Optimistic Greedy Strategies for Partially Known Submodular Functions

Andrew Downie, Bahman Gharesifard, and Stephen L. Smith

***Abstract*— We consider a class of submodular maximization problems in which decision-makers have limited access to the objective function. We explore scenarios where the decision-maker has access to only k-wise information about the objective function; that is, they can evaluate the submodular objective function on sets of size at most k. We begin with a negative result that no algorithm using only k-wise information can guarantee performance better than k/n, where n is the size of the selected set. We present an algorithm that utilizes only k-wise information about the function and characterizes its performance relative to the optimal, which depends on a new notion of curvature of the submodular function. Finally, we present an experiment in maximum entropy sampling that highlight the approximation performance of our proposed algorithm.**

I. Introduction

Submodular maximization has recently generated interest in many decision-making problems, as it can provide strong performance guarantees for computationally difficult problems. Submodular functions are set functions that exhibit the property of diminishing returns. Submodular optimization is a well-studied subject, as these functions model many real-world problems in controls [1], [2], robotics [3], [4], [5], data processing [6], [7] and machine learning [8], [9].

Recently, information constraints on decision makers are explored in distributed submodular maximization, where they collaboratively maximize a submodular function. Each agent has access to their own set of actions and can observe a limited number of decisions made by other agents [10], [11], [12], [13], [14]. In contrast, we consider the case where each decision-maker has limited access to the function f itself.

Submodular functions can also effectively model objectives in many sensing applications. Some example applications include sensor selection for Kalman filtering [15], [16], [17], surveillance [18] and target tracking [19]. Another common sensing application is environment monitoring and event detection [20]. In these scenarios, the goal is to select a set sensors to maximize the information gained from observations made by the selected set. In this work, we test our algorithm on the maximum entropy sampling problem, similar to the one found in [9].

One practical difficulty in implementing algorithms for submodular maximization in complex settings is that the required function evaluations are computationally expensive. This can be attributed to the large-scale characteristics of the system [21], application-specific constraints such as communication constraints [22], or the type of data the objective function is evaluating [20]. In its most common form, the submodular function is treated as a *value oracle*, which is repeatedly queried by a greedy strategy to maximize the objective function. Therefore, it is inherently assumed that one can evaluate the function for sets of any size. In practice, however, it may only be possible to evaluate the functions on smaller set sizes due to computation cost or limitations imposed. Consider the setting where a company is selecting locations for several new retail stores. The total revenue received by a set of store locations can be modelled as a submodular function: As more stores are added, the marginal benefit of adding a new store is reduced. In the classical greedy algorithm for submodular maximization, we assume we have access to a value oracle to evaluate subsets of store locations. Armed with this oracle, we iteratively add a new store s_k to the existing set $\{s_1, \dots, s_{k-1}\}$ by selecting the location s_k that maximizes the marginal benefit $f(s_1, \dots, s_{k-1}, s) - f(s_1, \dots, s_{k-1})$. To evaluate this quantity, the oracle must accurately model the revenue of k stores, which can be challenging in practice due to their complex interactions: for example, s_k may reduce the revenue at some s_i, which then may affect some other store's revenue.

Motivated by the lack of access to the full value oracle in practical settings, in this paper, we seek to determine how well we can approximate the maximum value of a submodular function when we have access to a limited set of function values. Suppose we can access function values for single elements $f(s_i)$ and for pairs of elements $f(s_i, s_j)$. We refer to this as *pairwise information*. In the motivating example, this corresponds to knowing the total revenue for a single store and the total revenue for any two stores together and nothing more. Note that this restriction on information is severe. A submodular function on a base set of N elements can be represented as a look-up table with 2^N values. If only singleton and pairwise information are available, this means we have access to only $N(N+1)/2$ values. In this work we focus on the case of k-wise information, where we can evaluate any set of size at most k. Greedy algorithms have also been applied to submodular maximization in applications including 1) large-scale distributed coverage, where low-power robots seek locations that maximize the area covered by their sensors [23], [24]; 2) large-scale transportation networks, where sensing infrastructure locations must be selected to best observe large traffic networks [25] (a similar

This research is supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

A. Downie and S. L. Smith are with are with the Electrical and Computer Engineering at the University of Waterloo, Waterloo, ON, Canada {adownie,stephen.smith}@uwaterloo.ca

B. Gharesifard is with the Department of Electrical and Computer Engineering at the University of California, Los Angeles gharesifard@ucla.edu

problem is explored in detail in the simulation results of this paper); and 3) or sensor selection for state estimation of a linear dynamical system [26].

In these applications, function evaluations may be limited due to computational resources, modelling challenges, sensor limitations, or communication constraints. For example, in sensor coverage, a function evaluation requires an agent to compute the set difference between their own sensor footprint and the union of all other sensor footprints. This requires accurate position information for all other agents. In contrast, pairwise function evaluations require only the distance to each agent (provided by a range sensor), and computation is limited to the intersecting two sets. In transportation networks, accurately modelling the interactions of a large number of agents poses similar challenges to the revenue example above. This paper sheds light onto the fundamental question of how well one can optimize a submodular function when given only limited access to its values.

Statement of Contributions: We consider the submodular maximization problem where information about the underlying function is limited, in that we only have access to evaluations of sets of size at most k. Let X be the base set of elements we are optimizing over and n be the maximum number of elements that can be in our solution set. We begin with a negative result; namely, there exists a class submodular functions for which no algorithm subject to this information constraint can guarantee performance better than k/n of optimal. In light of this, we propose a class of functions where we can upper bound the marginal gains of the objective function in terms of k-wise information. We proposed a simple simple greedy algorithm that utilize only k-wise information. We introduce a new notion of curvature named the k-Marginal Curvature which capture "how k-wise submodular" a function is. We then adapt a previous result for approximate value oracles to prove performance bounds for the algorithm in terms of our new notion of curvature. The new notion of curvature provides a new way to understand submodular functions and may be of independent interest. Finally, we show experimental results that highlights the effectiveness of the algorithm.

An extended version of this work, which includes all proofs is available in [27].

II. Problem Definition and Inapproximability

Let X be a set of elements and 2^X be the power set of those elements. A set function $f : 2^X \to \mathbb{R}_{\geq 0}$ is submodular if the following property of diminishing returns holds: For all $A \subseteq B \subseteq X$ and $x \in X \backslash B$ we have

$$f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B).$$

We refer to $f(A \cup \{x\}) - f(A)$ as the marginal return of x given A, denoted by $f(x|A)$. For simplicity, we denote the objective value of a singleton $f(\{x\})$ by $f(x)$. We also denote marginal return of x with respect to a singleton set $A = \{y\}$ by $f(x|y)$ and refer to it as the pairwise marginal return of x given y. In addition to submodularity, throughout this paper, we assume that the functions satisfy

1) Monotonicity: For all $A \subseteq B \subseteq X$, $f(A) \leq f(B)$,
2) Normalization: $f(\emptyset) = 0$.

Another property of submodular functions we utilize is the notion of curvature [28], [29]. The curvature of a submodular function f is defined as

$$c = 1 - \min_{A \subseteq X, x \in X \setminus A} \frac{f(x|A)}{f(x)}. \tag{1}$$

Note that if the value of $c = 0$, the function is modular.

We start by recalling the problem of maximizing a submodular function over the uniform matriod. Let X be a set of elements and let $f : 2^X \to \mathbb{R}_{\geq 0}$ be a monotone normalized submodular function. We wish to solve the following problem:

$$\begin{aligned} &\max_{S \subseteq X} f(S) \\ &\text{s.t. } |S| \leq n. \end{aligned} \tag{2}$$

This is the classical submodular maximization problem that can be solved to an approximation factor of $(1 - 1/e)$ using the following simple greedy algorithm, see [30]:

$$\begin{aligned} x_i &\in \underset{x \in X \setminus S_{i-1}}{\arg\max}\ f(x|S_{i-1}) \\ S_i &= S_{i-1} \cup \{x_i\}, \end{aligned} \tag{3}$$

where i is the iteration of the algorithm and S_i is the solution produced after i iterations. In what follows, we often refer to this strategy as the full information greedy algorithm. A key focus of our contributions is to understand the limitations of algorithms that only have access to partial information about the objective function. We make this precise in the next definition.

Definition II.1. (k-wise Information) Given a submodular function f, the k-wise information set is defined as the set of tuples $\{(S, f(S)) | S \subseteq X, |S| \leq k\}$. When $k = 2$, we refer to this as pairwise information.

In this work we consider algorithms that take as input a base set of elements X, a function $f : 2^X \to \mathbb{R}_{\geq 0}$ as a black box, and a number n and which outputs a subset of S with cardinality $|S| \leq n$ in finite time. An algorithm that has access to k-wise information can utilize the black box for f only to evaluate f on sets of size k or less. We denote the class of such algorithms by $\Pi_{\text{k-wise}}$, or Π_{pairwise} when $k = 2$.

The main objective that we have in mind is to study Problem 2 with such limitations. We now present a negative result that addresses the inapproximability of this problem.

Proposition II.2. *Consider Problem 2 with k-wise information. Then for every algorithm $\pi \in \Pi_{\text{k-wise}}$, there exists a normalized, monotone and submodular function $f : 2^X \to \mathbb{R}$ with $|X| \geq 2n$ such that*

$$f(S^\pi) \leq \frac{k}{n} f(S^*),$$

where S^π is the solution constructed by π and S^* is the optimal solution.

Proof. We begin by constructing a normalized, monotone submodular function f. Consider a set X that is partitioned into two disjoint sets $X = V \cup V^*$, where $|V^*| = n$ and $|V| \geq n$. We define the function $f : 2^X \to \mathbb{R}_{\geq 0}$ as:

$$f(S) = \min\{|S \cap V|, k\} + |S \cap V^*|. \tag{4}$$

This function is normalized and monotone, and given k, it assigns a value of k to all sets of size k. The set V can be thought of as the general set and V^* is a special set where you are guaranteed to get value if you selected an element from V^*. The function, counts the number of elements of S that are in V^*. However, for all sets S where $|S| \leq k$ get mapped to their cardinality.

We now show that f is also submodular. Consider any two sets $A \subset B \subset X$ and an element $x \in X \setminus B$. We show that

$$f(x|A) \geq f(x|B).$$

First notice that $f(x|A)$ and $f(x|B)$ are each either 0 or 1, since adding an element can increase the function value by at most one. There are two cases to consider:

Case 1 ($x \in V^*$): In this case $f(x|A) = 1$, since $A \cup \{x\}$ has one more element in V^* than A. Since $f(x|B) \leq 1$, the result follows.

Case 2 ($x \in V$): We assume that $f(x|B) = 1$, as otherwise, the result holds. Since $f(x|B) = 1$ and $x \in V$, we must have $|B \cap V| < k$. But this implies that $|A \cap V| < k$ since $A \subseteq B$. Thus $f(x|A) = 1$ and the result holds.

For any set S with $|S| \leq k$ we have that $f(S) = |S|$ which reveals no information on which elements of S are in V or V^*. Hence, for any algorithm in $\Pi_{\text{k-wise}}$, the values returned by the black box for f are independent of the choice of V and V^*. Given an algorithm π in $\Pi_{\text{k-wise}}$, we can provide the black box containing k-wise information (i.e., a black box implementing $f(S) = |S|$) and observe the resulting solution S^π. We then select any $V^* \subset X$ of size n such that $S^\pi \cap V^* = \emptyset$. Equation (4) then defines a function such that S^π is a solution returned by π with value $f(S^\pi) = k$, and $S^* = V^*$ is the optimal solution with value $f(S^*) = n$. This proves the desired result. ■

This result highlights the challenges that arise under k-wise information constraints. In the following sections, we characterize functions where the marginals with respect to sets of size k, or smaller, are informative of the higher order marginals using new notions of curvature.

III. Limited Information Algorithm

Our main objective in what follows is to leverage k-wise information to find an approximate solution to Problem 2.

A. Optimistic Algorithm

A natural strategy is to greedily select elements that maximize the *estimated* marginal return using only k-wise information. First, given $S \subseteq X$ and $x \in X \backslash S$ note that

$$\min_{A \subseteq S, |A| < k} f(x|A) \geq f(x|S), \tag{5}$$

which holds by submodularity of f, because for all $A \subseteq S$, we have $f(x|A) \geq f(x|S)$. We will define a simple estimate of the marginal returns of f as the left hand side of (5):

$$\bar{f}_k(x|S) := \min_{A \subseteq S, |A| < k} f(x|A).$$

The k-wise marginal for all $A \subseteq S$ upper bounds $f(x|S)$ and hence we choose the minimum as it is the best available estimate of the true value of $f(x|A)$. In a nearly identical style to the classical greedy strategy, we now define an algorithm as follows:

$$\begin{aligned} x_i &\in \operatorname*{arg\,max}_{x \in X \backslash S_{i-1}} \bar{f}_k(x|S_{i-1}) \\ S_i &= S_{i-1} \cup \{x_i\}. \end{aligned} \tag{6}$$

Throughout this paper, we will refer to (6) as the *optimistic* algorithm. In essence, the optimistic algorithm aims to greedily select elements with maximum *potential* marginal return.

B. Approximate Value Oracles

To characterize the performance of the optimistic algorithm given by (6), we consider the problem through the lens of maximizing a submodular objective function via surrogate objective functions. Following [29], we will discuss how to determine performance guarantees when using such surrogates.

Let $S \subseteq X$ be the result set produced by an algorithm. Suppose we have an ordering of the elements of $S = \{x_1, \ldots, x_n\}$, then we work with the set $S_i = \{x_1, \ldots, x_i\}$. Now let $\{x_1^g, \ldots, x_n^g\} \subseteq X$ be such that each x_i^g maximizes the marginal return of f conditioned on S_{i-1}, i.e.,

$$x_i^g \in \operatorname*{arg\,max}_{x \in X \backslash S_{i-1}} f(x|S_{i-1}) \tag{7}$$

The set $\{x_1^g, \ldots, x_n^g\}$ represents the elements that a greedy algorithm with full information about the objective f would have selected if it had previously selected S_{i-1}. Using these values, we can now measure the quality of a given algorithm's choices compared to that of an algorithm with full information about the objective. We do this by finding $\alpha_i \in \mathbb{R}_+$, for $i \in \{1, \ldots, n\}$ such that

$$\alpha_i f(x_i|S_{i-1}) \geq f(x_i^g|S_{i-1}). \tag{8}$$

By the greedy choice property of x_i^g, we have that

$$f(x_i|S_{i-1}) \leq f(x_i^g|S_{i-1}).$$

Hence, $\alpha_i \geq 1$, for all $i \in \{1, \ldots, n\}$. From this point on, we call each α_i the approximation factor associated with x_i.

In the general framework proposed in [29], the objective is to greedily maximize multiple surrogate objective functions, and to use these to generate approximate solutions. For our problem of submodular maximization with only pairwise information, we simply maximize using a single surrogate function $\bar{f}_k(x|S)$. We provide a simplified version of [29, Theorem 1] as follows.

Theorem III.1. *Suppose that $S = \{x_1, \dots, x_n\} \subseteq X$ is the set of elements selected by an algorithm and $\{\alpha_1, \dots, \alpha_n\}$ are the set of approximation factors that satisfy* (8). *Let S^* be the optimal solution to Problem 2. Then*

$$f(S) \geq \left(1 - e^{-\frac{1}{n}\sum_{i=1}^{n}\frac{1}{\alpha_i}}\right) f(S^*). \tag{9}$$

Note that Theorem III.1 relies on f being a normalized, monotone and submodular function, and hence the result can be applied to any algorithm for Problem 2, not just algorithms that only have access to k-wise information. An interesting remark about Theorem III.1 is that the performance bound depends essentially on the average of the approximation factors. Some of these factors could be large compared to the others, but as long as most of them are small, good performance is maintained.

Remark III.2 (On The Ordering Of The Result Set)*.* As described in this section, we impose an ordering of the elements produced by an algorithm. From the proof provided in [27], the result holds for any ordering. Therefore, the tightest bound produced by an algorithm is achieved by taking the maximum over all permutations of the output set of the algorithm. From a practical perspective, computing the bound for every permutation is infeasible. The algorithms we present later in this work, select elements iteratively. Therefore, the ordering we impose on elements in the resulting solution is the order that the elements were selected.

C. Optimistic Algorithm Approximation Performance

We aim to provide approximation guarantees for the optimistic algorithm. To give an intuition for what we are about to present, we consider the following example.

Example III.3. Consider the scenario depicted in Figure 1. Suppose we only have access to pairwise information i.e., $k = 2$, here we wish to select four sensors to maximize the area of their combined footprints. One of the simplest algorithms that satisfies the pairwise information constraint is the *uninformed* greedy strategy:

$$x_i \in \underset{x \in X \setminus S_{i-1}}{\arg\max}\, f(x) \tag{10}$$
$$S_i = S_{i-1} \cup \{x_i\}.$$

We refer to this algorithm as *uninformed* because it only use the most basic information about f which it's values evaluated on single elements. For the uninformed greedy strategy, the scenario described in Figure 1 could potentially lead to poor performance. This strategy cannot distinguish between its choices and therefore could select four sensors that almost perfectly overlap with each other (i.e., in the same pile), resulting in a low objective value. Alternatively, if we had used the optimistic algorithm, once one element is selected from a pile, the 2-wise upper bound on the other elements in a pile would be low. In later iterations, the optimistic algorithm would avoid selecting elements in piles where elements have been previously selected from. Interestingly, we see that for each i, the difference between $f(x_i|S_{i-1})$ and $\bar{f}_2(x_i|S_{i-1})$ is small. We notice that in

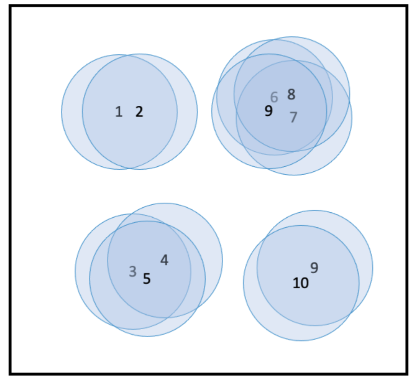

Fig. 1. Example sensor coverage configuration where the optimistic algorithm performs better than uninformed greedy strategy

these scenarios, the value of $\bar{f}_2(x_i|S_{i-1})$ provides accurate information about the value of $f(x_i|S_{i-1})$. This is the idea that we want to capture in the following result.

Theorem III.4. *Let $S_{i-1} \subseteq X$ be the partial solution of optimistic algorithm after $(i-1)$ iterations, and let $x_i \in X$ be the element selected during the ith iteration. Then we have that*

$$\alpha_i = \begin{cases} 1 & i \leq k \\ \frac{\bar{f}_k(x_i|S_{i-1})}{f(x_i|S_{i-1})} & i > k \end{cases} \tag{11}$$

satisfy (8) *for all $i \leq n$.*

Proof. Let x_i^g be the true greedy choice at iteration i given S_{i-1}. For $i \leq k$ we have that $f(x_i|S_{i-1}) = f(x_i^g|S_{i-1})$ by the definition of $\bar{f}_k(x_i|S_{i-1})$. Therefore we have, $\alpha_1 = \cdots = \alpha_k = 1$. The minimum possible approximation factor we have can be written as

$$\alpha_i^{\min} = \frac{f(x_i^g|S_{i-1})}{f(x_i|S_{i-1})} \tag{12}$$

Any approximation factor α_i such that $\alpha_i \geq \alpha_i^{\min}$ will satisfy equation (8). We will now upper bound $\alpha_i^{\min}$ as follows.

$$\begin{aligned} \alpha_i^{\min} &= \frac{f(x_i^g|S_{i-1})}{f(x_i|S_{i-1})} \\ &\leq \frac{\bar{f}_k(x_i^g|S_{i-1})}{f(x_i|S_{i-1})} \qquad (13) \\ &\leq \frac{\bar{f}_k(x_i|S_{i-1})}{f(x_i|S_{i-1})}, \qquad (14) \end{aligned}$$

where equation (13) holds by the definition of the upper bound. Equation (14) holds by the greedy choice property of the k-wise optimistic algorithm. Therefore, the right hand side of (14) is a valid approximation factor, which we set α_i to, concluding the result. ∎

The following corollary is an immediate consequence.

Corollary III.5. *Let $S \subseteq X$ be the solution produced by the optimistic algorithm and $S_{i-1} \subseteq S$ be the partial solution after $(i-1)$ iterations of the optimistic algorithm and let*

$x_i \in S$ *be the element selected at the* i*-th iteration, then we have*

$$f(S) \geq \left(1 - e^{-\frac{1}{n}\left(k + \sum_{i=k+1}^{n} \frac{f(x_i|S_{i-1})}{\bar{f}_k(x_i|S_{i-1})}\right)}\right) f(S^*). \quad (15)$$

We see that the approximation performance of the algorithm is dictated by the sum in the exponent. We can interpret the exponent as the mean of the set

$$\left\{1, 1, \frac{f(x_3|S_2)}{\bar{f}_k(x_3|S_2)}, \ldots, \frac{f(x_n|S_{n-1})}{\bar{f}_k(x_n|S_{n-1})}\right\}.$$

This implies that, to get adequate performance from the optimistic algorithm, we need the value of $\bar{f}_k(x_i|S_{i-1})$ to be close to $f(x_i|S_{i-1})$ on *average*.

The term $\frac{f(x_i|S_{i-1})}{\bar{f}_k(x_i|S_{i-1})}$ is closely related to the traditional notion of curvature, leading us to the next definition.

Definition III.6 (k-Marginal Curvature)**.** The k-marginal curvature of f given $S \subseteq X$ and $x \in X \backslash S$ is defined as

$$c_k(x|S) = 1 - \max_{A \subseteq S, |A| < k} \frac{f(x|S)}{f(x|A)}. \quad (16)$$

Note that $c_k(x|S) = 1 - \frac{f(x|S)}{\bar{f}_k(x|S)}$.

Remark III.7. This k-marginal curvature characterizes the relationship between the values of the k-wise upper bounds $\bar{f}_k(x|S)$ and true values of $f(x|S)$. Note that there exist functions where the values of the k-marginal curvatures can be close to 0 even though the value of traditional curvature is close to 1. The sensor coverage function, described in Figure 1, is an example of a function where the traditional curvature is close to 1 and the values of the 2-marginal curvatures are close to 0.

This allows us to rewrite (15) as follows:

$$f(S) \geq \left(1 - e^{-\frac{1}{n}\left(k + \sum_{i=k+1}^{n} 1 - c_k(x_i|S_{i-1})\right)}\right) f(S^*). \quad (17)$$

We characterize the worst-case performance in terms of the average of the k-marginal curvatures, which capture the intuition from Example III.3.

We can now compare the performance of the k-wise optimistic strategy when more information is provided to the algorithm, using the k-marginal curvatures. We see that

$$c_k(x|S) \geq c_l(x|S),$$

for all $l > k$. Therefore, (17) will provide a tighter bound for higher values of k.

D. Computational Cost of k-wise Optimistic Algorithm

Having access to k-wise information provides us with stronger approximation bounds, but we trade off computation performance. We are required to compute the minimum marginal overall subsets $A \subseteq S_{i-1}$, where $|A| < k$ for each $x \in X$. When $k \leq |S_i|$, we need to check $\binom{|S_i|}{k-1}$ subsets of S to find the minimum. This becomes expensive to do computationally as S_{i-1} grows larger. If $k = 3$, the computation of each marginal is quadratic in $|S_{i-1}|$ and can be expensive to compute. In general, a simple implementation of the k-wise optimistic algorithm has a time complexity of $\mathcal{O}\left(|X| \cdot n \cdot \binom{n}{k-1}\right)$. From a practical perspective, we can actually compute the 2-wise optimistic algorithm efficiently which is discussed [27].

E. Constant Factor Approximation

From the analysis in Sections III-C, we notice that the performance bound for the algorithm uses the function evaluations of the the elements selected in the set. This may not be desirable in some applications due to the fact that the user will have to run the algorithm (which could potentially be expensive) to know the performance of the optimistic algorithms. We can provide a bound that is weaker but does not require us to know what $x_1, \ldots, x_n \in X$ are before computing.

Let us define a similar notion of curvature as k-marginal curvature, closer to the tradition notion (1).

Definition III.8 (Total k-Marginal Curvature)**.** Let f be a submodular function then, the curvature $\bar{c}_k$ of f is defined as

$$\bar{c}_k = 1 - \min_{S \subseteq X, x \in X \backslash S} \frac{f(x|S)}{\bar{f}_k(x|S)}. \quad (18)$$

Using this notion, we can follow a similar process to arrive at a approximation bound for the k-wise optimistic algorithm; we state this result next.

Theorem III.9. *Let* $S \subseteq X$ *be the solution produced by the* k*-wise optimistic algorithm, and* $S^* \subseteq X$ *be the solution to Problem 2 then we have*

$$f(S) \geq \left(1 - e^{-\left(1 - \frac{n-k}{n}\bar{c}_k\right)}\right) f(S^*) \geq \left(1 - e^{-(1-\bar{c}_k)}\right) f(S^*). \quad (19)$$

Proof. Let $S = \{x_1, \ldots, x_n\}$ be the solution produced by the k-wise optimistic algorithm, and $S_i = \{x_1, \ldots, x_i\}$. Let x_i^g be the true greedy as defined in (7). Let $\alpha_1, \ldots, \alpha_n$ be the approximation factors for the solution S. For $i \leq k$ we have that $x_i = x_i^g$ by the definition of $\bar{f}_k(x_i|S_{i-1})$. Therefore we have, $\alpha_i = 1$ for all $i \leq k$. The minimum possible approximation factor we have can be written as

$$\alpha_i^{\min} = \frac{f(x_i^g|S_{i-1})}{f(x_i|S_{i-1})}. \quad (20)$$

Any approximation factor α_i such that $\alpha_i \geq \alpha_i^{\min}$ will satisfy equation (8). We will now upper bound $\alpha_i^{\min}$ as follows for $i > k$:

$$\alpha_i^{\min} = \frac{f(x_i^g|S_{i-1})}{f(x_i|S_{i-1})} \leq \frac{\bar{f}_k(x_i^g|S_{i-1})}{f(x_i|S_{i-1})} \quad (21)$$

$$\leq \frac{\bar{f}_k(x_i|S_{i-1})}{f(x_i|S_{i-1})} \leq \max_{\bar{S} \subseteq X, x \in X \backslash \bar{S}} \frac{\bar{f}_k(x|\bar{S})}{f(x|\bar{S})}, \quad (22)$$

where (21) holds by the definition of the k-wise upper bound, and (22) holds by the greedy choice property of the k-wise optimistic algorithm. To utilize Theorem III.1 for producing an approximation bound, we compute

$$\frac{1}{\alpha_i} = \frac{1}{\max_{\bar{S}\subseteq X, x\in X\setminus\bar{S}} \frac{\bar{f}_k(x|\bar{S})}{f(x|\bar{S})}}$$
$$= \min_{\bar{S}\subseteq X, x\in X\setminus\bar{S}} \frac{f(x|\bar{S})}{\bar{f}_k(x|\bar{S})} = 1 - \bar{c}_k, \tag{23}$$

where (23) holds since $\bar{f}_k(x|\bar{S}) \geq f(x|\bar{S})$ for all $\bar{S} \subseteq X, x \in X\setminus\bar{S}$ and (23) holds by the definition of total k marginal curvature. We will now substitute in each $\frac{1}{\alpha_i}$ into Theorem III.1 and simplify to arrive at the approximation bound:

$$f(S) \geq \left(1 - e^{-\left(1-\frac{n-k}{n}\bar{c}_k\right)}\right) f(S^*) \geq \left(1 - e^{-(1-\bar{c}_k)}\right) f(S^*),$$

where the second inequality holds since $1 - \frac{n-k}{n}\bar{c}_k \geq 1 - \bar{c}_k$, concluding the proof. ■

Theorem III.9 shows that the k-wise optimistic algorithm provides a constant factor approximation for Problem 2 which is dependent on this new notion of curvature. Although, the performance bounds are not as strong as Corollary III.5, this fact does show that the fundamental quantity that underlies the performance of the k-wise optimistic algorithm is the total k marginal curvature. We do not provide experiments showing the approximation bounds produced by Theorem III.9 since the bound produced by Corollary III.5 is stronger.

IV. Simulation Results

To illustrate the performance of our proposed k-wise optimistic algorithm, we consider the problem of maximum entropy sampling problem with Gaussian Radial Basis Function (RBF) Kernels [9], and make a comparison to the full information greedy strategy.

A. Maximum Entropy Sampling Problem

In maximum entropy sampling, we are given N random variables and the goal is to select a subset of size n that is most informative. This is often done by maximizing the Boltzman-Shannon entropy [31]. Let $C \in \mathbb{R}^{N\times N}$ be the covariance matrix of the N random variables, where each random variable has a unique row and column in the covariance matrix. We let $C[S,T]$ denote the principle submatrix of C with rows indexed by $S \subseteq [N]$ and columns indexed by subset $T \subseteq [N]$. We wish to select a set of indices $S \subseteq [N]$, such that the corresponding random variables have maximum entropy. Assuming that the random variables are jointly Gaussian distributed, this problem is equivalent to selecting a set S of size n to maximize

$$f(S) = \log\det(C[S,S]).$$

Following [9], we assume that C takes the form of a Gaussian RBF kernel matrix and associate each random variable with a corresponding vector $v_1,\ldots,v_N \in \mathbb{R}^d$. Then, each entry of C is defined as $C_{i,j} = \exp(-\gamma\|v_i - v_j\|^2)$ where γ is a scaling parameter. This function is thoroughly explored in [9]. The function f is a normalized, monotone and submodular when the minimum eigenvalue of C is greater 1 which can be done by scaling C.

B. Results

To compare the k-wise optimistic algorithm to the full information greedy strategy we computed 10 covarience matrices by randomly generating 100 uniformly distributed points in $v_1\ldots,v_{100} \in [0,3)\times[0,3)$. We then executed the full information greedy algorithm, and the k-wise optimistic algorithm for $k = 2,3,5$ on f using the 10 different covarience matrices. We selected up 20 to elements and computed both the objective value and performance bound after n iterations. Figure 2 highlight the effectiveness of

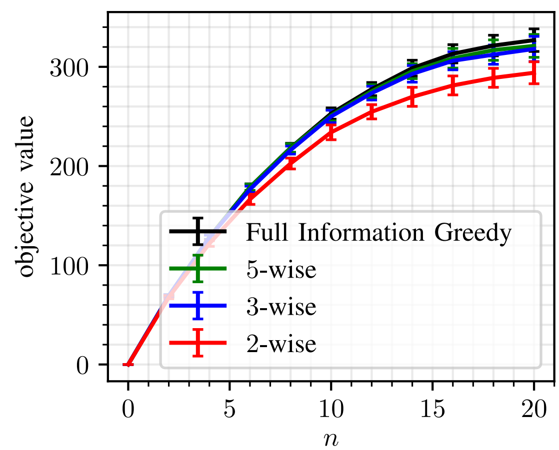

Fig. 2. Average entropy of n variables selected by algorithms. 5,3,2-wise algorithms refer to the k-wise optimistic algorithms. The errors bars represent 1 standard deviation from the mean value.

the k-wise algorithm relative to greedy strategy with full access to the objective function. We see that with access to 3 and 5-wise information the optimistic algorithm is nearly as effective as the full information greedy strategy. The 2-wise greedy strategy has slightly weaker performance, but for low values of n is competitive with the full information greedy strategy.

Figure 3 looks at the average performance bound as a function of n. We see when k increases, the optimistic greedy strategies performance degrades much slower as n increases. The guaranteed approximation ratios for the pairwise optimistic strategy degrade rapidly as n increases. On the other hand, even though the performance guarantees degrade as n increases, we see from Figure 2 that the performance relative to the greedy strategy is not degrading. This suggests that the bounds produced by Corollary III.5 can be loose relative to the average performance. This is a common observation made when experimentally testing greedy strategies for submodular maximization [9].

V. Conclusions

In this work we introduced the problem of maximizing a submodular function with limited function access. We showed that for a general submodular function, we cannot guarantee strong approximation guarantees. In light of this we propose a simple algorithm that leverages only k-wise

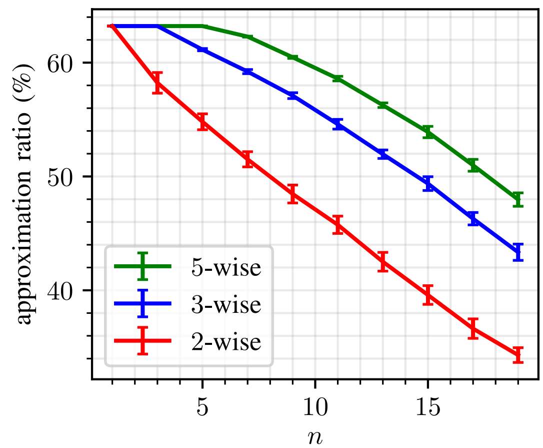

Fig. 3. Average approximation ratio for each k-wise optimistic algorithms after selecting n random variables. Bounds were computed using Corollary III.5.

information that can provide a constant factor approximation to the problem. We also see in practice, that the k-wise algorithms can be nearly as effective as the fully information greedy strategy. The optimistic algorithm is relatively simple, and we are interested in knowing if we develop a more sophisticated approach, could we provide improved performance guarantees for the problem. In the future we would like to explore how the limited information greedy strategy can be extend to the distributed scenario described in [10], [11].

References

[1] Z. Liu, A. Clark, P. Lee, L. Bushnell, D. Kirschen, and R. Poovendran, "Submodular optimization for voltage control," *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 502–513, 2018.

[2] J. Qin, I. Yang, and R. Rajagopal, "Submodularity of storage placement optimization in power networks," *IEEE Transactions on Automatic Control*, vol. 64, no. 8, pp. 3268–3283, 2019.

[3] M. Roberts, S. Shah, D. Dey, A. Truong, S. Sinha, A. Kapoor, P. Hanrahan, and N. Joshi, "Submodular trajectory optimization for aerial 3d scanning," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5334–5343.

[4] A. Krause and C. Guestrin, "Submodularity and its applications in optimized information gathering," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 4, Jul. 2011.

[5] D. Sahabandu, L. Niu, A. Clark, and R. Poovendran, "Scalable planning in multi-agent mdps," in *IEEE Conference on Decision and Control (CDC)*, 2021, pp. 5932–5939.

[6] K. Wei, R. Iyer, and J. Bilmes, "Submodularity in data subset selection and active learning," in *International Conference on Machine Learning*, vol. 37, Lille, France, 07–09 Jul 2015, pp. 1954–1963.

[7] H. Lin and J. Bilmes, "Multi-document summarization via budgeted maximization of submodular functions," in *Human Language Technologies: Conference of the North American Chapter of the Association for Computational Linguistics*, Los Angeles, CA, Jun. 2010, pp. 912–920.

[8] S. Stan, M. Zadimoghaddam, A. Krause, and A. Karbasi, "Probabilistic submodular maximization in sub-linear time," in *International Conference on Machine Learning*, Sydney, Australia, Aug 2017, pp. 3241–3250.

[9] D. Sharma, A. Kapoor, and A. Deshpande, "On greedy maximization of entropy," in *International Conference on Machine Learning*, 2015, pp. 1330–1338.

[10] B. Gharesifard and S. L. Smith, "Distributed submodular maximization with limited information," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 4, pp. 1635–1645, 2018.

[11] D. Grimsman, M. S. Ali, J. P. Hespanha, and J. R. Marden, "The impact of information in distributed submodular maximization," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 4, pp. 1334–1343, 2019.

[12] D. Grimsman, M. R. Kirchner, J. P. Hespanha, and J. R. Marden, "The impact of message passing in agent-based submodular maximization," in *IEEE Conference on Decision and Control*, 2020, pp. 530–535.

[13] H. Sun, D. Grimsman, and J. R. Marden, "Distributed submodular maximization with parallel execution," in *American Control Conference*, 2020, pp. 1477–1482.

[14] N. Rezazadeh and S. S. Kia, "Multi-agent maximization of a monotone submodular function via maximum consensus," in *IEEE Conference on Decision and Control (CDC)*, 2021, pp. 1238–1243.

[15] L. F. O. Chamon, G. J. Pappas, and A. Ribeiro, "The mean square error in kalman filtering sensor selection is approximately supermodular," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 343–350.

[16] A. Hashemi, M. Ghasemi, H. Vikalo, and U. Topcu, "Randomized greedy sensor selection: Leveraging weak submodularity," *IEEE Transactions on Automatic Control*, vol. 66, no. 1, pp. 199–212, 2021.

[17] S. T. Jawaid and S. L. Smith, "Submodularity and greedy algorithms in sensor scheduling for linear dynamical systems," *Automatica*, vol. 61, pp. 282–288, 2015.

[18] X. Sun, C. G. Cassandras, and X. Meng, "A submodularity-based approach for multi-agent optimal coverage problems," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 4082–4087.

[19] L. Zhou, V. Tzoumas, G. J. Pappas, and P. Tokekar, "Resilient active target tracking with multiple robots," *IEEE Robotics and Automation Letters*, vol. 4, no. 1, pp. 129–136, 2019.

[20] A. Krause, J. Leskovec, C. Guestrin, J. VanBriesen, and C. Faloutsos, "Efficient sensor placement optimization for securing large water distribution networks," *Journal of Water Resources Planning and Management*, vol. 134, no. 6, pp. 516–526, 2008.

[21] A. Badanidiyuru, B. Mirzasoleiman, A. Karbasi, and A. Krause, "Streaming submodular maximization: Massive data summarization on the fly," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2014, p. 671–680.

[22] M. Corah and N. Michael, "Distributed submodular maximization on partition matroids for planning on large sensor networks," in *IEEE Conference on Decision and Control*, 2018, pp. 6792–6799.

[23] M. Rubenstein, C. Ahler, N. Hoff, A. Cabrera, and R. Nagpal, "Kilobot: A low cost robot with scalable operations designed for collective behaviors," *Robotics and Autonomous Systems*, vol. 62, no. 7, pp. 966–975, 2014.

[24] X. Sun, C. G. Cassandras, and X. Meng, "A submodularity-based approach for multi-agent optimal coverage problems," in *IEEE Conference on Decision and Control*, 2017, pp. 4082–4087.

[25] N. Mehr and R. Horowitz, "A submodular approach for optimal sensor placement in traffic networks," in *2018 Annual American Control Conference (ACC)*, June 2018, pp. 6353–6358.

[26] A. Hashemi, M. Ghasemi, H. Vikalo, and U. Topcu, "Randomized greedy sensor selection: Leveraging weak submodularity," *IEEE Transactions on Automatic Control*, vol. 66, no. 1, pp. 199–212, 2020.

[27] A. Downie, B. Gharesifard, and S. L. Smith, "Submodular maximization with limited function access," *arXiv preprint arXiv:2201.00724*, 2022.

[28] T. Friedrich, A. Göbel, F. Neumann, F. Quinzan, and R. Rothenberger, "Greedy maximization of functions with bounded curvature under partition matroid constraints," *AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 2272–2279, Jul. 2019.

[29] K. Wei, R. Iyer, and J. Bilmes, "Fast multi-stage submodular maximization," in *International Conference on International Conference on Machine Learning*, 2014, pp. 1494–1502.

[30] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—I," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[31] J. Lee, "Maximum entropy sampling," in *Encyclopedia of Environmetrics*, 2002, vol. 3, pp. 1229–1234.