



On Maximizing Sums of Non-monotone Submodular and Linear Functions

Benjamin Qi¹

Received: 16 February 2023 / Accepted: 9 October 2023 / Published online: 13 November 2023
© The Author(s) 2023

Abstract

We study the problem of Regularized Unconstrained Submodular Maximization (RegularizedUSM) as defined by Bodek and Feldman (Maximizing sums of non-monotone submodular and linear functions: understanding the unconstrained case, [arXiv:2204.03412](https://arxiv.org/abs/2204.03412), 2022): given query access to a non-negative submodular function $f: 2^{\mathcal{N}} \rightarrow \mathbb{R}_{\geq 0}$ and a linear function $\ell: 2^{\mathcal{N}} \rightarrow \mathbb{R}$ over the same ground set \mathcal{N} , output a set $T \subseteq \mathcal{N}$ approximately maximizing the sum $f(T) + \ell(T)$. An algorithm is said to provide an (α, β) -approximation for RegularizedUSM if it outputs a set T such that $\mathbb{E}[f(T) + \ell(T)] \geq \max_{S \subseteq \mathcal{N}} [\alpha \cdot f(S) + \beta \cdot \ell(S)]$. We also consider the setting where S and T are constrained to be independent in a given matroid, which we refer to as Regularized *Constrained* Submodular Maximization (RegularizedCSM). The special case of RegularizedCSM with monotone f has been extensively studied (Sviridenko et al. in *Math Oper Res* 42(4):1197–1218, 2017; Feldman in *Algorithmica* 83(3):853–878, 2021; Harshaw et al., in: *International conference on machine learning*, PMLR, 2634–2643, 2019), whereas we are aware of only one prior work that studies RegularizedCSM with non-monotone f (Lu et al. in *Optimization* 1–27, 2023), and that work constrains ℓ to be non-positive. In this work, we provide improved (α, β) -approximation algorithms for both RegularizedUSM and RegularizedCSM with non-monotone f . Specifically, we are the first to provide nontrivial (α, β) -approximations for RegularizedCSM where the sign of ℓ is unconstrained, and the α we obtain for RegularizedUSM improves over (Bodek and Feldman in *Maximizing sums of non-monotone submodular and linear functions: understanding the unconstrained case*, [arXiv:2204.03412](https://arxiv.org/abs/2204.03412), 2022) for all $\beta \in (0, 1)$. We also prove new inapproximability results for RegularizedUSM and RegularizedCSM, as well as 0.478-inapproximability for maximizing a sub-

A conference version of this paper that focused on results from Sects. 4 and 8 previously appeared in ISAAC 2022.

✉ Benjamin Qi
bqi343@gmail.com

¹ Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA 02139, USA

modular function where S and T are subject to a cardinality constraint, improving a 0.491-inapproximability result due to Oveis Gharan and Vondrak (in: Proceedings of the twenty-second annual ACM-SIAM symposium on discrete algorithms, SIAM, pp 1098–1116, 2011).

Keywords Submodular maximization · Regularization · Continuous greedy · Double greedy · Inapproximability

1 Introduction

Submodularity is a property satisfied by many fundamental set functions, including coverage functions, matroid rank functions, and directed cut functions. Optimization of submodular set functions has found a wealth of applications in machine learning, including the spread of influence in social networks [1], sensor placement [2], information gathering [3], document summarization [4–6], image segmentation [7], and multi-object tracking [8], among others (see Krause and Golovin [9] for a survey).

Problems involving the maximization of non-negative submodular functions can be classified as either *unconstrained* or *constrained*. In the unconstrained case, the objective is to return a set in the domain of the function approximately maximizing the function; we refer to this problem as USM. In the most commonly studied form of constrained submodular maximization, the returned set is subject to a “matroid constraint,” which means that the returned set is constrained to be independent in a given matroid. We refer to this form of constrained submodular maximization as CSM. The simplest nontrivial example of a matroid constraint is a *cardinality* constraint, which means that an upper bound is given on the allowed size of the returned set. Additionally, we refer to the special case where the function f we are maximizing is monotone as “monotone CSM.”

Approximation algorithms for both USM and CSM have been studied extensively. Say that an algorithm running in polynomial time with respect to the size of the ground set provides an α -approximation if it returns a set with expected value at least α times that of the optimum. For USM, a 0.5-approximation algorithm was provided by Buchbinder et al. [10]. For monotone CSM, a $(1 - e^{-1})$ -approximation was achieved by Nemhauser et al. [11] using a *greedy* algorithm for the special case of cardinality constraints and later generalized by Calinescu et al. [12] to matroid constraints using the *continuous greedy* algorithm. For general CSM, the *measured continuous greedy* algorithm of Feldman et al. [13] achieves an $e^{-1} > 0.367$ -approximation, and a subsequent algorithm due to Buchbinder and Feldman [14] achieves a 0.385-approximation.

To bound how far the algorithms from the previous paragraph are from optimal, corresponding inapproximability results have been shown. Say that a problem is α -inapproximable if no algorithm running in sub-exponential time with respect to the size of the ground set can provide an α -approximation. The first two approximation factors from the previous paragraph are in fact the best achievable; $(0.5 + \epsilon)$ -inapproximability and $(1 - e^{-1} + \epsilon)$ -inapproximability for any $\epsilon > 0$ were shown by Feige et al. [15] and Nemhauser and Wolsey [16], respectively, using ad hoc methods. On the other hand, the best achievable approximability for general CSM remains open; the best-known

inapproximability factor is 0.478 due to Oveis Gharan and Vondrak [17] using the *symmetry gap* technique of Vondrak [18]. This technique has the advantage of being able to succinctly re-prove the inapproximability results of [15, 16] and many others.

In this work, we study approximation algorithms for maximizing the sum of a non-negative submodular function f and a linear function ℓ . Sviridenko et al. [19] were the first to study algorithms for the sum $f + \ell$ in the case of f monotone, in order to provide improved approximation algorithms for monotone CSM with bounded *curvature*. Here, the curvature $c \in [0, 1]$ of a non-negative monotone submodular function g is roughly a measure of how far g is from being linear. They provided a $(1 - c/e - \epsilon)$ -approximation algorithm and a complementary $(1 - c/e + \epsilon)$ -inapproximability result. The idea of the algorithm is to decompose g into $f + \ell$ and show that an approximation factor of $1 - e^{-1}$ can be achieved with respect to f and an approximation factor of 1 can be achieved with respect to ℓ simultaneously. Formally, if \mathcal{I} is the independent set family of a matroid, the algorithm computes a set $T \in \mathcal{I}$ that satisfies

$$\mathbb{E}[g(T)] = \mathbb{E}[f(T) + \ell(T)] \geq \max_{S \in \mathcal{I}} [(1 - e^{-1} - \epsilon)f(S) + (1 - \epsilon)\ell(S)] \quad (1.1)$$

by first “guessing” the value of $\ell(S)$, and then running the continuous greedy algorithm. Subsequently, Feldman eliminated the need for the guessing step and the dependence on $\epsilon\ell(S)$ by introducing a *distorted objective* [20]. Many faster algorithms and practical applications for the case of f monotone have since been introduced [21–23]. Note that ℓ has several potential interpretations; while setting ℓ to be non-negative provides improved approximations for monotone submodular functions with low curvature, setting ℓ to be non-positive allows it to serve as a *regularizer* or *soft constraint* that favors returning smaller sets as suggested by Harshaw et al. [21].

On the other hand, we know of only two prior works that study RegularizedUSM where f is not constrained to be monotone. Bodek and Feldman [24] were the first to consider the case where f is non-monotone and the sign of ℓ is unconstrained. They defined and studied the problem of Regularized Unconstrained Submodular Maximization (RegularizedUSM):

Definition 1.1 (RegularizedUSM [24]) Given query access to a (not necessarily monotone) non-negative submodular function $f: 2^{\mathcal{N}} \rightarrow \mathbb{R}_{\geq 0}$ and a linear function $\ell: 2^{\mathcal{N}} \rightarrow \mathbb{R}$ over the same ground set \mathcal{N} , an algorithm is said to provide an (α, β) -approximation for RegularizedUSM if it outputs a set $T \subseteq \mathcal{N}$ such that $\mathbb{E}[f(T) + \ell(T)] \geq \max_{S \subseteq \mathcal{N}} [\alpha \cdot f(S) + \beta \cdot \ell(S)]$.

The main approximation result of [24] is the first non-trivial approximation algorithm for RegularizedUSM with f non-monotone and the sign of ℓ unconstrained. Specifically, they used *non-oblivious local search* to provide $(\alpha(\beta) - \epsilon, \beta - \epsilon)$ -approximations for RegularizedUSM for all $\beta \in (0, 1]$, where $\alpha(\beta) \triangleq \beta(1 - \beta)/(1 + \beta)$ [24, Theorem 1.2]. They also proved inapproximability results for the cases of ℓ non-negative and ℓ non-positive using the symmetry gap technique of Vondrak [18]. In particular, they showed $(1 - e^{-\beta} + \epsilon, \beta)$ -inapproximability for monotone f and non-positive ℓ for all $\beta \geq 0$ [24, Theorem 1.1], essentially matching the $(1 - e^{-\beta} - \epsilon, \beta)$ -approximability provided by Lu et al.’s *distorted measured contin-*

uous greedy algorithm [25] (note that (α, β) -inapproximability is defined in the same way as α -inapproximability).

In this work, we present improved approximability and inapproximability results for RegularizedUSM as well as the setting where S and T are subject to a matroid constraint, which we define analogously as Regularized Constrained Submodular Maximization (RegularizedCSM):

Definition 1.2 (RegularizedCSM) Given query access to a (not necessarily monotone) non-negative submodular function $f: 2^{\mathcal{N}} \rightarrow \mathbb{R}_{\geq 0}$ and a linear function $\ell: 2^{\mathcal{N}} \rightarrow \mathbb{R}$ over the same ground set \mathcal{N} , as well as a matroid with family of independent sets denoted by \mathcal{I} also over the same ground set, an algorithm is said to provide an (α, β) -approximation for RegularizedCSM if it outputs a set $T \in \mathcal{I}$ such that $\mathbb{E}[f(T) + \ell(T)] \geq \max_{S \in \mathcal{I}}[\alpha \cdot f(S) + \beta \cdot \ell(S)]$.

The only prior work considering RegularizedCSM for non-monotone f that we are aware of is that of Lu et al. [25], which as noted by [24] achieves $(\beta e^{-\beta} - \epsilon, \beta)$ -approximations for RegularizedCSM for all $\beta \in [0, 1]$, but only when ℓ is constrained to be non-positive.

Organization of the Paper. We present the definitions and notation used throughout this work in Sect. 2 and summarize our results in Sect. 3. Sections 4, 5, 6, 7 and 8 prove the results introduced in Sect. 3. We conclude with a discussion of open problems in Sect. 9.

2 Preliminaries

Set Functions. Let $\mathcal{N} \triangleq \{u_1, u_2, \dots, u_n\}$ denote the *ground set*. A set function $f: 2^{\mathcal{N}} \rightarrow \mathbb{R}$ is said to be *submodular* if for every two sets $S, T \subseteq \mathcal{N}$, $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$. Equivalently, f is said to be submodular if it satisfies the property of “diminishing returns.” That is, for every two sets $S \subseteq T \subseteq \mathcal{N}$ and an element $u \in \mathcal{N} \setminus T$, $f(u|S) \geq f(u|T)$, where $f(u|S) \triangleq f(S \cup \{u\}) - f(S)$ is the *marginal value* of u with respect to S . We use $f(u)$ as shorthand for $f(\{u\})$. All submodular functions are implicitly assumed to be non-negative unless otherwise stated.

A set function f is said to be *monotone* if for every two sets $S \subseteq T \subseteq \mathcal{N}$, $f(S) \leq f(T)$. A set function ℓ is said to be *linear* if there exist values $\{\ell_u \in \mathbb{R} \mid u \in \mathcal{N}\}$ such that for every set $S \subseteq \mathcal{N}$, $\ell(S) = \sum_{u \in S} \ell_u$. When considering the sum of a non-negative submodular function f and a linear function ℓ whose sign is unconstrained, define $\ell_+(S) \triangleq \sum_{u \in S} \max(\ell_u, 0)$ and $\ell_-(S) \triangleq \sum_{u \in S} \min(\ell_u, 0)$ to be the components of ℓ with positive and negative sign, respectively.

Value Oracles. We make the standard assumption that an algorithm for the sum $f + \ell$ does not have direct access to the representation of f ; instead, it may obtain information about f only through a *value oracle*. Given any query set $S \subseteq \mathcal{N}$, a value oracle for f returns $f(S)$ in polynomial time. The coefficients of ℓ are directly provided to the algorithm.

Multilinear Extensions. All vectors of reals are in bold (e.g., \mathbf{x}). Given two vectors $\mathbf{x}, \mathbf{y} \in [0, 1]^{\mathcal{N}}$, we define $\mathbf{x} \vee \mathbf{y}$, $\mathbf{x} \wedge \mathbf{y}$, and $\mathbf{x} \circ \mathbf{y}$ to be the coordinate-wise maximum,

minimum, and multiplication, respectively, of \mathbf{x} and \mathbf{y} . We also define $\mathbf{x} \setminus \mathbf{y} \triangleq \mathbf{x} - \mathbf{x} \wedge \mathbf{y}$. Given a set function $f: 2^{\mathcal{N}} \rightarrow \mathbb{R}$, its *multilinear extension* is the function $F: [0, 1]^{\mathcal{N}} \rightarrow \mathbb{R}$ defined by $F(\mathbf{x}) = \mathbb{E}[f(\mathcal{R}(\mathbf{x}))]$, where $\mathcal{R}(\mathbf{x})$ is a random subset of \mathcal{N} including every element $u \in \mathcal{N}$ with probability x_u , independently. One can verify that F is a multilinear function of its arguments as well an extension of f in the sense that $F(\mathbf{1}_S) = f(S)$ for every set $S \subseteq \mathcal{N}$. Here, $\mathbf{1}_S$ is the *characteristic vector* of the set S ; that is, the vector with value 1 at each $u \in S$ and 0 at each $u \in \mathcal{N} \setminus S$.

Matroid Polytopes. A matroid \mathcal{M} may be specified by a pair of a ground set \mathcal{N} and a family \mathcal{I} of independent sets. The *matroid polytope* \mathcal{P} corresponding to \mathcal{M} is defined to be $\text{conv}(\{\mathbf{1}_S \mid S \in \mathcal{I}\})$, where conv denotes the convex hull. Due to the matroid axioms, \mathcal{P} is guaranteed to be *down-closed*; that is, $0 \leq \mathbf{x} \leq \mathbf{y}$ and $\mathbf{y} \in \mathcal{P}$ imply $\mathbf{x} \in \mathcal{P}$. It is also well-known that \mathcal{P} is *solvable*; that is, linear functions can be maximized over \mathcal{P} in polynomial time [12, Section 2.3]. For CSM and RegularizedCSM, we let OPT denote any set such that $OPT \in \mathcal{I}$ (equivalently, $\mathbf{1}_{OPT} \in \mathcal{P}$), while for USM and RegularizedUSM, we let OPT denote any subset of \mathcal{N} . For example, in the context of CSM, $\mathbb{E}[f(T)] \geq \alpha f(OPT)$ is equivalent to $\forall S \in \mathcal{I}, \mathbb{E}[f(T)] \geq \alpha f(S)$. **Miscellaneous.** We let ϵ denote any positive real. Many of our algorithms are “almost” (α, β) approximations in the sense that they provide an $(\alpha - \epsilon, \beta)$ -approximation in $\text{poly}(n, \frac{1}{\epsilon})$ time for any $\epsilon > 0$. Similarly, some of our results show $(\alpha + \epsilon, \beta)$ -inapproximability for any $\epsilon > 0$.

2.1 Technical Tools

We obtain our results by carefully combining the following known techniques.

To show approximability, the main techniques we use are the *measured continuous greedy* introduced by Feldman et al. [13] and used by [14, 25], as well as the *distorted objective* introduced by Feldman [20] and used by [25]. For some of our approximability results, we additionally require the analysis of the 0.385-approximation algorithm for CSM due to Buchbinder et al. [14] and the “guessing step” used by Sviridenko et al. [19].

To show inapproximability, the main technique we use is the symmetry gap of Vondrak [18], and most of our symmetry gap constructions are based on those of Oveis Gharan and Vondrak [17].

Additional notes on all relevant prior work can be found in Sect. A.1.

3 Our Contributions

Our results are organized into five sections. Section 4 contains our inapproximability result for CSM. The remaining sections contain our results for RegularizedUSM and RegularizedCSM divided by the assumptions they make about ℓ . Specifically, Sect. 5 covers non-negative ℓ , Sects. 6 and 7 cover non-positive ℓ , and Sect. 8 covers arbitrary ℓ .

Section 4: Inapproximability of Maximization with Cardinality Constraint

We start with the inapproximability of CSM. Oveis Gharan and Vondrak [17] used a symmetry gap construction [18] to prove 0.491-inapproximability of CSM in the special case where the matroid constraint is a cardinality constraint. Our first result improves the inapproximability factor for a cardinality constraint to 0.478 using a modified construction, matching the factor of the current best inapproximability result for CSM in the general case (also due to [17]).

Theorem 4.1 *There exist instances of the problem $\max\{f(S) : S \subseteq \mathcal{N} \text{ and } |S| \leq w\}$ where f is non-negative submodular such that a 0.478-approximation would require exponentially many value queries.*

Section 5: Non-positive ℓ

The results of this section are summarized in Fig. 1. In Sect. 5.1, we present improved $(\alpha(\beta), \beta)$ -approximations for RegularizedUSM for all $\beta \geq 0$ and RegularizedCSM for all $\beta \in [0, 1]$,¹ Previously, the best known result for both RegularizedUSM and RegularizedCSM was $\alpha(\beta) = \beta e^{-\beta} - \epsilon$ due to Lu et al. [25]. This function achieves its maximum value at $\alpha(1) = e^{-1} - \epsilon > 0.367$. We improve the approximation factor for RegularizedCSM to $\alpha(1) > 0.385$, matching the best-known approximation factor for CSM without a regularizer due to Buchbinder and Feldman [14]. Additionally, we show that larger values of $\alpha(\beta)$ are approximable for RegularizedUSM when $\beta > 1$. The idea is to combine the “guessing step” of Sviridenko et al. [19] with a generalization of the *aided measured continuous greedy* algorithm of Buchbinder and Feldman [14].

Theorem 5.1 *For RegularizedUSM with non-positive ℓ , an $(\alpha(\beta), \beta)$ -approximation algorithm exists for any $(\alpha(\beta), \beta)$ in Table 1. In particular, $\alpha(1) > 0.385$ and $\alpha(1.3) > 0.398$. When $\beta \leq 1$, there is an algorithm for RegularizedCSM that achieves the same approximation factor.*

A natural follow-up question is whether there is a $(0.5, \beta)$ -approximation algorithm for RegularizedUSM with non-positive ℓ for some β . Although it is unclear whether this is the case for general f , we use linear programming to show this result when f is an undirected or directed cut function (Propositions 5.4 and 5.5).

In Sect. 5.2, we use the symmetry gap technique to demonstrate improved inapproximability for RegularizedUSM with non-positive ℓ . The previous best inapproximability results were [24, Theorem 1.1] near $\beta = 0$ and [24, Theorem 1.3] near $\beta = 1$. Our result, which generalizes the construction from Sect. 4, beats or matches both of these theorems for all β .

Theorem 5.6 *There are instances of RegularizedUSM with non-positive ℓ such that $(\alpha(\beta), \beta)$ is inapproximable for any $(\alpha(\beta), \beta)$ in Table 2. In particular, $\alpha(0) \approx 0$, matching the result of [24, Theorem 1.1], and $\alpha(1) < 0.478$, matching the result of [24, Theorem 1.3].*

¹ For RegularizedCSM it is possible to consider $\beta > 1$, but the α obtained by our algorithm does not increase as β increases past one.

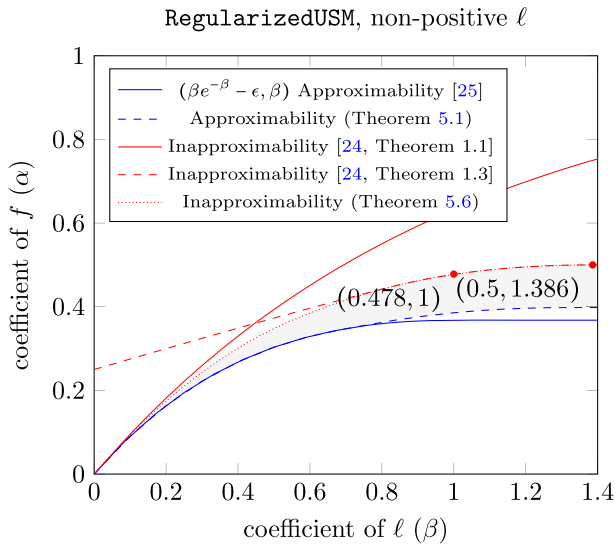


Fig. 1 Graphical depiction of results for Sect. 5. Following the convention of [24], the x and y axes represent the coefficients of ℓ and f , respectively. We use blue for approximation algorithms and red for inapproximability results, and the shaded area represents the gap between the best-known approximation algorithms and inapproximability results. Theorem 5.6 unifies the two inapproximability theorems from [24]. $(0.5, 2 \ln 2 - \epsilon \approx 1.386)$ -inapproximability is due to Proposition 5.9. For RegularizedCSM, the results are the same for $\beta \leq 1$

We conclude this section by showing that taking the limit of Theorem 5.6 as $\alpha(\beta) \rightarrow 0.5$ implies $(0.5, 2 \ln 2 - \epsilon \approx 1.386)$ -inapproximability (Proposition 5.9).

Section 6: Non-negative ℓ , RegularizedUSM

The results of this section are summarized in Figs. 2 and 3.

It is easy to check that Theorem 5.1 may be modified to obtain guarantees for RegularizedUSM with non-negative ℓ , which we do in Sect. 6.2. But before that, in Sect. 6.1, we take a slight detour and reanalyze the guarantee for this task provided by the randomized double greedy algorithm of [10] (RandomizedDG), which achieves the best-known $(\alpha(\beta), \beta)$ -approximations near $\beta = 3/4$. We also reanalyze the guarantee of the deterministic variant of double greedy from the same paper (DeterministicDG).

Recall that DeterministicDG achieves a $1/3$ -approximation for USM, while RandomizedDG achieves a $1/2$ -approximation for USM in expectation. Bodek and Feldman [24] extended these guarantees to RegularizedUSM with non-negative ℓ , showing that DeterministicDG simultaneously achieves $(\alpha, 1 - \alpha)$ -approximations for all $\alpha \in [0, 1/3]$, and that RandomizedDG simultaneously achieves $(\alpha, 1 - \alpha/2)$ -approximations for all $\alpha \in [0, 1/2]$. In Sect. 6.1, we show improved approximation factors for a variant of DeterministicDG and the original RandomizedDG:

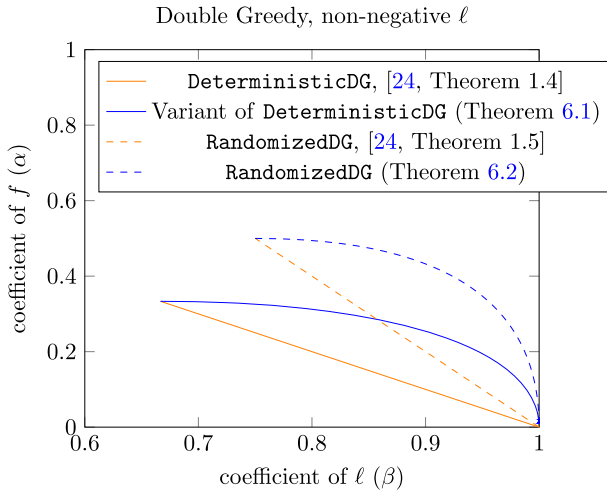


Fig. 2 Graphical depiction of results for Sect. 6.1. For RandomizedDG, we obtain higher α for all $\beta \in (3/4, 1)$, and for DeterministicDG, we obtain higher α for all $\beta \in (2/3, 1)$

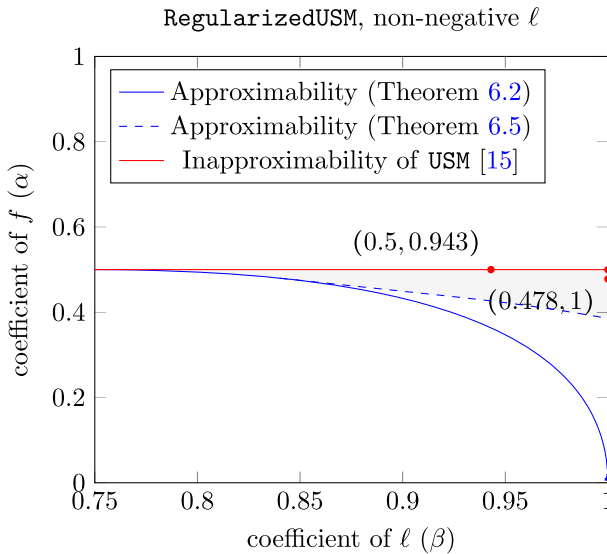


Fig. 3 Graphical depiction of results for Sect. 6. $(0.5 + \epsilon)$ -inapproximability is due to [15] and $(0.4998 + \epsilon, 1)$ -inapproximability is due to [24, Lemma 6.3]. $(0.478, 1)$ -inapproximability is due to Theorem 6.8, and $(0.5, 2\sqrt{2}/3 + \epsilon)$ -inapproximability is due to Theorem 6.9

- Improved analysis of a variant of DeterministicDG (Theorem 6.1). For any $r \geq 1$, we describe a variant of DeterministicDG that simultaneously achieves $(0, 1)$ and $(\frac{1}{r+1+r^{-1}}, \frac{r+1}{r+1+r^{-1}})$ -approximations. For $r = 1$, the variant is actually just the original DeterministicDG.

- Improved analysis of `RandomizedDG` (Theorem 6.2). We then show that `RandomizedDG` simultaneously achieves $\left(\frac{2}{r+2+r^{-1}}, \frac{r+2}{r+2+r^{-1}}\right)$ -approximations for all $r \geq 1$.

Observe that for both `DeterministicDG` and `RandomizedDG`, increasing r improves the dependence of the approximation on ℓ but decreases the dependence on f . Setting $r = 1$ recovers the guarantees of [24]. We also provide examples showing that neither `DeterministicDG` nor `RandomizedDG` achieve (α, β) -approximations better than Theorems 6.1 and 6.2 in Propositions 6.3 and 6.4, respectively.

In Sect. 6.2 we provide improved approximation algorithms for non-negative ℓ near $\beta = 1$ by combining the results of Sects. 5.1 and 6.1:

Theorem 6.5 *An $(\alpha(\beta), \beta)$ -approximation algorithm for `RegularizedUSM` with non-negative ℓ exists for any $(\alpha(\beta), \beta)$ in Table 3. In particular, the $\alpha(\beta)$ obtained for $\beta \geq 0.85$ is superior to that of Theorem 6.2, and $\alpha(1) > 0.385$, matching the approximation factor of Theorem 5.1.*

In Sect. 6.3, we use the symmetry gap technique to prove both $(0.478, 1 - \epsilon)$ - and $(0.5, 2\sqrt{2}/3 \approx 0.943 + \epsilon)$ -inapproximability (Theorems 6.8 and 6.9). These results are much stronger than [24, Theorem 1.6], which only proved $(0.4998 + \epsilon, 1)$ -inapproximability. Again, our constructions are variants of that used in Sect. 4.

Section 7: Non-negative ℓ , `RegularizedCSM`

The results of this section are summarized in Fig. 4. In Sect. 7.1 we combine the *distorted measured continuous greedy* of [25] with the *aided measured continuous greedy* of [14] to show the following.

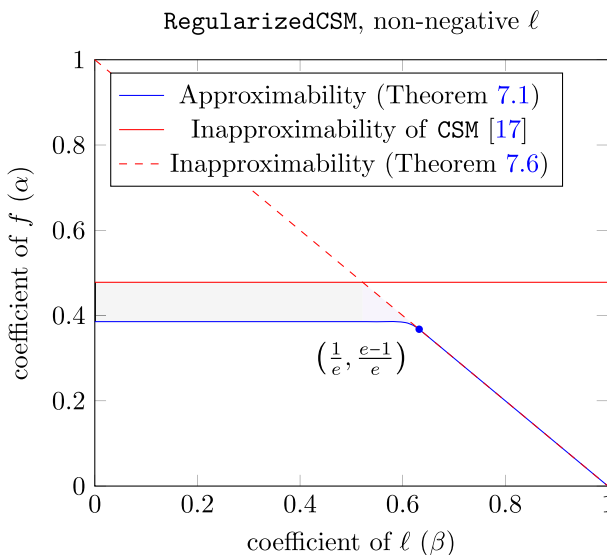


Fig. 4 Graphical depiction of results for Sect. 7. Recall that [17, Theorem 5.4] showed 0.478-inapproximability for `CSM`

Theorem 7.1 For RegularizedCSM with non-negative ℓ , there is a $(\alpha(\beta) - \epsilon, \beta - \epsilon)$ approximation algorithm for all $\beta \in [0, 1]$ where α is a decreasing concave function satisfying $\alpha(0.385) > 0.385$, $\alpha(0.6) > 0.384$, $\alpha(1 - e^{-1}) = e^{-1}$, and $\alpha(1) = 0$.

Note that $\alpha(0.385) > 0.385$ matches the (trivial) result of directly applying the algorithm of [14] to $f + \ell$. In Sect. 7.2, we prove a complementary inapproximability result showing that our algorithm is tight for $\beta \geq 1 - e^{-1}$.

Theorem 7.6 (Inapproximability of RegularizedCSM Near $\beta = 1$) For any $0 \leq \beta \leq 1$, there exist instances of RegularizedCSM with non-negative ℓ such that a $(1 - \beta + \epsilon, \beta)$ -approximation would require exponentially many value queries.

Section 8: Unconstrained ℓ

The results of this section are summarized in Figs. 5 and 6.

In Sect. 8.1, we present the first nontrivial (α, β) -approximation algorithm for RegularizedCSM where the sign of ℓ is unconstrained. Furthermore, the α we obtain for RegularizedUSM improves over that of [24] for all $\beta \in (0, 1)$.

Theorem 8.1 For all $t \geq 0$, there is a $(\frac{te^{-t}}{t+e^{-t}} - \epsilon, \frac{t}{t+e^{-t}})$ -approximation algorithm for RegularizedUSM. This algorithm achieves the same approximation guarantee for RegularizedCSM when $t \leq 1$.

For certain values of β , we can achieve greater α for RegularizedCSM than that guaranteed by Theorem 8.1. Because the improvement is marginal and we do not have

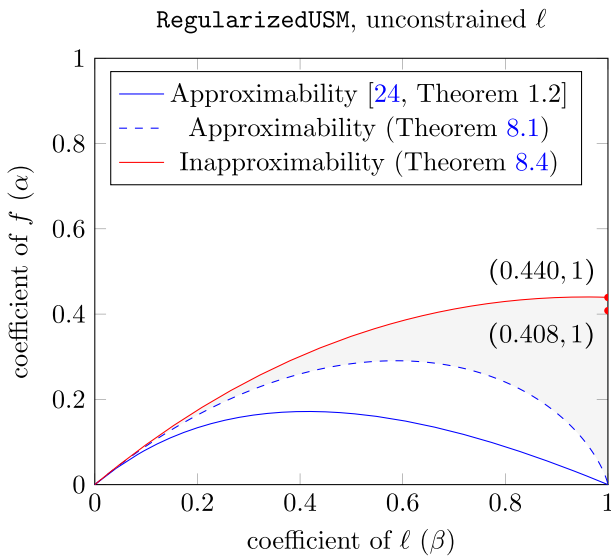


Fig. 5 Graphical depiction of results from Sect. 8 relating to RegularizedUSM. The $(\frac{\beta(1-\beta)}{1+\beta} - \epsilon, \beta - \epsilon)$ -approximation for RegularizedUSM is due to [24, Theorem 1.2]. The red dot corresponds to $(0.408, 1)$ -inapproximability, which is due to Theorem 8.5

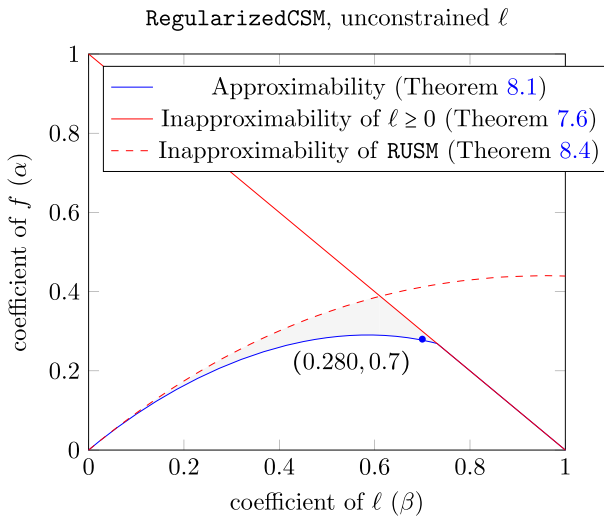


Fig. 6 Graphical depiction of results from Sect. 8 relating to RegularizedCSM. The blue dot corresponds to $(0.280, 0.7)$ -approximability, which is due to Theorem 8.3

a closed form, our following result addresses only the specific case of $\beta = 0.7$. Note that Theorem 8.1 guarantees a $(0.277, 0.7)$ -approximation (by setting $t \approx 0.925$).

Theorem 8.3 *There is a $(0.280, 0.7)$ -approximation algorithm for RegularizedCSM.*

In Sect. 8.2, we prove two theorems pertaining to improved inapproximability for RegularizedUSM. The former generalizes Theorem 1.3 of [24] in order to show improved inapproximability for a range of β when ℓ is not necessarily constrained to be non-positive.

Theorem 8.4 (Inapproximability of RegularizedUSM) *There are instances of RegularizedUSM where $(\alpha(\beta), \beta)$ is inapproximable for any $(\alpha(\beta), \beta)$ in Table 4. In particular, $(0.440, 1)$ is inapproximable.*

The latter shows stronger inapproximability specifically near $\beta = 1$.

Theorem 8.5 (Inapproximability of RegularizedUSM, $\beta = 1$) *There are instances of RegularizedUSM where $(0.408, 1)$ is inapproximable.*

The best prior $(\alpha, 1)$ -inapproximability result for RegularizedUSM is $(0.478, 1)$ due to Theorem 1.3 of [24], matching the 0.478 -inapproximability result for CSM due to Oveis Gharan and Vondrak [17]. We note that as Bodek and Feldman [24] show inapproximability specifically for the case of non-positive ℓ , it is not too surprising that we can show improved inapproximability for general ℓ . Notably, the gap between the best approximability and inapproximability results for RegularizedUSM remains quite large; in fact, it remains unclear whether an $(\epsilon, 1)$ -approximation algorithm exists for any $\epsilon > 0$.

4 Inapproximability of Maximization with Cardinality Constraint

In this section, we prove Theorem 4.1:

Theorem 4.1 *There exist instances of the problem $\max\{f(S) : S \subseteq \mathcal{N} \text{ and } |S| \leq w\}$ where f is non-negative submodular such that a 0.478-approximation would require exponentially many value queries.*

First, we provide the relevant definitions for proving inapproximability using the symmetry gap technique from Vondrak [18].

Definition 4.2 (Symmetrization [18]) Let \mathcal{G} be a group of permutations over \mathcal{N} . For $\mathbf{x} \in [0, 1]^{\mathcal{N}}$, define the “symmetrization of \mathbf{x} ” as $\bar{\mathbf{x}} = \mathbb{E}_{\sigma \in \mathcal{G}}\{\sigma(\mathbf{x})\}$, where $\sigma \in \mathcal{G}$ is uniformly random and $\sigma(\mathbf{x})$ denotes \mathbf{x} with coordinates permuted by σ .

Definition 4.3 (Symmetry Gap [18]) Let $\max\{f(S) : S \in \mathcal{F} \subseteq 2^{\mathcal{N}}\}$ be strongly symmetric with respect to a group \mathcal{G} of permutations over \mathcal{N} , meaning that for all $\sigma \in \mathcal{G}$ and $S \subseteq 2^{\mathcal{N}}$, $f(S) = f(\sigma(S))$ and $S \in \mathcal{F} \Leftrightarrow S' \in \mathcal{F}$ whenever $\bar{\mathbf{1}}_S = \bar{\mathbf{1}}_{S'}$. Here, $\sigma(S) = \{\sigma(i) : i \in S\}$. Define $P(\mathcal{F}) = \text{conv}(\{\mathbf{1}_I : I \in \mathcal{F}\})$ to be the polytope associated with \mathcal{F} . Let $\mathbf{OPT} \triangleq \max_{\mathbf{x} \in P(\mathcal{F})} F(\mathbf{x})$ and $\bar{\mathbf{OPT}} \triangleq \max_{\mathbf{x} \in P(\mathcal{F})} F(\bar{\mathbf{x}})$. Then the symmetry gap of $\max\{f(S) : S \in \mathcal{F}\}$ is defined as $\gamma \triangleq \frac{\mathbf{OPT}}{\bar{\mathbf{OPT}}}$.

Lemma 4.4 (Inapproximability due to Symmetry Gap [18]) *Let $\max\{f(S) : S \in \mathcal{F}\}$ be an instance of non-negative submodular maximization, strongly symmetric with respect to \mathcal{G} , with symmetry gap γ . Let \mathcal{C} be the class of instances $\max\{\tilde{f}(S) : S \in \tilde{\mathcal{F}}\}$ where \tilde{f} is non-negative submodular and $\tilde{\mathcal{F}}$ is a refinement of \mathcal{F} . Then for every $\epsilon > 0$, any (even randomized) $(1+\epsilon)\gamma$ -approximation algorithm for the class \mathcal{C} would require exponentially many queries to the value oracle for $\tilde{f}(S)$.*

The formal definition of refinement can be found in [18]. The important thing to note is that $\tilde{\mathcal{F}}$ satisfies the same properties as \mathcal{F} . In particular, $\tilde{\mathcal{F}}$ preserves cardinality and matroid independence constraints. Before proving Theorem 4.1, we start with a related lemma.

Lemma 4.5 (Inapproximability of Cardinality Constraint on Subset of Domain) *Let T be some subset of the ground set \mathcal{N} . There exist instances of the problem $\max\{f(S) : S \subseteq \mathcal{N} \wedge |S \cap T| \leq w\}$ such that a 0.478-approximation would require exponentially many value queries.*

Proof It suffices to provide \mathcal{F} , f , and \mathcal{G} satisfying the definitions of Lemma 4.4 with symmetry gap $\gamma < 0.478$. The construction is identical to that of [17, Theorem 5.4], except we omit $|S \cap \{a, b\}| \leq 1$ from the definition of \mathcal{F} . Specifically, letting $a_{1\dots k}$ be shorthand for a_1, a_2, \dots, a_k , we define $\mathcal{N} \triangleq \{a, b, a_{1\dots k}, b_{1\dots k}\}$ and

$$\mathcal{F} \triangleq \{S \mid S \subseteq \mathcal{N} \wedge |S \cap \{a_{1\dots k}, b_{1\dots k}\}| \leq 1\} \tag{4.1}$$

instead of:

$$\mathcal{F}_{orig} \triangleq \{S \mid S \subseteq \mathcal{N} \wedge |S \cap \{a, b\}| \leq 1 \wedge |S \cap \{a_1, \dots, a_k, b_1, \dots, b_k\}| \leq 1\}. \tag{4.2}$$

Recall that Theorem 5.4 of [17] defines the submodular function f as the sum of the weighted cut functions of two directed hyperedges $(\{a_1, a_2, \dots, a_k\}, a)$, $(\{b_1, b_2, \dots, b_k\}, b)$ and the undirected edge (a, b) (see Fig. 4 of [17] for an illustration). Specifically, the weighted cut function on the directed hyperedge $(\{a_1, a_2, \dots, a_k\}, a)$ contributes $\kappa \triangleq 0.3513$ to the value of $f(S)$ if $S \cap \{a_1, \dots, a_k\} \neq \emptyset$ and $a \notin S$, and 0 otherwise. The weighted cut function on the directed hyperedge $(\{b_1, b_2, \dots, b_k\}, b)$ is defined in the same way. Finally, the weighted cut function on the undirected edge (a, b) contributes $1 - \kappa$ if $|S \cap \{a, b\}| = 1$ and 0 otherwise. Thus, the multilinear extension of f is as follows:

$$\begin{aligned} F(\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_{a_1, \dots, k}, \mathbf{x}_{b_1, \dots, k}) \\ \triangleq (1 - \kappa)(\mathbf{x}_a(1 - \mathbf{x}_b) + \mathbf{x}_b(1 - \mathbf{x}_a)) \\ + \kappa \left[\left(1 - \prod_{i=1}^k (1 - \mathbf{x}_{a_i})\right) (1 - \mathbf{x}_a) + \left(1 - \prod_{i=1}^k (1 - \mathbf{x}_{b_i})\right) (1 - \mathbf{x}_b) \right]. \end{aligned} \tag{4.3}$$

As in [17, Lemma 5.1], we let \mathcal{G} be the group of permutations generated by $\{\sigma_1, \sigma_2\}$, where

$$\sigma_1(a) = b, \sigma_1(b) = a, \sigma_1(a_i) = b_i, \sigma_1(b_i) = a_i \tag{4.4}$$

swaps the two hyperedges, and

$$\sigma_2(a) = a, \sigma_2(b) = b, \sigma_2(a_i) = a_i \pmod{k+1}, \sigma_2(b_i) = b_i \tag{4.5}$$

rotates the tail vertices of the first hyperedge. It is easy to check that (f, \mathcal{F}) are strongly symmetric with respect to both σ_1 and σ_2 , and that the symmetrization of \mathbf{x} is as follows:

$$\bar{\mathbf{x}} = \mathbb{E}_{\sigma \in \mathcal{G}} [\sigma(\mathbf{x})] = \begin{cases} \bar{x}_a = \bar{x}_b = \frac{\mathbf{x}_a + \mathbf{x}_b}{2} \\ \bar{x}_{a_1} = \dots = \bar{x}_{a_k} = \bar{x}_{b_1} = \dots = \bar{x}_{b_k} = \frac{\sum_{i=1}^k (\mathbf{x}_{a_i} + \mathbf{x}_{b_i})}{2k}. \end{cases} \tag{4.6}$$

Observe that

$$\mathbf{OPT} \geq \max_{S \in \mathcal{F}} f(S) \geq f(\{a, b_1\}) = (1 - \kappa) + \kappa = 1. \tag{4.7}$$

Defining $q \triangleq \frac{\mathbf{x}_a + \mathbf{x}_b}{2}$ and $p \triangleq \frac{\sum_{i=1}^k (\mathbf{x}_{a_i} + \mathbf{x}_{b_i})}{2}$, the maximum of F over all symmetric \mathbf{x} is thus:

$$\overline{\mathbf{OPT}} = \max_{\mathbf{x} \in P(\mathcal{F})} F(\bar{\mathbf{x}}) = \max_{\mathbf{x} \in P(\mathcal{F})} F(q, q, \overbrace{p/k, p/k, \dots, p/k}^{2k \text{ times}}) \triangleq \max_{\mathbf{x} \in P(\mathcal{F})} \hat{F}(q, p)$$

$$\begin{aligned}
 &= (1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - (1 - p/k)^k) \\
 &\approx (1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p})
 \end{aligned}
 \tag{4.8}$$

where the approximate equality holds as $k \rightarrow \infty$. Now,

$$\begin{aligned}
 \overline{\text{OPT}} &= \max_{\mathbf{x} \in P(\mathcal{F})} F(\bar{\mathbf{x}}) = \max_{p \leq 1/2} \hat{F}(q, p) = \max_{p, q \leq 1/2} \hat{F}(q, p) \\
 &= \max_{\mathbf{x} \in P(\mathcal{F}_{orig})} F(\bar{\mathbf{x}}) < 0.478.
 \end{aligned}
 \tag{4.9}$$

The third equality holds (i.e., adding the constraint $q \leq 1/2$ has no effect) since $\hat{F}(q, p) \leq \hat{F}(1 - q, p)$ for $q \in (1/2, 1]$, while the inequality holds due to the proof of [17, Theorem 5.4] and may be verified using a numerical optimizer. So the symmetry gap $\gamma = \frac{\overline{\text{OPT}}}{\text{OPT}}$ is less than 0.478, as desired. \square

Now, to show Theorem 4.1, all we need to do is to convert the cardinality constraint on $S \cap T$ in Lemma 4.5 into a cardinality constraint on all of S .

Proof of Theorem 4.1 Again, it suffices to provide \mathcal{F} , f , and \mathcal{G} satisfying the definitions of Lemma 4.4 with symmetry gap $\gamma < 0.478$. We start with the construction from Lemma 4.5, replace each element a_i and b_i with t copies $a_{i,1}, \dots, a_{i,t}$, and $b_{i,1}, \dots, b_{i,t}$ and set $w \triangleq t + 1$. Letting $a_{1\dots k,1\dots t}$ be shorthand for $\{a_{ij} : i \in [1, k] \wedge j \in [1, t]\}$, we redefine f such that F is as follows:

$$\begin{aligned}
 &F(\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_{a_{1\dots k,1\dots t}}, \mathbf{x}_{b_{1\dots k,1\dots t}}) \\
 &\triangleq (1 - \kappa)(\mathbf{x}_a(1 - \mathbf{x}_b) + \mathbf{x}_b(1 - \mathbf{x}_a)) \\
 &\quad + \kappa \left[\left(1 - \prod_{i=1}^k \left(1 - \frac{\sum_{j=1}^t \mathbf{x}_{a_{i,j}}}{t} \right) \right) (1 - \mathbf{x}_a) \right. \\
 &\quad \left. + \left(1 - \prod_{i=1}^k \left(1 - \frac{\sum_{j=1}^t \mathbf{x}_{b_{i,j}}}{t} \right) \right) (1 - \mathbf{x}_b) \right].
 \end{aligned}
 \tag{4.10}$$

Importantly, f remains non-negative submodular and symmetric, with the new symmetrization being as follows for an appropriate choice of \mathcal{G} :

$$\bar{\mathbf{x}} = \mathbb{E}_{\sigma \in \mathcal{G}} [\sigma(\mathbf{x})] = \begin{cases} \bar{x}_a = \bar{x}_b = \frac{\mathbf{x}_a + \mathbf{x}_b}{2} \\ \bar{x}_{a_{1,1}} = \dots = \bar{x}_{a_{k,t}} = \bar{x}_{b_{1,1}} = \dots = \bar{x}_{b_{k,t}} \\ = \frac{\sum_{i=1}^k \sum_{j=1}^t (\mathbf{x}_{a_{i,j}} + \mathbf{x}_{b_{i,j}})}{2kt}. \end{cases}
 \tag{4.11}$$

For example, we may define \mathcal{G} to be the group generated by $\{\sigma_1, \sigma_2, \sigma_3\}$ where

- σ_1 swaps a with b and $a_{i,j}$ with $b_{i,j}$;
- σ_2 takes $a_{i,j}$ to $a_{i \pmod k + 1, j}$ and leaves all other vertices unchanged;
- σ_3 takes $a_{1,j}$ to $a_{1, j \pmod t + 1}$ and leaves all other vertices unchanged.

It can be verified that $F(\bar{\mathbf{x}})$ may be written in terms of the same function of two variables $\hat{F}(q, p)$ from Lemma 4.5. Let q be as defined above and redefine $p \triangleq \frac{\sum_{i=1}^k \sum_{j=1}^t (\mathbf{x}_{a_{ij}} + \mathbf{x}_{b_{ij}})}{2t}$, so that:

$$\begin{aligned} F(\bar{\mathbf{x}}) &\triangleq F(q, q, \overbrace{p/k, p/k, \dots, p/k}^{2kt \text{ times}}) \triangleq \hat{F}(q, p) \\ &= (1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - (1 - p/k)^k) \\ &\approx (1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p}) \end{aligned} \tag{4.12}$$

where the approximate equality holds as $k \rightarrow \infty$, same as before.

To finish, we must show that the symmetry gap of f with respect to \mathcal{F} remains less than 0.478 as $t \rightarrow \infty$. As in the proof of Lemma 4.5,

$$\text{OPT} \geq \max_{S: |S| \leq t+1} f(S) \geq f(\{a, b_{11}, \dots, b_{1t}\}) = 1, \tag{4.13}$$

$$\overline{\text{OPT}} = \max_{\sum \mathbf{x}_i \leq t+1} F(\bar{\mathbf{x}}) = \max_{\sum \mathbf{x}_i \leq t+1} \hat{F}(q, p) \leq \max_{p \leq \frac{t+1}{2t}} \hat{F}(q, p) \approx \max_{p \leq 1/2} \hat{F}(q, p) < 0.478, \tag{4.14}$$

where the gap between the two sides of the approximate equality goes to 0 as $t \rightarrow \infty$ because \hat{F} is Lipschitz continuous and its domain is bounded. So the symmetry gap $\gamma = \frac{\overline{\text{OPT}}}{\text{OPT}}$ is again less than 0.478, as desired. \square

5 Non-positive ℓ

The results of this section are summarized in Fig. 1.

5.1 Approximation Algorithms

In this subsection, we provide improved approximations for general f (Theorem 5.1) as well as for f a cut function (Propositions 5.4 and 5.5).

Theorem 5.1 *For RegularizedUSM with non-positive ℓ , an $(\alpha(\beta), \beta)$ -approximation algorithm exists for any $(\alpha(\beta), \beta)$ in Table 1. In particular, $\alpha(1) > 0.385$ and $\alpha(1.3) > 0.398$. When $\beta \leq 1$, there is an algorithm for RegularizedCSM that achieves the same approximation factor.*

We start with a special case of Theorem 5.1.

Lemma 5.2 *There is a $(0.385, 1)$ approximation algorithm for RegularizedCSM when ℓ is non-positive.*

Proof The idea is to combine the “guessing step” of Sviridenko et al. [19] with the 0.385-approximation for CSM due to Buchbinder and Feldman [14] (which actually

Table 1 $(\alpha(\beta), \beta)$ -Approximations for RegularizedUSM with non-positive ℓ (Theorem 5.1)

β	$\beta e^{-\beta}$	$\alpha(\beta)$
0.7	0.3476	0.3478
0.8	0.3595	0.3630
0.9	0.3659	0.3757
1.0	0.3679	0.3856
1.1	0.3662	0.3925
1.2	0.3614	0.3967
1.3	0.3543	0.3982
1.4	0.3452	0.3982

For comparison, the previous best-known approximation factors of [25] are included in the second column. These results are depicted graphically in Fig. 1
 In all tables, all values are rounded to the nearest multiple of 10^{-4}

provides a $(0.385 + \epsilon)$ -approximation for any $\epsilon \leq 0.0006$. Recall that [19] achieves a $(1 - \frac{1}{e} - \epsilon, 1)$ -approximation for monotone f and non-positive ℓ . The idea is that if we know the value of $\ell(OPT)$, we can run the algorithm of [14] on the intersection $\mathcal{P} \cap \{\mathbf{x} : L(\mathbf{x}) \geq \ell(OPT)\}$, which is down-closed and solvable because \mathcal{P} is down-closed and solvable, and the same is true for $\{\mathbf{x} : L(\mathbf{x}) \geq \ell(OPT)\}$. This will guarantee finding $\mathbf{x} \in \mathcal{P}$ such that $\mathbb{E}[F(\mathbf{x})] \geq 0.385f(OPT)$ and $\mathbb{E}[L(\mathbf{x})] \geq \ell(OPT)$. Afterward, we can use *pipage rounding* to round \mathbf{x} to an integral solution $T \in \mathcal{I}$ [18]. Specifically, given $\mathbf{x} \in \mathcal{P}$, pipage rounding generates $T \in \mathcal{I}$ such that $\mathbb{E}[\mathbf{1}_T] = \mathbf{y}$ and $\mathbb{E}[F(\mathbf{1}_T) + L(\mathbf{1}_T)] \geq F(\mathbf{x}) + L(\mathbf{x})$.

Of course, we do not actually know what the value of $\ell(OPT)$ is. To guarantee that we run [14] on the intersection $\mathcal{P} \cap \{\mathbf{x} : L(\mathbf{x}) \geq w\}$ for some w satisfying $\ell(OPT) \geq w \geq \ell(OPT)(1 + \epsilon)$, it suffices to try setting w equal to each of the $\mathcal{O}(\frac{n^2}{\epsilon})$ values in the following set:

$$\{0\} \cup \left\{ \ell(u) \cdot k \mid u \in \mathcal{N}, k \in \mathcal{Z} \text{ and } k \in \left[\left\lceil \frac{1}{\epsilon} \right\rceil, \left\lceil \frac{n}{\epsilon} \right\rceil \right] \right\}. \tag{5.1}$$

For at least one of these values of w (“guesses”), we will have $\mathbb{E}[F(\mathbf{x})] \geq (0.385 + \epsilon)f(OPT)$ (if $\epsilon \leq 0.0006$) and $\mathbb{E}[L(\mathbf{x})] \geq (1 + \epsilon)\ell(OPT)$. Combining these guarantees shows that rounding \mathbf{x} provides a $(0.385 + \epsilon, 1 + \epsilon)$ approximation, which in turn implies a $(0.385, 1)$ approximation because

$$\begin{aligned} & \max(0, (0.385 + \epsilon)f(OPT) + (1 + \epsilon)\ell(OPT)) \\ & \geq \frac{(0.385 + \epsilon)f(OPT) + (1 + \epsilon)\ell(OPT)}{1 + \epsilon} \\ & \geq 0.385f(OPT) + \ell(OPT). \end{aligned} \tag{5.2}$$

□

Before proving Theorem 5.1, we start by briefly reviewing the main algorithm from [14] when executed on a solvable down-closed polytope \mathcal{P} . First, it uses a *local search* to generate $\mathbf{z} \in \mathcal{P}$ such that both of the following inequalities hold with high probability:

$$F(\mathbf{z}) \geq \frac{1}{2}F(\mathbf{z} \wedge \mathbf{1}_{OPT}) + \frac{1}{2}F(\mathbf{z} \vee \mathbf{1}_{OPT}) - o(1) \cdot f(OPT), \tag{5.3}$$

$$F(\mathbf{z}) \geq F(\mathbf{z} \wedge \mathbf{1}_{OPT}) - o(1) \cdot f(OPT). \tag{5.4}$$

Then it executes [14, Algorithm 2], *Aided Measured Continuous Greedy*, to generate $\mathbf{y} \in \mathcal{P}$ such that

$$\begin{aligned} \mathbb{E}[F(\mathbf{y})] \geq e^{t_s-1} \cdot [(2 - t_s - e^{-t_s} - o(1)) \cdot f(OPT) - (1 - e^{-t_s}) \cdot F(\mathbf{z} \wedge \mathbf{1}_{OPT}) \\ - (2 - t_s - 2e^{-t_s}) \cdot F(\mathbf{z} \vee \mathbf{1}_{OPT})]. \end{aligned} \tag{5.5}$$

Finally, assuming \mathcal{P} is the matroid polytope corresponding to a family of independent sets \mathcal{I} , pipage rounding may be used to convert both \mathbf{y} and \mathbf{z} to integral solutions $y \in \mathcal{I}$ and $z \in \mathcal{I}$ such that $\mathbb{E}[f(y)] \geq F(\mathbf{y})$ and $\mathbb{E}[f(z)] \geq F(\mathbf{z})$, and the solution from y and z with the larger value of f will be returned.²

To obtain improved approximation bounds, we need the following generalization of Eq. (5.5):

Lemma 5.3 (Generalization of Aided Measured Continuous Greedy) *If we run Aided Measured Continuous Greedy given a fractional solution \mathbf{z} and a polytope \mathcal{P} for a total of t_f time, where $t_f \geq t_s$, it will generate $\mathbf{y} \in t_f\mathcal{P} \cap [0, 1]^{\mathcal{N}}$ such that*

$$\begin{aligned} \mathbb{E}[F(\mathbf{y})] \geq e^{-t_f} [(e^{t_s} + t_f e^{t_s} - t_s e^{t_s} - 1 - o(1))f(OPT) \\ + (-e^{t_s} + 1)F(\mathbf{z} \wedge \mathbf{1}_{OPT}) \\ + (-e^{t_s} - t_f e^{t_s} + t_s e^{t_s} + 1 + t_f)F(\mathbf{z} \vee \mathbf{1}_{OPT})] \end{aligned} \tag{5.6}$$

Note that this matches term by term with Eq. (5.5) when $t_f = 1$.

Proof Sketch By [14], proving the conclusion for integral sets Z implies the conclusion for fractional \mathbf{z} . So it suffices to prove the following.

$$\begin{aligned} \mathbb{E}[F(\mathbf{y}(t_f))] \geq e^{-t_f} [(e^{t_s} + t_f e^{t_s} - t_s e^{t_s} - 1 - o(1))f(OPT) \\ + (-e^{t_s} + 1)f(OPT \cap Z) \\ + (-e^{t_s} - t_f e^{t_s} + t_s e^{t_s} + 1 + t_f)f(OPT \cup Z)]. \end{aligned} \tag{5.7}$$

The idea of the original *aided measured continuous greedy* is to run *measured continuous greedy* for t_s time only on the elements of $\mathcal{N} \setminus Z$, and then for $1 - t_s$ additional time with all elements of \mathcal{N} . Working out what happens when we run it for a total of t_f instead of 1 time is just a matter of going through the equations from [14, Section 4] and making minor changes. The remainder of the proof is deferred to Sect. A.3.

² Actually, [14] analyze their algorithm assuming z is returned with probability $p = 0.23$ and otherwise y , but this distinction is of little consequence.

Proof of Theorem 5.1 Our algorithm for RegularizedUSM is as follows:

1. As in Lemma 5.2, first guess the value of $\ell(OPT)$ to within a factor of $1 + \epsilon$, and then replace \mathcal{P} with $\mathcal{P} \cap \{\mathbf{x} : L(\mathbf{x}) \geq (1 + \epsilon)\ell(OPT)\}$.
2. Generate \mathbf{z} using the local search procedure on (f, \mathcal{P}) described by [14].
3. Run aided continuous greedy given \mathbf{z} for all pairs³

$$(t_s, t_f) \in \mathcal{T} \triangleq \left\{ \left(\frac{x}{20}, \frac{y}{20} \right) \mid (x, y) \in \mathcal{Z}^2 \text{ and } 0 \leq x \leq y \leq 40 \right\}. \tag{5.8}$$

4. Round \mathbf{z} from step 1 and all fractional solutions found in step 2 to valid integral solutions. Note that by replacing \mathbf{z} with $R(\mathbf{z})$, the value of $F + L$ is preserved in expectation.
5. Return the solution from step 4 with the maximum value, or the empty set if none of these solutions has positive value. Let OPT' be the expected value of this solution.

For a fixed $\beta \geq 0$, we can compute the maximum $\alpha(\beta)$ such that an $(\alpha(\beta) - \mathcal{O}(\epsilon), \beta)$ -approximation is guaranteed by solving the following linear program:

$$\begin{aligned} & \max x_1 \\ & \text{s.t. } (x_1, x_2, x_3, x_4) \in \text{conv}(\{(0, 0, 0, 0), (0, 0.5, 0.5, 1), (0, 1, 0, 1)\} \cup \\ & \qquad \qquad \qquad \{ (e^{t_s-t_f} + t_f e^{t_s-t_f} - t_s e^{t_s-t_f} - e^{-t_f}, \\ & \qquad \qquad \qquad - e^{t_s-t_f} + e^{-t_f}, \\ & \qquad \qquad \qquad - e^{t_s-t_f} - t_f e^{t_s-t_f} + t_s e^{t_s-t_f} + e^{-t_f} + t_f e^{-t_f}, \\ & \qquad \qquad \qquad t_f) \mid (t_s, t_f) \in \mathcal{T} \}) \\ & \text{and } x_2 \geq 0, x_3 \geq 0, x_4 \leq \beta. \end{aligned} \tag{5.9}$$

Any point (x_1, x_2, x_3, x_4) within the convex hull satisfies:

$$OPT' \geq x_1 f(OPT) + x_2 F(\mathbf{z} \wedge \mathbf{1}_{OPT}) + x_3 F(\mathbf{z} \vee \mathbf{1}_{OPT}) + x_4 L(OPT) \tag{5.10}$$

if we ignore the $o(1)$ terms contributed by Lemma 5.3 and Eqs. (5.3) and (5.4) and take the limit as $\epsilon \rightarrow 0$. The points determining the hull are as follows:

- $(0, 0, 0, 0)$ corresponds to returning the empty set.
- $(0, 0.5, 0.5, 1)$ corresponds to \mathbf{z} satisfying Eq. (5.3).
- $(0, 1, 0, 1)$ corresponds to \mathbf{z} satisfying Eq. (5.4).
- The remaining vertices of the hull correspond to running Lemma 5.3 on \mathcal{P} given \mathbf{z} for all $(t_s, t_f) \in \mathcal{T}$.

Adding the constraints $x_2, x_3 \geq 0$ and $x_4 \leq \beta$ ensures that $OPT' \geq x_1 F(\mathbf{z}) + \beta L(OPT)$. The results of solving this program with CVXPY [26] for $\beta \in [0, 1.5]$ are displayed in Fig. 1. In particular, $\alpha(1) \geq 0.385$ and the maximum value of α is obtained around $\alpha(1.3) \geq 0.398$.

³ For improved approximations, a finer discretization can be chosen, but we found the benefit of doing so to be negligible.

For the case of `RegularizedCSM`, the reasoning is almost the same, but to ensure that all points returned by Lemma 5.3 lie within \mathcal{P} , we only include pairs in \mathcal{T} with $t_f \leq 1$ in step 3, and pipage rounding with respect to the *original* \mathcal{P} (not $\mathcal{P} \cap \{\mathbf{x} : L(\mathbf{x}) \geq (1 + \epsilon)\ell(OPT)\}$), which is not necessarily a matroid polytope) must be used for step 4. The results turn out to be identical to those displayed in Fig. 1 for $\beta \leq 1$. \square

Next, we state better approximation results for f an undirected and directed cut function, respectively. The proofs, which use linear programming, are deferred to Sect. A.3. We note that linear programming was previously used to provide a 0.5-approximation for `MAX-DICUT` by Trevisan [27] and later by Halperin and Zwick [28].

Proposition 5.4 *There is a (0.5, 1)-approximation algorithm for `RegularizedCSM` when ℓ has arbitrary sign and f is the cut function of a weighted undirected graph (V, E, w) ; that is, for all $S \subseteq V$,*

$$f(S) \triangleq \sum_{ab \in E} w_{ab} \cdot [|S \cap \{a, b\}| = 1], \quad (5.11)$$

where each edge weight w_{ab} is non-negative.

Note that while our above result for undirected cut functions applies to `RegularizedCSM`, our subsequent result for directed cut functions only applies to `RegularizedUSM`.

Proposition 5.5 *There is a (0.5, 1)-approximation algorithm for `RegularizedUSM` when ℓ has arbitrary sign and f is the cut function of a weighted directed graph (V, E, w) ; that is, for all $S \subseteq V$,*

$$f(S) = \sum_{ab \in E} w_{ab} \cdot [a \in S \text{ and } b \notin S], \quad (5.12)$$

where each edge weight w_{ab} is non-negative.

5.2 Inapproximability

In this subsection, we prove Theorem 5.6. Recall from Fig. 1 that it unifies the guarantees of [24, Theorem 1.1] and [24, Theorem 1.3].

Theorem 5.6 *There are instances of `RegularizedUSM` with non-positive ℓ such that $(\alpha(\beta), \beta)$ is inapproximable for any $(\alpha(\beta), \beta)$ in Table 2. In particular, $\alpha(0) \approx 0$, matching the result of [24, Theorem 1.1], and $\alpha(1) < 0.478$, matching the result of [24, Theorem 1.3].*

The idea behind the proof of Theorem 5.6 is to extend the symmetry gap construction of [24, Theorem 1.3], which in turn is a modification of the 0.478-inapproximability result of [17] used in Sect. 4.

Table 2 Inapproximability of $(\alpha(\beta), \beta)$ -approximations for RegularizedUSM with non-positive ℓ

β	$\alpha(\beta)$ [24, Theorem 1.3]	$\alpha(\beta)$ (Theorem 5.6)	κ	ℓ_p	ℓ_q
0.1	0.2750	0.0935	0.6705	-0.6095	-0.2680
0.2	0.2998	0.1743	0.6513	-0.5322	-0.2192
0.3	0.3245	0.2433	0.6498	-0.4705	-0.1505
0.4	0.3488	0.3008	0.6506	-0.4207	-0.0893
0.5	0.3728	0.3477	0.6484	-0.3800	-0.0410
0.6	0.3964	0.3846	0.6388	-0.3400	0.0000
0.7	0.4195	0.4162	0.5811	-0.2887	0.0000
0.8	0.4420	0.4420	0.5088	-0.2289	0.0000
0.9	0.4621	0.4621	0.4335	-0.1766	0.0000
1.0	0.4773	0.4773	0.3515	-0.1294	0.0000

The parameters κ, ℓ_p, ℓ_q are as described in the proof of Theorem 5.6. These results are depicted graphically in Fig. 1

Before proving Theorem 5.6, we state a generalization of the symmetry gap technique to $f + \ell$ sums that we use for Theorem 5.6 and the rest of our inapproximability results.

Definition 5.7 [18] We say that $\max_{S \in \mathcal{F}} [f(S) + \ell(S)]$ is *strongly symmetric* with respect to a group of permutations \mathcal{G} if $\ell(S) = \ell(\sigma(S))$ for all $\sigma \in \mathcal{G}$ and (f, \mathcal{F}) are strongly symmetric with respect to \mathcal{G} as defined in Definition 4.3.

Lemma 5.8 (Inapproximability of (α, β) Approximations) *Let $\max_{S \in \mathcal{F}} [f(S) + \ell(S)]$ be an instance of submodular maximization with f non-negative submodular and ℓ linear, strongly symmetric with respect to a group of permutations \mathcal{G} . For any two constants $\alpha, \beta \geq 0$, if*

$$\max_{\mathbf{x} \in P(\mathcal{F})} [F(\mathbf{x}) + L(\mathbf{x})] < \max_{S \in \mathcal{F}} [\alpha f(S) + \beta \ell(S)], \tag{5.13}$$

then no polynomial-time algorithm for RegularizedCSM can guarantee a (α, β) -approximation. The same inapproximability holds for RegularizedUSM by setting $\mathcal{F} = 2^{\mathcal{N}}$.

Proof Theorem 3.1 of [24] proves this lemma only for the special case of $\mathcal{F} = 2^{\mathcal{N}}$ because the proof of [24, Lemma A.3] cites a special case of [18, Lemma 3.3] that only applies for $\mathcal{F} = 2^{\mathcal{N}}$. It suffices to modify the proof to cite the full version of [18, Lemma 3.3] instead. □

Proof of Theorem 5.6 Set f to be the same as defined in Lemma 4.5. Now apply Lemma 5.8 with $S = \{a, b_1\}$. For a fixed β , we can show (α, β) -inapproximability using this method if it is possible to choose ℓ and κ such that the following inequality is true:

$$\max_{\mathbf{x} \in P(\mathcal{F})} [F(\mathbf{x}) + L(\mathbf{x})] < \alpha f(\{a, b_1\}) + \beta \ell(\{a, b_1\})$$

$$= \alpha + \beta \ell(\{a, b_1\}) \tag{5.14}$$

$$\implies \max_{\mathbf{x} \in P(\mathcal{F})} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] - \beta \ell(\{a, b_1\}) < \alpha. \tag{5.15}$$

Our goal is now to minimize the LHS of the above inequality. Theorem 1.3 of [24] sets $\ell_a = \ell_b = 0$, and then chooses κ and $\ell_{a_{1\dots k}} = \ell_{b_{1\dots k}} \triangleq \ell_p \leq 0$ in order to minimize the quantity

$$\begin{aligned} & \max_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] - \beta \ell(\{a, b_1\}) \\ &= \max_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] - \beta \ell_p \\ &= \max_{0 \leq q \leq 1, 0 \leq p} [(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p}) + 2p\ell_p] - \beta \ell_p. \end{aligned} \tag{5.16}$$

However, choosing $\ell_a = \ell_b \triangleq \ell_q$ to be negative rather than zero gives superior bounds for small β . That is, our goal is to compute

$$\begin{aligned} & \min_{0 \leq \kappa \leq 1, \ell_q \leq 0, \ell_p \leq 0} \left[\max_{0 \leq q \leq 1, 0 \leq p} [(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p}) \right. \\ & \quad \left. + 2p\ell_p + 2q\ell_q] - \beta(\ell_p + \ell_q) \right]. \end{aligned} \tag{5.17}$$

We can approximate the optimal value by brute forcing over a range of (κ, ℓ_q, ℓ_p) . For $\beta \in \{0.8, 0.9, 1.0\}$, it is optimal to set $\ell_q = 0$, and our guarantee is the same as that of [24, Theorem 1.3]. Our results for $\beta \in \{0.6, 0.7\}$ are stronger than those of [24, Theorem 1.3] even though they also set $\ell_q = 0$, because that theorem actually only considers $\ell_p \geq -0.5$ and $\kappa \leq 0.5$. □

Next, we consider the limit of Theorem 5.6 as $\alpha(\beta) \rightarrow 0.5$. Note that this is not a new result in the sense that [24, Theorem 1.3] can already prove it when the parameters ℓ_p and κ are chosen appropriately, but we nevertheless believe it is worth explicitly stating.

Proposition 5.9 *For any $\epsilon > 0$, there are instances of RegularizedUSM with non-positive ℓ such that $(0.5, 2 \ln 2 - \epsilon \approx 1.386)$ is inapproximable.*

Proof To find the maximum β such that we can show $(0.5, \beta)$ -inapproximability using the construction of Theorem 5.6, our goal is to choose $\kappa \in (0, 0.5)$ and $\ell_p < 0$ such that β is maximized in

$$\begin{aligned} & \max_{0 \leq q \leq 1, 0 \leq p} [(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p}) + 2p\ell_p] - \beta \ell_p < 0.5 \\ \iff \beta & < \frac{0.5 - \max_{0 \leq q \leq 1, 0 \leq p} [(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p}) + 2p\ell_p]}{-\ell_p}. \end{aligned} \tag{5.18}$$

We can rewrite half the expression within the max as

$$\begin{aligned} & \max_{q, 0 \leq p} \left[(1 - \kappa)q(1 - q) + \kappa(1 - q)(1 - e^{-p}) + p\ell_p \right] \\ &= \max_{q, 0 \leq p} \left[-q^2(1 - \kappa) + q(1 - 2\kappa + \kappa e^{-p}) + \kappa(1 - e^{-p}) + p\ell_p \right] \\ &= \max_{0 \leq p} \left[\frac{(1 - 2\kappa + \kappa e^{-p})^2}{4(1 - \kappa)} + \kappa(1 - e^{-p}) + p\ell_p \right], \end{aligned} \tag{5.19}$$

so the RHS of Eq. (5.18) becomes:

$$\begin{aligned} & \frac{2 \cdot \min_{0 \leq p} \left[\frac{1 - \kappa - (1 - 2\kappa + \kappa e^{-p})^2}{4(1 - \kappa)} - \kappa(1 - e^{-p}) - p\ell_p \right]}{-\ell_p} \\ &= \frac{2 \cdot \min_{0 \leq p} \left[\frac{\kappa(2e^{-p} - 1) - \kappa^2 e^{-2p}}{4(1 - \kappa)} - p\ell_p \right]}{-\ell_p}. \end{aligned} \tag{5.20}$$

Next, we claim that for any $p^* > 0$, it is possible to choose $\ell_p < 0$ such that the numerator of Eq. (5.20) reaches its minimum at $p = p^*$. Define the function $h(p) \triangleq \frac{\kappa(2e^{-p} - 1) - \kappa^2 e^{-2p}}{4(1 - \kappa)}$. It suffices to check that h is decreasing at $p = 0$ and concave up for $p \geq 0$; that is, $\frac{d}{dp} h(p) \Big|_{p=0} < 0$ and $\frac{d^2}{dp^2} \left[\frac{\kappa(2e^{-p} - 1) - \kappa^2 e^{-2p}}{4(1 - \kappa)} \right] > 0$ for all $p \geq 0$. Both of these inequalities follow from the assumption $\kappa \in (0, 0.5)$.

Finally, when $p^* < \ln 2$, $2e^{-p^*} > 1$, implying that $h(p^*) > 0$ when κ is sufficiently close to 0. For such p^* , the RHS of Eq. (5.20) becomes

$$\frac{2(h(p^*) - p^*\ell_p)}{-\ell_p} \geq \frac{-2p^*\ell_p}{-\ell_p} = 2p^*, \tag{5.21}$$

so the RHS of Eq. (5.18) can be made arbitrarily close to $2 \ln 2$. □

6 Non-negative ℓ : RegularizedUSM

6.1 Approximations with Double Greedy

In this subsection, we show improved approximability for DeterministicDG and RandomizedDG in Theorems 6.1 and 6.2, and then show that both of these results are tight in Propositions 6.3 and 6.4. The results of this subsection are summarized in Fig. 2.

First, we briefly review the behavior of the original DeterministicDG and RandomizedDG of [10] when executed on a non-negative submodular function g , as well as their approximation factors.

The Algorithm:

The algorithm will construct sequences of sets X_i, Y_i for $i \in [0, n]$. First, $X_0 \triangleq \emptyset$ and $Y_0 \triangleq \mathcal{N}$. Then for each i from 1 to n , execute the following two steps:

1. Compute the marginal gains $g(u_i|X_{i-1}) = g(X_{i-1} \cup \{u_i\}) - g(X_{i-1})$ and $g(u_i|Y_{i-1} \setminus \{u_i\}) = g(Y_{i-1}) - g(Y_{i-1} \setminus \{u_i\})$. By the original proof of double greedy,

$$g(u_i|X_{i-1}) - g(u_i|Y_{i-1} \setminus \{u_i\}) \geq 0 \quad (6.1)$$

holds by submodularity.

2. Based on the marginal gains, either set $(X_i, Y_i) = (X_{i-1} \cup \{u_i\}, Y_{i-1})$ or $(X_i, Y_i) = (X_{i-1}, Y_{i-1} \setminus \{u_i\})$.

- In `DeterministicDG`, the first event occurs if $g(u_i|X_{i-1}) \geq -g(u_i|Y_{i-1} \setminus \{u_i\})$.
- In `RandomizedDG`, the first event occurs with probability proportional to $a_i \triangleq \max(g(u_i|X_{i-1}), 0)$, while the second event occurs with probability proportional to $b_i \triangleq \max(-g(u_i|Y_{i-1} \setminus \{u_i\}), 0)$. In the edge case where $a_i = b_i = 0$, it does not matter which event occurs.

Finally, the algorithm returns $X_n = Y_n$.

The Approximation Factors:

Let $OPT_i \triangleq (OPT \cup X_i) \cap Y_i$, so that $OPT_0 = OPT$ while $OPT_n = X_n = Y_n$. For `DeterministicDG`, it can be shown via exhaustive casework that:

$$g(OPT_{i-1}) - g(OPT_i) \leq (g(X_i) - g(X_{i-1})) + (g(Y_i) - g(Y_{i-1})), \quad (6.2)$$

while for `RandomizedDG`, it can similarly be shown that:

$$\mathbb{E}[g(OPT_{i-1}) - g(OPT_i)] \leq \frac{1}{2} \mathbb{E}[(g(X_i) - g(X_{i-1})) + (g(Y_i) - g(Y_{i-1}))]. \quad (6.3)$$

Summing Eq. (6.2) from $i = 1$ to $i = n$ gives

$$\begin{aligned} g(OPT) - g(X_n) &= g(OPT_0) - g(OPT_n) \\ &\leq g(X_n) - g(X_0) + g(Y_n) - g(Y_0) \\ &\leq 2g(X_n) - g(\mathcal{N}) \\ \implies g(X_n) &\geq \frac{g(OPT) + g(\mathcal{N})}{3}, \end{aligned} \quad (6.4)$$

whereas summing Eq. (6.3) from $i = 1$ to $i = n$ gives

$$\begin{aligned} \mathbb{E}[g(OPT) - g(X_n)] &= \mathbb{E}[g(OPT_0) - g(OPT_n)] \\ &\leq \mathbb{E}\left[\frac{1}{2}(2g(X_n) - g(\mathcal{N}))\right] \\ \implies \mathbb{E}[g(X_n)] &\geq \frac{2g(OPT) + g(\mathcal{N})}{4}. \end{aligned} \quad (6.5)$$

Equations (6.4) and (6.5) imply that if we substitute $f + \ell$ in place of g , `DeterministicDG` and `RandomizedDG` provide $(1/3, 2/3)$ - and $(1/2, 3/4)$ -approximations for `RegularizedUSM`, respectively, because $\ell(OPT) \leq$

$\ell(\mathcal{N})$. Showing improved (α, β) -approximations for `DeterministicDG` (`RandomizedDG`) when $\alpha < \frac{1}{3}$ ($\alpha < \frac{1}{2}$) is just a matter of modifying Eq. (6.2) (Eq. (6.3)).

Theorem 6.1 *For RegularizedUSM with non-negative ℓ and any $r \geq 1$, there exists a variant of `DeterministicDG` that simultaneously achieves $(0, 1)$ and $(\frac{1}{r+1+r^{-1}}, \frac{r+1}{r+1+r^{-1}})$ -approximations (and consequently, (α, β) -approximations for all (α, β) on the segment connecting these two points as well). For $r = 1$, the variant is actually just the original `DeterministicDG`.*

Proof Modify step 2 of `DeterministicDG` so that the first event occurs if $g(u_i|X_{i-1}) \geq -rg(u_i|Y_{i-1} \setminus \{u_i\})$. We claim that the following modified version of Eq. (6.2) now holds:

$$g(OPT_{i-1}) - g(OPT_i) \leq r^{-1}(g(X_i) - g(X_{i-1})) + r(g(Y_i) - g(Y_{i-1})). \tag{6.6}$$

First we show that Eq. (6.6) implies an $(\frac{1}{r+1+r^{-1}}, \frac{r+1}{r+1+r^{-1}})$ -approximation. Summing it from $i = 1$ to $i = n$ gives:

$$\begin{aligned} g(OPT) - g(X_n) &= g(OPT_0) - g(OPT_n) \\ &\leq r^{-1}(g(X_n) - g(X_0)) + r(g(Y_n) - g(Y_0)) \\ &\leq (r^{-1} + r)g(X_n) - rg(\mathcal{N}) \end{aligned} \tag{6.7}$$

$$\begin{aligned} \implies g(X_n) &\geq \frac{g(OPT)}{r+1+r^{-1}} + \frac{r}{r+1+r^{-1}}g(\mathcal{N}) \\ &\geq \frac{f(OPT)}{r+1+r^{-1}} + \frac{r+1}{r+1+r^{-1}} \cdot \ell(OPT), \end{aligned} \tag{6.8}$$

as desired. Now we show Eq. (6.6). First, we consider the case $g(u_i|X_{i-1}) \geq -rg(u_i|Y_{i-1} \setminus \{u_i\})$. This assumption implies that $Y_i = Y_{i-1}$, so the last part of Eq. (6.6) drops out.

1. If $u_i \in OPT_{i-1}$, then $OPT_i = OPT_{i-1}$, and Eq. (6.6) reduces to $0 \leq g(u_i|X_{i-1})$, which holds by combining Eq. (6.1) with the assumption.
2. If $u_i \notin OPT_{i-1}$, then $OPT_i = OPT_{i-1} \cup \{u_i\}$, then Eq. (6.6) reduces to

$$-g(u_i|OPT_{i-1}) \leq r^{-1}g(u_i|X_{i-1}). \tag{6.9}$$

Since $OPT_{i-1} \subseteq Y_{i-1} \setminus \{u_i\}$, the LHS of this inequality is at most $-g(u_i|Y_{i-1} \setminus \{u_i\})$ by submodularity. On the other hand, the RHS of this inequality is at least $-g(u_i|Y_{i-1} \setminus \{u_i\})$ by assumption.

On the other hand, if $g(u_i|X_{i-1}) < -rg(u_i|Y_{i-1} \setminus \{u_i\})$, then $X_i = X_{i-1}$, and the first part of the RHS of Eq. (6.6) drops out.

1. If $u_i \notin OPT_{i-1}$, then $OPT_i = OPT_{i-1}$, and Eq. (6.6) reduces to $0 \leq -g(u_i|Y_{i-1} \setminus \{u_i\})$, which holds by combining Eq. (6.1) with the assumption.

2. If $u_i \in OPT_{i-1}$, then $OPT_i = OPT_{i-1} \setminus \{u_i\}$, then Eq. (6.6) reduces to

$$g(u_i | OPT_i) \leq -rg(u_i | Y_{i-1} \setminus \{u_i\}). \tag{6.10}$$

Since $X_i \subseteq OPT_i$, the LHS of this inequality is at most $g(u_i | X_i)$ by submodularity. On the other hand, the RHS of this inequality is greater than $g(u_i | X_{i-1})$ by assumption.

It remains to show that this algorithm simultaneously achieves a $(0, 1)$ -approximation. Because $g(u_i | X_i) - g(u_i | Y_i \setminus \{u_i\}) \geq 0$, $\max(g(u_i | X_i), -rg(u_i | Y_i \setminus \{u_i\})) \geq 0$. Next,

- If $g(u_i | X_i) \geq -rg(u_i | Y_i \setminus \{u_i\})$, then $g(u_i | X_i) \geq 0$, $Y_{i+1} = Y_i$ and

$$g(X_{i+1}) = g(u_i | X_i) + g(X_i) \geq g(X_i). \tag{6.11}$$

- Otherwise, if $g(u_i | X_i) < -rg(u_i | Y_i \setminus \{u_i\})$, then $g(u_i | Y_i \setminus \{u_i\}) \leq 0$, $X_{i+1} = X_i$ and

$$g(Y_{i+1}) = g(Y_i) - g(u_i | Y_i \setminus \{u_i\}) \geq g(Y_i). \tag{6.12}$$

Thus, the values of both $g(X_i)$ and $g(Y_i)$ are increasing over the course of the algorithm, so:

$$g(X_n) = g(Y_n) \geq g(Y_{n-1}) \geq \dots \geq g(Y_0) \geq \ell(\mathcal{N}) \geq \ell(OPT). \tag{6.13}$$

□

The reasoning for RandomizedDG, which we show next, is very similar.

Theorem 6.2 *Running RandomizedDG on $f + \ell$ simultaneously achieves an $\left(\frac{2}{r+2+r^{-1}}, \frac{r+2}{r+2+r^{-1}}\right)$ -approximation for all $r \geq 1$ for RegularizedUSM with non-negative ℓ .*

Proof We claim that the following modified version of Eq. (6.3) holds for any $r > 0$:

$$\mathbb{E}[g(OPT_{i-1}) - g(OPT_i)] \leq \frac{1}{2} \mathbb{E} \left[r^{-1}(g(X_i) - g(X_{i-1})) + r(g(Y_i) - g(Y_{i-1})) \right]. \tag{6.14}$$

As in the proof of Theorem 6.1, it is easy to check that Eq. (6.14) implies the conclusion:

$$\begin{aligned} \mathbb{E}[g(OPT) - g(X_n)] &= \mathbb{E}[g(OPT_0) - g(OPT_n)] \\ &\leq \mathbb{E} \left[\frac{1}{2} \left((r^{-1} + r)g(X_n) - rg(\mathcal{N}) \right) \right] \\ \implies \mathbb{E}[g(X_n)] &\geq \frac{2}{r + 2 + r^{-1}} \cdot g(OPT) + \frac{r}{r + 2 + r^{-1}} \cdot g(\mathcal{N}) \end{aligned} \tag{6.15}$$

$$\geq \frac{2}{r + 2 + r^{-1}} \cdot f(OPT) + \frac{r + 2}{r + 1 + r^{-1}} \cdot \ell(OPT) \tag{6.16}$$

It remains to show Eq. (6.14). We note that in the edge case $a_i = b_i = 0$, Eq. (6.1) implies that $g(u_i|X_{i-1}) = g(u_i|Y_{i-1} \setminus \{u_i\}) = 0$, so the inequality reduces to $0 \leq 0$. Otherwise, recall that the original proof of double greedy upper bounded the LHS of Eq. (6.14) by

$$\mathbb{E}[g(OPT_{i-1}) - g(OPT_i)] \leq \frac{a_i b_i}{a_i + b_i}. \tag{6.17}$$

On the other hand, we can lower bound twice the RHS by

$$\begin{aligned} & \mathbb{E}[r^{-1}(g(X_i) - g(X_{i-1})) + r(g(Y_i) - g(Y_{i-1}))] \\ &= r^{-1} \cdot \frac{a_i}{a_i + b_i} (g(X_{i-1} \cup \{u_i\}) - g(X_{i-1})) \\ & \quad + r \cdot \frac{b_i}{a_i + b_i} (g(Y_{i-1} \setminus \{u_i\}) - g(Y_{i-1})) \\ &= \frac{r^{-1} a_i^2}{a_i + b_i} + \frac{r b_i^2}{a_i + b_i} \geq \frac{2a_i b_i}{a_i + b_i}, \end{aligned} \tag{6.18}$$

where the last step follows from the AM-GM inequality as in the original proof. Equation (6.14) follows. \square

Next, we prove that `DeterministicDG` and `RandomizedDG` do no better than the bounds we just showed. Recall that [24, Theorem 1.4] proved that the original `DeterministicDG` is an (α, β) -approximation algorithm whenever $\alpha \leq \frac{1}{3}$ and $\alpha + \beta \leq 1$. To show that this analysis is tight, it suffices to check that whenever $\alpha > \frac{1}{3}$ or $\alpha + \beta > 1$, there are instances where `DeterministicDG` does not achieve the desired approximation factor. The inequality $\alpha > \frac{1}{3}$ holds by [10, Theorem II.3], while $\alpha + \beta > 1$ holds by applying the following proposition with $r = 1$:

Proposition 6.3 *For any $r \geq 1$ and $\epsilon > 0$, there are instances of `RegularizedUSM` with non-negative ℓ where the variant of `DeterministicDG` described in the proof of Theorem 6.1 does not achieve an (α, β) -approximation for any (α, β) above the line connecting $(0, 1)$ and $(\frac{1}{r+1+r^{-1}}, \frac{r+1}{r+1+r^{-1}})$.*

Proof The points (α, β) lying above the line connecting $(0, 1)$ and $(\frac{1}{r+1+r^{-1}}, \frac{r+1}{r+1+r^{-1}})$ are precisely those that satisfy $\alpha + \beta r = r + \epsilon$ for some $\epsilon > 0$. Define $f(S)$ to be the sum of two weighted cut functions:

$$\mathcal{N} \triangleq \{u_1, u_2\} \tag{6.19}$$

$$f(S) \triangleq (r + \epsilon/2) \cdot [u_1 \in S \text{ and } u_2 \notin S] + 1 \cdot [u_2 \in S \text{ and } u_1 \notin S] \tag{6.20}$$

$$\ell(u_1) = 0, \ell(u_2) = r \tag{6.21}$$

The weights of the directed edges are chosen such that if the variant of `DeterministicDG` considers u_1 before u_2 , it will compute

$$g(u_1|X_0) = r + \epsilon/2 > -r g(u_1|Y_0 \setminus \{u_1\}) = r, \tag{6.22}$$

so it will return a set T satisfying $u_1 \in T$, implying that $f(T) + \ell(T) \leq r + \epsilon/2$ regardless of whether $u_2 \in T$ or not. If we define $OPT \triangleq \{u_2\}$, then $f(OPT) = 1$ and $\ell(OPT) = r$, so we get

$$f(T) + \ell(T) = r + \epsilon/2 < r + \epsilon = \alpha + \beta r = \alpha f(OPT) + \beta \ell(OPT) \tag{6.23}$$

implying that an (α, β) -approximation is not achieved. □

Next, we generalize the construction of Proposition 6.3 to show that Theorem 6.2 is tight for RandomizedDG.

Proposition 6.4 *For any $r \geq 1$ and $\epsilon > 0$, there are instances of RegularizedUSM with non-negative ℓ where RandomizedDG does not provide an $(\alpha, \beta) = \left(\frac{2}{r+2+r^{-1}} + \epsilon, \frac{r+2}{r+2+r^{-1}}\right)$ -approximation.*

Proof Define $f(S)$ to be the sum of $2(n - 1)$ weighted directed cut functions:

$$f(S) = \frac{1}{n - 1} \left[\sum_{i=1}^{n-1} (r[u_i \in S \wedge u_n \notin S] + [u_n \in S \wedge u_i \notin S]) \right] \tag{6.24}$$

and $\ell(u_1) = \ell(u_2) = \dots = \ell(u_{n-1}) = 0, \ell(u_n) = r - 1$. For each $i \in [1, n - 1]$, RandomizedDG will compute $a_i = \frac{r}{n-1}$ and $b_i = \frac{1}{n-1}$, so it will include each of $u_1 \dots u_{n-1}$ in its returned set X_n independently with probability $\frac{r}{r+1}$ each. Thus, for any $\epsilon > 0$, the following inequality holds by a Chernoff bound for sufficiently large n :

$$\Pr \left[\left| \frac{|X_n \cap \{u_1, \dots, u_{n-1}\}|}{n - 1} - \frac{r}{r + 1} \right| \geq \frac{\epsilon}{2r} \right] = o(1), \tag{6.25}$$

Assuming $\left| \frac{|X_n \cap \{u_1, \dots, u_{n-1}\}|}{n - 1} - \frac{r}{r + 1} \right| < \frac{\epsilon}{2r}$ holds, it follows that

$$f(X_n) + \ell(X_n) < r \cdot \left(\frac{r}{r + 1} + \frac{\epsilon}{2r} \right) = \frac{r^2}{r + 1} + \frac{\epsilon}{2}. \tag{6.26}$$

regardless of whether u_n is included in X_n or not. On the other hand, if we define $OPT \triangleq \{u_n\}$, then

$$\begin{aligned} \alpha f(OPT) + \beta \ell(OPT) &= \alpha + (r - 1)\beta \\ &= \frac{2}{r + 2 + r^{-1}} + \epsilon + \frac{(r - 1)(r + 2)}{r + 2 + r^{-1}} \\ &= \frac{r^2 + r}{r + 2 + r^{-1}} + \epsilon \\ &= \frac{r^2}{r + 1} + \epsilon. \end{aligned} \tag{6.27}$$

As $f(X_n) + \ell(X_n) < \alpha f(OPT) + \beta \ell(OPT) - \frac{\epsilon}{2}$ with high probability and $f(X_n)$ is bounded above by a constant independent of n , $\mathbb{E}[f(X_n) + \ell(X_n)] < \alpha f(OPT) + \beta \ell(OPT)$ for sufficiently large n , implying that `RandomizedDG` does not provide an (α, β) approximation for this instance. \square

Unfortunately, neither version of double greedy achieves any (α, β) -approximation when ℓ is non-positive rather than non-negative. We defer further discussion to Sect. A.2.

6.2 Additional Approximation Algorithms

In this subsection, we prove Theorem 6.5, which improves upon Theorem 6.2 for β close to one. The results of this subsection and the next are summarized in Fig. 3.

Theorem 6.5 *An $(\alpha(\beta), \beta)$ -approximation algorithm for `RegularizedUSM` with non-negative ℓ exists for any $(\alpha(\beta), \beta)$ in Table 3. In particular, the $\alpha(\beta)$ obtained for $\beta \geq 0.85$ is superior to that of Theorem 6.2, and $\alpha(1) > 0.385$, matching the approximation factor of Theorem 5.1.*

First, we show that the result for $\beta = 1$ easily follows from Lemma 5.2.

Lemma 6.6 *For `RegularizedUSM` with non-negative ℓ , there is a $(0.385, 1)$ -approximation algorithm.*

Proof Define $g(S) \triangleq f(\mathcal{N} \setminus S)$, which is also non-negative submodular. Then apply Lemma 5.2 on $(g, -\ell)$ to find $T \subseteq \mathcal{N}$ such that

$$\begin{aligned} \mathbb{E}[g(T) - \ell(T)] &\geq \max_S [0.385g(S) - \ell(S)] \\ &= \max_S [0.385f(S) - \ell(\mathcal{N} \setminus S)] \\ &= \max_S [0.385f(S) + \ell(S)] - \ell(\mathcal{N}). \end{aligned} \tag{6.28}$$

Setting $T' = \mathcal{N} \setminus T$, we have

$$\begin{aligned} \mathbb{E}[f(T') + \ell(T') - \ell(\mathcal{N})] &= \mathbb{E}[f(T') - \ell(\mathcal{N} \setminus T')] \\ &= \mathbb{E}[g(T) - \ell(T)] \end{aligned}$$

Table 3 $(\alpha(\beta), \beta)$ -approximations for `RegularizedUSM` with non-negative ℓ

β	$\alpha(\beta)$ (Theorem 6.5)	$\alpha(\beta)$ (Theorem 6.2)
0.85	0.4749	0.4746
0.9	0.4493	0.4325
0.95	0.4226	0.3472
1	0.3856	0

These results are depicted graphically in Fig. 3

$$\geq \max_S [0.385 f(S) + \ell(S)] - \ell(\mathcal{N}). \tag{6.29}$$

Adding $\ell(\mathcal{N})$ to both sides, we conclude that

$$\mathbb{E}[f(T') + \ell(T')] \geq \max_S [0.385 f(S) + \ell(S)]. \tag{6.30}$$

So an algorithm returning T' would achieve a $(0.385, 1)$ -approximation as desired. \square

For β close to one, we can obtain better (α, β) -approximations than what Theorem 6.2 alone provides by combining double greedy with the following corollary of Lemma 6.6:

Corollary 6.7 *An (α, β) -approximation algorithm for RegularizedUSM for the case of ℓ non-positive may be used to return a set $T \subseteq \mathcal{N}$ such that*

$$\mathbb{E}[f(T)] \geq \alpha f(OPT) + \beta \ell(OPT) + (1 - \beta)\ell(\mathcal{N}). \tag{6.31}$$

for the case of ℓ non-negative.

Proof The proof is very similar to the above; the RHSes of Eqs. (6.28) and (6.29) become $\alpha f(OPT) + \beta \ell(OPT) - \beta \ell(\mathcal{N})$. \square

Now we can prove Theorem 6.5 by combining Corollary 6.7 with Theorem 6.2.

Proof of Theorem 6.5 Our algorithm returns the best of the solutions returned by the following two algorithms:

1. Randomized double greedy on $f + \ell$, whose guarantee is given by Eq. (6.15)
2. Corollary 6.7 using Theorem 5.1 for $\beta \in \mathcal{T} \triangleq \{(\alpha(1 + 0.01x), 1 + 0.01x) \mid x \in \{0, 1, 2, \dots, 30\}\}$

As with Theorem 5.1, for a fixed β we can lower bound the $\alpha(\beta)$ guaranteed by the algorithm above by the solution to the following linear program after choosing the set \mathcal{R} appropriately:

$$\begin{aligned} & \max x_1 \\ & \text{s.t. } (x_1, x_2, x_3) \in \text{conv} \left(\{(\alpha', \beta', 1 - \beta') \mid (\alpha', \beta') \in \mathcal{T} \text{ and} \right. \\ & \qquad \qquad \qquad \left. \exists (\alpha', \beta')\text{-approximation algorithm for } \ell \leq 0\} \cup \right. \\ & \qquad \qquad \qquad \left. \left\{ \left(\frac{2}{(r + 1/r)^2}, \frac{2}{(r + 1/r)^2}, \frac{r^2}{(r + 1/r)^2} \right) \mid r \in \mathcal{R} \right\} \right) \\ & \text{and } x_2 + x_3 \geq \beta, x_3 \geq 0 \end{aligned} \tag{6.32}$$

Let \mathbf{OPT}' denote the expected value of the returned solution. Any point (x_1, x_2, x_3) within the convex hull satisfies the following inequality:

$$\mathbf{OPT}' \geq x_1 f(OPT) + x_2 \ell(OPT) + x_3 \ell(\mathcal{N}). \tag{6.33}$$

The conditions $x_2 + x_3 \geq \beta, x_3 \geq 0$ ensure that $\mathbf{OPT}' \geq x_1 f(OPT) + \beta \ell(OPT)$. \square

6.3 Inapproximability

In this subsection, we prove Theorems 6.8 and 6.9.

Theorem 6.8 *For some $\epsilon > 0$, there are instances of RegularizedUSM with non-negative ℓ such that $(0.478, 1 - \epsilon)$ is inapproximable.*

Note that this is much stronger than the $(0.4998 + \epsilon, 1)$ -inapproximability provided by [24, Lemma 6.3].

Proof We start by showing $(0.478, 1)$ -inapproximability, which is easier. First, we claim that any $(\alpha, 1)$ -approximation algorithm for the RegularizedUSM instance $(f(\mathcal{N} \setminus S), -\ell(S))$ immediately implies a $(\alpha, 1)$ -approximation algorithm for $(f(S), \ell(S))$. Letting $\mathcal{N} \setminus T$ be the set returned by the former approximation algorithm, we find

$$\begin{aligned} \mathbb{E}[f(\mathcal{N} \setminus (\mathcal{N} \setminus T)) - \ell(\mathcal{N} \setminus T)] &\geq \alpha f(\mathcal{N} \setminus OPT') - \ell(OPT') \\ \implies \mathbb{E}[f(T) - \ell(\mathcal{N}) + \ell(T)] &\geq \alpha f(\mathcal{N} \setminus OPT') - \ell(OPT') \end{aligned} \tag{6.34}$$

Substituting $OPT' = \mathcal{N} \setminus OPT$ gives

$$\begin{aligned} \mathbb{E}[f(T) - \ell(\mathcal{N}) + \ell(T)] &\geq \alpha f(OPT) - \ell(\mathcal{N}) + \ell(OPT) \\ \implies \mathbb{E}[f(T) + \ell(T)] &\geq \alpha f(OPT) + \ell(OPT). \end{aligned} \tag{6.35}$$

Note that when ℓ is set to be non-negative, this means that any $(\alpha, 1)$ -approximation algorithm for ℓ non-positive implies an $(\alpha, 1)$ -approximation algorithm for ℓ non-negative. Similarly, by setting ℓ to be non-positive, we get the implication in the opposite direction. This means that $(\alpha, 1)$ -inapproximability results for one sign of ℓ can be converted to corresponding inapproximability results for the other sign of ℓ . Thus, the $(0.478, 1)$ -inapproximability result for non-positive ℓ implies the same inapproximability result for non-negative ℓ .

The slightly stronger result of $(0.478, 1 - \epsilon)$ inapproximability for some $\epsilon > 0$ follows from modifying the symmetry gap construction of Theorem 5.6. Let (f_-, ℓ_-) be the f and ℓ defined in the proof of Theorem 5.6 for $\beta = 1$. Then let

$$f(S) \triangleq f_-(\mathcal{N} \setminus S), \ell(S) \triangleq -\ell_-(S). \tag{6.36}$$

For k sufficiently large, this instance shows $(\alpha, 1)$ -inapproximability for some $\alpha < 0.478$. Furthermore, if we fix k to be constant, then the desired result follows; specifically, we can choose some $\epsilon > 0$ such that

$$\alpha f(OPT) + \ell(OPT) = 0.478f(OPT) + (1 - \epsilon)\ell(OPT), \tag{6.37}$$

showing $(0.478, 1 - \epsilon)$ -inapproximability as desired. □

Next, we provide an inapproximability result for $\alpha = 0.5$ by fixing $k = 2$ in the construction for Theorem 6.8.

Theorem 6.9 *For any $\epsilon > 0$, there are instances of RegularizedUSM with non-negative ℓ such that $(0.5, 2\sqrt{2}/3 \approx 0.943 + \epsilon)$ is inapproximable.*

Proof Again, let (f_-, ℓ_-) be the f and ℓ defined in the proof of Theorem 5.6 with $\ell_q = 0$. Define

$$f(S) \triangleq f_-(\mathcal{N} \setminus S), \tag{6.38}$$

$$p \triangleq k - \frac{\sum_{i=1}^k (\mathbf{x}_{a_i} + \mathbf{x}_{b_i})}{2} \in [0, k], \tag{6.39}$$

$$q \triangleq 1 - \frac{\mathbf{x}_a + \mathbf{x}_b}{2} \in [0, 1]. \tag{6.40}$$

$$\ell(S) \triangleq -\ell_-(S) = \ell_p(2k - 2p) \tag{6.41}$$

where we may choose any real number $\ell_p > 0$. Applying Lemma 5.8, we find that the LHS is given by

$$\begin{aligned} & \max_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [(F + L)(\bar{\mathbf{x}})] \\ &= \max_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [F_-(\mathbf{1} - \bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] \\ &= \max_{0 \leq p \leq k, 0 \leq q \leq 1} \left[\left((1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - (1 - p/k)^k) \right) \right. \\ & \quad \left. + (-2p\ell_p + 2k\ell_p) \right] \\ &= \max_{0 \leq p \leq k, 0 \leq q \leq 1} \left[(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - (1 - p/k)^k) - 2p\ell_p \right] \\ & \quad + 2k\ell_p, \tag{6.42} \end{aligned}$$

while the RHS is bounded below by

$$(\alpha f + \beta \ell)(\mathcal{N} \setminus \{a, b\}) = \alpha + \beta [(2k - 1)\ell_p]. \tag{6.43}$$

Now fix $k = 2$ and $\alpha = 0.5$, and define

$$g(p) \triangleq \max_{0 \leq q \leq 1} \left[(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - (1 - p/k)^k) \right]. \tag{6.44}$$

Then the minimum β^* such that we can show $(\alpha, \beta^* + \epsilon)$ -inapproximability using this technique is given by

$$\max_{0 \leq p \leq k} [g(p) - 2p\ell_p] + 2k\ell_p = 0.5 + \beta^* [(2k - 1)\ell_p] \tag{6.45}$$

$$\implies \beta^* = \min_{0 \leq \kappa \leq 1, 0 < \ell_p} \left[\frac{\max_{0 \leq p \leq k} [g(p) - 2p\ell_p] + 2k\ell_p - 0.5}{(2k - 1)\ell_p} \right]. \tag{6.46}$$

Choose any $p^* \in (0, 2 - \sqrt{2})$, which automatically guarantees $1 - (1 - p^*/k)^k < \frac{1}{2}$. For any such p^* , we claim that there exist κ and ℓ_p such that

$$\max_{0 \leq p \leq k} [g(p) - 2p\ell_p] < 0.5 - 2p^*\ell_p. \tag{6.47}$$

The reason why Eq. (6.47) holds is that, for sufficiently small $\kappa > 0$, $g(p^*) < 0.5$, $g(p)$ is increasing with respect to p , and $g(p)$ is concave down with respect to p . Thus, we can always choose $\ell_p > 0$ so that $\operatorname{argmax}_{0 \leq p \leq k} [g(p) - 2p\ell_p] = p^*$. From Eq. (6.47) we can finish as follows:

$$\beta^* \leq \frac{2k\ell_p - 2p^*\ell_p}{(2k - 1)\ell_p} = \frac{4 - 2p^*}{3}. \tag{6.48}$$

Taking the limit as $p^* \rightarrow 2 - \sqrt{2}^-$ shows the inapproximability of $\beta^* = \frac{4 - 2(2 - \sqrt{2})}{3} + \epsilon = \frac{2\sqrt{2}}{3} + \epsilon$, as desired. \square

7 Non-negative ℓ : RegularizedCSM

The results of this section are summarized in Fig. 4.

7.1 Approximation Algorithms

In this subsection, we prove Theorem 7.1.

Theorem 7.1 *For RegularizedCSM with non-negative ℓ , there is a $(\alpha(\beta) - \epsilon, \beta - \epsilon)$ approximation algorithm for all $\beta \in [0, 1]$ where α is a decreasing concave function satisfying $\alpha(0.385) > 0.385$, $\alpha(0.6) > 0.384$, $\alpha(1 - e^{-1}) = e^{-1}$, and $\alpha(1) = 0$.*

Recall from the introduction that [25] introduced *distorted measured continuous greedy* and analyzed its guarantee for the case of non-positive ℓ . Our improved results are based on generalizing the analysis to the case where ℓ contains both positive and negative components.

Lemma 7.2 *For unconstrained ℓ and any $t_f \in [0, 1]$, there is a polynomial-time algorithm for RegularizedCSM that returns $T \in \mathcal{I}$ such that*

$$\mathbb{E}[f(T) + \ell(T)] \geq (t_f e^{-t_f} - o(1))f(OPT) + (1 - e^{-t_f} - o(1))\ell_+(OPT) + t_f \ell_-(OPT). \tag{7.1}$$

When $t_f > 1$, the algorithm provides the same approximation guarantee but is allowed to return any $T \subseteq \mathcal{N}$.⁴

⁴ Recall from Sect. 2 that ℓ_+ and ℓ_- are the components of ℓ with positive and negative sign, respectively.

Proof It suffices to show that for any $\epsilon > 0$, with high probability, Algorithm 1 from [25] generates $\mathbf{y}(t_f) \in [0, 1]^{\mathcal{N}}$ such that $\mathbf{y}(t_f) \in t_f \cdot \mathcal{P}$ and

$$F(\mathbf{y}(t_f)) + L(\mathbf{y}(t_f)) \geq t_f e^{-t_f} f(OPT) + (1 - e^{-t_f})\ell_+(OPT) + t_f \ell_-(OPT) - \mathcal{O}(\epsilon M) \tag{7.2}$$

in $\text{poly}(n, 1/\epsilon)$ time, where $M \triangleq \max\{\max_{u \in \mathcal{N}} f(u|\emptyset), -\min_{u \in \mathcal{N}} f(u|\mathcal{N}-u)\} > 0$. How to use $\mathbf{y}(t_f)$ to generate a set T satisfying the conditions in the statement of this lemma is standard and is deferred to the appendix.

First, we briefly review the *measured continuous greedy* algorithm introduced by Feldman et al. [13]. The idea is to continuously evolve a solution $\mathbf{y}(t)$ from time $t = 0$ to time $t = t_f$ such that $\mathbf{y}(t) \in (t \cdot \mathcal{P}) \cap ((1 - e^{-t}) \cdot [0, 1]^{\mathcal{N}})$. At all times, $\mathbf{y}'(t) = \mathbf{z}(t) \circ (\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t))$, where $\mathbf{z}(t) \in \mathcal{P}$. To transform this continuous process into an algorithm running in finite time, it is necessary to discretize time into timesteps of size δ , where δ evenly divides t_f . Then $\mathbf{y}(t + \delta) \triangleq \mathbf{y}(t) + \delta \mathbf{z}(t) \circ (\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t))$. How small δ needs to be to achieve the desired approximation factor is given by a polynomial in terms of n and ϵ .

Algorithm 1 of [25] combines measured continuous greedy with Feldman’s *distorted objective* [20]. Specifically, Algorithm 1 of [25] defines the objective at time t to be

$$\Phi(t) = (1 - \delta)^{(t_f-t)/\delta} F(\mathbf{y}(t)) + L(\mathbf{y}(t)) \approx e^{t-t_f} F(\mathbf{y}(t)) + L(\mathbf{y}(t)) \tag{7.3}$$

and chooses $\mathbf{z}(t)$ so that with high probability,

$$\Phi(t + \delta) - \Phi(t) \geq \delta [e^{-t_f} f(OPT) + \ell(OPT)] - \frac{\delta}{t_f} \cdot \mathcal{O}(\epsilon M), \tag{7.4}$$

assuming that ℓ is non-positive [25, Lemma 3.8]. Summing this inequality over all $\frac{t_f}{\delta}$ timesteps yields the desired result for non-positive ℓ .⁵

We claim that when the sign of ℓ is unconstrained, the following generalization of [25, Lemma 3.8] holds:

$$\Phi(t + \delta) - \Phi(t) \geq \delta [e^{-t_f} f(OPT) + (1 - \delta)^{t/\delta} \ell_+(OPT) + \ell_-(OPT)] - \frac{\delta}{t_f} \cdot \mathcal{O}(\epsilon M). \tag{7.5}$$

To show this, the only part of the proof of [25, Lemma 3.8] that needs to change is the part where [25, Lemma 3.7] is invoked. Lemma 3.7 of [25] states that for non-positive ℓ ,

$$L(\mathbf{y}(t + \delta)) - L(\mathbf{y}(t)) = \delta L(\mathbf{z}(t) \circ (\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t))) \geq \delta \langle \ell, \mathbf{z}(t) \rangle. \tag{7.6}$$

⁵ Actually, the authors of [25] only described their algorithm for the case of $t_f = 1$. Though as noted in [24], the proof of their algorithm easily generalizes to arbitrary t_f . Of course, the step size δ must be adjusted accordingly.

For unconstrained ℓ , we obtain the following inequality instead:

$$\begin{aligned} L(\mathbf{y}(t + \delta)) - L(\mathbf{y}(t)) &= \delta L(\mathbf{z}(t) \circ (\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t))) \\ &= \delta (\langle \ell_+, \mathbf{z}(t) \circ (\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t)) \rangle + \langle \ell_-, \mathbf{z}(t) \circ (\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t)) \rangle) \\ &\geq \delta (\langle \ell_+, \mathbf{z}(t) \rangle \cdot (1 - \delta)^{t/\delta} + \langle \ell_-, \mathbf{z}(t) \rangle), \end{aligned} \tag{7.7}$$

where the last inequality follows from [25, Lemma 3.1], which states that $\mathbf{y}_u(t) \leq 1 - (1 - \delta)^{\frac{t}{\delta}}$ for all $u \in \mathcal{N}$. It is easy to verify that Eq. (7.5) follows after substituting Eq. (7.7) in place of [25, Lemma 3.7] in the proof of [25, Lemma 3.8].

To go from Eq. (7.5) to the conclusion, we just need to check that $\sum_{i=0}^{t_f/\delta-1} \delta(1 - \delta)^i \geq 1 - (1 - \delta)^{t_f/\delta} \geq 1 - e^{-t_f}$. Thus, Eq. (7.2) has been proven. \square

Corollary 7.3 *When $\ell \geq 0$, there is a $(e^{-1} - \epsilon, 1 - e^{-1})$ -approximation algorithm for RegularizedCSM.*

Proof The result follows immediately from substituting $t_f = 1$ into Lemma 7.2. \square

Before proving Theorem 7.1, we will need two more lemmas. The first is very simple.

Lemma 7.4 (Trivial Approximation for RegularizedCSM) *When ℓ is unconstrained, there exists a $(0, 1)$ -approximation algorithm for RegularizedCSM.*

Proof Ignore f and maximize ℓ , which can be done in polynomial time as noted in the preliminaries. \square

The next lemma combines Lemma 7.2 with the *aided measured continuous greedy* used by [14].

Lemma 7.5 (Guarantee of Distorted Aided Measured Continuous Greedy) *Let ℓ be unconstrained. If we run Distorted Aided Measured Continuous Greedy given a fractional solution \mathbf{z} and a polytope \mathcal{P} for a total of t_f time, where $t_f \geq t_s$, it will generate $\mathbf{y} \in t_f \mathcal{P} \cap ((1 - e^{-t_f}) \cdot [0, 1]^{\mathcal{N}})$ such that*

$$\begin{aligned} \mathbb{E} [F(\mathbf{y}) + L(\mathbf{y})] &\geq e^{-t_f} [(e^{t_s} + t_f e^{t_s} - t_s e^{t_s} - 1 - o(1))f(OPT) \\ &\quad + (-e^{t_s} + 1)F(\mathbf{z} \wedge \mathbf{1}_{OPT}) \\ &\quad + (-e^{t_s} - t_f e^{t_s} + t_s e^{t_s} + 1 + t_f)F(\mathbf{z} \vee \mathbf{1}_{OPT})] \\ &\quad + (1 - e^{-t_f})L_+(\mathbf{1}_{OPT} \setminus \mathbf{z}) + (1 - e^{t_s - t_f})L_+(\mathbf{1}_{OPT} \wedge \mathbf{z}) \\ &\quad + t_f L_-(\mathbf{1}_{OPT} \setminus \mathbf{z}) + (t_f - t_s)L_-(\mathbf{1}_{OPT} \wedge \mathbf{z}). \end{aligned} \tag{7.8}$$

Note that the terms depending on f are precisely the same as those in Lemma 5.3.

The proof is deferred to Sect. A.3.

Proof of Theorem 7.1 The algorithm is similar to that of Theorem 5.1.

1. Run the trivial approximation algorithm (Lemma 7.4).

2. Generate \mathbf{z} using the local search procedure described by [14, Lemma 3.1] on $(f + \ell, \mathcal{P})$. This finds $\mathbf{z} \in \mathcal{P}$ such that

$$\begin{aligned} F(\mathbf{z}) + L(\mathbf{z}) &\geq \frac{(F + L)(\mathbf{z} \vee \mathbf{1}_{OPT}) + (F + L)(\mathbf{z} \wedge \mathbf{1}_{OPT})}{2} \\ &\quad - o(1) \cdot (f + \ell)(OPT) \\ &\geq \frac{1}{2}F(\mathbf{z} \vee \mathbf{1}_{OPT}) + \frac{1}{2}F(\mathbf{z} \wedge \mathbf{1}_{OPT}) + \frac{1}{2}\ell(OPT) \\ &\quad + \frac{1}{2}L(\mathbf{z} \wedge \mathbf{1}_{OPT}) - o(1) \cdot (f + \ell)(OPT), \end{aligned} \quad (7.9)$$

and

$$F(\mathbf{z}) + L(\mathbf{z}) \geq F(\mathbf{z} \wedge \mathbf{1}_{OPT}) + L(\mathbf{z} \wedge \mathbf{1}_{OPT}) - o(1) \cdot (f + \ell)(OPT). \quad (7.10)$$

Note that unlike Theorem 5.1, there is no guessing step.

3. Run distorted aided measured continuous greedy given \mathbf{z} (Lemma 7.5), for all pairs

$$(t_s, t_f) \in \mathcal{T} \triangleq \{(0.1x, 1) \mid 0 \leq x \leq 10\}. \quad (7.11)$$

4. Round \mathbf{z} from step 1 and all fractional solutions found in steps 2 and 3 to valid integral solutions using pipage rounding, which preserves the value of $F + L$ in expectation.
5. Return the solution from step 4 with the maximum value. Let \mathbf{OPT}' be the expected value of this solution.

As in the proof of Theorem 5.1, for a fixed β , we claim that to find a lower bound on α such that the following inequality is true:

$$\begin{aligned} \mathbf{OPT}' &\triangleq \max \left(\mathbb{E}[F(\mathbf{z}) + L(\mathbf{z})], \max_{(t_s, t_f) \in \mathcal{T}} (\mathbb{E}[F(\mathbf{y}_{t_s, t_f}) + L(\mathbf{y}_{t_s, t_f})]) \right) \\ &\geq \alpha F(OPT) + \beta \ell(OPT), \end{aligned} \quad (7.12)$$

it suffices to solve the following linear program:

$$\begin{aligned} &\max x_1 \\ &\text{s.t. } (x_1, x_2, x_3, x_4, x_5) \in \text{conv}(\{(0, 0, 0, 1, 1), (0, 0.5, 0.5, 0.5, 1), (0, 1, 0, 0, 1)\}) \cup \\ &\quad \{(e^{t_s - t_f} + t_f e^{t_s - t_f} - t_s e^{t_s - t_f} - e^{-t_f}, \\ &\quad \quad - e^{t_s - t_f} + e^{-t_f}, \\ &\quad \quad - e^{t_s - t_f} - t_f e^{t_s - t_f} + t_s e^{t_s - t_f} + e^{-t_f} + e^{-t_f} t_f, \\ &\quad \quad 1 - e^{-t_f}, \\ &\quad \quad 1 - e^{t_s - t_f}) \mid (t_s, t_f) \in \mathcal{T}\} \\ &\text{and } x_2 \geq 0, x_3 \geq 0, x_4 \geq \beta, x_5 \geq \beta. \end{aligned} \quad (7.13)$$

Any point $(x_1, x_2, x_3, x_4, x_5)$ within the convex hull satisfies:

$$\begin{aligned} \mathbf{OPT}' \geq & x_1 f(\mathbf{OPT}) + x_2 F(\mathbf{z} \wedge \mathbf{1}_{\mathbf{OPT}}) + x_3 F(\mathbf{z} \vee \mathbf{1}_{\mathbf{OPT}}) + x_4 L(\mathbf{1}_{\mathbf{OPT}} \setminus \mathbf{z}) \\ & + x_5 L(\mathbf{z} \wedge \mathbf{1}_{\mathbf{OPT}}) \end{aligned} \tag{7.14}$$

up to $o(1)$ terms. The points determining the hull are as follows:

- $(0, 0, 0, 1, 1)$ corresponds to Lemma 7.4.
- $(0, 0.5, 0.5, 0.5, 1)$ corresponds to Eq. (7.9).
- $(0, 1, 0, 0, 1)$ corresponds to Eq. (7.10).
- The remaining points correspond to Lemma 7.5 for all $(t_s, t_f) \in \mathcal{T}$.

The constraints $x_2, x_3 \geq 0$ ensure that

$$\mathbf{OPT}' \geq x_1 f(\mathbf{OPT}) + x_4 L(\mathbf{1}_{\mathbf{OPT}} \setminus \mathbf{z}) + x_5 L(\mathbf{z} \wedge \mathbf{1}_{\mathbf{OPT}}). \tag{7.15}$$

The constraints $\min(x_4, x_5) \geq \beta$ ensure that

$$\mathbf{OPT}' \geq x_1 f(\mathbf{OPT}) + \beta \ell(\mathbf{OPT}). \tag{7.16}$$

□

7.2 Inapproximability

In this subsection, we prove Theorem 7.6, which can be used to show that Theorem 7.1 is tight for $\beta \geq (e - 1)/e$. We then discuss whether the construction used in Theorem 7.6 could potentially be extended to RegularizedUSM.

Theorem 7.6 (Inapproximability of RegularizedCSM Near $\beta = 1$) *For any $0 \leq \beta \leq 1$, there exist instances of RegularizedCSM with non-negative ℓ such that a $(1 - \beta + \epsilon, \beta)$ -approximation would require exponentially many value queries.*

Proof Define $\alpha \triangleq 1 - \beta + \epsilon$. By Lemma 5.8, it suffices to construct a submodular function f satisfying

$$\max_{\mathbf{x} \in \mathcal{P}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] < \max_{S \in \mathcal{I}} [\alpha \cdot f(S) + \beta \cdot \ell(S)]. \tag{7.17}$$

We use the same f that Vondrak [18] uses for proving the inapproximability of maximization over matroid bases. Specifically, define $\mathcal{N} = \{a_1, \dots, a_k, b_1, \dots, b_k\}$ and let f correspond to the sum of directed cut functions of k disjoint arcs; that is, $f(S) \triangleq \sum_{i=1}^k [a_i \in S \text{ and } b_i \notin S]$. Its multilinear extension is $F(\mathbf{x}_{a_1 \dots a_k}, \mathbf{x}_{b_1 \dots b_k}) = \sum_{i=1}^k \mathbf{x}_{a_i} (1 - \mathbf{x}_{b_i})$. We define \mathcal{I} to consist of precisely the subsets of \mathcal{N} that contain at most one element from a_1, \dots, a_k and at most $k - 1$ elements from b_1, \dots, b_k , resulting in the following matroid independence polytope:

$$\mathcal{P} = \left\{ (\mathbf{x}_{a_i}, \mathbf{x}_{b_i}) \mid \sum_{i=1}^k \mathbf{x}_{a_i} \leq 1 \text{ and } \sum_{i=1}^k \mathbf{x}_{b_i} \leq k - 1 \right\}. \tag{7.18}$$

Finally, we define ℓ as $\ell(a_i) = 0$, $\ell(b_i) = \frac{1}{k}$. Then the RHS of Eq. (7.17) is at least:

$$\max_{S \in \mathcal{I}} [\alpha f(S) + \beta \ell(S)] \geq (\alpha f + \beta \ell)(\{a_1, b_2, b_3, \dots, b_k\}) = \alpha + \beta \cdot \frac{k-1}{k}, \quad (7.19)$$

while the LHS of Eq. (7.17) is:

$$\begin{aligned} \max_{\mathbf{x} \in \mathcal{P}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] &= \max_{0 \leq p \leq 1/k, 0 \leq q \leq (k-1)/k} [kp(1-q) + q] \\ &= \max_{0 \leq p \leq 1/k, 0 \leq q \leq (k-1)/k} [q(1-kp) + kp] \\ &= \max_{0 \leq p \leq 1/k} [(1-kp) + kp] = 1, \end{aligned} \quad (7.20)$$

where the third equality follows because the expression is always maximized by setting $q = \frac{k-1}{k}$. For sufficiently large k we have

$$\alpha + \beta \cdot \frac{k-1}{k} \geq (\alpha + \beta) \frac{k-1}{k} = (1 + \epsilon) \cdot \frac{k-1}{k} > 1. \quad (7.21)$$

Equation (7.17) follows. \square

In fact, the bound of Theorem 7.6 is (nearly) tight for β close to one.

Corollary 7.7 (Tight RegularizedCSM Near $\beta = 1$ for $\ell \geq 0$) *For all $\frac{e-1}{e} \leq \beta < 1$, there is a $(1 - \beta - \epsilon, \beta)$ -approximation algorithm for RegularizedCSM with non-negative ℓ , nearly matching the bound of Theorem 7.6.*

Proof The better of Corollary 7.3 and Lemma 7.4 will be an (α, β) -approximation for all (α, β) lying above the segment connecting $(\frac{1}{e} - \epsilon, \frac{e-1}{e} - \epsilon)$ and $(0, 1)$. \square

As the f used by Lemma 5.8 to prove Theorem 7.6 is just a directed cut function, it is natural to ask whether directed cut functions can be used by Lemma 5.8 to show improved inapproximability for RegularizedUSM. We build on Proposition 5.5 to show that doing so is impossible.

Proposition 7.8 *When ℓ is unconstrained, setting f to be a directed cut function in Lemma 5.8 cannot be used to show $(0.5, 1)$ -inapproximability for RegularizedUSM.*

The proof is deferred to Sect. A.3.

8 Unconstrained ℓ

The results of this section are summarized in Figs. 5 and 6.

8.1 Approximability

In this section, we prove Theorems 8.1 and 8.3.

Theorem 8.1 *For all $t \geq 0$, there is a $\left(\frac{te^{-t}}{t+e^{-t}} - \epsilon, \frac{t}{t+e^{-t}}\right)$ -approximation algorithm for RegularizedUSM. This algorithm achieves the same approximation guarantee for RegularizedCSM when $t \leq 1$.*

Algorithm 1 Simple RegularizedCSM (t)

- 1: Let $T \leftarrow$ the set returned by running Lemma 7.2 for $t_f = t$ time.
 - 2: Let $T' \leftarrow$ the set returned by Lemma 7.4 (trivial approximation).
 - 3: **if** $f(T) + \ell(T) \geq \ell(T')$ **then return** T .
 - 4: **else return** T' .
-

Proof It suffices to show that Algorithm 1 achieves the desired approximation factor. Disregard the factors of $o(1)$ in Lemma 7.2; they can be taken into account later at the cost of introducing the factor of ϵ . The set returned from Lemma 7.4 actually satisfies the stronger guarantee

$$\ell(T') \geq \ell_+(OPT), \tag{8.1}$$

because if we define $OPT^* \triangleq OPT \cap \{u : u \in \mathcal{N} \wedge \ell(u) > 0\}$, $OPT^* \in \mathcal{I}$ due to \mathcal{I} being downward-closed, and thus $\ell(T') \geq \ell(OPT^*) = \ell_+(OPT)$.

Next, add $t + e^{-t} - 1$ times Eq. (8.1) to the inequality of Lemma 7.2.

$$\begin{aligned} (t + e^{-t})\mathbb{E}[\max(f(T) + \ell(T), \ell(T'))] &\geq \mathbb{E}[f(T) + \ell(T)] + (t + e^{-t} - 1)\mathbb{E}[\ell(T')] \\ &\geq te^{-t}f(OPT) + t(\ell_+(OPT) + \ell_-(OPT)) \\ &= te^{-t}f(OPT) + t\ell(OPT). \end{aligned} \tag{8.2}$$

Dividing both sides by $t + e^{-t}$ gives the desired result after accounting for the factors of $o(1)$:

$$\mathbb{E}[\max(f(T) + \ell(T), \ell(T'))] \geq \left(\frac{te^{-t}}{t + e^{-t}} - \epsilon\right) f(OPT) + \frac{t}{t + e^{-t}} \ell(OPT). \tag{8.3}$$

□

Note that an analog of Corollary 7.7 (Tight RegularizedCSM Near $\beta = 1$ for $\ell \geq 0$) holds for unconstrained ℓ , though for a smaller range of β :

Corollary 8.2 (Tight RegularizedCSM Near $\beta = 1$) *There is a $(1 - \beta - \epsilon, \beta)$ -approximation algorithm for RegularizedCSM for any $\frac{\epsilon}{\epsilon+1} \leq \beta < 1$, almost matching the bound of Theorem 7.6.*

Proof Setting $t = 1$, the output of Theorem 8.1 is both a $\left(\frac{1}{e+1} - \epsilon, \frac{e}{e+1}\right)$ -approximation and a $(0, 1)$ -approximation for RegularizedCSM. Therefore it is also an (α, β) -approximation for all (α, β) lying above the segment connecting $\left(\frac{1}{e+1} - \epsilon, \frac{e}{e+1}\right)$ and $(0, 1)$. \square

However, our result is not tight for $\beta < e/(e + 1)$; it turns out that it is possible to do a little better than Theorem 8.1 for β near 0.7 by making use of Lemma 7.5.

Theorem 8.3 *There is a $(0.280, 0.7)$ -approximation algorithm for RegularizedCSM.*

Proof The algorithm is Theorem 7.1 augmented to use the guessing step from Theorem 5.1. That is, we start by guessing the value of $\ell_-(OPT)$ to within a factor of $1 + \epsilon$ and replacing \mathcal{P} with $\mathcal{P} \cap \{\mathbf{x} : L_-(\mathbf{x}) \geq (1 + \epsilon)\ell_-(OPT)\}$ as in Theorem 5.1, and then run Theorem 7.1.

To analyze the guarantee of this algorithm, we set up a linear program similar to that of Theorem 7.1 with two additional variables x_6 and x_7 corresponding to $L_-(\mathbf{1}_{OPT} \setminus \mathbf{z})$ and $L_-(\mathbf{z} \wedge \mathbf{1}_{OPT})$, respectively. Again, we ignore terms that are $o(1)$ and those depending on ϵ .

max x_1

$$\begin{aligned}
 \text{s.t. } (x_1, x_2, x_3, x_4, x_5, x_6, x_7) \in & \text{conv}(\{(0, 0, 0, 1, 1, 0, 0), \\
 & (0, 0.5, 0.5, 0.5, 1, 1, 1), \\
 & (0, 1, 0, 0, 1, 0, 1)\} \cup \\
 & \{(e^{t_s-t_f} + t_f e^{t_s-t_f} - t_s e^{t_s-t_f} - e^{-t_f}, \\
 & -e^{t_s-t_f} + e^{-t_f}, \\
 & -e^{t_s-t_f} - t_f e^{t_s-t_f} + t_s e^{t_s-t_f} + e^{-t_f} + e^{-t_f} t_f, \\
 & 1 - e^{-t_f}, 1 - e^{t_s-t_f}, t_f, t_f - t_s) \mid (t_s, t_f) \in \mathcal{T}\}) \\
 \text{and } x_2 \geq 0, x_3 \geq 0, x_4 \geq \beta, x_5 \geq \beta, x_6 \leq \beta, x_7 \leq \beta. & \tag{8.4}
 \end{aligned}$$

Any point $(x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ within the convex hull satisfies:

$$\begin{aligned}
 OPT' \geq & x_1 f(OPT) + x_2 F(\mathbf{z} \wedge \mathbf{1}_{OPT}) + x_3 F(\mathbf{z} \vee \mathbf{1}_{OPT}) \\
 & + x_4 L(\mathbf{1}_{OPT} \setminus \mathbf{z}) + x_5 L(\mathbf{z} \wedge \mathbf{1}_{OPT}) + x_6 L_-(\mathbf{1}_{OPT} \setminus \mathbf{z}) \\
 & + x_7 L_-(\mathbf{z} \wedge \mathbf{1}_{OPT}) \tag{8.5}
 \end{aligned}$$

up to $o(1)$ terms. The points determining the hull are as follows:

- $(0, 0, 0, 1, 1, 0, 0)$ corresponds to Lemma 7.4.
- $(0, 0.5, 0.5, 0.5, 1, 1, 1)$ corresponds to Eq. (7.9). Note that this inequality holds only because of the guessing step.
- $(0, 1, 0, 0, 1, 0, 1)$ corresponds to Eq. (7.10).
- The remaining vertices correspond to Lemma 7.5.

Choosing $\mathcal{T} = \{(0.205, 0.955)\}$ and solving the linear program gives $x_1 \geq 0.280$ as desired. \square

8.2 Inapproximability

In this subsection, we prove Theorems 8.4 and 8.5. Note that Theorem 7.6 cannot possibly apply to RegularizedUSM because Theorem 8.1 achieves $(1 - \beta + \epsilon, \beta)$ -approximations for β close to one. Unfortunately, we are unable to prove $(1, \epsilon)$ -inapproximability of RegularizedUSM, but we modify Theorem 5.6 to show improved inapproximability for unconstrained ℓ than for ℓ non-negative or ℓ non-positive.

Theorem 8.4 (Inapproximability of RegularizedUSM) *There are instances of RegularizedUSM where $(\alpha(\beta), \beta)$ is inapproximable for any $(\alpha(\beta), \beta)$ in Table 4. In particular, $(0.440, 1)$ is inapproximable.*

Proof Set f to be the same as defined in Lemma 4.5, and define $S \triangleq \{a, b_1\}$. For a fixed β , we can show (α, β) -inapproximability using Lemma 5.8 if it is possible to choose ℓ and κ such that:

$$\max_{\mathbf{x} \in [0,1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] < \alpha f(\{a, b_1\}) + \beta \ell(\{a, b_1\}) = \alpha + \beta \ell(\{a, b_1\}), \tag{8.6}$$

which is equivalent to

$$\max_{\mathbf{x} \in [0,1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] - \beta \ell(\{a, b_1\}) < \alpha. \tag{8.7}$$

For a fixed β , our goal is to choose ℓ and κ to minimize the LHS of the above inequality. Theorem 1.3 of [24] sets $\ell_a = \ell_b = 0$, and then chooses κ and $\ell_{a_1 \dots k} = \ell_{b_1 \dots k} \triangleq \ell_p$ in order to minimize the quantity

$$\begin{aligned} \max_{\mathbf{x} \in [0,1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] - \beta \ell(\{a, b_1\}) &= \max_{\mathbf{x} \in [0,1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] - \beta \ell_p \\ &\approx \max_{0 \leq q \leq 1, 0 \leq p} [(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p}) + 2p\ell_p] - \beta \ell_p \end{aligned} \tag{8.8}$$

Table 4 Inapproximability of $(\alpha(\beta), \beta)$ -approximations for RegularizedUSM with unconstrained ℓ (Theorem 8.4)

β	$\alpha(\beta)$	κ	ℓ_p	ℓ_q
0.1	0.0935	0.6708	-0.6064	-0.2644
0.2	0.1743	0.6598	-0.5370	-0.2102
0.3	0.2432	0.6577	-0.4809	-0.1504
0.4	0.3008	0.6519	-0.4261	-0.0940
0.5	0.3476	0.6465	-0.3778	-0.0407
0.6	0.3844	0.6390	-0.3309	0.0128
0.7	0.4114	0.6315	-0.2881	0.0655
0.8	0.4293	0.6226	-0.2498	0.1155
0.9	0.4383	0.6144	-0.2142	0.1668
1.0	0.4390	0.6022	-0.1819	0.2152

These results are depicted graphically in Fig. 5

However, allowing $\ell_a = \ell_b \triangleq \ell_q$ to be nonzero gives better bounds for all β . That is, our goal is to compute

$$\min_{0 \leq \kappa \leq 1, \ell_q, \ell_p} \left[\max_{0 \leq q \leq 1, 0 \leq p} [(1 - \kappa)2q(1 - q) + \kappa 2(1 - q)(1 - e^{-p}) + 2p\ell_p + 2q\ell_q] - \beta(\ell_p + \ell_q) \right]. \tag{8.9}$$

We can approximate the optimal value by brute forcing over a range of (κ, ℓ_q, ℓ_p) . The best triples we found are displayed in Table 4. Allowing ℓ_q to be negative gives superior bounds for β near zero, while allowing ℓ_q to be positive gives superior bounds for β near one. In particular, for $\beta = 1$, taking $\kappa = 0.6022$, $\ell_p = -0.1819$, and $\ell_q = 0.2152$ gives $\alpha(\beta) \approx 0.4390 < 0.440$. \square

We can do slightly better than Theorem 8.4 for β very close to one with a construction inspired by [24, Theorem 1.6].

Theorem 8.5 (Inapproximability of RegularizedUSM, $\beta = 1$) *There are instances of RegularizedUSM where $(0.408, 1)$ is inapproximable.*

Proof Again, we use Lemma 5.8. Let $\mathcal{N} \triangleq \{a_{1\dots k}, b_{1\dots k}\}$, and define f as the directed cut function of a generalized hyperedge $(a_{1\dots k}; b_{1\dots k})$; that is, the generalized hyperedge is said to be cut by S if S contains at least one of the tails of the hyperedge $(a_{1\dots k})$ but not all of the heads of the hyperedge $(b_{1\dots k})$:

$$f(S) \triangleq [S \cap \{a_{1\dots k}\} \neq \emptyset] \cdot [\{b_{1\dots k}\} \not\subset S]. \tag{8.10}$$

Also define $\ell(a_i) = -0.2037$, $\ell(b_i) = 0.2037$, $p \triangleq \sum_{i=1}^k \mathbf{x}_{a_i}$ and $q \triangleq k - \sum_{i=1}^k \mathbf{x}_{b_i}$, and \mathcal{G} such that a_1, \dots, a_k and b_1, \dots, b_k are symmetric. Then as $k \rightarrow \infty$,

$$F(\bar{\mathbf{x}}) = \left(1 - (1 - p/k)^k\right) \left(1 - (1 - q/k)^k\right) \approx (1 - e^{-p})(1 - e^{-q}). \tag{8.11}$$

Now,

$$\begin{aligned} \max_{\mathbf{x} \in [0,1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] &= \max_{p,q \geq 0} [(1 - e^{-p})(1 - e^{-q}) - 0.2037(p + q) + 0.2037k] \\ &= 0.2037k, \end{aligned} \tag{8.12}$$

where the last equality follows since the maximum is attained at $p = q = 0$, which may be verified using a numerical optimizer. On the other hand,

$$\max_S [\alpha f(S) + \ell(S)] \geq (f + \ell)(\{a_1, b_{1\dots k-1}\}) = \alpha + 0.2037(k - 2). \tag{8.13}$$

It follows from Lemma 5.8 that we have shown $(\alpha, 1)$ -inapproximability for any α satisfying

$$0.2037k < \alpha + 0.2037(k - 2) \implies \alpha > 0.4074. \tag{8.14}$$

\square

9 Open Problems

For all of the settings we consider, there is still a range of β for which there is a gap between the highest $\alpha(\beta)$ known to be approximable and the lowest $\alpha(\beta)$ known to be inapproximable. The open problems discussed below focus more on RegularizedUSM than RegularizedCSM due to this gap being larger for RegularizedUSM.

9.1 Approximability

Section 5: Non-positive ℓ .

Theorem 5.1 attains bounds for RegularizedUSM with $\alpha \geq 0.398$. What is the maximum α such that an (α, β) approximation exists for some β ? In particular, is $\alpha = 0.5$ achievable?

Section 8: Unconstrained ℓ .

Is there an algorithm that achieves an $(\epsilon, 1)$ approximation for RegularizedUSM? Recall that for RegularizedCSM this was achievable when ℓ was restricted to be non-positive or non-negative (Lemmas 5.2 and 6.6, respectively), but not in the case where ℓ can have arbitrary sign (Theorem 7.6).

Section A.2: Online RegularizedUSM.

Can the α in Proposition A.4 be improved? Is there an online algorithm that works for general non-monotone f and ℓ non-positive? We note that the *semi-streaming* algorithms studied by Kazemi et al. [22] and Nikolakaki et al. [23] provide a $(0.5, 1)$ -approximation algorithm for RegularizedUSM when f is monotone. For non-monotone USM, simply selecting each element of f with probability 0.5 achieves a 0.25-approximation [15]. For non-monotone CSM where the constraint is a cardinality constraint, Buchbinder et al. [29] provide an online algorithm achieving a competitive ratio of $\frac{56}{627} > 0.0893$ when *preemption* is allowed.

9.2 Inapproximability

Table 5 summarizes some of the best-known inapproximability results and their corresponding approximation guarantees. The gaps between approximability and inapproximability are particularly large in the second and fourth rows, corresponding to RegularizedUSM for $\ell \leq 0$ and unconstrained ℓ , respectively. Most of our inapproximability results are applications of the symmetry gap technique to modified versions of [17, Theorem 5.4]. Perhaps further reducing these gaps is simply a matter of finding a different construction to apply the symmetry gap technique to.

Supplementary information

Python code for all theorems requiring computational verification can be found at <https://github.com/bqi343/maximizing-sums>.

Table 5 Gaps between current approximability and inapproximability

Section	Problem	Inapproximability	Approximability	Source of Inapproximability
Sect. 4	CSM	0.478	0.385	[17, Theorem 5.4]
Sect. 5	RegularizedUSM, $\ell \leq 0$	$(0.5, 2 \ln 2 - \epsilon)$	None for $\alpha = 0.5$	Proposition 5.9
Sect. 6	RegularizedUSM, $\ell \geq 0$	$(0.5, 2\sqrt{2}/3 + \epsilon)$	$(0.5, 0.75)$	Theorem 6.9
Sect. 8	RegularizedUSM	$(0.408, 1)$	$(0, 1)$	Theorem 8.5

Acknowledgements I thank Tasuku Soma for mentoring me through the MIT Undergraduate Research Opportunities Program, as well as Moran Feldman for providing many helpful suggestions on an earlier version of this paper.

Author Contributions Benjamin Qi wrote the manuscript.

Funding Open Access funding provided by the MIT Libraries. This work was funded by the MIT Department of Mathematics.

Declarations

Conflict of interest None.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix A

A.1 Prior Work

We outline the general idea for all *continuous greedy* algorithms because our results build on them.

A.1.1 Submodular Maximization

For surveys of submodular maximization results, see Krause and Golovin [9] or Buchbinder and Feldman [30]. Here, we present the prior work most relevant to our results.

f **Monotone (Constrained):**

It is well-known that a simple *greedy* algorithm achieves a $(1 - \frac{1}{e})$ -approximation for maximizing monotone submodular functions subject to a cardinality constraint [11], and that this approximation factor is optimal [16].

Calinescu et al. [12] introduced the *continuous greedy* algorithm, which achieves a $(1 - \frac{1}{e})$ -approximation for maximizing the multilinear extension of a monotone submodular function over a solvable down-closed polytope \mathcal{P} . The idea is to continuously evolve a fractional solution $\mathbf{y}(t)$ from “time” $t = 0$ to $t = 1$ such that

$$\mathbf{y}(t) \in t \cdot \mathcal{P} \quad \text{and} \quad F(\mathbf{y}(t)) \geq (1 - e^{-t})f(OPT). \quad (\text{A1})$$

This continuous process can be discretized into a polynomial number of steps at the cost of a negligible loss in the approximation factor. If \mathcal{P} is the *matroid polytope* corresponding to a matroid $\mathcal{M} = (\mathcal{N}, \mathcal{I})$, then *pipage rounding* may be used to round

the fractional solution $\mathbf{y}(1)$ to an independent set $S \in \mathcal{I}$ such that $\mathbb{E}[f(S)] \geq F(\mathbf{y}(1))$ [18].

f Non-monotone (Unconstrained):

Feige et al. [15] showed that no polynomial-time algorithm may provide a $(0.5 + \epsilon)$ -approximation for maximizing a non-monotone submodular function. Buchbinder et al. [10] later discovered a *randomized double greedy* algorithm that achieves a 0.5-approximation in expectation. The idea is to iterate through the elements of the ground set \mathcal{N} in arbitrary order, and for each one choose whether or not to include it in the returned set with some probability.

f Non-monotone (Constrained):

Feldman et al. [13] showed a $1/e > 0.367$ -approximation for maximizing the *multilinear extension* of a non-monotone submodular function over a solvable down-closed polytope \mathcal{P} using a *measured continuous greedy*. The idea is to continuously evolve a fractional solution $\mathbf{y}(t)$ from $t = 0$ to $t = 1$ such that

$$\mathbf{y}(t) \in (t \cdot \mathcal{P}) \cap ((1 - e^{-t}) \cdot [0, 1]^{\mathcal{N}}) \quad \text{and} \quad F(\mathbf{y}(t)) \geq te^{-t} f(OPT). \quad (\text{A2})$$

As with the original continuous greedy, the fractional solution $\mathbf{y}(1)$ can be rounded to an integer solution when \mathcal{P} is a matroid polytope. Additionally, when f is monotone, measured continuous greedy provides the same approximation guarantee as continuous greedy.

The approximation factor was later improved by Buchbinder and Feldman [14] to 0.385. The idea is to first run *local search* on the multilinear extension F to find a “locally optimal” fractional solution $\mathbf{z} \in \mathcal{P}$, round \mathbf{z} to a set Z , and then run a measured continuous greedy “aided” by Z . Either Z will be a 0.385-approximation in expectation, or the set returned by *aided measured continuous greedy* will be. The aided measured continuous greedy consists of running measured continuous greedy from $t = 0$ to $t = t_s$ on $\mathcal{N} \setminus Z$, followed by running measured continuous greedy from $t = t_s$ to $t = 1$ on the entire ground set \mathcal{N} , where $t_s = 0.372$. The optimal value of t_s was determined by solving a non-convex optimization problem.

On the inapproximability side, Oveis Gharan and Vondrak [17] showed that no polynomial-time algorithm may achieve a 0.478-approximation for maximizing a non-negative submodular function subject to a matroid independence constraint or a 0.491-approximation for maximizing a non-negative submodular function subject to a cardinality constraint using the *symmetry gap* framework of Vondrak [18]. The symmetry gap framework may also be used to succinctly re-prove the optimality of the $1 - \frac{1}{e}$ and $\frac{1}{2}$ approximation factors for monotone and non-monotone maximization, respectively, which were previously proved by ad hoc methods. The idea is that given a maximization problem with a symmetry gap of $\gamma \in (0, 1)$, we can construct a family of pairs of functions that require exponentially many value oracle queries to distinguish but whose maxima differ by a factor of γ . This in turn shows the impossibility of a $(\gamma + \epsilon)$ -approximation.

A.1.2 Regularized Submodular Maximization

Monotone f:

Sviridenko et al. [19] first presented an $(1 - 1/e - \epsilon, 1 - \epsilon)$ -approximation algorithm for RegularizedCSM involving a step where the value of $\ell(OPT)$ needs to be “guessed” to within a factor of $1 + \epsilon$, followed by continuous greedy on $\mathcal{P} \cap \{\mathbf{x} : L(\mathbf{x}) \geq \ell(OPT)\}$. Afterward, if \mathcal{P} is a matroid independence polytope, \mathbf{x} can be rounded to a set S such that $\mathbf{1}_S \in \mathcal{P}$ using pipage rounding such that $\mathbb{E}[f(S) + \ell(S)] \geq F(\mathbf{x}) + L(\mathbf{x})$.

Feldman [20] later combined continuous greedy with the notion of a *distorted objective* that initially places higher weight on the linear term and increases the weight on the submodular term over time. This *distorted continuous greedy* achieves the same approximation factor as [19] without the need for the guessing step. The idea is to continuously evolve a fractional solution $\mathbf{y}(t)$ from $t = 0$ to $t = t_f$ such that

$$\mathbf{y}(t) \in t \cdot \mathcal{P} \quad \text{and} \quad G_t(\mathbf{y}(t)) \geq (e^{t-t_f} - e^{-t_f})f(OPT) + t\ell(OPT), \tag{A3}$$

where $G_t(\mathbf{y}) \triangleq e^{t-t_f}F(\mathbf{y}) + L(\mathbf{y})$ is the distorted objective at time t .⁶ For $t_f = 1$, this gives a $(1 - 1/e - \epsilon, 1)$ -approximation, eliminating the ϵ in the linear term that appears in the bound of [19] due to the guessing step.

Using the symmetry gap technique [18], Bodek and Feldman [24, Theorem 1.1] proved that no $(1 - e^{-\beta} + \epsilon, \beta)$ -approximation algorithm for RegularizedUSM exists for any $\beta \geq 0$, even when ℓ is constrained to be non-positive (see Fig. 1 for an illustration). This matches the guarantee of *distorted continuous greedy*, which achieves a $(1 - e^{-\beta} - \epsilon, \beta)$ -approximation for RegularizedCSM whenever $\beta \in [0, 1]$. When ℓ is constrained to be non-positive, Lu et al. [25] achieve a $(1 - e^{-\beta} - \epsilon, \beta)$ -approximation for RegularizedCSM for any $\beta \geq 0$ using *distorted measured continuous greedy* (described below). For the remainder of this section, f is not necessarily monotone.

Non-positive ℓ :

Lu et al. [25] presented a $(\beta e^{-\beta} - \epsilon, \beta)$ -approximation algorithm for RegularizedCSM combining the measured and distorted continuous greedies mentioned above due to Feldman et al. [13, 20]. The idea is to continuously evolve a solution $\mathbf{y}(t)$ from $t = 0$ to $t = t_f$ such that

$$\mathbf{y}(t) \in (t \cdot \mathcal{P}) \cap ((1 - e^{-t}) \cdot [0, 1]^N) \tag{A4}$$

and

$$G_t(\mathbf{y}(t)) \geq te^{-t_f}f(OPT) + t\ell(OPT), \tag{A5}$$

where $G_t(\mathbf{y}) = e^{t-t_f}F(\mathbf{y}) + L(\mathbf{y})$ as in distorted continuous greedy above. Setting $t_f = \beta$ gives the desired approximation factor. Note that when $\ell = 0$, the guarantee of *distorted measured continuous greedy* becomes the same as *measured continuous greedy*. As noted in the previous paragraph, the approximation guarantee of this algorithm becomes the same as Feldman’s distorted continuous greedy when f is monotone.

Bodek and Feldman [24, Theorem 1.3] proved $(\alpha(\beta), \beta)$ -inapproximability for RegularizedUSM for all $\beta \geq 0$, where $\alpha(\beta)$ is an increasing function satisfying

⁶ Actually, the original paper shows this only for $t_f = 1$, but as noted by [24] this can easily be generalized.

$\alpha(1) \approx 0.478$, matching the best known inapproximability bound for maximizing a submodular function subject to a matroid constraint [17] (see Fig. 1 for an illustration).

Non-negative ℓ :

Bodek and Feldman [24, Theorem 1.5] showed that Buchbinder et al.'s *double greedy* [10] is simultaneously a $(\alpha, 1 - \alpha/2)$ -approximation for RegularizedUSM for any $\alpha \in [0, 0.5]$, and that a $(0.4998 + \epsilon, 1)$ -approximation for RegularizedUSM is impossible [24, Theorem 1.6].

Unconstrained ℓ :

Bodek and Feldman [24, Theorem 1.2] presented a $(\frac{\beta(1-\beta)}{1+\beta} - \epsilon, \beta - \epsilon)$ -approximation for RegularizedUSM using a *local search* technique.

Sun et al. [31] presented an algorithm for RegularizedCSM where the sign of ℓ is unconstrained which turns out to be identical to that of Lu et al. [25]. They showed that their algorithm outputs $\mathbf{x} \in \mathcal{P}$ such that $F(\mathbf{x}) + L(\mathbf{x}) \geq \max_{S \in \mathcal{P}} \left[\left(\frac{1}{e} - \epsilon\right) \cdot f(S) + \left(\frac{\beta(S)-e}{e(\beta(S)-1)}\right) \ell(S) \right]$, where $\beta(S) \triangleq \frac{\sum_{u \in S \cap \mathcal{N}^+} \ell(u)}{-\sum_{u \in S \cap \mathcal{N}^-} \ell(u)} \geq 0$. Note that when $\ell \leq 0$, $\beta(S) = 0$ and the coefficient of $\ell(S)$ is 1, recovering the approximation guarantee of Lu et al. [25]. However, this is not quite an (α, β) -approximation algorithm when ℓ is allowed to have arbitrary sign since $\beta(S)$ is not constant. Furthermore, the expression $\frac{\beta(S)-e}{e(\beta(S)-1)}$ could potentially be negative, which is problematic.

A.2 Online Algorithms for RegularizedUSM

Here, we discuss whether the general class of *online* algorithms can achieve approximation factors for RegularizedUSM when ℓ is not necessarily non-negative. First, we formally define the notion of online algorithms in the context of $f + \ell$ sums with f a directed cut function.

Definition A.1 (*Online Algorithms for Directed Cuts* [32]) When f is a directed cut function, we say that an algorithm is *online* in the sense of Bar-Noy and Lampis [32] if it works in the following setting:

1. The vertices of the ground set are revealed in the order u_1, u_2, \dots, u_n .
2. The algorithm is provided with ℓ_u , the total in-degree of u ($in(u)$), the total out-degree of u ($out(u)$), as well as the edges between u and all previously revealed vertices, only after u is revealed.
3. The algorithm makes an irreversible decision about whether to include u in the returned set before any vertices after u are revealed.

We note that both DeterministicDG and RandomizedDG are examples of online algorithms. Huang and Borodin [33] extended the notion of online algorithms to general non-monotone submodular f , though we do not consider their extension here. We first show that deterministic online algorithms cannot achieve any (α, β) -approximation.

Proposition A.2 *There are instances of RegularizedUSM with f a directed cut function and ℓ non-positive such that no deterministic online algorithm can provide a (α, β) -approximation for any $\alpha > 0$.*

Proof Suppose that after u_1 is revealed, the algorithm is provided with $in(u_1) = 1$, $out(u_1) = \alpha/2$, and $\ell(u_1) = 0$.

1. If the algorithm includes u_1 in the returned set, then this algorithm fails to provide the desired approximation factor on the following instance:

$$f(S) = \alpha/2 \cdot [u_1 \in S \text{ and } u_2 \notin S] + [u_2 \in S \text{ and } u_1 \notin S], \ell(u_2) = 0, \quad (A6)$$

since it outputs a set with value at most $\alpha/2$, whereas if we let $OPT \triangleq \{u_2\}$ then $f(OPT) = 1$ and $\ell(OPT) = 0$, implying that $\alpha f(OPT) + \beta \ell(OPT) = \alpha > \alpha/2$.

2. On the other hand, if the algorithm does not include u_1 in the returned set, then this algorithm fails to provide the desired approximation factor on the following instance:

$$f(S) = \alpha/2 \cdot [u_1 \in S \text{ and } u_2 \notin S] + [u_2 \in S \text{ and } u_1 \notin S], \ell(u_2) = -1, \quad (A7)$$

since it outputs a set with value 0 whereas if we let $OPT \triangleq \{u_1\}$ then $f(OPT) = \alpha/2$ and $\ell(OPT) = 0$, implying that $\alpha f(OPT) + \beta \ell(OPT) = \alpha/2 > 0$. \square

We next show that `RandomizedDG` does not achieve any (α, β) -approximation by adapting the proof of Proposition 6.4.

Proposition A.3 *There are instances of RegularizedUSM with f a directed cut function and ℓ non-positive such that `RandomizedDG` does not provide any (α, β) -approximation for any $\alpha > 0$.*

Proof Define f to be the same as in Proposition 6.4, $\ell(u_n) = 0$, and $\ell(u_i) = \frac{1-r}{n-1}$ for all $i \in [1, n - 1]$. Then

$$\max_S [\alpha f(S) + \beta \ell(S)] \geq (\alpha f + \beta \ell)(\{u_n\}) = \alpha \cdot 1 + \beta \cdot 0 = \alpha. \quad (A8)$$

For each $i \in [1, n - 1]$,

$$a_i = \ell(u_i) + f(u_i|\emptyset) = \frac{1-r}{n-1} + \frac{r}{n-1} = \frac{1}{n-1} \quad (A9)$$

and

$$b_i = -\ell(u_i) - f(u_i|\mathcal{N} \setminus \{u_i\}) = \frac{r-1}{n-1} + \frac{1}{n-1} = \frac{r}{n-1}. \quad (A10)$$

So by similar reasoning as the proof of Proposition 6.4, the fraction f of $\{u_1, \dots, u_{n-1}\}$ selected by double greedy will be close to $\frac{1}{r+1}$ with high probability. If u_n is included in the returned set, then the value of the set will be $f(1-r) + (1-f) \approx \frac{1}{r+1}$, whereas if u_n is not, then the value of the set will be $f \approx \frac{1}{r+1}$. So regardless of whether double greedy chooses to include u_n in the returned set or not, the returned set will have expected value at most $\frac{1}{r+1} + \epsilon < \alpha$ when both r and n are sufficiently large. \square

On the other hand, there are randomized algorithms that achieve (α, β) -approximations. In fact, the algorithm we provide next is *oblivious* in the sense of Feige and Shlomo [34]; that is, it uses only information local to each vertex.

Proposition A.4 *For any $\beta \in [0, 1]$, there is an oblivious $(\beta(1 - \beta), \beta)$ -approximation algorithm for RegularizedUSM with f a directed cut function and ℓ having arbitrary sign.*

Proof For each vertex v , select it with probability β if $(1 - \beta) \cdot \text{out}(v) + \ell(v) \geq 0$, and 0 otherwise. Then

$$\begin{aligned} \beta(1 - \beta)f(OPT) + \beta\ell(OPT) &\leq \sum_{v \in OPT} (\beta(1 - \beta) \cdot \text{out}(v) + \beta\ell(v)) \\ &\leq \sum_{v \in V} \beta \max((1 - \beta) \cdot \text{out}(v) + \ell(v), 0), \quad (\text{A11}) \end{aligned}$$

and the last expression lower bounds the expected value of the solution returned by the randomized algorithm since every vertex is *not* selected with probability at least $1 - \beta$. \square

A.3 Omitted Proofs

Proof of Lemma 5.3 (Remainder) We modify the non-formal proof of [14]. This non-formal proof uses some simplifications such as allowing direct oracle access to the multilinear extension F and giving the algorithm in the form of a continuous-time algorithm, but these simplifications may be removed using known techniques at the cost of introducing the $o(1)$ into the guarantee [14].

By [14, Lemma 4],

$$\frac{dF(\mathbf{y}(t))}{dt} \geq \begin{cases} F(\mathbf{y}(t) \vee \mathbf{1}_{OPT \setminus Z}) - F(\mathbf{y}(t)) & t \in [0, t_s) \\ F(\mathbf{y}(t) \vee \mathbf{1}_{OPT}) - F(\mathbf{y}(t)) & t \in [t_s, t_f) \end{cases}. \quad (\text{A12})$$

Also, by [14, Lemma 5], for every time $t \in [0, t_f)$ and set $A \subseteq \mathcal{N}$ it holds that:

$$F(\mathbf{y}(t) \vee \mathbf{1}_A) \geq \left(e^{-\max\{0, t-t_s\}} - e^{-t} \right) [f(A) - f(A \cup Z)] + e^{-t} f(A). \quad (\text{A13})$$

So then by [14, Corollary 1], plugging in $A = OPT \setminus Z$ and $A = OPT$ for $t \in [0, t_s)$ and $t \in [t_s, t_f)$, respectively, gives us

$$\begin{aligned} \frac{dF(\mathbf{y}(t))}{dt} &\geq \begin{cases} f(OPT \setminus Z) - (1 - e^{-t})f(OPT \cup Z) & t \in [0, t_s) \\ e^{t_s-t} f(OPT) - (e^{t_s-t} - e^{-t})f(OPT \cup Z) & t \in [t_s, t_f) \end{cases} - F(\mathbf{y}(t)) \\ &\triangleq G(t) - F(\mathbf{y}(t)). \quad (\text{A14}) \end{aligned}$$

By submodularity of f , we may replace $f(OPT \setminus Z)$ with $f(OPT) - f(OPT \cap Z)$ in $G(t)$. Then

$$\begin{aligned}
 F(\mathbf{y}(t_f)) &\geq e^{-t_f} \left[\int_0^{t_f} e^t G(t) dt + F(\mathbf{y}(0)) \right] \\
 &= e^{-t_f} \left(\int_0^{t_s} e^t [f(OPT) - f(OPT \cap Z) - (1 - e^{-t})f(OPT \cup Z)] dt \right. \\
 &\quad \left. + \int_{t_s}^{t_f} e^t [e^{t_s-t} f(OPT) - (e^{t_s-t} - e^{-t})f(OPT \cup Z)] dt \right). \tag{A15}
 \end{aligned}$$

After evaluating and rearranging this final expression, we can see that this matches Eq. (5.7). □

Proof of Proposition 5.4 Consider the following linear program:

$$\begin{aligned}
 \text{maximize}_{\mathbf{x} \in \mathcal{P}} \quad & \frac{1}{2} \hat{f}(\mathbf{x}) + L(\mathbf{x}) \triangleq \frac{1}{2} \max_{\mathbf{c}} (\sum_{ab \in E} w_{ab} \mathbf{c}_{ab}) + L(\mathbf{x}) \\
 \text{subject to} \quad & \mathbf{c}_{ab} \geq 0 \\
 & \mathbf{c}_{ab} \leq \mathbf{x}_a + \mathbf{x}_b \\
 & \mathbf{c}_{ab} \leq 2 - \mathbf{x}_a - \mathbf{x}_b
 \end{aligned} \tag{A16}$$

Here, \mathbf{c}_{ab} corresponds to whether the edge (a, b) was cut. Note that $\hat{f}(\mathbf{1}_S) = f(S)$ for all $S \subseteq \mathcal{N}$, meaning that \hat{f} is an extension of f (though not multilinear). Furthermore, since f is an *undirected* cut function,

$$\forall (\mathbf{x}_a, \mathbf{x}_b) \in [0, 1]^2, \mathbf{x}_a(1 - \mathbf{x}_b) + (1 - \mathbf{x}_a)\mathbf{x}_b \geq \frac{1}{2} \min(\mathbf{x}_a + \mathbf{x}_b, 2 - \mathbf{x}_a - \mathbf{x}_b), \tag{A17}$$

implying that $F(\mathbf{x}) \geq \frac{1}{2} \hat{f}(\mathbf{x})$ for all $\mathbf{x} \in [0, 1]^{\mathcal{N}}$. Equation (A17) can be verified by first replacing $(\mathbf{x}_a, \mathbf{x}_b)$ with $(1 - \mathbf{x}_a, 1 - \mathbf{x}_b)$ if $\mathbf{x}_a + \mathbf{x}_b > 1$, and then performing the following sequence of computations:

$$\begin{aligned}
 \mathbf{x}_a(1 - \mathbf{x}_b) + (1 - \mathbf{x}_a)\mathbf{x}_b &= \mathbf{x}_a + \mathbf{x}_b - 2\mathbf{x}_a\mathbf{x}_b \\
 &\geq \mathbf{x}_a + \mathbf{x}_b - \frac{(\mathbf{x}_a + \mathbf{x}_b)^2}{2} \\
 &= (\mathbf{x}_a + \mathbf{x}_b) \left(1 - \frac{\mathbf{x}_a + \mathbf{x}_b}{2} \right) \\
 &\geq \frac{\mathbf{x}_a + \mathbf{x}_b}{2}. \tag{A18}
 \end{aligned}$$

Let \mathbf{x}^* be a solution attaining the optimal value for Eq. (A16), which can be found using any LP solver (e.g. using the ellipsoid method). Then

$$F(\mathbf{x}^*) + L(\mathbf{x}^*) \geq \frac{1}{2} \hat{f}(\mathbf{x}^*) + L(\mathbf{x}^*)$$

$$\begin{aligned}
&= \max_{\mathbf{x} \in \mathcal{P}} \left[\frac{1}{2} \hat{f}(\mathbf{x}^*) + L(\mathbf{x}^*) \right] \\
&\geq \max_{S \in \mathcal{I}} \left[\frac{1}{2} f(S) + \ell(S) \right] \\
&= \frac{1}{2} f(OPT) + \ell(OPT). \tag{A19}
\end{aligned}$$

Thus, \mathbf{x}^* achieves the desired approximation factor. We can finish by using pipage rounding to round \mathbf{x}^* to an integral solution within \mathcal{I} that preserves the value of $f + \ell$ in expectation.

Proof of Proposition 5.5 Consider a linear program similar to the one in the proof of Proposition 5.4.

$$\begin{aligned}
&\text{maximize}_{\mathbf{x} \in [0,1]^{\mathcal{N}}} && \frac{1}{2} \hat{f}(\mathbf{x}) + L(\mathbf{x}) \triangleq \frac{1}{2} \max_{\mathbf{c}} \left(\sum_{ab \in E} w_{ab} \mathbf{c}_{ab} \right) + L(\mathbf{x}) \\
&\text{subject to} && \mathbf{c}_{ab} \geq 0 \\
&&& \mathbf{c}_{ab} \leq \mathbf{x}_a \\
&&& \mathbf{c}_{ab} \leq 1 - \mathbf{x}_b
\end{aligned} \tag{A20}$$

Unfortunately, it does *not* suffice to just find any \mathbf{x}^* that attains the optimum value and apply pipage rounding. The reason for this is that $F(\mathbf{x}) \not\geq \frac{1}{2} \hat{f}(\mathbf{x})$ in general. However, it can be verified that

1. $F(\mathbf{x}) \geq \frac{1}{2} \hat{f}(\mathbf{x})$ when \mathbf{x} is *half-integral*; that is, $\mathbf{x}_u \in \{0, 0.5, 1\}$ for all $u \in \mathcal{N}$. This inequality can easily be verified for the cut function of a single directed edge, and thus extends to sums of cut functions.
2. The (\mathbf{x}, \mathbf{c}) polytope defined by the constraints in Eq. (A20) is bounded, and all its vertices are half-integral. Note that this property would no longer hold if the constraint $\mathbf{x} \in \mathcal{P}$ was included, which is why Proposition 5.5 does not apply to RegularizedCSM.

Both of these properties were previously used by Halperin and Zwick's combinatorial 0.5-approximation to MAX-DICUT [28]. Thus, the remainder of our algorithm is identical to that in the proof of Proposition 5.4, except we additionally require that the point \mathbf{x}^* returned by the LP solver is a vertex of the polytope to guarantee that $F(\mathbf{x}) \geq \frac{1}{2} \hat{f}(\mathbf{x})$. \square

Proof of Lemma 7.2 (Omitted Details) We need to show that we can use an algorithm that outputs $\mathbf{y}(t_f)$ satisfying Eq. (7.2) to generate $\mathbf{y} \in t_f \cdot \mathcal{P}$ such that

$$\mathbb{E}[F(\mathbf{y}) + L(\mathbf{y})] \geq (t_f e^{-t_f} - o(1)) f(OPT) + (1 - e^{-t_f} - o(1)) \ell_+(OPT) + t_f \ell_-(OPT), \tag{A21}$$

where the RHS of Eq. (A21) is identical to that of Eq. (7.1). Once we have \mathbf{y} satisfying Eq. (A21), we can use *pipage rounding* to round \mathbf{y} to an integral solution $T \in \mathcal{I}$ if $t_f \leq 1$ or $T \subseteq \mathcal{N}$ otherwise [18]. Specifically, given $\mathbf{y} \in \mathcal{P}$, pipage rounding generates $T \in \mathcal{I}$ such that $\mathbb{E}[\mathbf{1}_T] = \mathbf{y}$ and $\mathbb{E}[F(\mathbf{1}_T) + L(\mathbf{1}_T)] \geq F(\mathbf{y}) + L(\mathbf{y})$.

We note that $\mathbf{y}(t_f)$ satisfying Eq. (7.2) with high probability does not necessarily satisfy Eq. (A21) if either:

1. $M = \omega(f(OPT) + \ell_+(OPT))$.
2. Equation (7.2) does not hold.

Algorithm 2 Modified Distorted Measured Continuous Greedy (t)

- 1: Order the elements of \mathcal{N} such that $f(u_1) \leq f(u_2) \leq \dots \leq f(u_n)$.
 - 2: **for** $i = 0$ to n **do**
 - 3: Let $\mathbf{y}_i \leftarrow$ the result of running [25, Algorithm 1] on $\mathcal{N}_i = \{u_1, u_2, \dots, u_i\}$.
 - 4: Let $T_i \leftarrow$ the result of applying pipage rounding to \mathbf{y}_i .
 - 5: **end for**
 - 6: **return** the set T_i maximizing $f(T_i) + \ell(T_i)$.
-

However, we claim that the output of Algorithm 2 *does* satisfy the conditions of Lemma 7.2. It suffices to show that the two issues mentioned above are now resolved.

1. Assume

$$\emptyset \neq OPT \in \operatorname{argmax}_{OPT \subseteq \mathcal{N}} [t_f e^{-t_f} f(OPT) + (1 - e^{-t_f}) \ell_+(OPT) + t_f \ell_-(OPT)]. \tag{A22}$$

Defining $j \triangleq \max\{i \mid u_i \in OPT\}$, it follows that

$$\begin{aligned} t_f e^{-t_f} f(OPT) + (1 - e^{-t_f}) \ell_+(OPT) &\geq t_f e^{-t_f} \max(f(u_j), f(\emptyset)) \\ &\geq t_f e^{-t_f} \frac{M_j}{j}, \end{aligned} \tag{A23}$$

where M_j is the value of M when restricted to \mathcal{N}_j . Thus, $M_j \leq n(f(OPT) + \ell_+(OPT))$. By choosing $\epsilon = o(\frac{1}{n})$ in [25, Algorithm 1], Eq. (7.2) implies that \mathbf{y}_j satisfies Eq. (A21) with high probability.

2. Regardless of whether Eq. (7.2) holds, the set T returned by Algorithm 2 always satisfies $f(T) + \ell(T) \geq f(T_0) + \ell(T_0) \geq 0$. Thus, if T satisfies Lemma 7.2 with high probability, it also satisfies Lemma 7.2 in expectation.

Proof of Lemma 7.5 As with the proof of Lemma 5.3, we only present an informal proof assuming direct oracle access to the multilinear extension F and giving the algorithm in the form of a continuous-time algorithm. The techniques mentioned in [14, 25] can be used to formalize this at the cost of introducing the $o(1)$ term. As before, it suffices to prove the conclusion for integral sets Z .

Let $G(\mathbf{y}(t)) \triangleq e^{t-t_f} F(\mathbf{y}(t)) + L(\mathbf{y}(t))$ be the value of the distorted objective at time t . Then

$$\begin{aligned} \frac{dG(\mathbf{y}(t))}{dt} &\geq e^{t-t_f} \cdot \begin{cases} f(OPT \setminus Z) - (1 - e^{-t})f(OPT \cup Z) & t \in [0, t_s) \\ e^{t_s-t} f(OPT) - (e^{t_s-t} - e^{-t})f(OPT \cup Z) & t \in [t_s, t_f) \end{cases} \\ &\quad + e^{-t} \cdot \begin{cases} \ell_+(OPT \setminus Z) & t \in [0, t_s) \\ \ell_+(OPT \setminus Z) + e^{t_s} \ell(OPT \cap Z) & t \in [t_s, t_f) \end{cases} \end{aligned}$$

$$+ \begin{cases} \ell_-(OPT \setminus Z) & t \in [0, t_s) \\ \ell_-(OPT) & t \in [t_s, t_f] \end{cases} \tag{A24}$$

Here, the first and third terms of the summation correspond directly to those of the original aided measured continuous greedy, while the second comes from observing that $y_u(t) \leq 1 - e^{-t}$ for $u \in OPT \setminus Z$ and $y_u(t) \leq 1 - e^{-\max(t-t_s, 0)}$ for $u \in OPT \cap Z$.

To lower bound $G(y(t_f))$, we can integrate Eq. (A24) from $t = 0$ to $t = t_f$. As expected, the dependence on f turns out to be the same as Lemma 5.3. \square

Proof of Proposition 7.8 Our goal is to show that when f is a directed cut function,

$$\max_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] \geq \max_{S \subseteq \mathcal{N}} [0.5f(S) + \ell(S)]. \tag{A25}$$

The idea is to first construct an auxiliary function $\hat{f}(\mathbf{x})$ satisfying the following properties (note that we do not capitalize \hat{f} since it is not a multilinear extension):

1. The function \hat{f} is symmetric; that is, $\hat{f}(\mathbf{x}) = \hat{f}(\bar{\mathbf{x}})$ for all $\mathbf{x} \in [0, 1]^{\mathcal{N}}$.
2. The function \hat{f} upper bounds f ; that is, $\hat{f}(\mathbf{1}_S) \geq f(S)$ for all $S \subseteq \mathcal{N}$.
3. There exists $\mathbf{x}^* \in \operatorname{argmax}_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [0.5\hat{f}(\mathbf{x}) + L(\mathbf{x})]$ such that $\mathbf{x}^* = \bar{\mathbf{x}}^*$ and $F(\mathbf{x}^*) \geq 0.5\hat{f}(\mathbf{x}^*)$.

Assuming that all these properties hold, we find:

$$\begin{aligned} \max_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [F(\bar{\mathbf{x}}) + L(\bar{\mathbf{x}})] &\geq \max_{\mathbf{x} \in [0, 1]^{\mathcal{N}}} [0.5\hat{f}(\mathbf{x}) + L(\mathbf{x})] && \text{(by properties 1 and 3)} \\ &\geq \max_{S \subseteq \mathcal{N}} [0.5\hat{f}(\mathbf{1}_S) + \ell(S)] \\ &\geq \max_{S \subseteq \mathcal{N}} [0.5f(S) + \ell(S)] && \text{(by property 2).} \end{aligned} \tag{A26}$$

Before defining \hat{f} , we examine the symmetrization operator $\bar{\mathbf{x}}$. Recall that symmetrization is defined with respect to a permutation group \mathcal{G} . Partition the ground set into K subsets $\mathcal{N} = \mathcal{N}_1 \cup \mathcal{N}_2 \dots \cup \mathcal{N}_K$, where \cup denotes the disjoint union of two sets, and define the function $g : \mathcal{N} \rightarrow \{1, 2, \dots, K\}$ to be the mapping from every element of the ground set to the subset that contains it. This mapping satisfies the property that $g(u_i) = g(u_j)$ if and only if there exists a permutation $\sigma \in \mathcal{G}$ such that $\sigma(u_i) = u_j$. Observe that

$$\bar{x}_{u_i} = \operatorname{avg}_{g(u_i)}(\mathbf{x}) \triangleq \frac{\sum_{u \in \mathcal{N}_{g(u_i)}} x_u}{|\mathcal{N}_{g(u_i)}|}; \tag{A27}$$

that is, the value at u_i in $\bar{\mathbf{x}}$ is just the average of the values in \mathbf{x} of all u in the same subset as u_i .

Next, we define \hat{f} in terms of $\operatorname{avg}_1(\mathbf{x}), \operatorname{avg}_2(\mathbf{x}), \dots, \operatorname{avg}_K(\mathbf{x})$, which guarantees that property 1 is satisfied. For all $1 \leq i, j \leq K$, define $w_{ij} \geq 0$ as the sum of the

weights of the edges directed from \mathcal{N}_i to \mathcal{N}_j (where i can equal j). Then

$$\hat{f}(\mathbf{x}) \triangleq \sum_{i=1}^K \sum_{j=1}^K w_{ij} \min(\text{avg}_i(\mathbf{x}), 1 - \text{avg}_j(\mathbf{x})). \quad (\text{A28})$$

It remains to show that properties 2 and 3 are satisfied.

Property 2:

Since every subset \mathcal{N}_i is symmetric, the proportion of edges from \mathcal{N}_i to \mathcal{N}_j that are cut by S is bounded above by the proportion of elements in \mathcal{N}_i contained within S (which is precisely $\text{avg}_i(\mathbf{x})$) as well as the proportion of elements in \mathcal{N}_j not contained within S (which is precisely $1 - \text{avg}_j(\mathbf{x})$). This implies that $\hat{f}(\mathbf{1}_S) \geq f(S)$ for all S .

Property 3:

Similar to the proof of Proposition 5.5, it can be shown that there exists a half-integral tuple $(\text{avg}_1(\mathbf{x}), \text{avg}_2(\mathbf{x}), \dots, \text{avg}_K(\mathbf{x}))$ at which $0.5\hat{f}(\mathbf{x}) + L(\mathbf{x})$ is maximized. This tuple corresponds to a half-integral and symmetric \mathbf{x}^* . As in Proposition 5.5, it is easy to check that $F(\mathbf{x}^*) \geq 0.5\hat{f}(\mathbf{x}^*)$, completing the proof. \square

References

1. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146 (2003)
2. Krause, A., Singh, A., Guestrin, C.: Near-optimal sensor placements in gaussian processes: theory, efficient algorithms and empirical studies. *J. Mach. Learn. Res.* **9**(2) (2008)
3. Krause, A., Guestrin, C.: Submodularity and its applications in optimized information gathering. *ACM Trans. Intell. Syst. Technol. (TIST)* **2**(4), 1–20 (2011)
4. Lin, H., Bilmes, J.: A class of submodular functions for document summarization. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 510–520 (2011)
5. Wei, K., Liu, Y., Kirchhoff, K., Bilmes, J.: Using document summarization techniques for speech data subset selection. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 721–726 (2013)
6. Gygli, M., Grabner, H., Van Gool, L.: Video summarization by learning submodular mixtures of objectives. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3090–3098 (2015)
7. Jegelka, S., Bilmes, J.: Submodularity beyond submodular energies: coupling edges in graph cuts. In: CVPR 2011, pp. 1897–1904. IEEE (2011)
8. Shen, J., Liang, Z., Liu, J., Sun, H., Shao, L., Tao, D.: Multiobject tracking by submodular optimization. *IEEE Trans. Cybern.* **49**(6), 1990–2001 (2018)
9. Krause, A., Golovin, D.: Submodular function maximization. *Tractability* **3**, 71–104 (2014)
10. Buchbinder, N., Feldman, M., Naor, J., Schwartz, R.: A tight linear time (1/2)-approximation for unconstrained submodular maximization. In: 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science, pp. 649–658 (2012). <https://doi.org/10.1109/FOCS.2012.73>
11. Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.: An analysis of approximations for maximizing submodular set functions—I. *Math. Program.* **14**(1), 265–294 (1978)
12. Calinescu, G., Chekuri, C., Pal, M., Vondrák, J.: Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.* **40**(6), 1740–1766 (2011)
13. Feldman, M., Naor, J., Schwartz, R.: A unified continuous greedy algorithm for submodular maximization. In: 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science, pp. 570–579. IEEE (2011)

14. Buchbinder, N., Feldman, M.: Constrained submodular maximization via a nonsymmetric technique. *Math. Oper. Res.* **44**(3), 988–1005 (2019)
15. Feige, U., Mirokni, V.S., Vondrák, J.: Maximizing non-monotone submodular functions. *SIAM J. Comput.* **40**(4), 1133–1153 (2011)
16. Nemhauser, G.L., Wolsey, L.A.: Best algorithms for approximating the maximum of a submodular set function. *Math. Oper. Res.* **3**(3), 177–188 (1978)
17. Oveis Gharan, S., Vondrák, J.: Submodular maximization by simulated annealing. In: *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1098–1116. SIAM (2011)
18. Vondrák, J.: Symmetry and approximability of submodular maximization problems. *SIAM J. Comput.* **42**(1), 265–304 (2013)
19. Sviridenko, M., Vondrák, J., Ward, J.: Optimal approximation for submodular and supermodular optimization with bounded curvature. *Math. Oper. Res.* **42**(4), 1197–1218 (2017)
20. Feldman, M.: Guess free maximization of submodular and linear sums. *Algorithmica* **83**(3), 853–878 (2021)
21. Harshaw, C., Feldman, M., Ward, J., Karbasi, A.: Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications. In: *International Conference on Machine Learning*, pp. 2634–2643. PMLR (2019)
22. Kazemi, E., Minaee, S., Feldman, M., Karbasi, A.: Regularized submodular maximization at scale. In: *International Conference on Machine Learning*, pp. 5356–5366. PMLR (2021)
23. Nikolakaki, S.M., Ene, A., Terzi, E.: An efficient framework for balancing submodularity and cost. In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1256–1266 (2021)
24. Bodek, K., Feldman, M.: Maximizing sums of non-monotone submodular and linear functions: understanding the unconstrained case. [arXiv:2204.03412](https://arxiv.org/abs/2204.03412) (2022). A conference version appeared in *ESA 2022*
25. Lu, C., Yang, W., Gao, S.: Regularized nonmonotone submodular maximization. *Optimization* 1–27 (2023)
26. Diamond, S., Boyd, S.: CVXPY: a python-embedded modeling language for convex optimization. *J. Mach. Learn. Res.* **17**(83), 1–5 (2016)
27. Trevisan, L.: Parallel approximation algorithms by positive linear programming. *Algorithmica* **21**(1), 72–88 (1998)
28. Halperin, E., Zwick, U.: Combinatorial approximation algorithms. In: *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, vol. 103, p. 1. SIAM (2001)
29. Buchbinder, N., Feldman, M., Schwartz, R.: Online submodular maximization with preemption. In: *Proceedings of the Twenty-sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1202–1216. SIAM (2014)
30. Buchbinder, N., Feldman, M.: Submodular functions maximization problems. (2018)
31. Sun, X., Xu, D., Zhou, Y., Wu, C.: Maximizing modular plus non-monotone submodular functions. [arXiv:2203.07711](https://arxiv.org/abs/2203.07711) (2022)
32. Bar-Noy, A., Lampis, M.: Online maximum directed cut. *J. Comb. Optim.* **24**(1), 52–64 (2012)
33. Huang, N., Borodin, A.: Bounds on double-sided myopic algorithms for unconstrained non-monotone submodular maximization. In: *International Symposium on Algorithms and Computation*, pp. 528–539. Springer (2014)
34. Feige, U., Jozeph, S.: Oblivious algorithms for the maximum directed cut problem. *Algorithmica* **71**(2), 409–428 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.