Incremental Submodular Maximization: Better Than Greedy

Marcin Bienkowski
University of Wrocław, Poland

Joakim Blikstad
CWI, Amsterdam, The Netherlands

Jarosław Byrka
University of Wrocław, Poland

Martín Costa
University of Warwick, UK

Yann Disser
TU Darmstadt, Germany

Annette Lutz
TU Darmstadt, Germany

Abstract

We consider submodular maximization under increasing cardinality constraint and ask for a good incremental solution, i.e., an ordering of the ground set such that each prefix of the ordering yields a good solution for its respective cardinality. A classical result in this setting is that the greedy algorithm achieves a competitive ratio, i.e., an approximation guarantee across all cardinalities, of $\mathrm{e}/(\mathrm{e}-1) \approx 1.582$. No better general guarantee was previously known. We present an adaptive scaling algorithm achieving a competitive ratio of 1.373. We complement our result by a lower bound of 1.25 on the best possible deterministic competitive ratio for incremental submodular maximization.

2012 ACM Subject Classification Theory of computation → Submodular optimization and polymatroids; Theory of computation → Online algorithms

Keywords and phrases Submodular maximization, incremental optimization, competitive analysis

Digital Object Identifier 10.4230/LIPIcs.ESA.2026.134

Related Version *Full Version*: `https://arxiv.org/abs/2606.28558` [4]

Funding This research has been initiated at the AlgUW Workshop (Bedlewo 09.2025), supported by the Excellence Initiative – Research University (IDUB) funds of the University of Warsaw.
Marcin Bienkowski: Supported by Polish National Science Centre grant 2022/45/B/ST6/00559.
Jarosław Byrka: Supported by Polish National Science Centre grant 2020/39/B/ST6/01641.
Yann Disser: Supported by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through subproject A09 of CRC/TRR154.
Annette Lutz: Supported by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through subproject A09 of CRC/TRR154.

1 Introduction

We consider cardinality constrained maximization problems of the form

$$S_k^\star \coloneqq \arg\max\{f(S) \mid S \subseteq \mathcal{U}, |S| \leq k\}, \tag{1}$$

with a monotone[1], submodular[2] and non-negative objective $f\colon 2^{\mathcal{U}} \to \mathbb{R}_{\geq 0}$ over a finite ground set $\mathcal{U}$ of size $n \coloneqq |\mathcal{U}|$, and a given cardinality $k \in [n] \coloneqq \{1, \dots, n\}$. It is a classical result that the greedy algorithm, that extends the solution S by the element $e \in \mathcal{U}$ maximizing $f(S \cup \{e\})$ in each step, produces a solution that is within a factor of $\frac{\mathrm{e}}{\mathrm{e}-1} \approx 1.582$ of the optimum [17], and that this is best-possible for polynomial time algorithms [14, 20], even if f is a coverage function [13] (assuming $\mathrm{P} \neq \mathrm{NP}$).

A major advantage of the greedy algorithm, besides being natural, simple and efficient, lies in the fact that it produces an *incremental* solution, i.e., that its solutions to different cardinalities share the same prefix in terms of the order in which elements are included. This means that greedy solutions can be gradually implemented as the cardinality bound increases, which is relevant, e.g., in large infrastructure projects that are implemented over time, but must already be useful long before their completion.

This raises the natural question whether better incremental solutions than greedy are possible if we do not require efficient computation of the solution. Sacrificing efficiency for quality of the solution is particularly well-motivated for long-term projects where computation time is short compared to the implementation timeframe of the solution. Also, outside the worst case, incremental solutions better than the greedy guarantee may sometimes be possible to compute in polynomial time.

We adopt the framework of incremental maximization introduced by Bernstein et al. [2] to address this question. Formally, an *incremental solution* is given by a sequence of nested solutions $\emptyset = S_0 \subseteq \cdots \subseteq S_n = \mathcal{U}$, where each S_i with $|S_i| = i$ is a solution to (1) for cardinality $k = i$. We can also identify an incremental solution with an ordering $\{e_1, \dots, e_n\} = \mathcal{U}$ of the elements in the ground set $\mathcal{U}$, where $e_i = S_i \setminus S_{i-1}$. We say that the incremental solution is *ρ-competitive* if

$$\max_{k \in \{1,\dots,n\}} \frac{f(S_k^\star)}{f(S_k)} \leq \rho,$$

and define the *competitive ratio*[3] of an algorithm as the infimum over all $\rho \geq 1$ such that the algorithm guarantees a ρ-competitive solution.

In these terms, the greedy algorithm is well known to have competitive ratio $\frac{\mathrm{e}}{\mathrm{e}-1}$. We ask for the best possible competitive ratio for incremental submodular maximization, i.e., for the *price of incrementality* that quantifies the loss in solution quality that we have to accept if we require a consistent solution for all cardinalities simultaneously.

Our Results. Our main result is the first algorithm (cf. Algorithm 1) to beat the greedy competitive ratio of $\mathrm{e}/(\mathrm{e}-1) \approx 1.582$ for the incremental submodular maximization problem.

---

[1] $f\colon 2^{\mathcal{U}} \to \mathbb{R}_{\geq 0}$ is monotone if $A \subseteq B \subseteq \mathcal{U} \Longrightarrow f(A) \leq f(B)$.

[2] $f\colon 2^{\mathcal{U}} \to \mathbb{R}_{\geq 0}$ is submodular if $A, B \in \mathcal{U} \Longrightarrow f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$.

[3] We use the notion of *competitive* ratio in contrast to *approximation* ratio to indicate that we do not require polynomial time. Incremental maximization can also be phrased as an online problem, where the only information revealed online in step i is whether or not $k \geq i$ (much like in ski rental).

▶ **Theorem 1.1.** *Clever-Greedy has a competitive ratio of* $\rho \approx 1.373$ *for incremental submodular maximization.*

In particular, our analysis is tight, up to the numerical precision of our analysis.

Note that our algorithm is not efficient but rather provides a proof that a good incremental solution always *exist*, i.e., a good consistent solution for all cardinalities. In fact, no better approximation algorithm than greedy for submodular maximization under even a fixed cardinality constraint is possible with a sub-exponential number of value-oracle accesses to f [14, 20]. In accordance to this, our algorithm requires access to optimum solutions for each cardinality. While this makes our algorithm inefficient in general, it may be efficiently implemented in special cases and it may be operated with heuristical solutions. We therefore believe the algorithm to be interesting beyond just providing a theoretical certificate for the existence of good incremental solutions.

Previous algorithms in the framework of incremental maximization were based on static scaling strategies that assemble optimum solutions of exponentially growing sizes in phases. Our algorithm is the first one that fully adapts its scaling behavior to the instance and the first scaling algorithm to (deterministically) beat a factor of $\varphi+1 \approx 2.618$ for any (nontrivial) incremental maximization variant [2, 11, 21].

We complement our result with an unconditional lower bound on the existence of good incremental solutions by providing an instance that does not admit incremental solutions that approximates the optimum to a factor better than 1.25 across all cardinalities.

▶ **Theorem 1.2.** *Every deterministic algorithm for incremental submodular maximization has a competitive ratio of at least* $\frac{5}{4} = 1.25$*.*

We leave closing the gap $[1.25, 1.373]$ between our bounds as an open question.

Techniques. Our algorithm is motivated by an interpretation of the classical analysis of the greedy algorithm by Nemhauser et al. [17]. The key property of monotone submodular functions exploited in this analysis is the fact that, for every set S that does not already contain the optimum solution $S_k^\star$, there always exists an element $e \in S_k^\star \setminus S$ with marginal return

$$f(S \cup \{e\}) - f(S) \geq \frac{1}{k} \cdot \big(f(S_k^\star \cup S) - f(S)\big) \geq \frac{1}{k} \cdot \big(f(S_k^\star) - f(S)\big).$$

In other words, there is always an element that covers at least a $1/k$ fraction of the missing value relative to the optimum. This is most intuitive if f is a coverage function, but holds for every monotone submodular function. By definition, if S is the partial solution so far, the greedy algorithm includes an element not worse than the element e above. It follows that the part of the optimum in the solution S_i after step i that is not yet accounted for is bounded by

$$f(S_k^\star) - f(S_i) \leq \left(1 - \frac{1}{k}\right) \cdot (f(S_k^\star) - f(S_{i-1})) \leq \ldots \leq \left(1 - \frac{1}{k}\right)^k \cdot f(S_k^\star) \leq \frac{1}{\mathrm{e}} \cdot f(S_k^\star),$$

which yields the greedy competitive ratio of

$$\max_k \frac{f(S_k^\star)}{f(S_i)} \leq \frac{\mathrm{e}}{\mathrm{e}-1} \approx 1.58.$$

We can interpret the analysis (not the actual greedy algorithm) as an algorithm that extends the current solution S by a single element towards the solution $S_k^\star \cup S$. The key idea of our approach is to adapt this behavior by extending the solution by multiple elements towards a fixed $S_k^\star \cup S$, which is possible, since for every integer $\ell \leq |S_k^\star \setminus S|$, there exist incremental sets $E_1, E_2, \dots, E_{|S_k^\star \setminus S|} \subseteq S_k^\star \setminus S$, where $|E_\ell| = \ell$ and

$$f(S \cup E_\ell) - f(S) \geq \frac{\ell}{k} \cdot \big(f(S_k^\star \cup S) - f(S)\big).$$

This yields an algorithm that operates in phases, adding k_j elements towards a fixed solution in phase j. Note that the algorithm is no longer greedy since E may not include elements of the largest individual marginal increase. In terms of its classical analysis, the greedy algorithm targets and builds towards a different solution in each step. Targeting instead a fixed solution for multiple steps promises to yield an improved guarantee for every cardinality of the form $k = \sum_j k_j$, since

$$f(S_k^\star) - f(S_k) \leq \prod_j \Big(1 - \frac{k_j}{k}\Big)^{k_j} \cdot f(S_k^\star) \leq \Big(1 - \frac{1}{k}\Big)^{k} \cdot f(S_k^\star),$$

i.e., strictly better than greedy if $k_j > 1$ for some j.

While long phases (large k_j) achieve improved guarantees upon completion of a phase, in the worst case, we only get a linear increase in objective value during a single phase, which may be worse than greedy during most of the phase. Previous approaches avoided this issue by completely separating the contributions in each phase and either using the contribution of the previously completed phase or of the current phase to estimate against the optimum solution for the current cardinality. The result were scaling algorithms that use phases of lengths (at least) δ^i with a carefully balanced factor δ [2, 11, 21]. However, such strategies fail to achieve competitive ratios better than even $\varphi + 1 \approx 2.618$ and only seem adequate for classes of objective functions beyond the submodular regime.

In contrast, based on the described perspective on the classical greedy analysis, we are able to integrate the accounting of the different phases, which means that we are able to use the total value achieved across all phases in our estimation. Our algorithm adapts the phase lengths dynamically so as to maximize the linear increase, i.e., the density, during a phase. The key insight in our analysis is that this results in the algorithm being able to afford long phases whenever $\textsc{Opt}(k) := f(S_k^\star)$ is nearly linear in k. In case $\textsc{Opt}(k)$ is far from linear, i.e., the slope of the concave envelope decreases quickly, we are able to exploit that the greedy solution is already better than $\mathrm{e}/(\mathrm{e}-1)$-competitive.

Related work. Bernstein et al. [2] initiated the study of incremental maximization with a monotone objective and presented a simple scaling algorithm that achieves a competitive ratio of $1 + \varphi \approx 2.618$ for accountable objectives, a property that generalizes submodularity. The best known lower bound in this regime is 2.246 by Disser et al. [9].

While, subadditivity[4] is a more direct generalization of submodularity and a more natural "diminishing returns" property, it was more elusive in terms of incremental guarantees. Recently, Disser and Weckbecker [11] were able to give the first competitive algorithm for this setting, which achieves a competitive ratio of at most $2 + \sqrt{2} \approx 3.414$. The algorithm again uses scaling by at least a constant factor, but further increases the lengths of phases if this is beneficial for the resulting density of the phase. Note that accountability and subadditivity are incomparable in the sense that either class captures functions not contained in the other.

[4] $f\colon 2^{\mathcal{U}} \to \mathbb{R}_{\geq 0}$ is subadditive if $A, B \subseteq \mathcal{U} \implies f(A) + f(B) \geq f(A \cup B)$.

Algorithm 1 Clever-Greedy($\mathcal{U}, f$).

$S \leftarrow \emptyset$, $j \leftarrow 0$, $k_0 \leftarrow 0$
while $S \neq \mathcal{U}$ **do**
  $j \leftarrow j + 1$
  take $k_j \in \arg\max_{k > k_{j-1}} (\textsc{Opt}(k) - \textsc{Opt}(k_{j-1}))/k$
  while $S^\star_{k_j} \setminus S \neq \emptyset$ **do**
    take $e \in \arg\max_{e' \in S^\star_{k_j} \setminus S} f(S \cup \{e'\})$
    $S \leftarrow S \cup \{e\}$

The analysis of the inherently incremental greedy algorithm for cardinality constrained maximization has a long history including classical results in combinatorial optimization. Rado [18] observed that greedy is optimal for modular objectives, and Nemhauser et al. [17] provided the well-known classical result that greedy achieves an $\frac{e}{e-1}$-approximation with nk value-oracle accesses to f (meaning that the value $f(S)$ is needed for nk different sets $S \subset \mathcal{U}$). Nemhauser and Wolsey [16] and Vondrak [20] showed that any better approximation algorithm requires an exponential number of value-oracle accesses. For the special case of coverage functions, Feige [13] used the PCP-theorem to establish conditional hardness of approximation to within a better factor than greedy.

The analysis of the greedy algorithm was generalized to functions of bounded submodularity ratio by Das and Kempe [8]. Bian et al. [3] proved this analysis tight and further generalized it to functions of bounded curvature. Elenberg et al. [12] further extended the results of Das and Kempe and Harshaw et al. [15] show no better approximation with polynomially many value-orcale accesses is possible for functions of bounded submodularity ratio. Moreover, augmentability was introduced by Bernstein et al. [2] as an alternative relaxation of submodularity. Bernstein et al. generalized the greedy analysis to this regime and Disser and Weckbecker [10] further parameterized the result and showed it to be tight.

Other variants of submodular maximization that have been considered include non-monotone settings [5, 6, 14], settings with knapsack [19] and matroid [7] constraints, and streaming models [1].

2 The Algorithm

Our algorithm Clever-Greedy (cf. Algorithm 1) constructs its solution S incrementally, starting from the empty set. It assumes knowledge of optimum solutions $\{S^\star_k\}_{k=1}^n$ and operates in *phases*. Let $k_0 = 0$. At the start of each phase j, we identify the solution $S^\star_{k_j}$, such that $k_j > k_{j-1}$ maximizes the value

$$\frac{\textsc{Opt}(k_j) - \textsc{Opt}(k_{j-1})}{k_j}.$$

Within the phase j, Clever-Greedy adds the elements of $S^\star_{k_j}$ to the solution, greedily taking the elements that increase the objective the most. The algorithm terminates once all elements have been added.

Algorithm 2 CLEVER-GREEDY$(\mathcal{U}, f)$, with variables to track phases and iterations.

$S^{(0)} \leftarrow \emptyset$, $j \leftarrow 0$, $k_0 \leftarrow 0$
while $S^{(j)} \neq \mathcal{U}$ **do**
  $j \leftarrow j+1$, $S_0^{(j)} \leftarrow S^{(j-1)}$
  take $k_j \in \arg\max_{k>k_{j-1}} (1/k) \cdot (\text{OPT}(k) - \text{OPT}(k_{j-1}))$
  $i \leftarrow 0$
  while $S^\star_{k_j} \not\subseteq S_i^{(j)}$ **do**
    take $e \in \arg\max_{e' \in S^\star_{k_j} \setminus S_i^{(j)}} f(S_i^{(j)} \cup \{e'\})$
    $S_{i+1}^{(j)} \leftarrow S_i^{(j)} \cup \{e\}$, $i \leftarrow i+1$
  $p_j \leftarrow i$, $S^{(j)} \leftarrow S_{p_j}^{(j)}$
$\ell \leftarrow j$
return $(S_1^{(1)}, \ldots, S_{p_1}^{(1)}, \ldots, S_1^{(\ell)}, \ldots, S_{p_\ell}^{(\ell)})$

3 Analysis

In this section, we prove the upper bound of Theorem 1.1 – CLEVER-GREEDY achieves a competitive ratio of $\rho < 1.3729$.

3.1 Notation

We start by introducing variables to keep track of the phases and iterations of the algorithm explicitly (cf. Algorithm 2 along with the following definitions). We let ℓ denote the number of phases (iterations of the outer loop), and p_j the number of steps during phase $j \in [\ell]$ (iterations of the inner loop). Observe that $1 \leq p_j \leq k_j$ and $\sum_{j=1}^{\ell} p_j = n = |\mathcal{U}|$.

We further use $S_i^{(j)}$ to denote the state of set S after iteration i of phase j, and let $S^{(j)} := S_{p_j}^{(j)}$ be the state of set S at the end of phase j. We also denote $S_0^{(j)} := S^{(j-1)}$, $S_i^{(j)} := S_{p_j}^{(j)}$ for $p_j < i \leq k_j$, and

$$\theta_j := \text{OPT}(k_j).$$

Finally, for every phase $j \in [\ell]$, we define its *density* as

$$\delta_j := \frac{1}{k_j} \cdot (\text{OPT}(k_j) - \text{OPT}(k_{j-1})) = \frac{1}{k_j} \cdot (\theta_j - \theta_{j-1})$$

It is easy to see that our algorithm ensures that densities are monotonically decreasing.

▶ **Observation 3.1.** *For each phase $j \in [\ell - 1]$, it holds that $\delta_j \geq \delta_{j+1}$.*

Proof. The algorithm definition implies $k_{j-1} < k_j$. Hence,

$$\begin{aligned} \delta_j &= \max_{k>k_{j-1}} \frac{1}{k} \cdot (\text{OPT}(k) - \theta_{j-1}) \geq \max_{k>k_{j-1}} \frac{1}{k} \cdot (\text{OPT}(k) - \theta_j) && (\text{by } \theta_{j-1} \leq \theta_j) \\ &\geq \max_{k>k_j} \frac{1}{k} \cdot (\text{OPT}(k) - \theta_j) = \delta_{j+1}. \end{aligned}$$ ◀

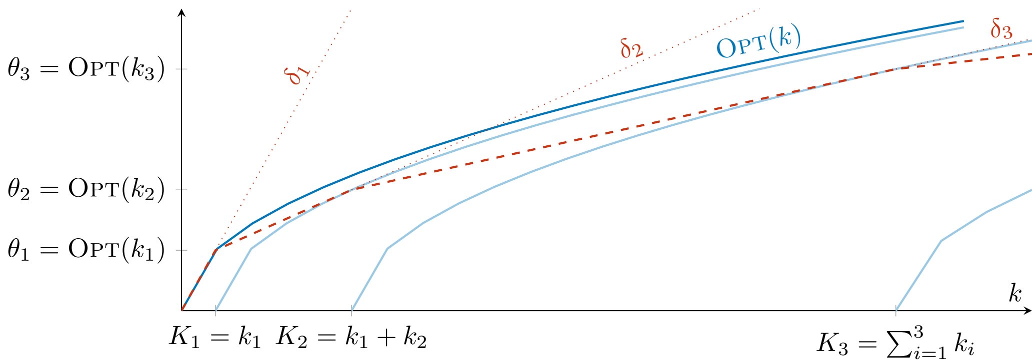

Figure 1 The phases of the algorithm: The dashed red curve describes the lower bound on the value of the algorithm given by Lemma 3.4. In phase j, the cardinality k_j is chosen such that the red curve is tangential to the translated optimum curve starting in $K_j = \sum_{i=1}^{j} k_i$ (solid blue plots) which corresponds to maximizing the density δ_j, i.e., the slope of the lower bound.

3.2 Basic Properties

We now prove a few properties that relate the gains of Clever-Greedy and Opt.

First, observe that at the end of phase j, Clever-Greedy has accumulated all elements of the optimum solution of size k_j. This happens after it collects $\sum_{i=1}^{j} p_i \leq \sum_{i=1}^{j} k_i$ elements.

▶ **Lemma 3.2.** *For each phase $j \in [\ell]$, we have $f(S^{(j)}) \geq \theta_j$.*

Proof. At the end of the j^{th} phase, we have added each element in $S^\star_{k_j}$ to the solution S. Thus, $S^{(j)} \supseteq S^\star_{k_j}$, and so $f(S^{(j)}) \geq f(S^\star_{k_j}) = \theta_j$ by the monotonicity of f. ◀

Fix any phase j of the algorithm and recall that the number of iterations in the phase is $p_j \leq k_j$. Lemma 3.2 lower-bounds Clever-Greedy's gain by θ_{j-1} and θ_j at the beginning and end of phase j, respectively. Linear interpolation of these two bounds evaluated after the i^{th} step of phase j yields a value of $\theta_{j-1} + (i/p_j) \cdot (\theta_j - \theta_{j-1}) \geq \theta_{j-1} + (i/k_j) \cdot (\theta_j - \theta_{j-1}) = \theta_{j-1} + i \cdot \delta_j$.

In the next lemma, we show that the submodularity of function f combined with greediness of Clever-Greedy guarantee at least this "interpolated gain" (cf. Figure 1). In the proof, we will use the following claim that follows directly by submodularity of f. For a proof of the claim, see the full version of the paper [4].

▷ Claim 3.3. For every two subsets $A \subsetneq B \subseteq \mathcal{U}$ there exists an element $e \in B \setminus A$, such that $f(A \cup \{e\}) - f(A) \geq (f(B) - f(A))/(|B| - |A|)$.

▶ **Lemma 3.4.** *For each phase $j \in [\ell]$ and $i \in [k_j]$, we have $f(S_i^{(j)}) \geq \theta_{j-1} + i \cdot \delta_j$.*

Proof. For $i \geq p_j$, we have $S_i^{(j)} = S^{(j)}$. By Lemma 3.2 and by the definition of density δ_j, it holds that $f(S_i^{(j)}) = f(S^{(j)}) \geq \theta_j = \theta_{j-1} + k_j \cdot \delta_j \geq \theta_{j-1} + i \cdot \delta_j$. Hence, in the remaining part of the proof, we focus on case $i \in [p_j]$.

Fix $t \in [p_j]$. By Claim 3.3 applied to sets $S_{t-1}^{(j)}$ and $S^{(j)}$, there exists an element $e_t \in S^{(j)} \setminus S_{t-1}^{(j)}$ such that

$$\frac{f(S^{(j)}) - f(S_{t-1}^{(j)})}{p_j - t + 1} \leq f(S_{t-1}^{(j)} \cup \{e_t\}) - f(S_{t-1}^{(j)}) \leq f(S_t^{(j)}) - f(S_{t-1}^{(j)}). \tag{2}$$

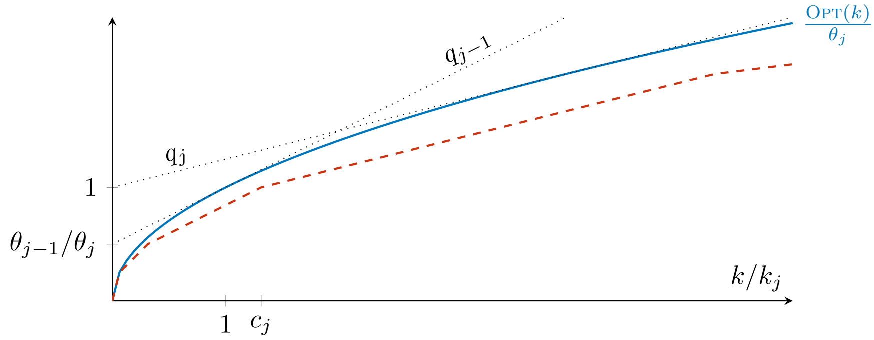

Figure 2 When analyzing the relation between c_j and the normalized density q_j in phase j in Lemma 3.7, we scale all values down by θ_j and all cardinalities by k_j. The solid blue plot depicts the rescaled optimum curve while the dashed red plot shows the rescaled lower bound. Note that the dotted slopes q_{j-1} and q_j are exactly the slopes of the rescaled lower bound before and after c_j.

Let $d_t \coloneqq f(S^{(j)}) - f(S_t^{(j)})$ and $g_t \coloneqq f(S_t^{(j)}) - f(S_{t-1}^{(j)})$. By (2), it holds that $d_t = d_{t-1} - g_t$ and $g_t \geq d_{t-1}/(p_j - t + 1)$. Thus, $d_t \leq d_{t-1} \cdot (1 - 1/(p_j - t + 1))$, and we have

$$d_i \leq d_0 \cdot \prod_{t=1}^{i} \left(1 - \frac{1}{p_j - t + 1}\right) = d_0 \cdot \prod_{t=1}^{i} \frac{p_j - t}{p_j - t + 1} = d_0 \cdot \frac{p_j - i}{p_j} \leq d_0 \cdot \left(1 - \frac{i}{k_j}\right). \tag{3}$$

By the definition of d_i and (3),

$$f(S^{(j)}) - f(S_i^{(j)}) = d_i \leq d_0 \cdot \left(1 - \frac{i}{k_j}\right) = \left(f(S^{(j)}) - f(S^{(j-1)})\right) \cdot \left(1 - \frac{i}{k_j}\right),$$

which can be rearranged to

$$\begin{aligned} f(S_i^{(j)}) &\geq \frac{i}{k_j} \cdot f(S^{(j)}) + \left(1 - \frac{i}{k_j}\right) \cdot f(S^{(j-1)}) \\ &\geq \frac{i}{k_j} \cdot \theta_j + \left(1 - \frac{i}{k_j}\right) \cdot \theta_{j-1} && \text{(by Lemma 3.2 and } i \leq k_j\text{)} \\ &= \theta_{j-1} + i \cdot \delta_j. \end{aligned}$$ ◀

Finally, we argue that the choice of the density made by Clever-Greedy can also be used to upper-bound the gain of Opt between phases.

▶ **Lemma 3.5.** *For each phase $t \in [\ell]$ and $k \in [n]$, we have $\textsc{Opt}(k) \leq \theta_t + (k - k_t) \cdot \delta_t$.*

Proof. We first argue that $(\textsc{Opt}(k) - \theta_{t-1})/k \leq (\theta_t - \theta_{t-1})/k_t$. This is immediately true if $k > k_{t-1}$ by the choice of k_t made by the algorithm. On the other hand, when $k \leq k_{t-1}$, the left hand side is at most 0 and the right hand side at least 0.

Next, the relation $(\textsc{Opt}(k) - \theta_{t-1})/k \leq (\theta_t - \theta_{t-1})/k_j$ can be rearranged to

$$\begin{aligned} \textsc{Opt}(k) &\leq \theta_{t-1} + \frac{k}{k_t} \cdot (\theta_t - \theta_{t-1}) = \theta_t + \left(\frac{k_t - k}{k_t}\right) \cdot (\theta_{t-1} - \theta_t) \\ &= \theta_t + (k_t - k) \cdot (-\delta_t). \end{aligned}$$ ◀

3.3 Inductive Analysis

In the analysis of phase j, we exploit the choice of cardinality k_j by Clever-Greedy together with the value θ_j of the optimal solution at this cardinality. To measure the progress of the algorithm in phase j it can be convenient to scale all cardinalities by k_j and all gains by θ_j. Accordingly, we define *normalized density* as

$$q_j \coloneqq \frac{k_j}{\theta_j} \cdot \delta_j = \frac{\theta_j - \theta_{j-1}}{\theta_j} = 1 - \frac{\theta_{j-1}}{\theta_j}.$$

We also define

$$K_j \coloneqq \textstyle\sum_{i=1}^{j} k_i \qquad \text{and} \qquad c_j = K_j / k_j.$$

See Figure 2 for an illustration of these definitions.

By Lemma 3.2, our algorithm reaches (or exceeds) the value θ_j after collecting $\sum_{i=1}^{j} p_i \leq \sum_{i=1}^{j} k_i = K_j$ elements, while this value is reached by Opt after collecting k_j elements. Hence, $c_j = K_j/k_j$ measures the multiplicative "cardinality overhead" our algorithm incurs in comparison to Opt. We now show that this overhead can be controlled, i.e., that we can link its values between consecutive phases, and the change of the overhead can be attributed to the normalized density.

More formally, we define the function $h\colon (0,1] \to \mathbb{R}$ as

$$h(q) \coloneqq \frac{156 + 1543 \cdot q - 550 \cdot q^2 - 149 \cdot q^3}{1000 \cdot q}.$$

We use some technical properties of function h, listed in the following claim. For a proof we refer to the full version of this paper [4].

We do not use the actual definition of function h in the remaining part of the analysis, but only the fact that it satisfies the properties of Claim 3.6 for a given value of ρ. Any other function (potentially giving a different ρ) satisfying these constraints can be applied in our analysis, yielding ρ as a bound on the competitive ratio of Clever-Greedy.

▷ Claim 3.6. Let $\rho = 1.3729$. Function h satisfies the following properties:

1. Function $h(q)$ is decreasing in q for $q > 0$ and $h(1) = 1$.
2. For all $x > 0$ and $q \in (0,1]$, we have
$$1 + \frac{h(q)}{1+x} \leq h\left(\frac{xq}{1+xq}\right).$$
3. For all $q \in (0,1]$ we have $1 + (h(q) - 1) \cdot q \leq \rho$.
4. For all $x \geq 0$ and $q \in (0,1]$, we have
$$1 - \frac{h(q) \cdot xq}{(1+xq)(1+x)} \geq \rho^{-1}.$$

This allows us to relate the cardinality overhead to the normalized density of a phase via h.

▶ **Lemma 3.7.** *In each phase $j \in [\ell]$, we have $c_j \leq h(q_j)$.*

Proof. We prove the lemma by induction on j. For the inductive basis ($j = 1$), we observe that $c_1 = 1 = h(1) \leq h(q_1)$.

Now, we assume the inductive hypothesis hold for phase $j-1$ and we show it for phase $j>1$. Let $x=(k_j-k_{j-1})/k_{j-1}$. Note that $x>0$ as $k_j>k_{j-1}$ by the choice of the algorithm. By Lemma 3.5 applied with $k=k_j$ and $t=j-1$, we obtain

$$\theta_j \leq \theta_{j-1}+(k_j-k_{j-1})\cdot\delta_{j-1}=\theta_{j-1}+(k_j-k_{j-1})\cdot\frac{\theta_{j-1}}{k_{j-1}}\cdot q_{j-1}=(1+x\cdot q_{j-1})\cdot\theta_{j-1}.$$

Using the bound above, we can bound the normalized density as

$$q_j=1-\frac{\theta_{j-1}}{\theta_j}\leq 1-\frac{\theta_{j-1}}{(1+x\cdot q_{j-1})\cdot\theta_{j-1}}=\frac{x\cdot q_{j-1}}{1+x\cdot q_{j-1}}. \tag{4}$$

We conclude that

$$\begin{aligned} c_j &= \frac{\sum_{i=0}^{j} k_i}{k_j} = 1+c_{j-1}\cdot\frac{k_{j-1}}{k_j} = 1+\frac{c_{j-1}}{1+x} \\ &\leq 1+\frac{h(q_{j-1})}{1+x} && \text{(by the inductive hypothesis)} \\ &\leq h\left(\frac{x\cdot q_{j-1}}{1+x\cdot q_{j-1}}\right) && \text{(by Property 2 of Claim 3.6)} \\ &\leq h(q_j), && \text{(by (4) and Property 1 of Claim 3.6)} \end{aligned}$$

which proves the inductive claim, and thus concludes the proof. ◀

3.4 Bounds on the Optimum

Recall that once the algorithm reaches the end of phase j, it has collected $\sum_{i=1}^{j} p_j \leq \sum_{i=1}^{j} k_i = K_j$ elements. Hence, our goal now is to upper-bound the gain of Opt once it has collected the same number of elements, as a benchmark for the gain of Clever-Greedy at that point.

For every phase $j\in[\ell-1]$, we define

$$k_j^* = k_j\cdot\frac{\delta_j}{\delta_j-\delta_{j+1}}.$$

Note that k_j^* is not necessarily an integer and may not correspond directly to a cardinality.

▶ **Lemma 3.8.** *For each phase $j\in[\ell-1]$, it holds that $k_j\leq k_j^*$ and $(k_j^*-k_j)\cdot\delta_j=k_j^*\cdot\delta_{j+1}$.*

Proof. By Observation 3.1, $\delta_j\geq\delta_{j-1}$, and thus $k_j^*\geq k_j$. The second property follows as $(k_j^*-k_j)\cdot\delta_j-k_j^*\cdot\delta_{j+1}=k_j^*\cdot(\delta_j-\delta_{j+1})-k_j\cdot\delta_j=0$. ◀

The intuition behind k_j^* is that it is the cardinality where the two bounds on $\textsc{Opt}(k)$ given by Lemma 3.5 applied with $t=j$ and $t=j+1$ intersect. To see this, recall that Lemma 3.5 applied with $t=j$ yields

$$\textsc{Opt}(k)\leq\theta_j+(k-k_j)\cdot\delta_j.$$

On the other hand, Lemma 3.5 applied with $t=j+1$ yields

$$\begin{aligned} \textsc{Opt}(k) &\leq \theta_{j+1}+(k-k_{j+1})\cdot\delta_{j+1} \\ &= \theta_j+k_{j+1}\cdot\delta_{j+1}+(k-k_{j+1})\cdot\delta_{j+1} \\ &= \theta_j+k\cdot\delta_{j+1}. \end{aligned}$$

We conclude that the above bounds coincide at $k_j^\star$ and use Lemma 3.8 to relate these bounds to ρ.

▶ **Lemma 3.9.** *Fix ρ as in Claim 3.6. For $j \in [\ell]$, we have*

$$\theta_j + (k_j^* - k_j) \cdot \delta_j = \theta_j + k_j^* \cdot \delta_{j+1} \le \rho \cdot (\theta_j + (k_j^* - K_j) \cdot \delta_{j+1}).$$

Proof. The first inequality is due to the definition of k_j^* and is restated in Lemma 3.8. Thus, it suffices to bound $\theta_j + (k_j^* - k_j) \cdot \delta_j$. Let $x = (k_j^* - k_j)/k_j$ be the normalized distance between k_j^* and k_j. Then,

$$\frac{k_j^* - k_j}{k_j^*} = \frac{x}{1+x} \qquad \text{and} \qquad \frac{k_j^*}{k_j} = 1 + x.$$

Moreover,

$$\theta_j + (k_j^* - k_j) \cdot \delta_j = \theta_j \cdot \left(1 + \frac{k_j^* - k_j}{k_j} \cdot q_j\right) = \theta_j \cdot (1 + xq_j). \tag{5}$$

On the other hand, by Lemma 3.8, we have $\delta_{j+1} = \delta_j \cdot (k_j^* - k_j)/k_j^* = q_j \cdot (\theta_j/k_j) \cdot (k_j^* - k_j)/k_j^*$, and thus

$$\begin{aligned}
\theta_j + (k_j^* - K_j) \cdot \delta_{j+1} &= \theta_j \cdot \left(1 + \frac{k_j^* - K_j}{k_j} \cdot \frac{k_j^* - k_j}{k_j^*} \cdot q_j\right) \\
&= \theta_j \cdot \left(1 + (1 + x - c_j) \cdot \frac{x}{1+x} \cdot q_j\right) \\
&= \theta_j \cdot \left(1 + xq_j - \frac{c_j \cdot xq_j}{1+x}\right).
\end{aligned} \tag{6}$$

Combining (5) with (6), we obtain

$$\begin{aligned}
\frac{\theta_j + (k_j^* - K_j) \cdot \delta_{j+1}}{\theta_j + (k_j^* - k_j) \cdot \delta_j} &= 1 - \frac{c_j \cdot xq_j}{(1 + xq_j)(1+x)} \\
&\ge 1 - \frac{h(q_j) \cdot xq_j}{(1 + xq_j)(1+x)} && \text{(by Lemma 3.7)} \\
&\ge 1/\rho. && \text{(by Property 4 of Claim 3.6)}
\end{aligned}$$ ◀

Now, we extend the good approximation result to all cardinalities within a phase. For this, we use linearity of the upper bound on the optimum value given in Lemma 3.5.

▶ **Lemma 3.10.** *Fix $t \in \{0, \ldots, \ell - 1\}$ and any real y such that $K_t \le y \le K_{t+1}$. Then, for every phase $j \in \{0, \ldots, \ell - 1\}$, it holds that $\theta_t + (y - K_t) \cdot \delta_{t+1} \le \theta_j + (y - K_j) \cdot \delta_{j+1}$.*

Proof. For any $r \in \{0, \ldots, \ell - 1\}$, we define $A_r \coloneqq \theta_r + (y - K_r) \cdot \delta_{r+1}$. We will show that $A_t \le A_j$ for any j. First, we observe that for each $r \ge 1$ it holds that

$$\begin{aligned}
A_r - A_{r-1} &= (\theta_r + (y - K_r) \cdot \delta_{r+1}) - (\theta_{r-1} + (y - K_{r-1}) \cdot \delta_r) \\
&= (\theta_r + (y - K_r) \cdot \delta_{r+1}) - (\theta_{r-1} + k_r \cdot \delta_r + (y - K_r) \cdot \delta_r) \\
&= (y - K_r) \cdot (\delta_{r+1} - \delta_r).
\end{aligned}$$

By Observation 3.1, it holds that $\delta_r \ge \delta_{r+1}$. Hence, $A_0 \ge A_1 \ge \cdots \ge A_t$ and $A_t \le A_{t+1} \le \cdots \le A_\ell$, which concludes the proof. ◀

▶ **Lemma 3.11.** *Fix ρ as in Claim 3.6. For all $k \in [n]$ and $j \in \{0, \ldots, \ell - 1\}$, it holds that*

$$\textsc{Opt}(k) \le \rho \cdot (\theta_j + (k - K_j) \cdot \delta_{j+1}).$$

Proof. Throughout the proof, we fix $j \in \{0, \ldots, \ell\}$. We prove the lemma by induction on k.

First, observe that for $k = 0$ Lemma 3.10, applied with $t = 0$ and $y = 0$, yields $\theta_j + (0 - K_j) \cdot \delta_{j+1} \geq \theta_0 + (0 - K_0) \cdot \delta_1 = \theta_0 = \textsc{Opt}(0)$, and thus the induction basis holds.

Assuming the lemma statement holds for all $k \leq k_t$ for some $t \in \{0, \ldots, \ell - 1\}$, we show that it holds for all $k' \in \{k_t + 1, k_t + 2, \ldots, k_{t+1}\}$. We have

$$\begin{aligned}
\theta_t + K_{t-1} \cdot \delta_t &= \theta_t + K_{t-1} \cdot (q_t / k_t) \cdot \theta_t && \text{(by definition of } q_j\text{)} \\
&= (1 + (c_t - 1) \cdot q_t) \cdot \theta_t && \text{(by definition of } c_j\text{)} \\
&\leq (1 + (h(q_t) - 1) \cdot q_t) \cdot \theta_t && \text{(by Lemma 3.7)} \\
&\leq \rho \cdot \theta_t. && \text{(by Property 3 of Claim 3.6)} \\
&= \rho \cdot (\theta_t + (K_t - K_t) \cdot \delta_{t+1}) \\
&\leq \rho \cdot (\theta_j + (K_t - K_j) \cdot \delta_{j+1}). && \text{(by Lemma 3.10 with } y = K_t\text{)} \qquad (7)
\end{aligned}$$

We now distinguish cases depending on whether or not $k' \leq K_t$.

Case 1. Assume $k' \leq K_t$. By induction, the statement holds for $k = k_t$, resulting in

$$\theta_t \leq \rho \cdot (\theta_j + (k_t - K_j) \cdot \delta_{j+1}). \qquad (8)$$

We fix $\alpha \in [0, 1]$ such that $k' = (1 - \alpha) \cdot k_t + \alpha \cdot K_t = k_t + \alpha \cdot K_{t-1}$. Then,

$$\begin{aligned}
\textsc{Opt}(k') &\leq \theta_t + (k' - k_t) \cdot \delta_t && \text{(by Lemma 3.5)} \\
&= (1 - \alpha) \cdot \theta_t + \alpha \cdot (\theta_t + K_{t-1} \cdot \delta_t) \\
&\leq (1 - \alpha) \cdot \rho \cdot (\theta_j + (k_t - K_j) \cdot \delta_{j+1}) \\
&\qquad + \alpha \cdot \rho \cdot (\theta_j + (K_t - K_j) \cdot \delta_{j+1}) && \text{(by (8) and (7))} \\
&= \rho \cdot (\theta_j + (k' - K_j) \cdot \delta_{j+1}).
\end{aligned}$$

Case 2. Assume $K_t < k' \leq k_{t+1}$. (This case holds vacuously if $K_t \geq k_{t+1}$). Note that $K_t < k' \leq k_{t+1} \leq K_{t+1}$.

Case 2.1. Consider the sub-case $k' \leq k_t^*$. In this case, we first observe that

$$\begin{aligned}
\theta_t + (k_t^* - k_t) \cdot \delta_t &\leq \rho \cdot (\theta_t + (k_t^* - K_t) \cdot \delta_{t+1}) \\
&\leq \rho \cdot (\theta_j + (k_t^* - K_j) \cdot \delta_{j+1}), \qquad (9)
\end{aligned}$$

where the first inequality is due to Lemma 3.9 and the second one follows from Lemma 3.10 applied with $y = k_t^*$. Next, we fix $\alpha \in [0, 1]$ such that $k' = (1 - \alpha) \cdot K_t + \alpha \cdot k_t^*$. Then,

$$\begin{aligned}
\textsc{Opt}(k') &\leq \theta_t + (k' - k_t) \cdot \delta_t && \text{(by Lemma 3.5)} \\
&= (1 - \alpha) \cdot (\theta_t + K_{t-1} \cdot \delta_t) + \alpha \cdot (\theta_t + (k_t^* - k_t) \cdot \delta_t) \\
&\leq (1 - \alpha) \cdot \rho \cdot (\theta_j + (K_t - K_j) \cdot \delta_{j+1}) \\
&\qquad + \alpha \cdot \rho \cdot (\theta_j + (k_t^* - K_j) \cdot \delta_{j+1}) && \text{(by (7) and (9))} \\
&= \rho \cdot (\theta_j + (k' - K_j) \cdot \delta_{j+1}).
\end{aligned}$$

Case 2.2. Consider the sub-case $k_t^* < k'$. Then,

$$\begin{aligned}
\textsc{Opt}(k') &\le \theta_{t+1} + (k' - k_{t+1}) \cdot \delta_{t+1} && \text{(by Lemma 3.5)}\\
&= \theta_t + k_{t+1} \cdot \delta_{t+1} + (k' - k_{t+1}) \cdot \delta_{t+1}\\
&= \theta_t + k' \cdot \delta_{t+1}\\
&= \theta_t + k_t^* \cdot \delta_{t+1} + (k' - k_t^*) \cdot \delta_{t+1}\\
&\le \rho \cdot (\theta_t + (k_t^* - K_t) \cdot \delta_{t+1}) + (k' - k_t^*) \cdot \delta_{t+1} && \text{(by Lemma 3.9)}\\
&\le \rho \cdot (\theta_t + (k' - K_t) \cdot \delta_{t+1}) && \text{(as } k' > k_t^* \text{ and } \rho \ge 1)\\
&\le \rho \cdot (\theta_j + (k' - K_j) \cdot \delta_{j+1}) && \text{(by Lemma 3.10 with } y = k').
\end{aligned}$$

In all cases, we showed that $\textsc{Opt}(k') \le \rho \cdot (\theta_j + (k' - K_j) \cdot \delta_{j+1})$, which concludes the induction step.

Finally, for $k' \in [n]$ with $k' \ge k_\ell$ we use that the statement holds for $k = k_\ell$ to infer

$$\begin{aligned}
\textsc{Opt}(k') &\le \theta_\ell + (k' - k_\ell) \cdot \delta_\ell && \text{(by Lemma 3.5)}\\
&\le \rho \cdot (\theta_j + (k_\ell - K_j) \cdot \delta_{j+1}) + (k' - k_\ell) \cdot \delta_\ell\\
&\le \rho \cdot (\theta_j + (k_\ell - K_j) \cdot \delta_{j+1} + (k' - k_\ell) \cdot \delta_\ell) && \text{(as } k' \ge k_\ell \text{ and } \rho \ge 1)\\
&\le \rho \cdot (\theta_j + (k' - K_j) \cdot \delta_{j+1}), && \text{(as } k' \ge k_\ell \text{ and } \delta_\ell \le \delta_{j+1})
\end{aligned}$$

which concludes the proof. ◀

By combining Lemma 3.4 and Lemma 3.11, we obtain our main result.

▶ **Theorem 1.1.** *Clever-Greedy has a competitive ratio of $\rho \approx 1.373$ for incremental submodular maximization.*

Proof. We fix ρ as in Claim 3.6, i.e., $\rho = 1.3729$. We fix an integer $k \in [n]$ and we will show that $f(S_k) \ge \textsc{Opt}(k)/\rho$.

If $k \le k_1$, then $f(S_k) = f(S_k^{(1)}) \ge k \cdot \delta_1 = \theta_1 + (k - k_1) \cdot \delta_1 \ge \textsc{Opt}(k)$, where the first and the second inequality follow by Lemma 3.4 and Lemma 3.5, respectively.

If $k \ge k_1$, we choose largest $t \in [\ell]$ satisfying $K_t \le k$. Then,

$$\begin{aligned}
\textsc{Opt}(k) &\le \rho \cdot (\theta_t + (k - K_t) \cdot \delta_{t+1}) && \text{(by Lemma 3.11 with } j = t)\\
&\le \rho \cdot f(S_i^{(j)}) && \text{(by Lemma 3.4 with } j = t+1 \text{ and } i = k - K_t)\\
&= \rho \cdot f(S_{k'}) && \text{(with } k' = \textstyle\sum_{r=1}^{j-1} p_r + i)\\
&\le \rho \cdot f(S_k). && \text{(as } k' \le K_t + i = k)
\end{aligned}$$ ◀

4 Lower Bounds

In this section, we provide the full description of our lower bounds. In Section 4.1, we provide a lower bound on the competitive ratio of any algorithm, proving the following theorem (restated).

▶ **Theorem 1.2.** *Every deterministic algorithm for incremental submodular maximization has a competitive ratio of at least $\frac{5}{4} = 1.25$.*

In Section 4.2, we give an improved lower bound against our algorithm, showing that the analysis of Clever-Greedy of Theorem 1.1 is tight.

▶ **Theorem 4.1.** *Algorithm 2 has a competitive ratio of at least 1.3724.*

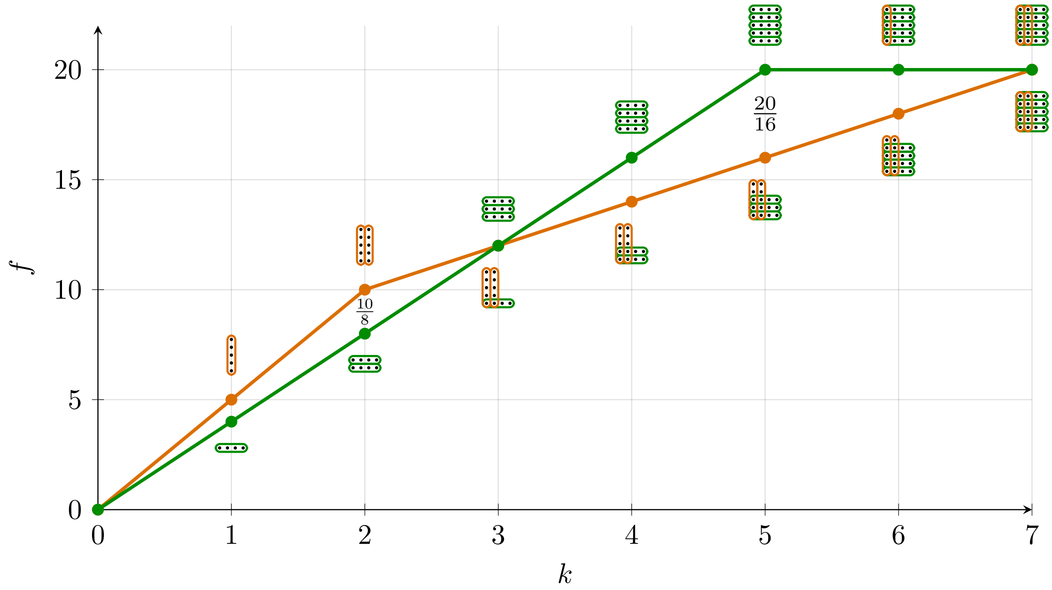

Figure 3 Illustration of the lower bound construction in Section 4.1. The dashed orange curve corresponds to an incremental solution that favors the sets C_j, while the solid green curve corresponds to an incremental solution that favors the sets R_i. No incremental solution can simultaneously be better than 1.25-competitive at $k = 2$ and $k = 5$. Other solutions than the ones depicted are dominated by one of the curves.

4.1 A Hard Instance for Incremental Algorithms

Consider a set $X \coloneqq \{x_{i,j} \mid i \in [5], j \in [4]\}$ of 20 elements, and the subsets $R_i = \{x_{i,j} \mid j \in [4]\}$ and $C_j = \{x_{i,j} \mid i \in [5]\}$. Let f be the coverage function defined over the collection of subsets $\mathcal{U} = \{R_1, \dots, R_5, C_1, C_2\}$. In other words, for every $S \subseteq \mathcal{U}$, we have $f(S) = \left|\bigcup_{E \in S} E\right|$.

The following lemma shows that *any* incremental solution for the function f has a competitive ratio of at least 1.25, proving Theorem 1.2. Consider Figure 3 alongside its proof.

▶ **Lemma 4.2.** *For every ordering $E_1, \dots, E_7$ of the subsets in $\mathcal{U}$, there exists $k \in [7]$ with*

$$\frac{\textsc{Opt}(k)}{f(E_1 \cup \dots \cup E_k)} \geq 1.25.$$

Proof. We first observe that, by symmetry between the C_j's and R_i's, we may restrict our analysis to orderings that include C_1, C_2 and R_1, R_2, R_3, R_4, R_5 in these orders, respectively. Suppose that $\{E_1, E_2\} \neq \{C_1, C_2\}$, i.e. that C_1 and C_2 are not the first two elements in the ordering $E_1, \dots, E_7$. In this case, $f(E_1 \cup E_2) \leq \max\{f(R_1, C_1), f(R_1, R_2)\} = 8$, and thus

$$\frac{\textsc{Opt}(2)}{f(E_1 \cup E_2)} \geq \frac{f(C_1 \cup C_2)}{f(E_1 \cup E_2)} = \frac{10}{8} = \frac{5}{4}.$$

Now, suppose that $\{E_1, E_2\} = \{C_1, C_2\}$. Then,

$$f(E_1 \cup \dots \cup E_5) = f(C_1 \cup C_2 \cup R_1 \cup R_2 \cup R_3) = 16,$$

yielding

$$\frac{\textsc{Opt}(5)}{f(E_1 \cup \dots \cup E_5)} \geq \frac{f(R_1 \cup \dots \cup R_5)}{f(E_1 \cup \dots \cup E_5)} = \frac{20}{16} = \frac{5}{4}.$$ ◀

4.2 A Lower Bound Against Our Algorithm

We begin by providing an informal overview of our lower bound in Section 4.2.1, followed by a formal description of the lower bound instance in Section 4.2.2.

4.2.1 Overview of the Lower Bound for Clever-Greedy

To demonstrate that the analysis of our algorithm is tight, we construct a hard instance for it, based on a geometric coverage function f defined over a collection of high-dimensional rectangles $\mathcal{U}$, so that $f(S)$ measures the volumes of the union of the rectangles in $S \subseteq \mathcal{U}$. The core idea is to exploit the density-based choice of Clever-Greedy by presenting it with a sequence of item groups that appear attractive locally but are inefficient globally.

The Instance $(\mathcal{U}, f)$. More formally, the instance uses a ground set $\mathcal{U}$ of $(q+1)$-dimensional rectangles, organized into groups $\mathcal{U}^{(1)}, \dots, \mathcal{U}^{(q)}$. The geometry of these rectangles is defined such that elements within a single group $\mathcal{U}^{(i)}$ are disjoint; they effectively partition the i^{th} dimension into s_i segments and cover a volume of v_i. However, elements from different groups overlap significantly. This specific geometric arrangement ensures that while a single group covers a volume efficiently on its own, adding elements from a later group $\mathcal{U}^{(j)}$ to an existing solution consisting of elements from $\mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(j-1)}$ provides diminishing marginal gains due to the heavy spatial overlap with the volume already covered by previous groups.

Intuitively, this construction generalizes the 2-dimensional hard instance from Section 4.1 (visualized in Figure 3). In the 2-dimensional case, the instance offers "columns" (C_j) which are dense for small cardinalities, but ultimately inefficient compared to the "rows" (R_i) that form the optimal solution for larger cardinalities. Analogously, our high-dimensional instance provides groups of elements aligned with the i^{th} dimension (group $\mathcal{U}^{(i)}$) which are locally dense, but less efficient than the groups aligned with the $(i+1)^{th}$ dimension (group $\mathcal{U}^{(i+1)}$) that constitute the optimal solution for the corresponding larger cardinality constraint s_{i+1}. This effectively generalizes the geometric conflict depicted in Figure 3 to q dimensions. In Section 4.2.2, we provide the precise definition of this lower bound instance.

The Behavior of Clever-Greedy on $(\mathcal{U}, f)$. By picking the appropriate values for s_i and v_i, we can ensure that Clever-Greedy behaves very predictably. Due to the constructed densities, the algorithm selects elements sequentially, exhausting the first group $\mathcal{U}^{(1)}$ before moving to $\mathcal{U}^{(2)}$, and so on (see Lemma A.8). Consequently, the algorithm accumulates value slowly relative to the number of items chosen. It reaches an objective value of v_i only after expending a cumulative cardinality of $\sum_{j=1}^{i} s_j$.

The Structure of the Optimal Solution. In contrast, the optimum solution with value v_i is extremely efficient. For a cardinality constraint of s_i, the optimum simply selects the group $\mathcal{U}^{(i)}$, achieving the objective value v_i using only s_i elements (see Lemma A.5). This discrepancy forces the algorithm to lag behind the optimum in terms of cardinality overhead. By analyzing the ratio between the values of the optimal solution and our algorithm's solution at the cardinality $k = s_q$, we establish a lower bound of 1.3724 (see Lemma 4.4). Figure 4 provides an illustration of the lower bound obtained by comparing the optimal solution with the solution maintained by our algorithm on the instance $(\mathcal{U}, f)$.

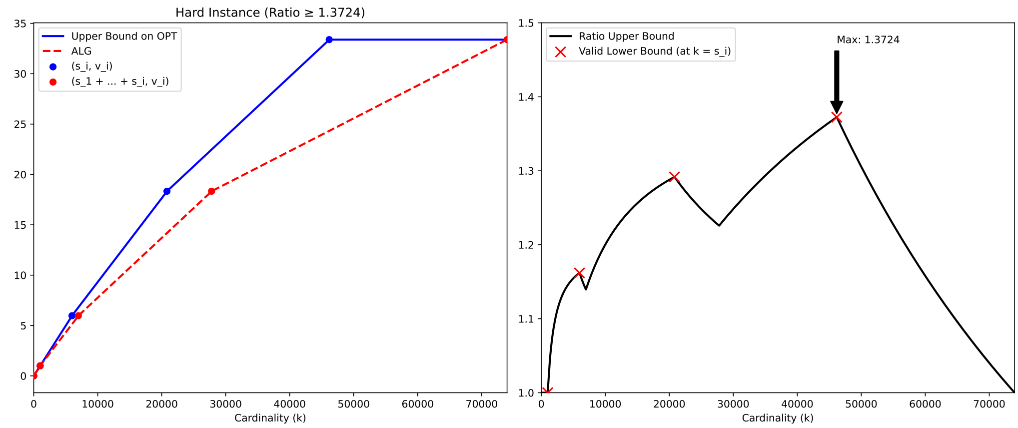

Figure 4 Visualization of the hard instance for our algorithm. **Left:** The solid blue line depicts the concave envelope of the points (s_i, v_i), which serves as an upper bound on $\textsc{Opt}(k)$ for all k, and equals $\textsc{Opt}(k)$ when $k = s_i$. The dashed red line (ALG) shows the algorithm's performance in this instance; it reaches the value v_i only after expending a cumulative cardinality of $\sum_{j=1}^{i} s_j$ (red dots). **Right:** The ratio between the upper bound on $\textsc{Opt}(k)$ and the algorithm's objective. The crosses mark the specific cardinalities $k = s_i$ where the upper bound coincides with $\textsc{Opt}(k)$, establishing the valid lower bound of ≥ 1.3724.

4.2.2 The Formal Lower Bound Instance

We begin by constructing a collection of subsets, which we will use in order to define a submodular function. Let $v_1, \dots, v_q \in \mathbb{R}_{>0}$ be a sequence of non-decreasing *volumes* and $s_1, \dots, s_q \in \mathbb{Z}_{>0}$ a sequence of non-decreasing *quantities*. For each $i \in [q], \ell \in [s_i]$, we define the set $E_\ell^{(i)} \subseteq \mathbb{R}^{q+1}$ as

$$E_\ell^{(i)} := [0, v_i) \times [0,1)^{i-1} \times \left[\frac{\ell - 1}{s_i}, \frac{\ell}{s_i}\right) \times [0,1)^{q-i}.$$

We let $\mathcal{U}^{(i)} := \{E_\ell^{(i)} \mid \ell \in [s_i]\}$ and $\mathcal{U} := \bigcup_{i \in [q]} \mathcal{U}^{(i)}$ denote the collection of these $(q+1)$-dimensional half-open rectangles. For notational convenience, we define $E^{(i)} := \bigcup_{\ell \in [s_i]} E_\ell^{(i)} = [0, v_i) \times [0,1)^q$, $v_0 := 0$, $s_0 := 0$, and let $n := |\mathcal{U}| = \sum_{i=1}^{q} s_i$ denote the size of the collection $\mathcal{U}$.

The proof of the following lemma is deferred to Appendix A.

▶ **Lemma 4.3.** *The collection of subsets $\mathcal{U}$ satisfies the following properties:*

1. *For all $1 \leq i \leq q$, $\ell, \ell' \in [s_i]$, we have $\operatorname{vol}(E_\ell^{(i)}) = v_i / s_i$ and $E_\ell^{(i)} \cap E_{\ell'}^{(i)} = \emptyset$ if $\ell \neq \ell'$.*
2. *For all $1 < i \leq q$, C an arbitrary union of subsets in $\mathcal{U}^{(1)} \cup \dots \cup \mathcal{U}^{(i-1)}$, and $E \in \mathcal{U}^{(i)}$, we have that $\operatorname{vol}(E \cap C) = \operatorname{vol}(C)/s_i$.*

The Lower Bound Instance. Consider sequences of non-decreasing volumes $v_1, \dots, v_q \in \mathbb{R}_{>0}$ and non-decreasing quantities $s_1, \dots, s_q \in \mathbb{Z}_{>0}$, satisfying the following condition for each $1 \leq i < q$:

$$\frac{v_{i+1} - v_{i-1}}{s_{i+1}} \leq \frac{v_i - v_{i-1}}{s_i}. \tag{10}$$

Let $\mathcal{U}$ be the collection of subsets defined by these sequences, and let f be the geometric coverage function defined over the collection of subsets $\mathcal{U}$. In other words, for some $S \subseteq \mathcal{U}$, we have $f(S) = \operatorname{vol}\left(\bigcup_{E \in S} E\right)$.

Our Lower Bound. The following lemma, which we prove in Appendix A.1, formalizes the intuition described in Section 4.2.1.

▶ **Lemma 4.4.** *For the objective f defined above such that* (10) *holds, Algorithm 2 produces a solution with competitive ratio at least*

$$v_q \cdot \left(v_{i^\star} + \frac{r^\star}{s_{i^\star+1}} \cdot (v_{i^\star+1} - v_{i^\star})\right)^{-1},$$

where $i^\star \in [q]$ is the unique value such that $s_1 + \cdots + s_{i^\star} + r^\star = s_q$, for $r^\star \le s_{i^\star+1}$.

By plugging in some specific values[5] for the v_i and s_i in Lemma 4.4, we can prove Theorem 4.1 (see Appendix A).

References

1. Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. Streaming submodular maximization: Massive data summarization on the fly. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 671–680, 2014. doi:10.1145/2623330.2623637.
2. Aaron Bernstein, Yann Disser, Martin Groß, and Sandra Himburg. General bounds for incremental maximization. *Mathematical Programming*, 191(2):953–979, 2022. doi:10.1007/s10107-020-01576-0.
3. Andrew An Bian, Joachim M. Buhmann, Andreas Krause, and Sebastian Tschiatschek. Guarantees for greedy maximization of non-submodular functions with applications. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 498–507, 2017. URL: https://proceedings.mlr.press/v70/bian17a.html.
4. Marcin Bienkowskis, Joakim Blikstad, Jarosław Byrka, Martín Costa, Yann Disser, and Annette Lutz. Incremental submodular maximization: Better than greedy. *arXiv preprint arXiv:2606.28558.*, 2026. arXiv:2606.28558.
5. Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1433–1452, 2014. doi:10.1137/1.9781611973402.106.
6. Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. A tight linear time (1/2)-approximation for unconstrained submodular maximization. *SIAM Journal on Computing*, 44(5):1384–1402, 2015. doi:10.1137/130929205.
7. Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011. doi:10.1137/080733991.
8. Abhimanyu Das and David Kempe. Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection. *Journal of Machine Learning Research*, 19(3):1–34, 2018. URL: https://jmlr.org/papers/v19/16-534.html.
9. Yann Disser, Max Klimm, Kevin Schewior, and David Weckbecker. Incremental maximization via continuization. In *Proceedings of the 50th International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 47:1–47:17, 2023. doi:10.4230/LIPIcs.ICALP.2023.47.
10. Yann Disser and David Weckbecker. Unified greedy approximability beyond submodular maximization. *SIAM Journal on Discrete Mathematics*, 38(1):348–379, 2024. doi:10.1137/22M1526952.

[5] Which we found using a computer.

11 Yann Disser and David Weckbecker. Incremental maximization for a broad class of objectives. In *Proceedings of the 33rd European Symposium on Algorithms (ESA)*, pages 92:1–92:14, 2025. doi:10.4230/LIPIcs.ESA.2025.92.

12 Ethan R. Elenberg, Rajiv Khanna, Alexandros G. Dimakis, and Sahand Negahban. Restricted strong convexity implies weak submodularity. *Annals of Statistics*, 46(6B):3539–3568, 2018. doi:10.1214/17-AOS1679.

13 Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998. doi:10.1145/285055.285059.

14 Uriel Feige, Vahab S. Mirrokni, and Jan Vondrák. Maximizing non-monotone submodular functions. *SIAM Journal on Computing*, 40(4):1133–1153, 2011. doi:10.1137/090779346.

15 Chris Harshaw, Moran Feldman, Justin Ward, and Amin Karbasi. Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pages 2634–2643, 2019. URL: http://proceedings.mlr.press/v97/harshaw19a.html.

16 George L. Nemhauser and Laurence A. Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of Operations Research*, 3(3):177–188, 1978. doi:10.1287/moor.3.3.177.

17 George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978. doi:10.1007/BF01588971.

18 Richard Rado. A theorem on independence relations. *The Quarterly Journal of Mathematics*, 13(1):83–89, 1942. doi:10.1093/qmath/os-13.1.83.

19 Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Operartions Research Letters*, 32:41–43, 2004. doi:10.1016/s0167-6377(03)00062-2.

20 Jan Vondrák. Symmetry and approximability of submodular maximization problems. *SIAM Journal on Computing*, 42(1):265–304, 2013. doi:10.1137/110832318.

21 David Weckbecker. *Competitive Analysis for Incremental Maximization*. PhD thesis, Technische Universität Darmstadt, 2024. doi:10.26083/tuprints-00026447.

A Missing Proofs in Lower Bound Analysis (Section 4.2)

By plugging in some specific values[6] for the v_i and s_i in Lemma 4.4, shown in Appendix A.1, we can prove Theorem 4.1.

▶ **Theorem 4.1.** *Algorithm 2 has a competitive ratio of at least* 1.3724.

Proof. Consider the following sequences for $q = 4$:

$$(s_i)_{i=1}^q = (10000, 59683, 208018, 461378),$$

$$(v_i)_{i=1}^q = (1.0000, 5.9683, 18.3164, 33.3561).$$

It is easy to verify that these sequences satisfy the condition from (10). We can also compute the values of $i^\star = 3$ and $r^\star = 183677$ from Lemma 4.4, yielding a lower bound on the competitive ratio of at least 1.3724. Figure 4 gives a plot with these values. ◀

▶ **Lemma 4.3.** *The collection of subsets $\mathcal{U}$ satisfies the following properties:*

1. *For all $1 \le i \le q$, $\ell, \ell' \in [s_i]$, we have $\mathrm{vol}\big(E_\ell^{(i)}\big) = v_i/s_i$ and $E_\ell^{(i)} \cap E_{\ell'}^{(i)} = \emptyset$ if $\ell \neq \ell'$.*
2. *For all $1 < i \le q$, C an arbitrary union of subsets in $\mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(i-1)}$, and $E \in \mathcal{U}^{(i)}$, we have that $\mathrm{vol}(E \cap C) = \mathrm{vol}(C)/s_i$.*

[6] Which we found using a computer.

Proof. For each $i \in [q]$, $\ell \in [s_i]$, we can see that

$$\mathrm{vol}\big(E^{(i)}_\ell\big) = v_i \cdot 1^{i-1} \cdot \frac{1}{s_i} \cdot 1^{q-i} = \frac{v_i}{s_i}.$$

Additionally, for any $\ell' \in [s_i]$ such that $\ell \neq \ell'$, we have that

$$E^{(i)}_\ell \cap E^{(i)}_{\ell'} = [0, v_i) \times [0, 1)^{i-1} \times \emptyset \times [0, 1)^{q-i} = \emptyset.$$

Now, consider some $1 < i \leq q$, a subset C obtained by taking an arbitrary union of subsets in $\mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(i-1)}$, and $E = E^{(i)}_\ell \in \mathcal{U}^{(i)}$. Let $C' \subseteq [0, v_{i-1}) \times [0, 1)^{i-1}$ denote the subset such that $C = C' \times [0, 1)^{q-i+1}$. We can see that

$$E \cap C = E \cap (C' \times [0, 1)^{q-i+1}) = C' \times \left[\frac{\ell - 1}{s_i}, \frac{\ell}{s_i}\right) \times [0, 1)^{q-i}.$$

Thus, we have that

$$\mathrm{vol}(E \cap C) = \mathrm{vol}(C') \cdot \frac{1}{s_i} = \mathrm{vol}(C) \cdot \frac{1}{s_i}.$$ ◀

A.1 Proof of Lemma 4.4

We begin by deriving some useful properties of the instance, then proceed to analyze the structure of the optimal solution, followed by an analysis of the solution constructed by Clever-Greedy.

A.1.1 Useful Properties of the Instance

The following claims are immediate applications of (10).

▷ Claim A.1. For each $1 \leq i \leq j \leq q$, we have that

$$\frac{v_j - v_{i-1}}{s_j} \leq \frac{v_i - v_{i-1}}{s_i}.$$

Proof. Let $i \in [q]$ and, for each $i \leq j \leq q$, define

$$R_j := \frac{v_j - v_{i-1}}{s_j}.$$

We prove by induction that $R_i \geq R_{i+1} \geq \cdots \geq R_q$. Suppose we have that $R_i \geq \cdots \geq R_j$, for some $i \leq j < q$. Then we have that

$$\begin{aligned} R_{j+1} &= \frac{v_{j+1} - v_{i-1}}{s_{j+1}} = \frac{v_{j+1} - v_{j-1}}{s_{j+1}} + \frac{v_{j-1} - v_{i-1}}{s_{j+1}} \\ &\leq \frac{v_j - v_{j-1}}{s_j} + \frac{v_{j-1} - v_{i-1}}{s_j} = \frac{v_j - v_{i-1}}{s_j} = R_j, \end{aligned}$$

where we are applying (10) and the fact that $s_j \leq s_{j+1}$. ◁

▷ Claim A.2. For each $1 \leq i < q$, we have that $v_i/s_i \geq v_{i+1}/s_{i+1}$.

Proof. We have that

$$\frac{v_i}{s_i} - \frac{v_{i+1}}{s_{i+1}} = \left(\frac{v_i - v_{i-1}}{s_i} - \frac{v_{i+1} - v_{i-1}}{s_{i+1}}\right) + v_{i-1} \cdot \left(\frac{1}{s_i} - \frac{1}{s_{i+1}}\right) \geq 0,$$

where the first term is greater than 0 by (10), and the second term since $s_{i+1} \geq s_i$. ◁

A.1.2 Analysis of the Optimal Solution

Given a subset $S \subseteq \mathcal{U}$, we define $\chi(S) \coloneqq \max\left\{j \in [q] \mid S \cap \mathcal{U}^{(j)} \neq \emptyset\right\}$, i.e., $\chi(S)$ is the maximum index $j \in [q]$ such that S contains a set from $\mathcal{U}^{(j)}$. Additionally, for each $j \in [q]$, we define $S_{\leq j} \coloneqq S \cap (\mathcal{U}^{(0)} \cup \cdots \cup \mathcal{U}^{(j)})$.

We begin by deriving the following useful identity for each $S \subseteq \mathcal{U}$.

▷ Claim A.3. For each $j \in [q]$, we have that

$$f(S_{\leq j}) = \frac{|S \cap \mathcal{U}^{(j)}|}{s_j} \cdot (v_j - f(S_{\leq j-1})) + f(S_{\leq j-1}). \tag{11}$$

Proof. For each $j' \in [q]$, let $a_{j'} \coloneqq |S \cap \mathcal{U}^{(j')}|$. Let $C \coloneqq \bigcup_{E \in S_{\leq j-1}} E$ and $B = \bigcup_{E \in S \cap \mathcal{U}^{(j)}} E$ denote the union of the sets in $S_{\leq j-1}$ and $S_{\leq j} \setminus S_{\leq j-1}$, respectively. Applying Lemma 4.3 and $\mathrm{vol}(C) = f(S_{\leq j-1})$ by definition, we have that

$$\begin{aligned} f(S_{\leq j}) = \mathrm{vol}(B \cup C) &= \mathrm{vol}(B) + \mathrm{vol}(C) - \mathrm{vol}(B \cap C) \\ &= \frac{a_j}{s_j} \cdot \mathrm{vol}(E^{(j)}) + \mathrm{vol}(C) - \sum_{E \in S \cap \mathcal{U}^{(j)}} \frac{1}{s_j} \cdot \mathrm{vol}(E \cap C) \\ &= \frac{a_j}{s_j} \cdot \mathrm{vol}(E^{(j)}) + f(S_{\leq j-1}) - \frac{a_j}{s_j} \cdot \mathrm{vol}(C) \\ &= \frac{a_j}{s_j} \cdot (v_j - f(S_{\leq j-1})) + f(S_{\leq j-1}). \end{aligned}$$ ◁

Using (11), we now prove the following key lemma.

▶ **Lemma A.4.** *Let $S \subseteq \mathcal{U}$ be a subset of size at most s_i. If $\chi(S) > i$, then there exists a subset $S' \subseteq \mathcal{U}$ of size $|S|$ such that $f(S') \geq f(S)$ and $\chi(S') < \chi(S)$.*

Proof. For each $j' \in [q]$, let $a_{j'} \coloneqq |S \cap \mathcal{U}^{(j')}|$. Now, let $j \coloneqq \chi(S)$. Let S' be the set obtained by adding any a_j elements from $\mathcal{U}^{(j-1)} \setminus S_{\leq j-1}$ to $S_{\leq j-1}$. Note that $|\mathcal{U}^{(j-1)} \setminus S_{\leq j-1}| \geq a_j$, so this is always possible. Clearly, $\chi(S') = j - 1 < j = \chi(S)$. It remains to show that $f(S') \geq f(S)$.

Proof that $f(S') \geq f(S)$. Similarly to (11), by applying Lemma 4.3, we can obtain the following identity:

$$f(S') = \frac{a_j}{s_{j-1}} \cdot (v_{j-1} - f(S_{\leq j-2})) + f(S_{\leq j-1}). \tag{12}$$

Now we can lower bound $f(S') - f(S)$ as follows:

$$
\begin{aligned}
f(S') - f(S) &= f(S') - f(S_{\leq j}) \\
&= \frac{a_j}{s_{j-1}} \cdot (v_{j-1} - f(S_{\leq j-2})) + f(S_{\leq j-1}) - \frac{a_j}{s_j} \cdot (v_j - f(S_{\leq j-1})) - f(S_{\leq j-1}) \\
&= \frac{a_j}{s_{j-1}} \cdot (v_{j-1} - f(S_{\leq j-2})) - \frac{a_j}{s_j} \cdot (v_j - f(S_{\leq j-1})) \\
&\geq \frac{a_j}{s_{j-1}} \cdot (v_{j-1} - f(S_{\leq j-2})) - \frac{a_j}{s_j} \cdot (v_j - f(S_{\leq j-2})) \\
&= a_j \cdot \left(\frac{v_{j-1}}{s_{j-1}} - \frac{v_j}{s_j} + \left(\frac{1}{s_j} - \frac{1}{s_{j-1}} \right) \cdot f(S_{\leq j-2}) \right) \\
&\geq a_j \cdot \left(\frac{v_{j-1}}{s_{j-1}} - \frac{v_j}{s_j} + \left(\frac{1}{s_j} - \frac{1}{s_{j-1}} \right) \cdot v_{j-2} \right) \\
&= a_j \cdot \left(\frac{v_{j-1} - v_{j-2}}{s_{j-1}} - \frac{v_j - v_{j-2}}{s_j} \right) \geq 0.
\end{aligned}
$$

The first inequality follows from $f(S_{\leq j-2}) \leq f(S_{\leq j-1})$ by monotonicity. The second inequality follows from $f(S_{\leq j-2}) \leq f(\mathcal{U}^{(j-2)}) = v_{j-2}$ and the observation that $(1/s_j - 1/s_{j-1}) \leq 0$ since $s_j \geq s_{j-1}$. The third inequality follows from (10). This concludes the proof. ◀

The following lemma shows that $\mathcal{U}^{(i)}$ is an optimal solution for $k = s_i$.

▶ **Lemma A.5.** *For each $i \in [q]$, we have that $\textsc{Opt}(s_i) = v_i$.*

Proof. We have that $v_i = \mathrm{vol}(E^{(i)}) = f(\mathcal{U}^{(i)}) \leq \textsc{Opt}(s_i)$, since $|\mathcal{U}^{(i)}| = s_i$. Thus, it suffices to show that $\textsc{Opt}(s_i) \leq v_i$.

Given any subset $S \subseteq \mathcal{U}$ of size s_i, we can repeatedly apply Lemma A.4 to the set S to obtain a subset $S' \subseteq \mathcal{U}$ of size s_i such that $f(S') \geq f(S)$ and $\chi(S') \leq i$. Since we have that $\bigcup_{E \in S'} E \subseteq \bigcup_{j=1}^{i} E^{(j)} = E^{(i)}$, it follows that

$$f(S) \leq f(S') \leq f(\mathcal{U}^{(i)}) = v_i. \qquad ◀$$

The proof of the following Lemma is deferred to the full version of this paper [4].

▶ **Lemma A.6.** *For each $k \in [n]$ such that $s_{i-1} \leq k \leq s_i$, we have that*

$$\textsc{Opt}(k) \leq v_{i-1} + \frac{k - s_{i-1}}{s_i - s_{i-1}} \cdot (v_i - v_{i-1}).$$

▷ Claim A.7. For each $i \in [q]$, we have that

$$\max_{k \in [n]} \left(\frac{\textsc{Opt}(k) - v_{i-1}}{k} \right) = \frac{v_i - v_{i-1}}{s_i}.$$

Proof. Let $R(k) \coloneqq (\textsc{Opt}(k) - v_{i-1})/k$ denote the ratio we wish to maximize. We begin by showing that $R(k)$ attains its maximum value for some k in $\{s_1, \ldots, s_q\} \subseteq [n]$.

Reducing the Search Space. We first note that $R(k)$ attains its maximum for some $k \leq s_q$ since $\textsc{Opt}(s_q) \geq f(E^{(q)}) = \textsc{Opt}(n)$. Furthermore, for $k \leq s_1$, we have that

$$R(k) = \frac{k \cdot (v_1/s_1) - v_{i-1}}{k} = \frac{v_1}{s_1} - \frac{v_{i-1}}{k} \leq R(s_1),$$

so $R(k)$ attains its maximum for some $k \geq s_1$. Now, consider some $k \in [s_1, s_q]$, and let $j > 1$ be an index such that $s_{j-1} \leq k \leq s_j$. By Lemma A.6, we have that

$$\textsc{Opt}(k) \leq L(k) \coloneqq v_{j-1} + \frac{k - s_{j-1}}{s_j - s_{j-1}}(v_j - v_{j-1}).$$

Now, consider the following upper bound on $R(k)$:

$$\tilde{R}(k) := \frac{L(k) - v_{i-1}}{k}.$$

Within the interval $[s_{j-1}, s_j]$, $L(k)$ is a line of the form $Ak + B$, so it follows that $\tilde{R}(k)$ can be expressed as $A + C/k$, which is a monotonic function of k. Since a monotonic function defined on a closed interval attains its maximum value at one of the endpoints, it follows that, for any k in this interval,

$$R(k) \leq \tilde{R}(k) \leq \max\left(\tilde{R}(s_{j-1}), \tilde{R}(s_j)\right)$$

Since $\tilde{R}(s_{j-1}) = R(s_{j-1})$ and $\tilde{R}(s_j) = R(s_j)$ by Lemma A.5, we have that

$$\max_{k \in [s_{j-1}, s_j]} R(k) = \max\{R(s_{j-1}), R(s_j)\}.$$

Consequently, the global maximum of $R(k)$ must occur for some $k \in \{s_1, \ldots, s_q\}$.

Identifying the Maximum Value. Now, consider some $j < i$. Since the v_j are non-decreasing, we have that

$$R(s_j) = (v_j - v_{i-1})/s_j \leq 0 \leq (v_i - v_{i-1})/s_i = R(s_i).$$

For $j > i$, it follows from Claim A.1 that $R(s_j) \leq R(s_i)$. Thus, we have that

$$\max_{1 \leq j \leq q} R(s_j) = R(s_i). \qquad \triangleleft$$

A.1.3 Analysis of Clever-Greedy

The following lemma summarizes the behavior of CLEVER-GREEDY on this lower bound instance.

▶ **Lemma A.8.** *Given the input $(\mathcal{U}, f)$, CLEVER-GREEDY runs for q phases, where the solution maintained by the algorithm at the end of the i^{th} phase is $\mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(i)} \subseteq \mathcal{U}$.*

Proof. We prove this lemma by induction on the phase i.

Inductive Step. Let $i \in [q]$ and consider the state of the algorithm at the start of phase i. Assume that the length of each phase $j < i$, k_j, was exactly s_j, and that the subset of elements added to the solution S during this phase was $\mathcal{U}^{(j)}$. Thus, the solution S at the start of phase i is $\mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(j)}$.[7] Now, the algorithm chooses k that maximizes the value of

$$\frac{\textsc{Opt}(k) - \textsc{Opt}(k_{i-1})}{k} = \frac{\textsc{Opt}(k) - v_{i-1}}{k} \qquad \text{(by Lemma A.5)}$$

and sets this to be k_i. By Claim A.7, the maximum value of this expression is $(v_i - v_{i-1})/s_i$, which is precisely the value of this expression when $k = s_i$. Thus, the algorithm sets $k_i = s_i$.[8] Since $\textsc{Opt}(s_i) = v_i = f(\mathcal{U}^{(i)})$, we have that $S^\star_{k_i} = \mathcal{U}^{(i)}$. The phase continues until $S^\star_{k_i} \subseteq S$. Since $\mathcal{U}^{(i)}$ is disjoint from the solution S at the start of the phase, the phase goes on for s_i iterations, and terminates with $S = \mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(i)}$.

[7] If $i = 0$, the solution S is empty, and we define $k_0 = 0$.
[8] We assume that we break ties arbitrarily in our favor.

Conclusion. At the end of phase q, we have that $S = \mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(q)} = \mathcal{U}$, and the algorithm terminates. ◀

A.1.4 Concluding the Proof of Lemma 4.4

We are now ready to prove Lemma 4.4 by upper bounding the objective of our algorithm and lower bounding the value of the optimal solution at some point during the execution of the algorithm.

Upper Bounding Our Objective. Let $i^\star \in [q]$ be the unique value such that $s_1 + \cdots + s_{i^\star} + r^\star = s_q$, for $r^\star \leq s_{i^\star+1}$, and let $\tau = s_q$. Consider the subset S consisting of the first τ elements in the incremental solution returned by Clever-Greedy when given $(f,\mathcal{U})$ as input. It follows from Lemma A.8 that

$$\mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(i^\star)} \subseteq S \subseteq \mathcal{U}^{(1)} \cup \cdots \cup \mathcal{U}^{(i^\star+1)}.$$

Thus, by Lemma 4.3, we have that

$$f(S) = \mathrm{vol}\left(E^{(i^\star)}\right) + \frac{r^\star}{s_{i^\star+1}} \cdot \left(\mathrm{vol}\left(E^{(i^\star+1)}\right) - \mathrm{vol}\left(E^{(i^\star)}\right)\right) = v_{i^\star} + \frac{r^\star}{s_{i^\star+1}} \cdot (v_{i^\star+1} - v_{i^\star})\,.$$

Lower Bounding the Optimum. It follows from Lemma A.5 that $\textsc{Opt}(\tau) = \textsc{Opt}(s_q) = v_q$.

Lower Bounding the Competitive Ratio. Putting these bounds together, it follows that the competitive ratio of our algorithm is at least

$$\frac{\textsc{Opt}(\tau)}{f(S)} = v_q \cdot \left(v_{i^\star} + \frac{r^\star}{s_{i^\star+1}} \cdot (v_{i^\star+1} - v_{i^\star})\right)^{-1}.$$