-endpoint trajectories, batch-endpoint vectors, manifests, source hashes, and deterministic figure-generation scripts.

Table 12: BC-UCB adaptive batch-parameter sensitivity.

λ	Single strong: regret	M	High: regret	M
0.25	161 ± 149	571.0	119 ± 361	615.2
0.5	242 ± 204	374.1	32 ± 26	403.1
1	164 ± 128	225.2	37 ± 29	280.1
2	214 ± 141	141.3	42 ± 27	183.2
4	293 ± 145	91.7	36 ± 20	117.3
8	380 ± 151	63.6	52 ± 20	80.5
16	522 ± 124	44.6	71 ± 18	62.4

Table 13: Baseline batch-endpoint placement diagnostic.

K	λ	Mean online batches	Baseline	Replayed	Geometric	BC-UCB
4	4	342.9	CTS	3570.7	3460.9	2338.7
4	4	342.9	TS-Cascade	5742.6	4855.3	
4	4	342.9	CascadeUCB1	22350.6	21744.8	
4	4	342.9	CascadeKL-UCB	19506.8	18691.4	
8	4	288.6	CTS	1110.9	1081.1	698.1
8	4	288.6	TS-Cascade	2249.0	2010.3	
8	4	288.6	CascadeUCB1	9108.8	7996.3	
8	4	288.6	CascadeKL-UCB	7752.2	6331.8	
4	1	862.1	CTS	3548.3	3471.0	1655.9
4	1	862.1	TS-Cascade	5306.1	4537.4	
4	1	862.1	CascadeUCB1	22506.1	16312.9	
4	1	862.1	CascadeKL-UCB	19218.7	12789.1	
8	1	790.4	CTS	1073.4	1088.1	495.6
8	1	790.4	TS-Cascade	2051.1	1850.0	
8	1	790.4	CascadeUCB1	8357.0	5824.8	
8	1	790.4	CascadeKL-UCB	6774.3	4294.3	