On Submodular Prophet Inequalities and Correlation Gap

Chandra Chekuri and Vasilis Livanos

University of Illinois at Urbana-Champaign *

Abstract. Prophet inequalities and secretary problems have been extensively studied in recent years due to their elegance, connections to online algorithms, stochastic optimization, and mechanism design problems in game theoretic settings. Rubinstein and Singla [31] developed a notion of *combinatorial* prophet inequalities in order to generalize the standard prophet inequality setting to combinatorial valuation functions such as submodular and subadditive functions. For non-negative submodular functions they demonstrated a constant factor prophet inequality for matroid constraints. Along the way they showed a variant of the correlation gap for non-negative submodular functions.

In this paper we revisit their notion of correlation gap as well as the standard notion of correlation gap and prove much tighter and cleaner bounds. Via these bounds and other insights we obtain substantially improved constant factor combinatorial prophet inequalities for both monotone and non-monotone submodular functions over any constraint that admits an Online Contention Resolution Scheme. In addition to improved bounds we describe efficient polynomial-time algorithms that achieve these bounds.

Keywords: combinatorial prophet inequality · submodularity · OCRS.

1 Introduction

Prophet inequalities arose from stochastic optimization and stopping theory in the '70s. In the basic setting there are n independent real-valued random variables $X_1, X_2, \ldots, X_n$ with prescribed distributions $\mathcal{D}_1, \ldots, \mathcal{D}_n$; they correspond to values of some items. An online algorithm (or agent) knows the distributions of the random variables a priori but sees their realizations in an *adversarial* order, and has to choose exactly one of them. The algorithm has to make an irrevocable decision on whether to accept an item or not when it arrives. In the single item setting the first accepted item stops the process. The algorithm's performance is measured with respect to the value of a prophet who gets to see all the realizations and then picks the variable with the largest value. The expected value of the prophet is $V^* = \mathbb{E}[\max_i X_i]$. An online algorithm is α-competitive if

* Dept. of Computer Science, University of Illinois, Urbana, IL 61801. {chekuri,livanos3}@illinois.edu. Work on this paper partially supported by NSF grant CCF-1910149.

its expected value is at least αV^*. Krengel and Sucheston [27] showed that 1/2 is the optimal competitive ratio for the single item setting.

There has been substantial recent interest in prophet inequalities in theoretical computer science, due to their strong connections to online mechanism design [22,9] (see [23,10] for surveys on Bayesian mechanism design). Subsequent work explored prophet inequality problems in more general settings. Of particular interest to us is the setting where multiple variables/items from $X_1, X_2, \ldots, X_n$ can be chosen such that the chosen items are feasible in some combinatorial constraint family. Two important examples are choosing k items for some integer $k \geq 1$ [4] and a further generalization where the items chosen are independent in a matroid[1] [26]. These generalizations had several motivations including algorithmic game theory, combinatorial optimization, stochastic optimization, and online algorithms. We refer the reader to surveys on prophet inequalities by Lucier [28] and Correa et al. [12] for several pointers to the extensive literature on these problems and related topics.

Combinatorial Prophet Inequalities: Prophet inequalities were studied with modular/additive objective functions, meaning that the total value of a subset S of variables is simply the sum of their individual values. However, more general combinatorial objective functions have several applications. For these objectives, the value of a subset of items from $\mathcal{N} = [n]$ is specified by a set function $f : 2^{\mathcal{N}} \to \mathbb{R}$. Prominent examples are submodular[2] and subadditive[3] set functions, and their special cases. Rubinstein and Singla introduced a model of combinatorial prophet inequalities [31] which is the main object of study in this paper. We restrict our attention to submodular objectives which form a rich class and, following [31], we refer to this as the Submodular Prophet Inequality (SPI) problem.

The model defined by Rubinstein and Singla is the following generalization of the standard prophet inequality problem. The input consists of n independent random variables $X_1, X_2, \ldots, X_n$. Unlike the standard prophet inequality where X_i is a real-valued random variable, in the combinatorial setting, each X_i is a discrete-valued random variable over a finite set $\mathcal{U}_i$. Thus $\mathcal{D}_i$ is a discrete probability distribution over $\mathcal{U}_i$. For technical reasons one assumes that $\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_n$ are mutually disjoint. Let $\mathcal{U} = \bigcup_i \mathcal{U}_i$. There is a non-negative submodular function $f : 2^{\mathcal{U}} \to \mathbb{R}_+$ defined over the ground set $\mathcal{U}$. As in the standard prophet inequality setting, the variables arrive in an adversarial order. After seeing the realization of a variable in the order, the algorithm has to make an irrevocable

[1] A constraint system $\mathcal{C}$ is downward-closed when for every feasible set A, all $B \subseteq A$ are also feasible. A matroid is a non-empty downward-closed constraint system where if $A, B \subseteq \mathcal{N}$ are both feasible and $|A| < |B|$, there exists an element $e \in B \setminus A$ such that $A \cup \{e\}$ is also feasible. The feasible sets of a matroid are called *independent sets*.

[2] A real-valued set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is submodular if and only if $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$ for all $A, B \subseteq \mathcal{N}$.

[3] A real-valued set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is subadditive if $f(A \cup B) \leq f(A) + f(B)$ for all $A, B \subseteq \mathcal{N}$. A non-negative submodular function is subadditive.

decision to accept it or not. Its goal is to maximize the value, $f(S)$, of the selected set $S \subseteq \mathcal{U}$. The input also specifies a downward-closed family of sets $\mathcal{I} \subseteq 2^{[n]}$, and the set S of chosen variables must belongs to $\mathcal{I}$. The prophet is allowed to optimize offline after seeing all the realizations and obtains a value equal to $\mathbb{E}[\max_{S\in\mathcal{I}} f(\bigcup_{i\in S}\{X_i\})]$. We say that an algorithm achieves a competitive ratio of α if the expected value of its choice is at least $\alpha \cdot \mathbb{E}[\max_{S\in\mathcal{I}} f(\bigcup_{i\in S}\{X_i\})]$. Observe that the SPI model generalizes the standard prophet inequality with additive functions and arbitrary downward-closed constraints.

We motivate a scenario for the SPI problem. Suppose we have an online advertising situation where one sees on each day (or time slot) a customer of some type drawn from a known distribution. It is natural to assume that customer types are discrete (or can be approximated by a discrete distribution). The agency has to irrevocably decide whether to show an ad to the customer when they arrive, but has budget constraints which dictate that at most k ads can be shown overall. This corresponds to picking k out of the n random variables. As for the value of serving the ads, a decreasing marginal utility for customers of the same type is a natural assumption and this is captured by submodular functions. The model proposed by [31] allows arbitrary submodular functions over $\mathcal{U}$, which allows for substantial generality.

Rubinstein and Singla presented an $O(1)$-competitive algorithm for SPI under a matroid constraint. The constant they obtained is very large (in the thousands, although they did not try to optimize it) and they did not consider or emphasize the computational aspects of the online algorithm. We note that prophet inequalities in the standard setting of modular/additive objectives are fairly small. For instance, the well-known result of Kleinberg and Weinberg [26] showed a bound of $1/2$ even for arbitrary matroid constraints, and it is also known that the bound for a cardinality constraint with k items is $(1 - O(1/\sqrt{k}))$ (hence it tends to 1 as $k \to \infty$) [4]. Moreover, [31] did not explicitly consider the case of monotone[4] submodular functions, and did not generalize the constraint family beyond a single matroid.

The approach of Rubinstein and Singla uses the standard relaxation and rounding paradigm and consists of two steps. First, they observe that an upper bound on the prophet's value for a given instance of SPI can be obtained by solving a mathematical programming relaxation with a continuous extension f^+ of f called the *concave closure*[5] However, solving and rounding fractional solutions with respect to f^+ is not feasible for several reasons. Instead, they obtain a fractional solution x to a different continuous extension of f, known as the *multilinear relaxation*[6] F, and then round it online via a rounding scheme known as an *Online Contention Resolution Scheme (OCRS)*, originally introduced by Feldman, Svensson and Zenklusen [20]. Informally, an OCRS for a constraint family is an online rounding scheme for a given polyhedral relaxation of the constraints that always returns a feasible set S, while providing some guarantees

[4] A real-valued set function f is monotone if $f(A) \leq f(B)$ whenever $A \subseteq B$.
[5] For the formal definition of the concave closure see Definition 2.
[6] For the formal definition of the multilinear relaxation see Definition 1.

with regards to the probability that each element is in S. To obtain a competitive ratio they relate $F(x)$ to $f^+(x)$. The ratio $\inf_{\boldsymbol{x}\in[0,1]^n} \frac{F(\boldsymbol{x})}{f^+(\boldsymbol{x})}$ is known as the *correlation gap* of f. An important result in submodular optimization is that the correlation gap for any monotone submodular function is at least $(1-1/e)$ [3,8,33]. However, it is known that, for general non-negative submodular functions (which can be non-monotone) the correlation gap can be arbitrarily small. Rubinstein and Singla circumvented this issue by using a variant of the standard correlation gap, showing that $\inf_{\boldsymbol{x}\in[0,1]^n} \frac{F_{\text{MAX}}(\boldsymbol{x})}{f^+(\boldsymbol{x})} \geq 1/200$, where F_{MAX} is the multilinear relaxation of $f_{\text{MAX}}(S) = \max_{T\subseteq S} f(T)$.

Our goal in this paper is to obtain improved bounds for SPI via a clean framework that applies to a wide variety of constraints. This question was explicitly raised by Lucier in his survey on prophet inequalities [28]. Another motivation, related to the goal of obtaining improved bounds, is to improve the best-known bounds for the correlation gap, which plays a crucial role in submodular optimization [11,34].

1.1 Our contributions

In this paper we make two high-level contributions:

1. We consider the correlation gap for non-negative submodular functions. For both the standard definition and the variant of [31], we obtain substantially improved bounds.
2. We address and provide improvements to three aspects of the SPI problem: (i) improved constants for the prophet inequalities for monotone and non-monotone functions, (ii) a clean black-box reduction to greedy Online Contention Resolution Schemes that allows one to obtain combinatorial prophet inequalities for various other constraints beyond a single matroid constraint and (iii) computational aspects of the prophet inequality that were not explicitly addressed in [31]. In essence, we answer the open question in [28] in the affirmative.

We now give a formal statement of our results. We refer the reader to Section 2.1 for some basic definitions and background on submodular functions, continuous extensions, the correlation gap and contention resolution schemes.

Correlation gap: For a non-negative submodular function, for any given $p \in [0,1]$, there is a simple instance with $n = 2$ where $F(\boldsymbol{x}) \leq (1-p)f^+(\boldsymbol{x})$, and this implies that, as p tends to 1, the correlation gap tends to 0. One way to overcome this is to restrict attention to settings where p is bounded away from 1. Nevertheless, there has been little work on precisely quantifying the correlation gap as a function of this parameter. Our first theorem addresses this.

Theorem 1. *Let $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be a non-negative submodular function and let $\boldsymbol{x} \in [0,1]^n$, where $n = |\mathcal{N}|$. Let $p = \max_i x_i$. Then $F(\boldsymbol{x}) \geq (1-p)(1-1/e)f^+(\boldsymbol{x})$. Given any $p \in [0,1]$ there are instances such that $F(\boldsymbol{x}) \leq (1-e^{-(1-p)})f^+(\boldsymbol{x})$.*

The upper bound of $(1-p)(1-1/e)$ is optimal when p is close to 0 or when p is close to 1. The lower bound on the gap that we show, $1-e^{-(1-p)}$, agrees nicely with the extremes, but we do not know whether it is the right bound for all ranges of p and leave it as an interesting open problem.

As stated earlier, Rubinstein and Singla [31] instead use the correlation gap of $f_{\max}(S) = \max_{T\subseteq S} f(T)$. $f_{\max}$ is monotone, but in general is not submodular, even when f is. It is shown in [31] that for any non-negative submodular function f, $\inf_{\boldsymbol{x}\in[0,1]^n} \frac{F_{\max}(\boldsymbol{x})}{f^+(\boldsymbol{x})} \geq 1/200$, where $F_{\max}$ is the multilinear extension of $f_{\max}$. For this variant of the correlation gap, we observe that known results on the Measured Continuous Greedy (MCG) algorithm [19,2] show that $F_{\max}(\boldsymbol{x}) \geq \frac{1}{e} f^+(\boldsymbol{x})$.

We strengthen this observation by considering the parameter p.

Theorem 2. *Let $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be a non-negative submodular function and let $\boldsymbol{x} \in [0,1]^n$, where $n = |\mathcal{N}|$. Let $p = \max_i x_i$. There exists a point $\boldsymbol{y} \in [0,1]^n$, where $\boldsymbol{y} \leq \boldsymbol{x}$ (coordinate wise), such that $F(\boldsymbol{y}) \geq \max\{\frac{1}{e}, (1-p-\frac{1}{e}(1+\ln(1-p)))\} f^+(\boldsymbol{x})$.*

We obtain the preceding theorem as a corollary of the following.

Theorem 3. *Let $p \in [0,1)$, f be a non-negative submodular function with multilinear extension F and $\mathcal{P}$ be a downward-closed solvable polytope[7] on $\mathcal{N}$, such that $\mathcal{P} \subseteq p \cdot [0,1]^{\mathcal{N}}$ (that is, if $\boldsymbol{z} \in \mathcal{P}$ then $z_i \leq p$ for all $i \in \mathcal{N}$). Then, the output of the Measured Continuous Greedy (MCG) algorithm on F and $\mathcal{P}$ at time $b \in [0,1]$ is a vector $\boldsymbol{x}(b) \in b \cdot \mathcal{P}$ such that*

$$F(\boldsymbol{x}(b)) \geq \begin{cases} b \cdot e^{-b} \cdot \max_{\boldsymbol{z}\in\mathcal{P}} f^+(\boldsymbol{z}), & 0 \leq b \leq \ln\left(\frac{1}{1-p}\right) \\ \left(1-p-e^{-b}\left(1+\ln\left(1-p\right)\right)\right) \cdot \max_{\boldsymbol{z}\in\mathcal{P}} f^+(\boldsymbol{z}), & \ln\left(\frac{1}{1-p}\right) \leq b \leq 1. \end{cases}$$

Theorem 2 on its own already improves the constants obtained by Rubinstein and Singla in [31] for the SPI problem. Theorems 1 and 3 are even more useful when p is small and we later that this can indeed be achieved in some cases, such as the SPI problem, via a reduction to an instance with many "copies" per element.

Submodular Prophet Inequality: For SPI we follow the high level framework of [31] via the correlation gap and greedy[8] Online Contention Resolution Schemes (OCRSs) [20]. We make two main contributions that lead to significantly improved constants as well as clarity on the parameters that affect them. Specifically, we utilize a simple reduction to a new instance of SPI with small p, allowing us to use Theorems 1 and 3 to improve upon the constant obtained by [31]. Our second contribution concerns the generalization of the approach of [31] to other feasibility constraints beyond matroids. The approach of Rubinstein and Singla

[7] Informally, a polytope $\mathcal{P}$ is solvable if one can efficiently do linear optimization over it. A formal definition is given in Section 2.

[8] The OCRSs that we will need in this paper are greedy OCRSs – for the formal definition see 6. We will sometimes abuse notation and omit the word greedy.

relies on the fact that, when the given constraint over $\mathcal{N}$ is a matroid, the resulting constraint over $\mathcal{U}$ is still a matroid, and thus the entire constraint admits a greedy OCRS [20]. However, this does not generalize to other feasibility constraints. To handle this, we design a modified algorithm that uses a given greedy OCRS for a constraint family over $\mathcal{N}$ in a black-box fashion and thus works for any constraint that admits a greedy OCRS.

The competitive ratio that we achieve for a particular constraint family is dictated by the OCRS available for that family. Because of this, any improvements to the guarantees of the known OCRSs, as well as any new OCRSs for other constraint families, immediately generalize and imply improvements to our results. The approximation quality of the OCRS is governed by two parameters $b, c \in [0,1]$ via the notion of (b,c)-selectability. For matroids there is $(b, 1-b)$-selectable OCRS for any $b \in [0,1]$, while there is a (b, e^{-2b})-OCRS for matching constraints and a $\left(1 - \frac{t}{\sqrt{k}}, 1 - \exp(\frac{-t^2}{4})\right)$-OCRS for the special case of a uniform matroid of rank k; see [20][9].

Theorem 4 (Informal). *For the SPI problem with a monotone submodular function f over a constraint family with a (b,c)-selectable OCRS, there is a $c \cdot (e^{-b} - \varepsilon)(1 - e^{-b})$-competitive algorithm for any fixed $\varepsilon > 0$. For non-negative submodular functions there is a $\frac{c}{4} \cdot (e^{-b} - \varepsilon)(1 - e^{-b})$-competitive algorithm for any fixed $\varepsilon > 0$. These competitive ratios can be achieved by an efficient randomized polynomial time algorithm, assuming value oracle access to f and efficiency of the corresponding OCRS.*

Our results hold in the setting of an almighty adversary who can adaptively decide the ordering of the variables based on the realization of all variables and the choices of the algorithm at each step. We note that the competitive ratios we obtain are explicit and relatively small. We summarize our concrete competitive ratios for several constraints of interest below. OCRSs for constraints can be composed nicely (similar to CRSs) and thus our black-box reduction is very useful.

Feasibility constraint	Competitive Ratio	
	Monotone Submodular	General Submodular
Uniform matroid of rank $k \to \infty$	1/4.3	1/17.2
Matroid	1/7.4	1/30
Matching	1/9.5	1/38
Knapsack	1/17.5	1/70
Intersection of k matroids	$\Omega(1/k)$	$\Omega(1/k)$

Table 1: A summary of our results for several feasibility constraints.

[9] For the uniform matroid of rank k the OCRS we claim is not in [20] but it is easy to derive and was explicitly done in an unpublished senior thesis [25].

1.2 Other related work

We have referred to recent surveys on prophet inequalities and and related models [28,12]. An older survey on prophet inequalities from a stopping theory point of view is due to Hill and Kertz [24]. Secretary problems are closely related to prophet inequalities. In the classical version, an online algorithm sees n adversarially chosen values in a *random order* and has to pick one item irrevocably, aiming to maximize the probability of picking the highest value. A classical result of Dynkin [16] shows an optimal competitive ratio of $1/e$. A survey on the secretary problem and variants is due to Dinitz [13]. The work here is connected to submodular optimization, stochastic optimization, online algorithms, and mechanism design which have extensive literature. It is infeasible to describe all the related work; Singla's thesis [32] touches upon several of these themes and has several pointers. Contention Resolution Schemes (CRSs) have found many applications since their introduction [11]; in fact Bayesian mechanism design, posted price mechanisms [9] and subsequent work by Yan [34], connecting mechanism design with the correlation gap, played an important role in [11]. Online CRSs were developed [20] with applications to Bayesian mechanism design as one of the main motivations and yield prophet inequalities in the modular case.

Submodular functions and constraints such as cardinality, matroids and others provide generality and computational tractability. It is possible to consider more general objective functions such as subadditive and monotone XOS functions, as well as more complex and general feasibility constraints. In such settings one can ignore computational considerations and focus on the online competitive ratio or assume access to a demand oracle (even though a demand oracle may be NP-Hard in general). We refer to [30,31] for some recent work and pointers. Such functions have also been considered under the related model of "combinatorial auctions" [14,15,5], in which a seller wants to sell distinct items to buyers that have combinatorial valuation functions for the items. The seller wishes to maximize either the social welfare or the revenue. In this model, Dütting, Feldman, Kesselheim and Lucier [14] obtained a 2-prophet inequality for submodular functions, while Dütting, Kesselheim and Lucier [15] obtained a $O(\log \log m)$ prophet inequality for subadditive functions. For the latter, the authors also show that achieving a constant factor prophet inequality for subadditive valuation functions is impossible via their techniques and requires a different approach.

Finally, in the setting where the arrival order of the random variables is chosen *uniformly at random*, one can obtain improved bounds via the use of Random Order CRSs, introduced in [2]. We discuss a variant of the SPI problem for this setting in brief in Section 5 and we refer the reader to a survey by Gupta and Singla for more information on random order models [21].

Organization: Section 2 introduces our notation and provides background on submodular functions, constraint systems and contention resolution schemes. Section 3 describes the relaxation of the prophet's objective. Section 4 describes the algorithm and analysis for SPI. We describe some open questions in Section 5.

The proof of Theorem 1 has been moved to Appendix A since we do not use it directly for our improvements in the bounds of the SPI problem.

2 Preliminaries

2.1 Background and Definitions

Let $\mathcal{N}$ be a finite ground set. A constraint family over $\mathcal{N}$ is simply a subset $\mathcal{I} \subseteq 2^{\mathcal{N}}$; a set $S \in \mathcal{I}$ is called feasible, while a set $S \notin \mathcal{I}$ is called infeasible. We are interested only in downward-closed families of constraints; $\mathcal{I}$ is downward-closed if and only if $A \in \mathcal{I}$ implies that any set $B \subset A$ is also in $\mathcal{I}$. Classical examples of downward-closed families include those induced by a matroid on $\mathcal{N}$ or intersections of several matroids on $\mathcal{N}$, independent sets of graphs, matchings in graphs and hypergraphs, boolean vectors that satisfy packing constraints of the form $A\boldsymbol{x} \leq b$ for non-negative A, b, among many others. We will use the terminology $(\mathcal{N}, \mathcal{I})$ to indicate a constraint family.

The maximum weight independent set problem over a constraint system $(\mathcal{N}, \mathcal{I})$ is the following: given $w : \mathcal{N} \to \mathbb{R}$ solve $\max_{S \in \mathcal{I}} w(S)$ where $w(S) = \sum_{e \in \mathcal{N}} w(e)$. Since many of these problems are NP-Hard, a common technique is to use polyhedral (or more generally convex) relaxations. We say $\mathcal{P} \subseteq [0,1]^{\mathcal{N}}$ is a *polyhedral relaxation* of $(\mathcal{N}, \mathcal{I})$ if $\mathcal{P}$ is a polyhedron and $\mathbb{1}_S \in \mathcal{P}$ for all $S \in \mathcal{I}$ (here $\mathbb{1}_S$ is the characteristic vector of S). We say that $\mathcal{P}$ is *solvable* if one can efficiently do linear optimization over $\mathcal{P}$, that is, given $w : \mathcal{N} \to \mathbb{R}$, there is a polynomial time algorithm that computes $\max_{\boldsymbol{x} \in \mathcal{P}} \sum_i w_i x_i$.

A real-valued set function $f : \{0,1\}^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ is called *submodular* if, for all $A, B \subseteq \mathcal{N}$, it satisfies $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$. f is *monotone* if $f(A) \leq f(B)$ for all $A \subseteq B$. In the rest of this paper we work with non-negative normalized functions that satisfy $f(\emptyset) = 0$ and $f(A) \geq 0$ for all $A \subseteq \mathcal{N}$. We often equate $\mathcal{N}$ with $[n] = \{1, 2, \ldots, n\}$. We use the terminology $S + i$ and $S - i$ as shorthands for $S \cup \{i\}$ and $S \setminus \{i\}$ respectively. The following continuous extensions of submodular functions to $[0,1]^{\mathcal{N}}$ play an important role in our discussion.

Definition 1 (Multilinear Extension). *Let $f : \{0,1\}^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$. For any $\boldsymbol{x} \in [0,1]^n$, let $S \sim \boldsymbol{x}$ denote a random set S that contains each element $i \in \mathcal{N}$ independently w.p. x_i. The* multilinear extension *of f is defined as*

$$F(\boldsymbol{x}) \coloneqq \mathop{\mathbb{E}}_{S \sim \boldsymbol{x}}[f(S)] = \sum_{S \subseteq N} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i).$$

It should be noted that via the multilinear relaxation, the polyhedral approach to approximation has been extended successfully to submodular function maximization [8,11,6].

Definition 2 (Concave Closure). *Let $f : \{0,1\}^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$. Moreover, let $\mathbb{1}_S$ denote the characteristic vector of a set $S \subseteq \mathcal{N}$ of length $n = |\mathcal{N}|$. For any*

$\boldsymbol{x} \in [0,1]^n$, *the* concave closure *of* f *is defined as*

$$f^+(\boldsymbol{x}) := \max_{\boldsymbol{a} \in [0,1]^{2^{\mathcal{N}}}} \left\{ \sum_{S \subseteq \mathcal{N}} a_S f(S) \;\middle|\; \sum_{S \subseteq \mathcal{N}} a_S = 1, \sum_{S \subseteq \mathcal{N}} a_S \mathbb{1}_S = \boldsymbol{x} \right\}.$$

$f^+(\boldsymbol{x})$ can be interpreted as the maximum expected value of $f(R)$ where R is generated by a distribution whose marginal values are given by $\boldsymbol{x}$. Since $F(\boldsymbol{x})$ corresponds to the product distribution defined by $\boldsymbol{x}$, which is a specific distribution, it follows that $F(\boldsymbol{x}) \leq f^+(\boldsymbol{x})$ for all $\boldsymbol{x}$. The *correlation gap*, introduced in the work of Agrawal, Ding, Saberi and Ye [3], provides an inequality in the opposite direction.

Definition 3 (Correlation Gap). *Let* $f : \{0,1\}^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ *be a set function and* F, f^+ *denote its multilinear relaxation and concave closure respectively. The correlation gap of* f *is defined as*

$$\inf_{\boldsymbol{x} \in [0,1]^{|\mathcal{N}|}} \frac{F(\boldsymbol{x})}{f^+(\boldsymbol{x})}.$$

It is easy to see that the correlation gap of modular/additive functions is 1. An important result in submodular optimization is that the correlation gap is at most $1 - 1/e$ for any monotone submodular function [3,8,33]. However, it is known that the correlation gap for general non-negative submodular functions (which can be non-monotone) can be arbitrarily small.

Contention Resolution Schemes: These are rounding schemes introduced in [11] for submodular function maximization. For the remaining definitions of this section, $R(\boldsymbol{x})$ denotes a random subset of $\mathcal{N}$ where each $i \in \mathcal{N}$ appears independently with probability x_i.

Definition 4 (Contention Resolution Scheme). *Let* $b, c \in [0,1]$. *A* (b,c)*-balanced* Contention Resolution Scheme π *for* $\mathcal{P}_{\mathcal{I}}$ *is a procedure that for every* $\boldsymbol{x} \in b \cdot \mathcal{P}_{\mathcal{I}}$ *and* $A \subseteq \mathcal{N}$, *returns a random set* $\pi_{\boldsymbol{x}}(A) \subseteq A \cap \mathit{support}(\boldsymbol{x})$ *and satisfies the following properties:*

1. $\pi_{\boldsymbol{x}}(A) \in \mathcal{I}$ *with probability* $1, \quad \forall A \subseteq \mathcal{N}, \boldsymbol{x} \in b \cdot \mathcal{P}_{\mathcal{I}}$, *and*
2. *for all* $i \in \mathit{support}(\boldsymbol{x})$, $\Pr[i \in \pi_{\boldsymbol{x}}(R(\boldsymbol{x})) \mid i \in R(\boldsymbol{x})] \geq c, \quad \forall \boldsymbol{x} \in b \cdot \mathcal{P}_{\mathcal{I}}$.

The scheme is said to be monotone *if* $\Pr[i \in \pi_{\boldsymbol{x}}(A_1)] \geq \Pr[i \in \pi_{\boldsymbol{x}}(A_2)]$ *whenever* $i \in A_1 \subseteq A_2$.

CRSs are offline rounding schemes. *Online Contention Resolution Schemes (OCRS)* were introduced by Feldman, Svensson and Zenklusen [20] to handle online settings such as the SPI problem where the arrival order of the elements is adversarial.

Definition 5 (Online Contention Resolution Scheme (OCRS)). *Let us consider the following online selection setting. A point $\boldsymbol{x} \in \mathcal{P}$ is given and let $R(\boldsymbol{x})$ be a random subset of active elements. The elements $e \in \mathcal{N}$ reveal one by one whether they are active, i.e., $e \in R(\boldsymbol{x})$, and the decision whether to select an active element is taken irrevocably before the next element is revealed. An* Online Contention Resolution Scheme *for $\mathcal{P}$ is an online algorithm that selects a subset $I \subseteq R(\boldsymbol{x})$ such that $\mathbb{1}_I \in \mathcal{P}$.*

Monotonicity of a CRS is important for rounding the multilinear relaxation of submodular functions [11], although such a condition is not needed for modular functions. In the online setting, [20] defines the notion of a *greedy* OCRS which is helpful in rounding for submodular functions.

Definition 6 (Greedy OCRS). *Let $\mathcal{P} \subseteq [0,1]^n$ be a relaxation for the feasible sets $\mathcal{F} \subseteq 2^{\mathcal{N}}$. A* greedy OCRS *$\pi$ for $\mathcal{P}$ is an OCRS that for any $\boldsymbol{x} \in \mathcal{P}$ defines a downward-closed subfamily of feasible sets $\mathcal{F}_{\pi,\boldsymbol{x}} \subseteq \mathcal{F}$, and an element e is selected when it arrives if, together with the already selected elements, the obtained set is in $\mathcal{F}_{\pi,\boldsymbol{x}}$. If the choice of $\mathcal{F}_{\pi,\boldsymbol{x}}$ given $\boldsymbol{x}$ is randomized, we talk about a randomized greedy OCRS; otherwise, we talk about a deterministic greedy OCRS.*

For a greedy OCRS, the quality of the approximation guaranteed with respect to the multilinear relaxation is governed by the notion of (b,c)-selectability [20].

Definition 7 ((b,c)-selectability). *Let $b, c \in [0,1]$. A greedy OCRS for $\mathcal{P}$ is (b,c)-selectable if for any $\boldsymbol{x} \in b \cdot \mathcal{P}$, we have*

$$\Pr\left[I \cup \{e\} \in \mathcal{F}_{\pi,\boldsymbol{x}} \quad \forall I \subseteq R(\boldsymbol{x}),\, I \in \mathcal{F}_{\pi,\boldsymbol{x}}\right] \geq c, \quad \forall e \in \mathcal{N}.$$

We introduce the following notation, which will be useful in our analysis when dealing with the input constraints.

Definition 8 (Blowup of a Ground Set). *Let $\mathcal{N}$ denote a finite set, and $\mathcal{N}'$ denote another finite set, to be defined, with $|\mathcal{N}| \leq |\mathcal{N}'|$. Suppose for each $e \in \mathcal{N}$ there is an associated finite non-empty set $A_e \subseteq \mathcal{N}'$ such that the sets $A_e, e \in \mathcal{N}$ are mutually disjoint. Let $\mathcal{A} = \{A_e \mid e \in N\}$. We call $\mathcal{N}' = \bigcup_{e \in N} A_e$ the* blowup *of $\mathcal{N}$ by $\mathcal{A}$.*

Definition 9 (Partition Extension of a Constraint). *Let $\mathcal{C} = (\mathcal{N}, \mathcal{I})$ be a downward-closed constraint family over $\mathcal{N}$. Consider a blowup $\mathcal{N}'$ of $\mathcal{N}$ induced by sets $A_e, e \in \mathcal{N}$. Consider the function $g : \mathcal{N}' \to \mathcal{N}$ where $g(e') = e$ if and only if $e' \in A_e$. The* partition extension *of $\mathcal{C}$, denoted by $\mathcal{C}'$, is a constraint family $(\mathcal{N}', \mathcal{I}')$ where*

$$\mathcal{I}'_A = \{S \subseteq \mathcal{N}' \mid g(S) \in \mathcal{I} \text{ and } \forall e \in \mathcal{N},\, |A_e \cap S| \leq 1\}.$$

2.2 Useful Lemmas

Below we state two lemmas regarding sampling and submodular functions that we need.

Lemma 1 (Lemma 2.2 from [7]). *Let $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be submodular. Denote by $A(p)$ a random subset of A where each element appears with probability at most p (not necessarily independently). Then,*

$$\mathbb{E}[f(A(p))] \geq (1 - p) \cdot f(\emptyset).$$

Lemma 2 (Lemma 2.2 from [33]). *Let $g : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be submodular. Denote by $A(p)$ a random subset of A where each element appears with probability exactly p (not necessarily independently). Then*

$$\mathbb{E}[g(A(p))] \geq (1 - p) \cdot g(\emptyset) + p \cdot g(A).$$

3 Submodular Prophet Inequality Problem

In the *Submodular Prophet Inequality (SPI)* problem, we are given n random variables $X_1, \dots, X_n$ following (known) distributions $D_1, \dots, D_n$, along with a constraint $\mathcal{C}$ on $\mathcal{N} = \{1, 2, \dots, n\}$. The random variables are arranged in adversarial (worst-case) order. Let $\mathcal{U}_i$ denote the image (range) of X_i, and $\mathcal{I}$ denote the feasible sets of $\mathcal{C}$.

The online algorithm starts with a set $S = \emptyset$ of selected elements and a set $Z = \emptyset$ of selected indices from $\mathcal{N}$. At the i-th time step, it is presented with the realization $e \in \mathcal{U}_i$ of X_i. At that moment, it has to decide irrevocably whether to include e in S (and hence i in Z) or not, subject to Z remaining feasible in $\mathcal{C}$. The algorithm is also given a non-negative submodular function $f : \mathcal{U} \to \mathbb{R}_{\geq 0}$, where $\mathcal{U} \triangleq \bigcup_{i=1}^{n} \mathcal{U}_i$. The algorithm's objective is to maximize $f(S)$, subject to Z being feasible in $\mathcal{C}$.

In this model, we are comparing against the almighty adversary who already knows all realizations and can adaptively change the order in which to reveal the random variables based on the algorithm's actions so far and also the random coins it uses (if the algorithm is randomized). The prophet/adversary will select the best possible set S^* according to the constraints with knowledge of the realizations. Thus, we compare the expected value of the online algorithm against the expected value of the prophet, which is

$$\mathrm{OPT} = \mathop{\mathbb{E}}_{\boldsymbol{X}} \left[\max_{T \in \mathcal{I}} f\left(\{X_i \mid i \in T\}\right) \right].$$

Later, we will use an OCRS to round the fractional solution we obtain in this section. Since f is defined over $\mathcal{U}$ but the constraint given is over $\mathcal{N}$, we cannot immediately apply an OCRS for rounding. To overcome this issue, we view $\mathcal{U}$ as the blowup of $\mathcal{N}$ with respect to $\{\mathcal{U}_i\}_{i=1}^{n}$. On each step i, only one element arrives. Therefore, our input constraint $\mathcal{C}$ is equivalent to a new constraint $\mathcal{C}'$ on $\mathcal{U}$, where we are allowed to pick only one element from each $\mathcal{U}_i$. Notice that this is exactly the partition extension $\mathcal{C}' = (\mathcal{U}, \mathcal{I}')$ of $\mathcal{C}$.

We also denote $\boldsymbol{X} = \{X_1, X_2, \dots, X_n\}$ and $\boldsymbol{D} = \{D_1, D_2, \dots, D_n\}$. For an element $e \in \mathcal{U}_i$ we let $D_i(e)$ denote the probability of e being realized; we also use the notation $\mathcal{D}(e)$ to denote the probability of $e \in \mathcal{U}$ when we do not need to specify the part it belongs to. Note that the elements within $\mathcal{U}_i$ are correlated and hence we do not have a product distribution on $\mathcal{U}$.

Algorithmic approach: Following the description in Section 1, we design an online algorithm following the general approach of [31] but with technical differences. First, we obtain a relaxation of the prophet's objective. Afterwards, to design an online algorithm, we obtain an *offline* fractional point $\boldsymbol{z}$ based on the input, and round it *online* using a greedy OCRS and other tools. In this section, we describe the relaxation of the prophet's objective and how to obtain an offline fractional point $\boldsymbol{z}$. The process of rounding $\boldsymbol{z}$ online using a greedy OCRS is presented in Section 4.

Before we proceed, we describe a simple but technically important reduction that allows us to obtain improved bounds.

Observation 5 (Reduction to small probabilities) *Let $I = (\mathcal{N}, \mathcal{U}, \boldsymbol{D}, \boldsymbol{X}, f, \mathcal{C})$ be an instance of the Submodular Prophet Inequality problem. For every fixed $\varepsilon > 0$, there is a reduction of I to another instance $I' = (\mathcal{N}, \mathcal{U}', \boldsymbol{D'}, \boldsymbol{Y}, g, \mathcal{C})$ of the SPI problem such that that (i) for all $e \in \mathcal{U}'$, $\boldsymbol{D'}(e) \leq \varepsilon$ and (ii) there exists an α-competitive algorithm for I if and only if there exists an α-competitive algorithm for I'.*

Remark 1. The reduction's simplicity may make the reader wonder why it is useful in achieving improved bounds. The reason is a combination of the model and the power of submodularity. The fact that we can only pick a single element from each $\mathcal{U}_i$ allows us to make copies of the elements, and we can use a derived submodular function to treat the copies as a single element.

We describe the reduction in Appendix B, and sketch its correctness since it is rather simple and easy to see, though tedious to formally prove. The reduction to small probabilities allows us to use improved correlation gaps, as well as obtain better bounds in the rounding algorithm.

3.1 An upper bound on the prophet's value:

Let $\mathcal{P}$ denote a solvable polyhedral relaxation of $\mathcal{C}$. Then one can easily develop a solvable polyhedral relaxation of $\mathcal{C}'$ as follows:

$$\mathcal{P}' = \left\{ \boldsymbol{y} \in [0,1]^{|\mathcal{U}|} \;\middle|\; \sum_{e \in \mathcal{U}_i} y_e = x_i, \;\; i \in [n], \boldsymbol{x} \in \mathcal{P} \right\}.$$

Consider any algorithm, including an offline algorithm, that computes a feasible output given the realizations of the random variables. For any fixed algorithm $\mathcal{A}$ (deterministic or randomized) we have a probability $p_{\mathcal{A}}(e)$ for each $e \in \mathcal{U}$ appearing in the output of $\mathcal{A}$. Since an element $e \in \mathcal{U}$ is realized with probability $\mathcal{D}(e)$, e cannot appear in the output of $\mathcal{A}$ with probability more than $\mathcal{D}(e)$. Moreover, for a given realization, each output of the algorithm is feasible. Putting these facts together we obtain the following observation.

Observation 6 *Let $\mathcal{A}$ be any online or offline algorithm for a given instance of the problem. Let $p_{\mathcal{A}}(e)$ denote the probability that e is in the output of $\mathcal{A}$. Then the vector $\boldsymbol{p}$ is in the polytope*

$$\mathcal{P}'' = \left\{ \boldsymbol{z} \in [0,1]^{|\mathcal{U}|} \;\middle|\; \boldsymbol{z} \in \mathcal{P}', z_e \leq \mathcal{D}(e) \; e \in \mathcal{U} \right\}.$$

We are now ready to proceed with the relaxation of the prophet's objective.

Claim. Consider an instance of the Submodular Prophet Inequality problem. Then

$$\max_{\boldsymbol{z} \in \mathcal{P}''} f^+(\boldsymbol{z}) \geq \mathrm{OPT}.$$

Proof. Fix an optimal strategy for the prophet and let $\boldsymbol{y^*} \in [0,1]^{|\mathcal{U}|}$ denote the vector of probabilities of the elements appearing in the output of the prophet's strategy. We have $\boldsymbol{y^*} \in \mathcal{P}''$. By the definition of the concave closure of f, $f^+(\boldsymbol{y^*})$ maximizes the value of f among all distributions with the marginals $\boldsymbol{y^*}$ (note that the distribution that achieves this may not be a feasible strategy for any algorithm). Therefore, $f^+(\boldsymbol{y^*}) \geq \mathrm{OPT}$, which also implies that $\max_{\boldsymbol{z} \in \mathcal{P}''} f^+(\boldsymbol{z}) \geq \mathrm{OPT}$.

3.2 Fractional Solution and Correlation Gap

From Claim 3.1, $\max_{\boldsymbol{z} \in \mathcal{P}''} f^+(\boldsymbol{z}) \geq \mathrm{OPT}$. Since OCRSs are designed to relate the quality of their output to that of the multilinear relaxation, we need to relate $F(\boldsymbol{z})$ to $f^+(\boldsymbol{z})$ and hence to OPT. We present two different ways to do this — via a direct correlation gap and via the Measured Continuous Greedy (MCG) algorithm — with the second yielding strictly better results than the first.

The direct correlation gap approach The first approach is not computationally efficient and relies on optimally solving the optimization problem $\max_{\boldsymbol{z} \in \mathcal{P}''} f^+(\boldsymbol{z})$. Let $\boldsymbol{z^*}$ be the optimum solution. We can then use the correlation gap to relate $F(\boldsymbol{z^*})$ to OPT. For monotone functions we have $F(\boldsymbol{z^*}) \geq (1-1/e) f^+(\boldsymbol{z^*}) \geq (1-1/e)\mathrm{OPT}$. For non-negative functions we can use Theorem 1, the proof of which, along with all results on the direct correlation gap approach, can be found in Appendix A. Following the reduction that we described earlier, we can assume that $z^*_e \leq \max_e \boldsymbol{D}(e) \leq \varepsilon$ for all e and this implies, via Theorem 1 that $F(\boldsymbol{z^*}) \geq (1-\varepsilon)(1-1/e) f^+(\boldsymbol{z^*}) \geq (1-\varepsilon)(1-1/e)\mathrm{OPT}$. In rounding it is useful to have a solution $\boldsymbol{z} \in b \cdot \mathcal{P}''$ for some parameter $b \in (0,1)$. One can of course use $\boldsymbol{z} = b\boldsymbol{z^*}$ and in this case, we can use the concavity of f^+ to see that $f^+(b\boldsymbol{z^*}) \geq b f^+(\boldsymbol{z^*})$, and then apply the correlation gap to $b\boldsymbol{z^*}$ to conclude that, in the monotone case, $F(b\boldsymbol{z^*}) \geq b(1-1/e) f^+(\boldsymbol{z^*}) \geq b(1-1/e)\mathrm{OPT}$ and, in the non-monotone case, $F(b\boldsymbol{z^*}) \geq b(1-\varepsilon)(1-1/e) f^+(\boldsymbol{z^*}) \geq b(1-\varepsilon)(1-1/e)\mathrm{OPT}$.

The measured continuous greedy approach The second approach is algorithmic and relies on the Measured Continuous Greedy (MCG) algorithm and its properties. We state two relevant known results about the algorithm. For these results as well as Theorem 3, we assume the submodular function f is given via a value oracle, and that the algorithms are randomized and run in polynomial time and are correct with high probability.

Lemma 3 (Lemma 4 of [1]). *Let f be a monotone submodular function with multilinear extension F, and let $\mathcal{P}$ be a solvable downward-closed polytope. Let $\boldsymbol{x}(b)$ be solution produced by the Continuous Greedy algorithm on F and $\mathcal{P}$ until time $b \in (0,1]$. Then (i) $\boldsymbol{x}(b) \in b \cdot \mathcal{P}$ and (ii) $F(\boldsymbol{x}(b)) \geq \left(1 - e^{-b} - o(1)\right) \cdot \max_{\boldsymbol{y} \in \mathcal{P}} f^+(\boldsymbol{y})$.*

For a general non-negative submodular function, the MCG algorithm achieves the following bound.

Lemma 4 (Lemma 8.3 of [2]). *Let $b \in [0,1]$, f be a non-negative submodular function with multilinear extension F, and let $\mathcal{P}$ be a solvable downward-closed polytope. Then, the solution $\boldsymbol{x}(b) \in [0,1]^n$ produced by the MCG algorithm satisfies (i) $\boldsymbol{x}(b) \in b \cdot \mathcal{P}$ and (ii) $F(\boldsymbol{x}(b)) \geq \left(b \cdot e^{-b} - \varepsilon\right) \cdot \max_{\boldsymbol{y} \in \mathcal{P}} f^+(\boldsymbol{y})$, for any fixed $\varepsilon > 0$.*

The two preceding lemmas are algorithmic. If $\mathcal{P}$ is solvable then the underlying algorithms can be implemented efficiently. Based on our reduction to small probabilities it is useful to consider whether the preceding lemmas can take advantage of this. No advantage is possible in the monotone setting, however, we show below that one can indeed take advantage of the reduction when f is non-monotone. We provide a refined analysis of the standard bound of the MCG algorithm, which depends on a parameter p that quantifies the maximum value of any coordinate that is feasible in the polytope. For small enough p, Theorem 3 constitutes an improvement over Lemma 4, which comprises the main result of this section. Notice that Theorem 2 follows from Theorem 3 by setting $b = 1$.

Theorem 3. *Let $p \in [0,1)$, f be a non-negative submodular function with multilinear extension F and $\mathcal{P}$ be a downward-closed solvable polytope[10] on $\mathcal{N}$, such that $\mathcal{P} \subseteq p \cdot [0,1]^{\mathcal{N}}$ (that is, if $\boldsymbol{z} \in \mathcal{P}$ then $z_i \leq p$ for all $i \in \mathcal{N}$). Then, the output of the Measured Continuous Greedy (MCG) algorithm on F and $\mathcal{P}$ at time $b \in [0,1]$ is a vector $\boldsymbol{x}(b) \in b \cdot \mathcal{P}$ such that*

$$F(\boldsymbol{x}(b)) \geq \begin{cases} b \cdot e^{-b} \cdot \max_{\boldsymbol{z} \in \mathcal{P}} f^+(\boldsymbol{z}), & 0 \leq b \leq \ln\left(\frac{1}{1-p}\right) \\ \left(1 - p - e^{-b}\left(1 + \ln\left(1-p\right)\right)\right) \cdot \max_{\boldsymbol{z} \in \mathcal{P}} f^+(\boldsymbol{z}), & \ln\left(\frac{1}{1-p}\right) \leq b \leq 1. \end{cases}$$

[10] Informally, a polytope $\mathcal{P}$ is solvable if one can efficiently do linear optimization over it. A formal definition is given in Section 2.

Remark 2. Notice that, for the SPI problem, due to our reduction, we can assume that all vectors $\boldsymbol{z} \in \mathcal{P}''$ have $z_i \leq \varepsilon'$ for all $i \in \mathcal{N}$, for any fixed constant $\varepsilon' > 0$. Therefore, for any fixed constant $\varepsilon > 0$, there exists an ε' such that

$$F(\boldsymbol{x}(b)) \geq \left(1 - e^{-b} - \varepsilon\right) \cdot \max_{\boldsymbol{z} \in \mathcal{P}} f^+(\boldsymbol{z}),$$

where $\boldsymbol{x}(b) \in b \cdot \mathcal{P}''$ is the output of the MCG algorithm at time b.

Before we proceed with the proof of Theorem 3, we briefly sketch the idea implicit in prior work [19,2] that implies $F_{\max}(\boldsymbol{x}) \geq \frac{1}{e} f^+(\boldsymbol{x})$. Consider a downward-closed polytope $\mathcal{P}$ defined by all points in $[0,1]^n$ dominated by the given point $\boldsymbol{x}$: $\mathcal{P} := \{\boldsymbol{y} \in [0,1]^n \mid \forall\, 1 \leq i \leq n,\ y_i \leq x_i\}$. Suppose we run the MCG algorithm on $\mathcal{P}$. From Lemma 8.3 of [2] for $b = 1$, for any $\varepsilon > 0$, the algorithm can be used to find a point $\boldsymbol{z}_\varepsilon \in \mathcal{P}$ such that $F(\boldsymbol{z}_\varepsilon) \geq \left(\frac{1}{e} - \varepsilon\right) \max_{\boldsymbol{y} \in \mathcal{P}} f^+(\boldsymbol{y}) \geq \left(\frac{1}{e} - \varepsilon\right) f^+(\boldsymbol{x})$. Since such a point $\boldsymbol{z}_\varepsilon \in \mathcal{P}$ exists for any $\varepsilon > 0$, by the compactness of $\mathcal{P}$ and the continuity of F and f^+, it follows that there exists a point $\boldsymbol{y} \in \mathcal{P}$ such that $F(\boldsymbol{y}) \geq \frac{1}{e} \cdot f^+(\boldsymbol{x})$. Also notice that $\boldsymbol{x} \in \mathcal{P}$, and thus

$$F_{\max}(\boldsymbol{x}) = \max_{\boldsymbol{z} \in \mathcal{P}} F(\boldsymbol{z}) \geq F(\boldsymbol{y}) \geq \frac{1}{e} \cdot f^+(\boldsymbol{x}).$$

To prove Theorem 3, which generalizes Lemma 8.3 in [2], we use the same proof outline as above, but in the algorithm's analysis, we take advantage of the fact that $\|\boldsymbol{x}\|_\infty \leq p$.

Proof of Theorem 3. Let $\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{z} \in \mathcal{P}} f^+(\boldsymbol{z})$. Recall that there exists $\boldsymbol{\alpha} \in [0,1]^{2^{\mathcal{N}}}$ such that

$$f^+(\hat{\boldsymbol{x}}) = \sum_{S \subseteq \mathcal{N}} \alpha_S f(S), \quad \sum_{S \subseteq \mathcal{N}} \alpha_S = 1 \quad \text{and} \quad \sum_{S \subseteq \mathcal{N}} \alpha_S \mathbb{1}_S = \hat{\boldsymbol{x}}.$$

From the analysis of Measured Continuous Greedy and the fact that $\boldsymbol{x}(b) \in \mathcal{P}$, we know that, at time b, for all $i \in \mathcal{N}$ we have

$$x_i(b) \leq \min\{1 - e^{-b}, p\}.$$

Let $\boldsymbol{x} = \boldsymbol{x}(b)$, and, for $S \subseteq \mathcal{N}$, consider a line of direction $\boldsymbol{d}_S = (\boldsymbol{x} \vee \mathbb{1}_S) - \boldsymbol{x} = (\mathbb{1}_S - \boldsymbol{x}) \vee \boldsymbol{0}$. Notice that $\boldsymbol{0} \leq \boldsymbol{d}_S \leq \mathbb{1}_S$ for all $S \subseteq \mathcal{N}$. From Section 2.3 of [8], it follows that

$$\boldsymbol{d}_S \cdot \nabla F(\boldsymbol{x}) \geq F(\boldsymbol{x} \vee \mathbb{1}_S) - F(\boldsymbol{x}).$$

Since f may not be monotone, $\nabla F(\boldsymbol{x})$ may have negative entries. Let $\boldsymbol{d}'_S$ be a vector obtained from $\boldsymbol{d}_S$ as follows: $(\boldsymbol{d}'_S)_i = (\boldsymbol{d}_S)_i$ if $\nabla F(\boldsymbol{x})_i \geq 0$, otherwise $(\boldsymbol{d}'_S)_i = 0$. We have $\boldsymbol{0} \leq \boldsymbol{d}'_S \leq \boldsymbol{d}_S$ and,

$$\boldsymbol{d}'_S \cdot \nabla F(\boldsymbol{x}) \geq \max\{0, \boldsymbol{d}_S \cdot \nabla F(\boldsymbol{x})\} \geq \max\{0, F(\boldsymbol{x} \vee \mathbb{1}_S) - F(\boldsymbol{x})\}.$$

Since $\boldsymbol{x}(b)_i \leq \min\{1 - e^{-b}, p\}$ for all $i \in \mathcal{N}$, by Lemma III.5 of [19], we have

$$F(\boldsymbol{x} \vee \mathbb{1}_S) \geq \left(1 - \min\{1 - e^{-b}, p\}\right) f(S).$$

Therefore,

$$\begin{aligned} \boldsymbol{d}'_S \cdot \nabla F(\boldsymbol{x}) &\geq \max\{0, (1-p)f(S) - F(\boldsymbol{x}), e^{-b} f(S) - F(\boldsymbol{x})\} \\ &\geq \max\{1-p, e^{-b}\} f(S) - F(\boldsymbol{x}). \end{aligned}$$

Next, let $\hat{\boldsymbol{d}} = \sum_{S \subseteq \mathcal{N}} \alpha_S \boldsymbol{d}'_S$. Since $\boldsymbol{d}_S \leq \mathbb{1}_S$ and $\boldsymbol{d}'_S \leq \boldsymbol{d}_S$, we have $\boldsymbol{d}'_S \leq \mathbb{1}_S$, and thus

$$\hat{\boldsymbol{d}} = \sum_{S \subseteq \mathcal{N}} \alpha_S \boldsymbol{d}'_S \leq \sum_{S \subseteq \mathcal{N}} \alpha_S \mathbb{1}_S = \hat{\boldsymbol{x}}.$$

Since $\mathcal{P}$ is downward-closed and $\hat{\boldsymbol{x}} \in \mathcal{P}$, we know that $\hat{\boldsymbol{d}} \in \mathcal{P}$. Therefore, from the above and the fact that $\boldsymbol{v}_{\max} = \arg\max_{\boldsymbol{v} \in \mathcal{P}} \boldsymbol{v} \cdot \nabla F(\boldsymbol{x})$, we have

$$\begin{aligned} \frac{dF(\boldsymbol{x}(b))}{db} &= \boldsymbol{v}_{\max}(\boldsymbol{x}) \cdot \nabla F(\boldsymbol{x}) \\ &\geq \hat{\boldsymbol{d}}_S \cdot \nabla F(\boldsymbol{x}) \\ &= \sum_{S \subseteq \mathcal{N}} \alpha_S \cdot \boldsymbol{d}'_S \cdot \nabla F(\boldsymbol{x}) \\ &\geq \sum_{S \subseteq \mathcal{N}} \alpha_S \left(\max\{1-p, e^{-b}\} f(S) - F(\boldsymbol{x}) \right) \\ &\geq \max\{1-p, e^{-b}\} \sum_{S \subseteq \mathcal{N}} \alpha_S f(S) - \sum_{S \subseteq \mathcal{N}} \alpha_S F(\boldsymbol{x}) \\ &\geq \max\{1-p, e^{-b}\} f^+(\hat{\boldsymbol{x}}) - F(\boldsymbol{x}). \end{aligned}$$

We proceed to solve the above differential inequality. For brevity, let $y = F(\boldsymbol{x})$. Then,

$$\begin{aligned} dy + y\, db &\geq f^+(\hat{\boldsymbol{x}}) \max\{1-p, e^{-b}\}\, db \\ e^b\, dy + y e^b\, db &\geq f^+(\hat{\boldsymbol{x}}) \max\{(1-p)e^b, 1\}\, db \\ d\left(y e^b\right) &\geq f^+(\hat{\boldsymbol{x}}) \max\{(1-p)e^b, 1\}\, db \\ y &\geq e^{-b} f^+(\hat{\boldsymbol{x}}) \int_0^b \max\{(1-p)e^u, 1\}\, du. \end{aligned} \tag{1}$$

Notice that, for $0 \leq u \leq \ln\left(\frac{1}{1-p}\right)$, we have $(1-p)e^u \leq 1$, while for $\ln\left(\frac{1}{1-p}\right) \leq u \leq 1$, we have $1 \leq (1-p)e^u$. Therefore, for $b \leq \ln\left(\frac{1}{1-p}\right)$, (1) becomes

$$y \geq e^{-b} f^+(\hat{\boldsymbol{x}}) \int_0^b 1\, du = b \cdot e^{-b} \cdot f^+(\hat{\boldsymbol{x}}),$$

whereas for $b \geq \ln\left(\frac{1}{1-p}\right)$, (1) becomes

$$y \geq e^{-b} f^+(\hat{\boldsymbol{x}}) \left(\int_0^{\ln\left(\frac{1}{1-p}\right)} 1\, du + \int_{\ln\left(\frac{1}{1-p}\right)}^b (1-p)e^u\, du \right)$$

$$= \left(1 - p - e^{-b}\left(1 + \ln\left(1 - p\right)\right)\right) f^{+}(\hat{\boldsymbol{x}}).$$

We conclude that

$$F(\boldsymbol{x}(b)) \geq \begin{cases} b \cdot e^{-b} \cdot \max_{\boldsymbol{z} \in \mathcal{P}} f^{+}(\boldsymbol{z}), & 0 \leq b \leq \ln\left(\frac{1}{1-p}\right) \\ \left(1 - p - e^{-b}\left(1 + \ln\left(1 - p\right)\right)\right) \cdot \max_{\boldsymbol{z} \in \mathcal{P}} f^{+}(\boldsymbol{z}), & \ln\left(\frac{1}{1-p}\right) \leq b \leq 1. \end{cases}$$

We summarize the results via both methods below. We observe that for both monotone and non-monotone functions the bounds are best when $p \to 0$, which we can ensure via the reduction. Once we make this assumption, the bounds provided by the correlation gap approach are essentially $(1 - 1/e)$ when $b = 1$ which is optimal. However, these bounds are matched by the Continuous Greedy approach. When $b < 1$, which will be the case when applying the rounding schemes, the bound in Lemma 3 and our new refined bound in Theorem 3 are superior and have the further advantage of being computable in polynomial time.

4 Rounding the fractional solution

In the preceding section we described ways to obtain a vector $\boldsymbol{z} \in b \cdot \mathcal{P}''$ for some $b \in [0, 1]$ such that $F(\boldsymbol{z}) \geq \alpha \cdot OPT$ for various constants α depending on the approach. In this section we show how to round $\boldsymbol{z}$ in an online fashion. We follow the high-level approach of [31] but refine it in several ways. We will use a greedy OCRS for $\mathcal{C}$ via the relaxation $\mathcal{P}$ as a black box. Recall that our rounding needs to produce a feasible set in $\mathcal{C}'$ with ground set $\mathcal{U}$, while the OCRS is for the constraint on the variables of $\mathcal{N}$. Moreover the distribution $\boldsymbol{D}$ is *not* a product distribution on $\mathcal{U}$. These are the technical challenges that need to be overcome in the algorithm and analysis. The quality of the output will depend on the properties of the OCRS for $\mathcal{P}$. We assume that the greedy OCRS for $\mathcal{P}$ is (b, c)-selectable, where c is some function of b. This depends on the specific constraint family $\mathcal{C}$ and the polyhedral relaxation $\mathcal{P}$. At the end of the section, we use known results to derive concrete competitive ratios for several constraint families of interest. We note that $\boldsymbol{z} \in \mathcal{P}''$, which also implies that $\boldsymbol{z} \in \mathcal{P}'$. For rounding purposes we only work with $\mathcal{P}'$ and $\mathcal{P}$; $\mathcal{P}''$ is only necessary to obtain an upper bound on OPT.

We rely on the certain parts of the analysis of OCRS for submodular function maximization from [20]. In the following, we will use π to denote the mapping function for the OCRS over the ground set $\mathcal{N}$ and the polytope $\mathcal{P}$. Technically the mapping π is a function of $\boldsymbol{x} \in \mathcal{P}$ and should be written as $\pi_{\boldsymbol{x}}$ but we omit $\boldsymbol{x}$ for notational simplicity. We also note that π can be randomized. An important definition from [20] in the analysis of OCRSs is the characteristic CRS of a greedy OCRS.

Definition 10 (Characteristic CRS of a greedy OCRS). *The* characteristic CRS $\bar{\pi}$ *of a greedy OCRS* π *for a polytope* $\mathcal{P}$ *is a CRS for the same polytope* $\mathcal{P}$. *It is defined for an input* $x \in \mathcal{P}$ *and a set* $A \subseteq \mathcal{N}$ *by* $\bar{\pi}(A) = \{e \in A \mid I \cup \{e\} \in \mathcal{F}_{\pi,\boldsymbol{x}},\ \forall I \subseteq A,\ I \in \mathcal{F}_{\pi,\boldsymbol{x}}\}$. *Notice that, if* π *is randomized, then* $\bar{\pi}$ *is randomized as well.*

We will also need the following known results from [20].

Observation 7 (Observation 3.3 of [20]) *For every set $A \subseteq \mathcal{N}$ and a characteristic CRS $\bar{\pi}$ of a greedy OCRS π, the set $\bar{\pi}(A)$ is always a subset of the elements selected by π when the active elements are the elements of A.*

Lemma 5 (Lemma 3.4 of [20]). *The characteristic CRS $\bar{\pi}$ of a (b, c)-selectable greedy OCRS π is (b, c)-balanced and monotone.*

For any $S \subseteq \mathcal{U}$, we define $S_\downarrow \subseteq \mathcal{N}$ to be the projection of S onto $\mathcal{N}$, i.e.

$$S_\downarrow \coloneqq \{i \in \mathcal{N} \mid S \cap \mathcal{U}_i \neq \emptyset\}.$$

Also, for a greedy OCRS π, we denote the characteristic CRS of π by $\bar{\pi}$. We now define a CRS π' for $\mathcal{P}'$ that we will need for our analysis later on. We define π' using the characteristic CRS $\bar{\pi}$ of π as follows. For any set $S \subseteq \mathcal{U}$,

$$\pi'(S) \coloneqq \bigcup_{\substack{i \in \bar{\pi}(S_\downarrow) \\ |S \cap \mathcal{U}_i| = 1}} (S \cap \mathcal{U}_i).$$

Intuitively, the characteristic CRS $\bar{\pi}$ of a greedy OCRS π returns, on input $A \subseteq \mathcal{N}$ the set of all elements in A that are in $\pi(A)$ regardless of the arrival order chosen by the adversary. Given a set $S \subseteq \mathcal{U}$, $\pi'(S)$ is equal to the union of at most one element from each $\mathcal{U}_i$, for all such i that are in the projection $S_\downarrow$ of S and are selected by $\bar{\pi}$ on input $S_\downarrow$. The next lemma relates the balance guarantee provided by π' given a selectability guarantee on π.

Lemma 6. *For any (b, c)-selectable greedy OCRS π for $\mathcal{P}$ and $\boldsymbol{z} \in \mathcal{P}'$, the CRS π' is monotone and $(b, c \cdot \gamma)$-balanced, where $\gamma = \min_{i \in \mathcal{N}} \prod_{e \in \mathcal{U}_i} (1 - z_e)$.*

Proof. First, notice that π' is a CRS, since $\pi'(S) \subseteq S$ for all $S \subseteq \mathcal{U}$. This follows immediately from the definition of π' as $S \cap \mathcal{U}_i \subseteq S$ for all $i \in \mathcal{N}, S \subseteq \mathcal{U}$.

Next, we show that π' is monotone. Fix an element $e \in S_1 \subseteq S_2 \subseteq \mathcal{U}$, and an instantiation of $\mathcal{F}_{\pi,\boldsymbol{x}}$ (this is relevant if the OCRS is randomized). Let $e \in \mathcal{U}_i$ for some $i \in \mathcal{N}$. Suppose $e \in \pi'(S_2)$. This implies that $|S_2 \cap \mathcal{U}_i| = 1$ and since $S_1 \subseteq S_2$ and $e \in S_1$, we have $|S_1 \cap \mathcal{U}_i| = 1$. Furthermore, we know that $i \in \bar{\pi}(S_{2\downarrow})$. Since $S_1 \subseteq S_2$, it follows that $S_{1\downarrow} \subseteq S_{2\downarrow}$. By Lemma 5, we know that $\bar{\pi}$ is monotone, and thus, since $i \in \bar{\pi}(S_{2\downarrow})$, it follows that $i \in \bar{\pi}(S_{1\downarrow})$. Therefore, we know that $e \in \pi'(S_1)$. Since $e \in \pi'(S_2)$ implies that $e \in \pi'(S_1)$, unconditioning over the instantiation of $\mathcal{F}_{\pi,\boldsymbol{x}}$ yields

$$\Pr[e \in \pi'(S_1)] \geq \Pr[e \in \pi'(S_2)].$$

We now show that π' is $(b, c \cdot \gamma)$-balanced, for $\gamma = \min_{i \in \mathcal{N}} \prod_{e \in \mathcal{U}_i} (1 - z_e)$. It suffices to show that, for any $e \in \mathcal{U}$

$$\Pr_{S \sim R(\boldsymbol{z})}[e \in \pi'(S) \mid e \in S] \geq c \cdot \gamma.$$

Notice that, for any realization S of $R(\boldsymbol{z})$, $e \in \pi'(S)$ if and only if $S \cap \mathcal{U}_i = \{e\}$ and $i \in \bar{\pi}(S_\downarrow)$. Thus,

$$\begin{aligned}
\Pr_{S\sim R(\boldsymbol{z})}[e \in \pi'(S) \mid e \in S] &= \Pr_{S\sim R(\boldsymbol{z})}[S \cap \mathcal{U}_i = \{e\} \wedge i \in \bar{\pi}(S_\downarrow) \mid e \in S] \\
&= \Pr_{S\sim R(\boldsymbol{z})}[S \cap \mathcal{U}_i = \{e\} \mid e \in S] \\
&\quad \cdot \Pr_{S\sim R(\boldsymbol{z})}[i \in \bar{\pi}(S_\downarrow) \mid S \cap \mathcal{U}_i = \{e\}, e \in S] \\
&= \Pr_{S\sim R(\boldsymbol{z})}[S \cap \mathcal{U}_i = \{e\} \mid e \in S] \\
&\quad \cdot \Pr_{S\sim R(\boldsymbol{z})}[i \in \bar{\pi}(S_\downarrow) \mid S \cap \mathcal{U}_i = \{e\}], \qquad (2)
\end{aligned}$$

where the last equality follows from the fact that, if $S \cap \mathcal{U}_i = \{e\}$, then $e \in S$. We lower bound each probability in (2) separately, starting from

$$\Pr_{S\sim R(\boldsymbol{z})}[S \cap \mathcal{U}_i = \{e\} \mid e \in S] = \prod_{e' \neq e, e' \in \mathcal{U}_i} (1 - z_{e'}) \geq \prod_{e' \in \mathcal{U}_i} (1 - z_{e'}) \geq \gamma. \qquad (3)$$

Also, notice that $\bar{\pi}$ is a CRS over $\mathcal{N}$ and does not depend on which $S \cap \mathcal{U}_i$ led to $i \in S_\downarrow$. Therefore,

$$\Pr[i \in \bar{\pi}(S_\downarrow) \mid i \in S_\downarrow] = \Pr[i \in \bar{\pi} \mid S \cap \mathcal{U}_i = T]$$

for all $T \subseteq \mathcal{U}_i$ such that $T \neq \emptyset$. Specifically, for $T = \{e\}$,

$$\Pr[i \in \bar{\pi} \mid S \cap \mathcal{U}_i = \{e\}] = \Pr[i \in \bar{\pi}(S_\downarrow) \mid i \in S_\downarrow] \geq c, \qquad (4)$$

where the last inequality follows from the fact that fact that $\bar{\pi}$ is (b, c)-balanced, by Lemma 5.

Combining (2), (3) and (4), we obtain

$$\Pr_{S\sim R(\boldsymbol{z})}[e \in \pi'(S) \mid e \in S] \geq c \cdot \gamma.$$

Remark 3. Notice that via Observation 5, we can assume without loss of generality that, for any fixed $\varepsilon' > 0$, $z_e \leq \varepsilon'$ for all $e \in \mathcal{U}$. By choosing ε' sufficiently small, for any fixed $\varepsilon > 0$ we have

$$\gamma = \min_{i\in\mathcal{N}} \prod_{e\in\mathcal{U}_i} (1 - z_e) \geq \min_{i\in\mathcal{N}} \left(\prod_{e\in\mathcal{U}_i} e^{-z_e} \right) - \varepsilon = \min_{i\in\mathcal{N}} \left(e^{-\sum_{e\in\mathcal{U}_i} z_e} \right) - \varepsilon \geq e^{-b} - \varepsilon,$$

where the last inequality follows from the fact that $\boldsymbol{z} \in b \cdot \mathcal{P}'$. Thus, $c \cdot \gamma \geq c \cdot (e^{-b} - \varepsilon)$, and we obtain the following as corollary: For any (b, c)-selectable greedy OCRS π for $\mathcal{P}$ and fixed $\varepsilon > 0$, π' defined earlier is a $\left(b, c\left(e^{-b} - \varepsilon\right)\right)$-balanced monotone CRS for $\mathcal{P}'$.

Now we are ready to describe our online algorithm. We describe and analyze the algorithms for monotone and non-monotone cases separately, since there are technical differences. The algorithms are similar to the one in [31], however, the main technical difference is that we use the OCRS for $\mathcal{N}$ as a black box; in [31] the authors use an OCRS over $\mathcal{U}$ since they work in the special case of matroids.

4.1 Monotone Non-Negative Submodular Functions

We assume we have already computed a vector $\boldsymbol{z} \in b \cdot \mathcal{P}''$ for some $b \in [0,1]$ such that $F(\boldsymbol{z}) \geq \alpha \cdot \mathrm{OPT}$ for some α. Note that the adversary is almighty and can alter the order in which it feeds the variables to the algorithm based on knowledge of the full realizations of the variables and the actions of the algorithm so far.

Let $\boldsymbol{z_i}$ denote the product distribution on $\mathcal{U}_i$ defined by marginals $z_i(e), e \in \mathcal{U}_i$. We write $R \sim \boldsymbol{z_i}$ to denote a random set $R \subseteq \mathcal{U}_i$ realized according to this product distribution, and we denote $z_i(e)$ by z_e when i is clear from context or irrelevant. Furthermore, let $\boldsymbol{x} \in [0,1]^n$ be the vector where $x_i = \Pr_{R \sim \boldsymbol{z_i}}[R \neq \emptyset] = 1 - \prod_{e \in \mathcal{U}_i}(1 - z_e)$, for all $i \in \mathcal{N}$. We assume that $\boldsymbol{x}$ is the input vector to our OCRS π_x for $\mathcal{P}$ and its characteristic CRS $\bar{\pi}_{\boldsymbol{x}}$. To simplify our notation, we denote $\pi_{\boldsymbol{x}}$ and $\bar{\pi}_{\boldsymbol{x}}$ by π and $\bar{\pi}$, respectively.

ALGORITHM 1: Algorithm for Monotone Non-Negative Submodular Functions

Monotone Rounding($\mathcal{U}, f, \boldsymbol{D}, \mathcal{C}, \pi, \boldsymbol{z}$)

$T_{\text{ALG}} = \emptyset$
for $h \leftarrow 1$ **to** n **do**
  Let X_i be variable that arrives on step h
  Let $e \in \mathcal{U}_i$ be the realization of X_i
  With probability $\frac{\Pr_{R \sim \boldsymbol{z_i}}[R = \{e\}]}{\mathcal{D}_i(e)}$, set $T_i \leftarrow \{e\}$
  Otherwise, set T_i to be a random subset R of $\mathcal{U}_i$, drawn according to $\boldsymbol{z}_i$, conditioned on $|R| \neq 1$
  if $T_i \neq \emptyset$ **then**
    Feed i to OCRS π for $\mathcal{P}$
    if π *accepts* i and $T_i = \{e\}$ **then**
      $T_{\text{ALG}} \leftarrow T_{\text{ALG}} \cup \{e\}$
    end
  end
end
Return T_{ALG}

The online algorithm on the h-th step receives a random variable X_i decided by the almighty adversary, and once X_i is received the algorithm also sees the realization $e \in \mathcal{U}_i$ of X_i according to the distribution $\mathcal{D}_i$. The online algorithm generates a random set $T_i \subseteq \mathcal{U}_i$ *after* seeing the realization e. The idea is that if one does not see the realization e of X_i, the distribution of T_i appears identical to the product distribution generated by $\boldsymbol{z_i}$. Note that, for $S \subseteq \mathcal{U}_i$, $\Pr_{R \sim \boldsymbol{z_i}}[R = S] = \prod_{e \in S} z_e \prod_{e \in \mathcal{U} \setminus S}(1 - z_e)$.

Lemma 7. *For any $i \in \mathcal{N}$ and $S \subseteq \mathcal{U}_i$,*

$$\Pr[T_i = S] = \Pr_{R \sim \boldsymbol{z_i}}[R = S].$$

Proof. Let $\mathcal{E}_e$ be the event that $e \in \mathcal{U}_i$ is the realization of X_i. Note that $\Pr[\mathcal{E}_e] = \mathcal{D}_i(e)$. Consider $S \subseteq \mathcal{U}_i$ such that $|S| \neq 1$. We see from the algorithm's description that

$$\Pr[T_i = S \mid \mathcal{E}_e] = \left(1 - \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = \{e\}\right]}{D_i(e)}\right) \cdot \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = S\right]}{1 - \Pr_R\left[\left|R\left(\boldsymbol{z}_i\right)\right| = 1\right]}.$$

Summing up over all realizations of X_i, we have that, for any S such that $|S| \neq 1$,

$$\begin{aligned}
\Pr\left[T_i = S\right] &= \sum_{e \in \mathcal{U}_i} D_i(e) \Pr\left[T_i = S \mid \mathcal{E}_e\right] \\
&= \sum_{e \in \mathcal{U}_i} D_i(e) \left(1 - \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = \{e\}\right]}{D_i(e)}\right) \cdot \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = S\right]}{1 - \Pr_R\left[\left|R\left(\boldsymbol{z}_i\right)\right| = 1\right]} \\
&= \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = S\right]}{1 - \Pr_R\left[\left|R\left(\boldsymbol{z}_i\right)\right| = 1\right]} \cdot \sum_{e \in \mathcal{U}_i} D_i(e) \left(1 - \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = \{e\}\right]}{D_i(e)}\right) \\
&= \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = S\right]}{1 - \Pr_R\left[\left|R\left(\boldsymbol{z}_i\right)\right| = 1\right]} \cdot \left(\sum_{e \in \mathcal{U}_i} D_i(e) - \sum_{e \in U_i} \Pr_R\left[R\left(\boldsymbol{z}_i\right) = \{e\}\right]\right) \\
&= \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = S\right]}{1 - \Pr_R\left[\left|R\left(\boldsymbol{z}_i\right)\right| = 1\right]} \cdot \left(1 - \sum_{e \in \mathcal{U}_i} \Pr_R\left[R\left(\boldsymbol{z}_i\right) = \{e\}\right]\right) \\
&= \frac{\Pr_R\left[R\left(\boldsymbol{z}_i\right) = S\right]}{1 - \Pr_R\left[\left|R\left(\boldsymbol{z}_i\right)\right| = 1\right]} \cdot \left(1 - \Pr_R\left[\left|R\left(\boldsymbol{z}_i\right)\right| = 1\right]\right) \\
&= \Pr_R\left[R\left(\boldsymbol{z}_i\right) = S\right].
\end{aligned}$$

Next, consider any set S with $|S| = 1$ and, without loss of generality, assume $S = \{e\}$ for some $e \in \mathcal{U}_i$. It can be seen from the algorithm description that $T_i = \{e\}$ if and only if e is the realization of X_i and the algorithm succeeds in Line 5 in setting $T_i = \{e\}$ which happens with probability $\frac{\Pr_{R \sim \boldsymbol{z}_i}[R = \{e\}]}{\mathcal{D}_i(e)}$. Hence

$$\Pr[T_i = \{e\}] = \mathcal{D}_i(e) \cdot \frac{\Pr_{R \sim \boldsymbol{z}_i}\left[R = \{e\}\right]}{\mathcal{D}_i(e)} = \Pr_{R \sim \boldsymbol{z}_i}\left[R = \{e\}\right],$$

as desired.

We now analyze the expected value of $f(T_{\text{ALG}})$ relying on the CRS π' that we set up (this is inspired by the use of characteristic CRS in [20]).

Lemma 8. *Given a (b, c)-selectable greedy OCRS π for $\mathcal{P}$, for any $\boldsymbol{z} \in b \cdot \mathcal{P}''$ and fixed $\varepsilon > 0$, Algorithm 1 returns a set $T_{ALG} \subseteq \mathcal{U}$ such that*

$$\mathbb{E}\left[f(T_{ALG})\right] \geq c\left(e^{-b} - \varepsilon\right) \cdot F(\boldsymbol{z}).$$

Proof. It is easy to see from the algorithm's description that, for any X_i, only the actual realization of X_i can be potentially chosen to be added to T_{ALG}.

Furthermore, the variables chosen by the algorithm are feasible in $\mathcal{C}$, since this is ensured by the OCRS.

Let T_i be the random set generated by the online algorithm for variable X_i. We see that T_i is independent of $T_{i'}$ for $i \neq i'$, due to independence of the realization of the random variables $X_1, \ldots, X_n$ and the independence of the coins used in the algorithm across all steps in $\mathcal{N}$. From Lemma 7, the distribution of T_i is according to the product distribution $R \sim \boldsymbol{z_i}$ over $\mathcal{U}_i$. Let $Q = \bigcup_{i=1}^{n} T_i$. It follows that Q is a random set drawn from the product distribution induced by $\boldsymbol{z}$ over $\mathcal{U}$. Consider the distribution of the set $Q_\downarrow \in \mathcal{N}$. Because of the product distribution of Q it can be see that the distribution of $Q_\downarrow$ is a product distribution on $\mathcal{N}$ where $i \in \mathcal{N}$ appears in $Q_\downarrow$ with probability $x_i = 1 - \prod_{e \in \mathcal{U}_i}(1 - z_e) \leq b$ since $z \in b \cdot \mathcal{P}''$. Note that the algorithm feeds $Q_\downarrow$ to the OCRS π which is (b, c)-selectable. Let $\bar{\pi}$ be the characteristic CRS of π.

Fix a realization S of Q, along with an instantiation of $\mathcal{F}_{\pi, \boldsymbol{x}}$. Notice $e \in T_{\text{ALG}} \cap \mathcal{U}_i$ if and only if $|S \cap \mathcal{U}_i| = \{e\}$ and $i \in \pi(S_\downarrow)$. In fact,

$$T_{\text{ALG}} = \bigcup_{\substack{i \in \pi(S_\downarrow) \\ |S \cap \mathcal{U}_i| = 1}} (S \cap \mathcal{U}_i),$$

by the description of Algorithm 1. By Observation 7, we have $\bar{\pi}(A) \subseteq \pi(A)$ for any $A \subseteq \mathcal{N}$, and thus $\pi'(S) \subseteq T_{\text{ALG}}$. Therefore, by the monotonicity of f, we have $f(T_{\text{ALG}}) \geq f(\pi'(S))$, and by unconditioning

$$\mathbb{E}\left[f\left(T_{\text{ALG}}\right)\right] \geq \mathbb{E}\left[f\left(\pi'(Q)\right)\right].$$

Finally, by Lemma 6 and Remark 3, we have that for any $\boldsymbol{z} \in b \cdot \mathcal{P}''$ and any fixed $\varepsilon > 0$,

$$\mathbb{E}\left[f\left(\pi'(Q)\right)\right] \geq c\left(e^{-b} - \varepsilon\right) \cdot F(\boldsymbol{z}),$$

which yields

$$\mathbb{E}\left[f\left(T_{\text{ALG}}\right)\right] \geq c\left(e^{-b} - \varepsilon\right) \cdot F(\boldsymbol{z}).$$

We are now ready for the main theorem of this section, which follows from Lemmas 8 and 3, and Claim 3.1.

Theorem 8. *Let $(\mathcal{N}, \boldsymbol{D}, \mathcal{C}, f)$ be an instance of the Submodular Prophet Inequality model and let OPT denote the prophet's value. Given a (b, c)-selectable greedy OCRS π for $\mathcal{P}$, for a non-negative monotone submodular function f, $\boldsymbol{z} \in b \cdot \mathcal{P}''$ and fixed $\varepsilon > 0$, Algorithm 1 returns a set T_{ALG} such that*

$$\mathbb{E}\left[f(T_{ALG})\right] \geq c\left(e^{-b} - \varepsilon\right)\left(1 - e^{-b}\right) \cdot \text{OPT}.$$

Next, we provide concrete results for several constraints, given an OCRS for these constraints. First, we summarize known greedy OCRSs for various constraints of interest below.

Lemma 9 (Theorem 1.1 from [20]). *There exist:*

– *For every $b \in [0,1]$, a $(b, 1-b)$-selectable deterministic greedy OCRS for matroid polytopes.*
– *For every $b \in [0,1]$, a (b, e^{-2b})-selectable randomized greedy OCRS for matching polytopes.*
– *For every $b \in [0, \frac{1}{2}]$, a $(b, \frac{1-2b}{2-2b})$-selectable randomized greedy OCRS for the natural relaxation of a knapsack constraint.*

By combining Lemma 9 with Theorem 8, we obtain the following corollary.

Corollary 1. *Let $(\mathcal{N}, \boldsymbol{D}, \mathcal{C}, f)$ be an instance of the Submodular Prophet Inequality model and let OPT denote the prophet's value. For a non-negative monotone submodular function f and any fixed $\varepsilon > 0$, Algorithm 1 returns a set T_{ALG} such that*

$$\mathop{\mathbb{E}}_{\boldsymbol{X},\mathcal{T}}[f(T_{ALG})] \geq (1-b)\left(e^{-b}-\varepsilon\right)\left(1-e^{-b}\right) \cdot OPT, \qquad \forall b \in [0,1],$$

if $\mathcal{C}$ is a matroid constraint

$$\mathop{\mathbb{E}}_{\boldsymbol{X},\mathcal{T}}[f(T_{ALG})] \geq e^{-2b}\left(e^{-b}-\varepsilon\right)\left(1-e^{-b}\right) \cdot OPT, \qquad \forall b \in [0,1],$$

if $\mathcal{C}$ is a matching constraint

$$\mathop{\mathbb{E}}_{\boldsymbol{X},\mathcal{T}}[f(T_{ALG})] \geq \frac{1-2b}{2-2b}\left(e^{-b}-\varepsilon\right)\left(1-e^{-b}\right) \cdot OPT, \qquad \forall b \in \left[0, \frac{1}{2}\right],$$

if $\mathcal{C}$ is a knapsack constraint

where $\mathcal{T} = \{T^1, \ldots, T^n\}$ denotes the set of random sets Algorithm 1 generates.

4.2 Non-Negative Submodular Functions

Below we describe the algorithm for non-negative functions. It is very similar to the monotone case except for a minor change in accepting an element e; in the final step, the algorithm tosses an additional random coin and accepts e with probability 1/2 (see Line 10 in the algorithm). This is inspired by the similar idea in [20] in handling non-monotone functions.

Notice that Lemmas 6 and 7 still hold, as they do not depend on the monotonicity of f. We present the following analogue of Lemma 8 for general submodular functions. The proof of the next lemma relies on an argument similar to that in [20].

Lemma 10. *Given a (b,c)-selectable greedy OCRS π for $\mathcal{P}$, for any $\boldsymbol{z} \in b \cdot \mathcal{P}''$ and fixed $\varepsilon > 0$, Algorithm 2 returns a set $T_{ALG} \subseteq \mathcal{U}$ such that*

$$\mathbb{E}\left[f(T_{ALG})\right] \geq \frac{c\left(e^{-b}-\varepsilon\right)}{4} \cdot F(\boldsymbol{z}).$$

Proof. At every step i, Algorithm 2 draws a random set T_i according to the product distribution on $\mathcal{U}_i$ with probabilities $\boldsymbol{z_i}$, by Lemma 7. Let $Q = \bigcup_{i \in \mathcal{N}} T_i$. Since the realizations between the steps are independent, Q is a random set

ALGORITHM 2: Algorithm for General Non-Negative Submodular Functions

GENERAL ROUNDING$(\mathcal{U}, f, \boldsymbol{D}, \mathcal{C}, \pi, \boldsymbol{z})$

$T_{\text{ALG}} = \emptyset$
for $h \leftarrow 1$ **to** n **do**
  Let X_i be variable that arrives on step h
  Let $e \in \mathcal{U}_i$ be the realization of X_i
  With probability $\frac{\Pr_{R \sim \boldsymbol{z_i}}[R=\{e\}]}{\mathcal{D}_i(e)}$, set $T_i \leftarrow \{e\}$
  Otherwise, set T_i to be a random subset R of $\mathcal{U}_i$, drawn according to $\boldsymbol{z}_i$, conditioned on $|R| \neq 1$
  if $T_i \neq \emptyset$ **then**
    Feed i to OCRS π for $\mathcal{P}$
    if π *accepts* i and $T_i = \{e\}$ **then**
      With probability $\frac{1}{2}$, $T_{\text{ALG}} \leftarrow T_{\text{ALG}} \cup \{e\}$
    end
  end
end
Return T_{ALG}

that follows the product distribution on $\mathcal{U}$ with probabilities $\boldsymbol{z}$. Fix a realization S of Q and an instantiation of $\mathcal{F}_{\pi,\boldsymbol{x}}$. Notice that $e \in T_{\text{ALG}} \cap \mathcal{U}_i$ if and only if $|S \cap \mathcal{U}_i| = 1$, $i \in \pi(S_\downarrow)$ and the coin toss of Line 10 succeeds. In fact, if we denote

$$W = \bigcup_{\substack{i \in \pi(S_\downarrow) \\ |S \cap \mathcal{U}_i| = 1}} (S \cap \mathcal{U}_i),$$

we have that $\mathbb{E}\left[f(T_{\text{ALG}})\right] = \mathbb{E}[f(W(1/2))]$, by the description of Algorithm 2. By Observation 7, we have $\bar{\pi}(A) \subseteq \pi(A)$ for any $A \subseteq \mathcal{N}$, and thus $\pi'(S) \subseteq W$. For ease of notation, we denote $\pi'(S)$ by L. For our fixed choice of S and $\mathcal{F}_{\pi,\boldsymbol{x}}$, L is deterministic. Therefore, we can think of $W(1/2)$ as obtained by first calculating a set $L(1/2)$ in which every element of L appears with probability $1/2$ independently, and then adding to it a random set $\Delta \subseteq \mathcal{U} \setminus L$. The almighty prophet can control the order in which the elements arrive, and thus can make the distribution of Δ depend on $L(1/2)$. However, Δ is guaranteed to contain every element with probability at most $1/2$, for every given realization of $L(1/2)$. Thus,

$$\begin{aligned}
\mathbb{E}\left[f(W(1/2)) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right] &= \mathbb{E}\left[f(L(1/2) \cup \Delta) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right] \\
&= \sum_{B \subseteq L} \Pr\left[L(1/2) = B \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right] \cdot \mathbb{E}\left[f(B \cup \Delta) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right] \\
&\geq \sum_{B \subseteq L} \Pr\left[L(1/2) = B \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right] \cdot \frac{\mathbb{E}\left[f(B) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right]}{2} \\
&= \frac{\mathbb{E}\left[f(L(1/2)) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right]}{2} \\
&= \frac{\mathbb{E}\left[f(L) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right]}{4},
\end{aligned}$$

where the first inequality follows from Lemma 1 since the function $h_B(T) = h(B \cup T)$ is non-negative and submodular for all $B \subseteq \mathcal{U}$, and the second inequality follows from Lemma 2. Taking an expectation over all possible realizations of S and $\mathcal{F}_{\pi,\boldsymbol{x}}$, we obtain

$$\begin{aligned}\mathbb{E}\left[f(W(1/2))\right] &= \mathop{\mathbb{E}}_{S,\mathcal{F}_{\pi,\boldsymbol{x}}}\left[\mathbb{E}\left[f(W(1/2)) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right]\right] \geq \mathop{\mathbb{E}}_{S,\mathcal{F}_{\pi,\boldsymbol{x}}}\left[\frac{\mathbb{E}\left[f(L) \mid S, \mathcal{F}_{\pi,\boldsymbol{x}}\right]}{4}\right] \\ &= \frac{\mathbb{E}\left[f(L)\right]}{4}.\end{aligned}$$

Finally, by Lemma 6 and Remark 3, we have that for any $\boldsymbol{z} \in b \cdot \mathcal{P}''$ and any fixed $\varepsilon > 0$,

$$\frac{\mathbb{E}\left[f(L)\right]}{4} \geq \frac{c\left(e^{-b} - \varepsilon\right)}{4} \cdot F(\boldsymbol{z}),$$

which implies

$$\mathbb{E}\left[f(T_{\mathrm{ALG}})\right] = \mathbb{E}\left[f(W(1/2))\right] \geq \frac{c\left(e^{-b} - \varepsilon\right)}{4} \cdot F(\boldsymbol{z}).$$

We are now ready to proceed with the main result for general submodular functions, which follows from Lemma 10, Theorem 3, and Claim 3.1.

Theorem 9. *Let $(\mathcal{N}, \boldsymbol{D}, \mathcal{C}, f)$ be an instance of the Submodular Prophet Inequality model and let OPT denote the prophet's value. Given a (b,c)-selectable greedy OCRS π for $\mathcal{P}$, for a non-negative submodular function f, $\boldsymbol{z} \in b \cdot \mathcal{P}''$ and fixed $\varepsilon > 0$, Algorithm 2 returns a set T_{ALG} such that*

$$\mathbb{E}\left[f(T_{ALG})\right] \geq \frac{c\left(e^{-b} - \varepsilon\right)}{4} \cdot \left(1 - e^{-b} - \varepsilon\right) \cdot OPT.$$

By combining Lemma 9 with Theorem 9, we obtain the following corollary.

Corollary 2. *Let $(\mathcal{N}, \boldsymbol{D}, \mathcal{C}, f)$ be an instance of the Submodular Prophet Inequality model and let OPT denote the prophet's value. For a non-negative submodular function f and any fixed $\varepsilon > 0$, Algorithm 2 returns a set T_{ALG} such that*

$$\mathbb{E}[f(T_{ALG})] \geq \frac{(1-b)\left(e^{-b} - \varepsilon\right)}{4} \cdot \left(1 - e^{-b} - \varepsilon\right) \cdot OPT, \qquad \forall b \in [0,1],$$

if $\mathcal{C}$ is a matroid constraint

$$\mathbb{E}[f(T_{ALG})] \geq \frac{e^{-2b}\left(e^{-b} - \varepsilon\right)}{4} \cdot \left(1 - e^{-b} - \varepsilon\right) \cdot OPT, \qquad \forall b \in [0,1],$$

if $\mathcal{C}$ is a matching constraint

$$\mathbb{E}[f(T_{ALG})] \geq \frac{(1-2b)\left(e^{-b} - \varepsilon\right)}{8 - 8b} \cdot \left(1 - e^{-b} - \varepsilon\right) \cdot OPT, \qquad \forall b \in \left[0, \frac{1}{2}\right],$$

if $\mathcal{C}$ is a knapsack constraint

where $\mathcal{T} = \{T^1, \ldots, T^n\}$ denotes the set of random sets Algorithm 1 generates.

5 Conclusion

We presented a general framework for submodular prophet inequalities in the model of [31] via greedy Online Contention Resolution Schemes and correlation gaps. The framework yields substantially improved constant factor competitive ratios for both monotone and general submodular functions, and can be implemented in polynomial time for many classes of interesting constraints. The framework resolves the open question posed in [28] regarding the model of [31].

Along the way, we strengthened the notion of correlation gap for non-negative submodular functions introduced in [31], and provided a fine-grained variant of the standard correlation gap. For both cases, our bounds are cleaner and tighter. Moreover, we presented a refined analysis of the Measured Continuous Greedy algorithm for polytopes with small coordinates and general non-negative submodular functions, showing that, for these cases, it yields a bound that matches the bound of Continuous Greedy for the monotone case.

An interesting open question is whether our fine-grained correlation gap for general non-negative submodular functions can be made tight. It is tempting to conjecture that the lower bound on the gap shown in Theorem 11 is tight for all values of p. We leave this as an interesting open problem to resolve.

A natural question for the prophet inequality setting is whether one can obtain better prophet inequalities when the arrival order of the random variables is chosen uniformly at random or even chosen by the algorithm. In [18], the authors introduce the *prophet secretary* model, combining the best of both the secretary and prophet inequality worlds. There has been much work on this model and we refer to [12,17] for several interesting results in this and related models. We can consider the *Submodular Prophet Secretary (SPS)* problem as a generalization of the standard prophet secretary problem. We note that one can obtain improved guarantees in the SPS problem by using a Random Order CRS instead of an OCRS, since our results utilize the given OCRS in a black-box manner.

In subsequent work, Qiu and Singla [29] obtained improved bounds for SPI via the use of *submodular dominance*. In particular, they obtain an optimal $1 - 1/e$ bound for uniform matroids in the monotone case. Obtaining optimal SPIs under different constraints remains an interesting open problem.

Acknowledgements The authors thank Sahil Singla for clarifications on [31], and Jan Vondrák for helpful discussions.

References

1. Adamczyk, M., Sviridenko, M., Ward, J.: Submodular stochastic probing on matroids. Mathematics of Operations Research **41**(3), 1022–1038 (2016). https://doi.org/10.1287/moor.2015.0766, https://doi.org/10.1287/moor.2015.0766
2. Adamczyk, M., Wlodarczyk, M.: Random order contention resolution schemes. In: 59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018. pp. 790–801 (2018).

https://doi.org/10.1109/FOCS.2018.00080, https://doi.org/10.1109/FOCS.2018.00080

3. Agrawal, S., Ding, Y., Saberi, A., Ye, Y.: Price of correlations in stochastic optimization. Operations Research **60**(1), 150–162 (2012), http://www.jstor.org/stable/41476345
4. Alaei, S.: Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. SIAM Journal on Computing **43**(2), 930–972 (2014). https://doi.org/10.1137/120878422, https://doi.org/10.1137/120878422
5. Assadi, S., Kesselheim, T., Singla, S.: Improved Truthful Mechanisms for Subadditive Combinatorial Auctions: Breaking the Logarithmic Barrier, pp. 653–661. Association for Computing Machinery (2021). https://doi.org/10.1137/1.9781611976465.40, https://epubs.siam.org/doi/abs/10.1137/1.9781611976465.40
6. Buchbinder, N., Feldman, M.: Submodular functions maximization problems. In: Gonzalez, T.F. (ed.) Handbook of Approximation Algorithms and Metaheuristics: Methologies and Traditional Applications, Volume 1. CRC Press, 2nd edn. (2018). https://doi.org/10.1201/9781351236423, https://doi.org/10.1201/9781351236423
7. Buchbinder, N., Feldman, M., Naor, J.S., Schwartz, R.: Submodular maximization with cardinality constraints. In: Proceedings of the Twenty-fifth Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 1433–1452. SODA '14, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2014), http://dl.acm.org/citation.cfm?id=2634074.2634180
8. Calinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a monotone submodular function subject to a matroid constraint. SIAM Journal on Computing **40**(6), 1740–1766 (2011). https://doi.org/10.1137/080733991, https://doi.org/10.1137/080733991
9. Chawla, S., Hartline, J.D., Malec, D.L., Sivan, B.: Multi-parameter mechanism design and sequential posted pricing. In: Proceedings of the Forty-Second ACM Symposium on Theory of Computing. p. 311–320. STOC '10, Association for Computing Machinery, New York, NY, USA (2010). https://doi.org/10.1145/1806689.1806733, https://doi.org/10.1145/1806689.1806733
10. Chawla, S., Sivan, B.: Bayesian algorithmic mechanism design. SIGecom Exch. **13**(1), 5–49 (Nov 2014). https://doi.org/10.1145/2692375.2692378, https://doi.org/10.1145/2692375.2692378
11. Chekuri, C., Vondrák, J., Zenklusen, R.: Submodular function maximization via the multilinear relaxation and contention resolution schemes. In: Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing. pp. 783–792. STOC '11, ACM, New York, NY, USA (2011). https://doi.org/10.1145/1993636.1993740, http://doi.acm.org/10.1145/1993636.1993740
12. Correa, J.R., Foncea, P., Hoeksma, R., Oosterwijk, T., Vredeveld, T.: Recent developments in prophet inequalities. SIGecom Exch. **17**(1), 61–70 (2018). https://doi.org/10.1145/3331033.3331039, https://doi.org/10.1145/3331033.3331039
13. Dinitz, M.: Recent advances on the matroid secretary problem. SIGACT News **44**(2), 126–142 (jun 2013). https://doi.org/10.1145/2491533.2491557, https://doi.org/10.1145/2491533.2491557
14. Dütting, P., Feldman, M., Kesselheim, T., Lucier, B.: Prophet inequalities made easy: Stochastic optimization by pricing nonstochastic inputs. SIAM Journal on Computing **49**(3), 540–582 (2020). https://doi.org/10.1137/20M1323850, https://doi.org/10.1137/20M1323850

15. Dütting, P., Kesselheim, T., Lucier, B.: An o (log log m) prophet inequality for subadditive combinatorial auctions. SIAM Journal on Computing **0**(0), FOCS20–239–FOCS20–275 (0). https://doi.org/10.1137/20M1382799, https://doi.org/10.1137/20M1382799
16. Dynkin, E.B.: The optimum choice of the instant for stopping a Markov process. Sov. Math., Dokl. **4**, 627–629 (1963)
17. Ehsani, S., Hajiaghayi, M., Kesselheim, T., Singla, S.: Prophet secretary for combinatorial auctions and matroids. In: Czumaj, A. (ed.) Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018. pp. 700–714. SIAM (2018). https://doi.org/10.1137/1.9781611975031.46, https://doi.org/10.1137/1.9781611975031.46
18. Esfandiari, H., Hajiaghayi, M., Liaghat, V., Monemizadeh, M.: Prophet secretary. SIAM Journal on Discrete Mathematics **31**(3), 1685–1701 (2017). https://doi.org/10.1137/15M1029394, https://doi.org/10.1137/15M1029394
19. Feldman, M., Naor, J.S., Schwartz, R.: A unified continuous greedy algorithm for submodular maximization. In: Proceedings of the 2011 IEEE 52Nd Annual Symposium on Foundations of Computer Science. pp. 570–579. FOCS '11, IEEE Computer Society, Washington, DC, USA (2011). https://doi.org/10.1109/FOCS.2011.46, http://dx.doi.org/10.1109/FOCS.2011.46
20. Feldman, M., Svensson, O., Zenklusen, R.: Online contention resolution schemes with applications to bayesian selection problems. SIAM J. Comput. **50**(2), 255–300 (2021). https://doi.org/10.1137/18M1226130, https://doi.org/10.1137/18M1226130
21. Gupta, A., Singla, S.: Random-Order Models, p. 234–258. Cambridge University Press (2021). https://doi.org/10.1017/9781108637435.015
22. Hajiaghayi, M.T., Kleinberg, R., Sandholm, T.: Automated online mechanism design and prophet inequalities. In: Proceedings of the 22Nd National Conference on Artificial Intelligence - Volume 1. pp. 58–65. AAAI'07, AAAI Press (2007), http://dl.acm.org/citation.cfm?id=1619645.1619656
23. Hartline, J.D.: Bayesian mechanism design. Foundations and Trends® in Theoretical Computer Science **8**(3), 143–263 (2013). https://doi.org/10.1561/0400000045, http://dx.doi.org/10.1561/0400000045
24. Hill, T.P., Kertz, R.P.: A survey of prophet inequalities in optimal stopping theory. Contemp. Math **125**, 191–207 (1992)
25. Kalhan, S.: A study on matroid prophet inequalities (May 2016), senior thesis, University of Illinois at Urbana-Champaign
26. Kleinberg, R., Weinberg, S.M.: Matroid prophet inequalities and applications to multi-dimensional mechanism design. Games Econ. Behav. **113**, 97–115 (2019). https://doi.org/10.1016/j.geb.2014.11.002, https://doi.org/10.1016/j.geb.2014.11.002
27. Krengel, U., Sucheston, L.: Semiamarts and finite values. Bull. Amer. Math. Soc. **83**(4), 745–747 (07 1977), https://projecteuclid.org:443/euclid.bams/1183538915
28. Lucier, B.: An economic view of prophet inequalities. SIGecom Exch. **16**(1), 24–47 (Sep 2017). https://doi.org/10.1145/3144722.3144725, https://doi.org/10.1145/3144722.3144725
29. Qiu, F., Singla, S.: Submodular dominance and applications. In: Chakrabarti, A., Swamy, C. (eds.) Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2022, September 19-21, 2022, University of Illinois, Urbana-Champaign, USA (Virtual Conference). LIPIcs, vol. 245, pp. 44:1–44:21. Schloss Dagstuhl - Leibniz-Zentrum für

Informatik (2022). https://doi.org/10.4230/LIPIcs.APPROX/RANDOM.2022.44, https://doi.org/10.4230/LIPIcs.APPROX/RANDOM.2022.44
30. Rubinstein, A.: Beyond matroids: Secretary problem and prophet inequality with general constraints. In: Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing. p. 324–332. STOC '16, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2897518.2897540, https://doi.org/10.1145/2897518.2897540
31. Rubinstein, A., Singla, S.: Combinatorial prophet inequalities. In: Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 1671–1687. SIAM (2017), longer ArXiv version is at http://arxiv.org/abs/1611.00665
32. Singla, S.: Combinatorial Optimization Under Uncertainty: Probing and Stopping-Time Algorithms. Ph.D. thesis, CMU (2018), http://reports-archive.adm.cs.cmu.edu/anon/2018/CMU-CS-18-111.pdf
33. Vondrák, J.: Submodularity in combinatorial optimization. Ph.D. thesis, Univerzita Karlova, Matematicko-fyzikální fakulta (2007)
34. Yan, Q.: Mechanism design via correlation gap. In: Randall, D. (ed.) Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011. pp. 710–719. SIAM (2011). https://doi.org/10.1137/1.9781611973082.56, https://doi.org/10.1137/1.9781611973082.56

A Correlation Gap for Non-Negative Submodular Functions

In this section we prove Theorems 1 and 3 on the correlation gap for non-negative submodular functions.

The correlation gap for monotone functions [8,33] used a continuous time argument by relating $F(\boldsymbol{x})$ and $f^+(\boldsymbol{x})$ via another continuous extension f^* and this was the same approach followed in [31]. We take a different approach. For the exact correlation gap in Theorem 1 we build on a proof for the monotone case from [11] which is less well-known; we adapt their proof for the non-negative case via the parameter p. The proof of Theorem 1 is qualitatively different from that of Theorem 3.

We split the proof into two parts, the upper bound and the lower bound, state them separately and give their proof. Before we begin, we present two lemmas that are useful in our analysis.

Lemma 11 (Lemma 4.3 from [33]). *Let $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be a submodular function, let $A_1, A_2, \ldots, A_k \subseteq \mathcal{N}$ be k (not necessarily disjoint) sets and let $A_1(p_1), A_2(p_2), \ldots, A_k(p_k)$ their independently sampled subsets, where each element of A_i appears in $A_i(p_i)$ with probability p_i, for all $1 \leq i \leq k$. Then*

$$\mathbb{E}\left[f\left(\bigcup_{i=1}^{k} A_i(p_i)\right)\right] \geq \sum_{I \subseteq [k]} \left(\prod_{j \in I} p_j \prod_{j \notin I} (1 - p_j) f\left(\bigcup_{j \in I} A_j\right)\right).$$

Lemma 12. *Let* $a_1 \geq \cdots \geq a_m \in \mathbb{R}_{\geq 0}$*, and* $q_1, \ldots, q_m \in [0,1]$ *such that* $\sum_{k=1}^{m} q_k = 1$*. Then*

$$\sum_{k=1}^{m} q_k \, a_k \prod_{j=1}^{k-1} (1 - q_j) \geq \left(1 - \frac{1}{e}\right) \cdot \sum_{j=1}^{m} q_j a_j .$$

Proof. Since the above inequality is linear in the parameters a_i, it suffices to prove it for the special case $a_1 = a_2 = \cdots = a_r = 1$ and $a_{r+1} = \cdots = a_m = 0$. (A general decreasing sequence of a_j can be obtained as a positive linear combination of such special cases). Hence, it remains to prove

$$\sum_{k=1}^{r} q_k \prod_{j=1}^{k-1} (1 - q_j) \geq \left(1 - \frac{1}{e}\right) \cdot \sum_{j=1}^{r} q_j .$$

We start from the left-hand side, which we expand to

$$\sum_{k=1}^{r} q_k \prod_{j=1}^{k-1} (1 - q_j) = 1 - \prod_{k=1}^{r} (1 - q_k) \geq 1 - \left(1 - \frac{1}{r} \sum_{k=1}^{r} q_k\right)^r ,$$

where the inequality follows from the arithmetic-geometric mean inequality. Finally, we use the concavity of $\phi_r(x) \coloneqq 1 - \left(1 - \frac{x}{r}\right)^r$, and the fact that $\phi_r(0) = 0$, to get

$$\phi_r(x) \geq \phi_r(1) \cdot x = \left(1 - \left(1 - \frac{1}{r}\right)^r\right) \cdot x$$

for $x \in [0,1]$. Since $\left(1 - \left(1 - \frac{1}{r}\right)^r\right) \geq \left(1 - \frac{1}{e}\right)$ for all r, we get

$$\phi_r(x) \geq \left(1 - \frac{1}{e}\right) \cdot x .$$

which implies that

$$\phi_r\left(\sum_{k=1}^{r} q_k\right) = 1 - \left(1 - \frac{1}{r} \sum_{k=1}^{r} q_k\right)^r \geq \left(1 - \frac{1}{e}\right) \cdot \sum_{k=1}^{r} q_k .$$

A.1 Upper bound

The proof of this upper bound is inspired by the proof in [11] for the monotone case, which is different from the earlier one in [33].

Theorem 10. *Let* $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ *be a non-negative submodular function, where* $n = |\mathcal{N}|$*. Let* $\boldsymbol{x} \in [0,1]^n$*, such that* $\boldsymbol{x} \leq p \cdot \mathbf{1}_{\mathcal{N}}$ *for some* $p \in [0,1]$*. Then,*

$$F(\boldsymbol{x}) \geq (1 - p) \left(1 - \frac{1}{e}\right) f^{+}(\boldsymbol{x}).$$

Proof. Consider a basic feasible solution $(q_j, A_j)_{j\in[m]}$ to the linear program that defines $f^+(\boldsymbol{x})$. In other words, $f^+(\boldsymbol{x}) = \sum_{j=1}^m q_j f(A_j)$, where $\sum_{j=1}^m q_j = 1$, $\sum_{j:i\in A_j} q_j = x_i$, for all i, and $q_j \geq 0$ for all j. Notice that, since we chose a basic feasible solution and the LP that defines $f^+(\boldsymbol{x})$ has only $n+1$ constraints, apart from the non-negativity constraints, we have $m \leq n+1$.

Next, consider the following process to generate a subset of elements. For each $j \in [m]$ sample independently each element of A_j with probability q_j. An element $i \in \mathcal{N}$ is not selected with probability equal to $\prod_{j:i\in A_j}(1-q_j)$, thus, i is selected with probability equal to $1-\prod_{j:i\in A_j}(1-q_j)$. Notice that we can assume without loss of generality that $q_j \neq 1$ for all j; if $q_j = 1$ for some j then that implies that $x_i = 1$ for every element $i \in A_j$, and $q_{j'} = 0$ for all $j' \neq j$, which then leads us to $F(\boldsymbol{x}) = f(A_j) = f^+(\boldsymbol{x})$.

However, we want to make each element i to be selected with probability exactly equal to $x_i = \sum_{j:i\in A_j} q_j$. To do this, we simply need to sample again each element i with probability r_i, where

$$1-(1-r_i)\cdot\prod_{j:i\in A_j}(1-q_j) = \sum_{j:i\in A_j} q_j. \tag{5}$$

It is easy to see that $0 \leq r_i \leq x_i \leq p$.

Consider the sampling scheme described above, and let R denote a random set created via this sampling scheme. Notice that in our sampling scheme, each element i is chosen independently with probability x_i, which implies that $\mathbb{E}_R[f(R)] = F(\boldsymbol{x})$.

We consider $m+n$ sets $B_1, B_2, \ldots, B_{m+n}$ where $B_j = A_j$ for $1 \leq j \leq m$, and $B_{m+i} = \{i\}$ for $1 \leq i \leq n$. Let $\mathcal{J}$ denote a random subset of $[m+n]$ obtained by including each $j \in \{1,2,\ldots,m\}$ independently with probability q_j and each $i \in \{m+1, m+2, \ldots, m+n\}$ independently with probability r_i. Also, let $R' \subseteq \mathcal{N}$ denote a random set where

$$R' = \bigcup_{j\in\mathcal{J}} B_j.$$

The next claim is based on the submodularity of f.

Claim.

$$F(\boldsymbol{x}) \geq \mathop{\mathbb{E}}_{\mathcal{J}}\left[f(R')\right].$$

Proof. Since $F(\boldsymbol{x}) = \mathbb{E}_R[f(R)]$, it suffices to show that

$$\mathop{\mathbb{E}}_{R}\left[f(R)\right] \geq \mathop{\mathbb{E}}_{\mathcal{J}}\left[f(R')\right].$$

We apply Lemma 11 for $k = m+n$, $A_j = B_j$ for $1 \leq j \leq m+n$, $p_j = q_j$ for $1 \leq j \leq m$, and $p_{m+i} = r_i$ for $1 \leq i \leq n$. Notice that

$$\mathop{\mathbb{E}}_{R}\left[f(R)\right] = \mathbb{E}\left[f\left(\bigcup_{i=1}^{k} B_i(p_i)\right)\right],$$

while

$$\mathop{\mathbb{E}}_{\mathcal{J}}[f(R')] = \sum_{I\subseteq[k]} \left(\prod_{j\in I} p_j \prod_{j\notin I} (1-p_j) f\left(\bigcup_{j\in I} B_j \right) \right),$$

and thus, by Lemma 11, we get

$$\mathop{\mathbb{E}}_{R}[f(R)] \geq \mathop{\mathbb{E}}_{\mathcal{J}}[f(R')] .$$

Claim.

$$\mathop{\mathbb{E}}_{\mathcal{J}}[f(R')] \geq (1-p)\left(1-\frac{1}{e}\right)\cdot f^{+}(\boldsymbol{x}).$$

Proof. Assume, without loss of generality, that $f(A_1) \geq \cdots \geq f(A_m)$. We analyze $\mathbb{E}[f(R')]$ by conditioning on the minimum index j that belongs to $\mathcal{J}$. For $k \in [m]$, let

$$J_k = \{ I \subseteq [m+n] \mid k \in I \text{ and } \ell \notin I, \forall \ell < k \} .$$

Furthermore, for $k \in [m]$ define the set function $g_k : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ where $g_k(S) = f(B_k \cup S)$ for all $S \subseteq \mathcal{N}$. It is easy to verify that g_k is non-negative and submodular because f is non-negative and submodular. $\mathcal{J} \in J_k$ implies that $B_k \subseteq R'$, hence,

$$\begin{aligned}\mathop{\mathbb{E}}_{\mathcal{J}}[f(R') \mid \mathcal{J} \in J_k] &= \mathop{\mathbb{E}}_{\mathcal{J}}[f(B_k \cup (R' \setminus B_k)) \mid \mathcal{J} \in J_k] \\ &= \mathop{\mathbb{E}}_{\mathcal{J}}[g_k(R' \setminus B_k) \mid \mathcal{J} \in J_k].\end{aligned}$$

For any fixed $i \in \mathcal{N}$ we analyze the probability that $i \in R' \setminus B_k$ conditioned on $\mathcal{J} \in J_k$. Using independence of the choice of each index in $\mathcal{J}$ we obtain the following.

$$\begin{aligned}\Pr_{\mathcal{J}}[i \in (R' \setminus B_k) \mid \mathcal{J} \in J_k] &= 1 - (1-r_i) \prod_{j: i\in A_j, k<j\leq m} (1-q_j) \\ &\leq 1 - (1-r_i) \prod_{j: i\in A_j, j\in[m]} (1-q_j) \\ &= x_i \leq p.\end{aligned}$$

Thus, applying Lemma 1 to g_k yields

$$\mathop{\mathbb{E}}_{\mathcal{J}}[g_k(R' \setminus B_k) \mid \mathcal{J} \in J_k] \geq (1-p) g_k(\emptyset) = (1-p) f(B_k).$$

Combining the above,

$$\mathop{\mathbb{E}}_{\mathcal{J}}[f(R') \mid \mathcal{J} \in J_k] \geq (1-p)\cdot f(B_k). \tag{6}$$

Also notice that

$$\Pr_{\mathcal{J}}[\mathcal{J} \in J_k] = \Pr_{\mathcal{J}}[k \in \mathcal{J}] \cdot \prod_{j=1}^{k-1} \left(1 - \Pr_{\mathcal{J}}[j \in \mathcal{J}]\right) = q_k \cdot \prod_{j=1}^{k-1} (1-q_j). \tag{7}$$

Therefore,

$$
\begin{aligned}
\mathop{\mathbb{E}}_{\mathcal{J}}[f(R')] &= \sum_{k=1}^{m} \Pr_{\mathcal{J}}[\mathcal{J} \in J_k] \cdot \mathop{\mathbb{E}}_{\mathcal{J}}[f(R') \mid \mathcal{J} \in J_k] \\
&\quad + \Pr_{\mathcal{J}}[\mathcal{J} \cap [m] = \emptyset] \cdot \mathop{\mathbb{E}}_{\mathcal{J}}[f(R') \mid \mathcal{J} \cap [m] = \emptyset] \\
&\geq \sum_{j=1}^{m} \Pr_{\mathcal{J}}[\mathcal{J} \in J_k] \cdot \mathop{\mathbb{E}}_{\mathcal{J}}[f(R') \mid \mathcal{J} \in J_k] \\
&\geq \sum_{k=1}^{m} \Pr_{\mathcal{J}}[\mathcal{J} \in J_k] \cdot (1-p) \cdot f(B_k) \\
&= (1-p) \sum_{k=1}^{m} q_k f(B_k) \prod_{j=1}^{k-1} (1-q_j), \qquad (8)
\end{aligned}
$$

where the first inequality follows from the non-negativity of f, the second inequality follows from (6) and the last equality from (7). However, for $1 \leq k \leq m$, we have $B_k = A_k$, and thus

$$\mathop{\mathbb{E}}_{\mathcal{J}}[f(R')] \geq (1-p) \sum_{k=1}^{m} q_k f(A_k) \prod_{j=1}^{k-1} (1-q_j).$$

Finally, utilizing Lemma 12 for $a_k = f(A_k)$, we get that

$$\sum_{k=1}^{m} q_k f(A_k) \prod_{j=1}^{k-1} (1-q_j) \geq \left(1 - \frac{1}{e}\right) \cdot \sum_{j=1}^{m} q_j f(A_j) = \left(1 - \frac{1}{e}\right) \cdot f^+(\boldsymbol{x}). \qquad (9)$$

Combining (8) and (9),

$$\mathop{\mathbb{E}}_{\mathcal{J}}[f(R')] \geq (1-p) \left(1 - \frac{1}{e}\right) \cdot f^+(\boldsymbol{x}).$$

Finally, combining Claims A.1 and A.1, we obtain

$$F(\boldsymbol{x}) \geq (1-p) \left(1 - \frac{1}{e}\right) \cdot f^+(\boldsymbol{x}),$$

which completes the proof.

A.2 Lower bound

A simple example on $n = 2$ shows that $F(\boldsymbol{x}) \leq (1-p) f^+(\boldsymbol{x})$; the function is the cut function of a directed graph on two vertices. For monotone functions, a simple coverage example shows that $F(\boldsymbol{x}) \leq (1 - 1/e) f^+(\boldsymbol{x})$. We combine and generalize these two examples to create an instance for non-monotone functions and obtain the following theorem.

Theorem 11. *There exists a non-negative submodular function $f : 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ such that, for any $0 \leq p \leq 1$, there exists an $\boldsymbol{x} \in [0,1]^n$ where $\|\boldsymbol{x}\|_\infty \leq p$ and*

$$F(\boldsymbol{x}) \leq \left(1 - e^{-(1-p)}\right) f^+(\boldsymbol{x}).$$

Proof. Consider the following graph $G = (V, E)$, where $V = \{u_1, \ldots, u_n, v\}$, and $E = \{(u_i, v) \mid 1 \leq i \leq n\}$. Let $x_{u_i} = \frac{1-p}{n}$ for all $i \in \{1, \ldots, n\}$ and $x_v = p$. We define a function $f : 2^V \to \mathbb{R}_{\geq 0}$ as follows

$$f(S) = \begin{cases} 1 & \text{if } v \notin S \text{ and } S \neq \emptyset, \\ 0 & \text{otherwise.} \end{cases}$$

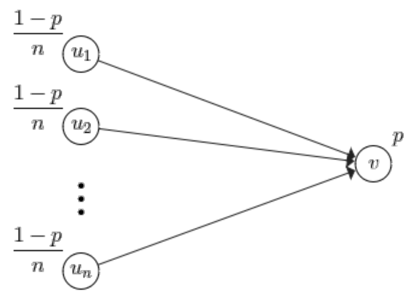

Fig. 1: Graph G which yields the desired lower bound.

It is easy to see that f is submodular. Notice that

$$f^+(\boldsymbol{x}) = 1 - p,$$

as the coefficients that maximize $\sum_S a_S f(S)$ subject to the constraints are $a_{\{v\}} = p$, $a_{\{u_i\}} = \frac{1-p}{n}$ for all $i \in \{1, \ldots, n\}$ and $a_S = 0$, for $|S| \neq 1$. In other words, $a_{\{u\}} = x_u$ for all $u \in V$, and $a_S = 0$, if $|S| \neq 1$.

Next, notice that, if $R(\boldsymbol{x}) \subseteq V$ is a random set, where each element $u \in V$ is sampled with probability x_u, then $f(R(\boldsymbol{x})) = 1$ if and only if v is not selected in $R(\boldsymbol{x})$, but at least one element of $V \setminus \{v\}$ is selected. Therefore,

$$F(\boldsymbol{x}) = \mathbb{E}[f(R(\boldsymbol{x}))] = (1-p) \cdot \left(1 - \left(1 - \frac{1-p}{n}\right)^n\right),$$

which implies that

$$\frac{F(\boldsymbol{x})}{f^+(\boldsymbol{x})} = \frac{(1-p) \cdot \left(1 - \left(1 - \frac{1-p}{n}\right)^n\right)}{1-p} = 1 - \left(1 - \frac{1-p}{n}\right)^n.$$

As $n \to \infty$, we get

$$\lim_{n\to\infty} 1 - \left(1 - \frac{1-p}{n}\right)^n = 1 - e^{-(1-p)}.$$

We conclude that, for any $0 \le p < 1$, when $x_i \le p$ for all i,

$$F(\boldsymbol{x}) \le \left(1 - e^{-(1-p)}\right) f^+(\boldsymbol{x}).$$

B Reduction to Small Probabilities

Observation 12 (Observation 5) *Let $I = (\mathcal{N}, \mathcal{U}, \boldsymbol{D}, \boldsymbol{X}, f, \mathcal{C})$ be an instance of the Submodular Prophet Inequality problem. For every fixed $\varepsilon > 0$, there is a reduction of I to another instance $I' = (\mathcal{N}, \mathcal{U}', \boldsymbol{D'}, \boldsymbol{Y}, g, \mathcal{C})$ of the SPI problem such that that (i) for all $e \in \mathcal{U}'$, $\boldsymbol{D'}(e) \le \varepsilon$ and (ii) there exists an α-competitive algorithm for I if and only if there exists an α-competitive algorithm for I'.*

Proof Sketch. Consider the original instance I and recall that each $\mathcal{D}_i$ is a probability distribution over $\mathcal{U}_i$. Our goal is to ensure that $\mathcal{D}_i(e) \le \varepsilon$ for every $e \in \mathcal{U}_i$. Suppose there is an element e such that $\mathcal{D}_i(e) > \varepsilon$. We obtain a new instance I' as follows. We replace $e \in \mathcal{U}_i$ by $h = \lceil 1/\varepsilon \rceil$ "copies" $e_1, e_2, \ldots, e_h$; let S_e denote this set of copies. Let $\mathcal{U}'_i$ be the new set of elements. We obtain a probability distribution $\mathcal{D}'_i : \mathcal{U}'_i \to [0,1]$ as follows. If $e' \in \mathcal{U}_i$ such that $e' \neq e$ then $\mathcal{D}'_i(e') = \mathcal{D}_i(e')$ (nothing changes for e'). For each copy e_j of e we set $\mathcal{D}'_i(e_j) = \mathcal{D}_i(e)/h$ and by our choice of h we have $\mathcal{D}'_i(e_j) \le 1/h \le \varepsilon$, for all $e_j \in S_e$. Thus, $\sum_{j=1}^{h} \mathcal{D}'_i(e_j) = \mathcal{D}_i(e)$. Since we replaced e by h copies of it, the ground set $\mathcal{U}$ changes to $\mathcal{U}'$ and we now define a new submodular function $g : \mathcal{U}' \to \mathbb{R}_+$ that is derived from f. The function g treats the copies of e as a "single" element and hence mimics f. More formally, for any $A \subseteq \mathcal{U}'$: $g(A) = f(A)$ if $A \cap S_e = \emptyset$, else $g(A) = f((A \setminus S_e) \cup \{e\})$. It is easy to verify that if f is non-negative and submodular, then g is also non-negative and submodular, and also inherits monotonicity from f. Let I' be the resulting modified instance. We observe that in I', the probability of an element from S_e being chosen is precisely equal to $\mathcal{D}_i(e)$ and hence the copies of e act as proxies for e and the submodular function g ensures that every copy behaves the same as e in f. Note that we crucially relied on the power of submodularity in this reduction. One can apply this reduction repeatedly to reduce all realization probabilities to at most ε. One also notices that the reduction is computationally efficient as a function of ε. For any fixed ε, the size of I' is at most $O(1/\varepsilon)$ times the size of I and a value oracle for f can be used to efficiently and easily obtain a value oracle for the new submodular function g.