Bregman Projections over Submodular Base Polytopes

Swati Gupta
swatig@mit.edu

Michel Goemans
goemans@math.mit.edu

Patrick Jaillet
jaillet@mit.edu

Massachusetts Institute of Technology

Abstract

A well-known computational bottleneck in various first order methods like mirror descent is that of computing a certain Bregman projection. We give a novel algorithm, INC-FIX, for computing these projections under separable mirror maps and more generally for minimizing separable convex functions over submodular base polytopes. For minimizing divergences onto cardinality-based submodular base polytopes defined on ground set E, we prove an $O(|E|^2)$ running time under any uniformly separable mirror map. This matches the running time of [9, 13] for projections under KL-divergence and squared Euclidean distance, recovers an algorithm from [16] for computing projections over the simplex.

1 Introduction

First order methods like mirror descent and its variants (exponential weights, projected gradient descent, lazy mirror descent, mirror prox, saddle-point mirror prox, stochastic gradient descent, online stochastic mirror descent etc) enjoy near-optimal regret bounds in online optimization and near-optimal convergence rates in convex optimization [28, 3, 27, 12, 10]. These methods are typically based on a strongly-convex[1] function ω, known as the mirror map (or the distance generating function), that must satisfy additional properties of divergence of the gradient. A well-known computational bottleneck in these algorithms is that of computing a generalized notion of projection (whenever the decision set is bounded), defined by the function $D_\omega(x, y) = \omega(x) - \omega(y) - \nabla\omega(y)^T(x - y)$, called the Bregman divergence of the mirror map. Squared Euclidean distance, KL-divergence, Itakura-Saito distance, Logistic loss, p-norm distance are some examples of Bregman divergences. For general convex sets, taking a Bregman projection is often a separable convex minimization problem. One could exploit the general machinery of convex optimization such as the ellipsoid algorithm, but the question is if we can do better by exploiting the structure of the set.

In this work, we consider polytopes that arise from submodular set functions, and model a number of important combinatorial concepts (see Table 1). We consider the problem of computing Bregman projections over these polytopes, and more generally the problem of minimizing separable convex functions over them. This problem becomes important whenever there is a combinatorial structure on the decision set (for instance, for online learning algorithms, structured regression, equilibria in games when a player plays combinatorial strategies).

In 1980, Fujishige gave the *monotone* algorithm to find the minimum norm point, i.e., $\min_{x\in X} \|x\|^2$ where X is a submodular base polytope [19]. There has been a large volume of work in the last 30 years [24, 25, 9] to develop faster combinatorial algorithms for separable convex minimization over submodular base polytopes, specifically focusing on the decomposition algorithm of [20]. These

[1] $f : X \to \mathbb{R}$ is κ-strongly convex w.r.t. $\|\cdot\|$ if $f(x) \geq f(y) + g^T(x-y) + \frac{\kappa}{2}\|x-y\|^2, \forall x, y \in X, g \in \partial f(x)$

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

approaches however rely on generating a sequence of violated inequalities and achieve a feasible solution only at the completion of the algorithm (characteristic of dual approaches).

We give a novel generalization of Fujishige's monotone algorithm, INC-FIX, that is fundamentally different from the known decomposition algorithm. It always maintains a feasible solution in the submodular polytope (characteristic of primal approaches). A useful property of our algorithm is that it can be terminated earlier to obtain a point in the submodular polytope, which can then be rounded (in multiple ways) to obtain a point in the base polytope. The key idea of the algorithm comes from first order optimality conditions, i.e. if a point x^* is a minimizer of a convex function $h : X \to \mathbb{R}$ over a convex set X, then it must hold that $\nabla h(x^*)^T(x^* - z) \leq 0$ for[2] all points $z \in X$. Read differently, if one somehow knew the value of $\nabla h(x^*) = c$ (say), then x^* would minimize the linear function $c^T z$ over $z \in X$. A simple greedy algorithm can be used to minimize linear cost functions over submodular polytopes [6]. We use this to construct a point x^* such that it is a minimizer of its gradient function $\nabla h(x^*)$. Using the recent result of [17], the INC-FIX algorithm can be shown to have a running time of $\tilde{O}(|E|^5\gamma + |E|^6)$ time, where γ is the time for one submodular function evaluation and E is the ground set of elements. Our approach provides useful lower bounds such that the algorithm can be terminated earlier with provable optimality gaps.

Further, we consider the special case of cardinality-based submodular functions[3]. We show that our algorithm can be implemented overall in $O(|E|^2)$ time under any uniformly separable mirror map of the form $\omega(x) = \sum_{e \in E} w(x(e))$ (Table 2), as long as a certain non-linear equation in a single parameter can be solved exactly[4]. This matches the running time of specialized algorithms [9] and [13] that work only for squared Euclidean distance and KL-divergence and recovers an algorithm from [16] for computing projections over the simplex. Our work, however, applies to any uniformly separable mirror map. We believe that the key ideas developed in these algorithms may be useful to compute projections under Bregman divergences over other polyhedra, as long as the linear optimization for those is well-understood. Note that our approach gives exact solutions[5] irrespective of the structure of the submodular function, as opposed to other approximate methods like Frank-Wolfe [18] that scale as $O(1/\epsilon)$ where ϵ is the gap from the optimal function value.

Table 1: Problems and the submodular functions (on ground set of elements E) that give rise to them.

Problem	**Submodular function,** $S \subseteq E$ **(unless specified)**
k out of n experts (k-simplex), $E = \{1, \dots, n\}$	$f(S) = \min\{\lvert S\rvert, k\}$
k-truncated permutations over $E = \{1, \dots, n\}$	$f(S) = (n-k)\lvert S\rvert$ for $\lvert S\rvert \leq k$, $f(S) = k(n-k) + \sum_{s=k+1}^{\lvert S\rvert}(n+1-s)$ if $\lvert S\rvert \geq k$
Spanning trees on $G = (V, E)$	$f(S) = \lvert V(S)\rvert - \kappa(S)$, $\kappa(S)$ is the number of connected components of S
Matroids over ground set E: $M = (E, \mathcal{I}), \mathcal{I} \subseteq 2^E$	$f(S) = r_M(S)$, the rank function of the matroid
Coverage of T: given $T_1, \dots, T_n \subseteq T$	$f(S) = \lvert \bigcup_{i \in S} T_i\rvert, E = \{1, \dots, n\}$
Cut functions on a directed graph $D = (V, E)$, $c : E \to \mathbb{R}_+$	$f(S) = c(\delta^{out}(S)), S \subseteq V$

Table 2: Examples of some popular uniform separable mirror maps and their corresponding divergences. $(w')^{-1} = w^{*\prime}$, where w^* is Legendre-Fenchel transform of w defined as $w^*(s) = \sup_{x \in \mathbb{R}} sx - w(x)$.

$\omega(x) = \sum w(x_e)$	$F_w = (w')^{-1}$	$D_\omega(x, y)$	Divergence
$\lVert x\rVert^2/2$	x	$\sum_e (x_e - y_e)^2/2$	Squared Euclidean Distance
$\sum_e x_e \log x_e - x_e$	e^x	$\sum_e \left(x_e \log(x_e/y_e) - x_e + y_e\right)$	Generalized KL-divergence
$-\sum_e \log x_e$	$-1/x$	$\sum_e \left(x_e/y_e - \log(x_e/y_e) - 1\right)$	Itakura-Saito Distance
$\sum_e x_e \log x_e + \sum_e (1 - x_e) \log(1 - x_e)$	$\frac{e^x}{1+e^x}$	$\sum_e x_e \log(x_e/y_e) + (1 - x_e) \log((1 - x_e)/(1 - y_e))$	Logistic Loss

Due to wide applicability of separable convex minimization over base polytopes, this work has applications in online combinatorial optimization ([31], [14], [13], [21, 1], [22]), obtaining equilibria for multi-player games ([15], [29], [8], [11]), sparse learning methods [2], computing bounds for the partition function of log-submodular distributions [5] and network analysis [26].

[2]Here, ∇h means gradient of the function h.

[3]A submodular function $f : 2^E \to \mathbb{R}$ is called cardinality-based if $f(S) = g(|S|)$ for some concave g.

[4]we assume oracle access to solutions of these equations to achieve an $O(|E|^2)$ running time

[5]given that the non-linear single variable equation can be solved exactly

2 Convex Minimization on Base Polytopes of Polymatroids

Consider a ground set E of elements on which the submodular function is defined. Let f be a submodular set function, i.e. $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$ for all $A, B \subseteq E$. Further, let f be monotone[6] and normalized ($f(\emptyset) = 0$). We can assume w.l.o.g. that $f(A) > 0$ for $A \neq \emptyset$. Given such a function f, the submodular polytope (or independent set polytope) is defined as $P(f) = \{x \in \mathbb{R}_+^{\mathrm{E}} : x(U) \leq f(U) \; \forall \; U \subseteq E\}$ and the base polytope as $B(f) = \{x \in \mathbb{R}_+^{\mathrm{E}} : x(E) = f(E), x(U) \leq f(U) \; \forall \; U \subseteq E\}$ [6]. For $x \in \mathbb{R}^E$, we use the shorthand $x(U)$ for $\sum_{e \in U} x(e)$ and by both $x(e)$ and x_e we mean the value of x on element e.

Let us consider any strongly convex separable function $h : \mathcal{D} \to \mathbb{R}$, defined over a convex open set $\mathcal{D} \subseteq \mathbb{R}^E$ such that $h(x) = \sum_{e \in E} h_e(x(e))$ (and some technical conditions like $P(f) \subseteq \overline{\mathcal{D}}$ are satisfied). For example, the squared Euclidean distance is $h(x) = \sum_e h_e(x(e)) = (x_e - y_e)^2$ with $\mathcal{D} = \mathbb{R}^E$, and KL-divergence is $h(x) = \sum_e x_e \ln(x_e/y_e)$ with $\mathcal{D} = \mathbb{R}_+^E$. We give the INC-FIX algorithm for computing $x^* = \arg\min_{x \in B(f)} \sum_{e \in E} h_e(x_e)$.

Key idea of the INC-FIX algorithm: The algorithm is iterative and maintains a vector $x \in P(f) \cap \mathcal{D}$. When considering x we associate a weight vector given by $\nabla h(x)$ and consider the set of minimum weight elements. We move x within $P(f)$ in a direction such that $(\nabla h(x))_e$ increases uniformly on the minimum weight elements, until one of two things happen: (i) either continuing further would violate a constraint defining $P(f)$, or (ii) the set of elements of minimum weight changes. If the former happens, we *fix* the tight elements and continue the process on non-fixed elements. If the latter happens, then we continue increasing the value of the elements in the modified set of minimum weight elements. The complete description of the INC-FIX algorithm is given in Algorithm 1 (in the appendix) and we include an example showing how the gradients are increased in each iteration in Figure 1 for developing intuition. The correctness of the algorithm follows from first order optimality conditions and Edmonds' greedy algorithm and crucially relies on the following theorem.

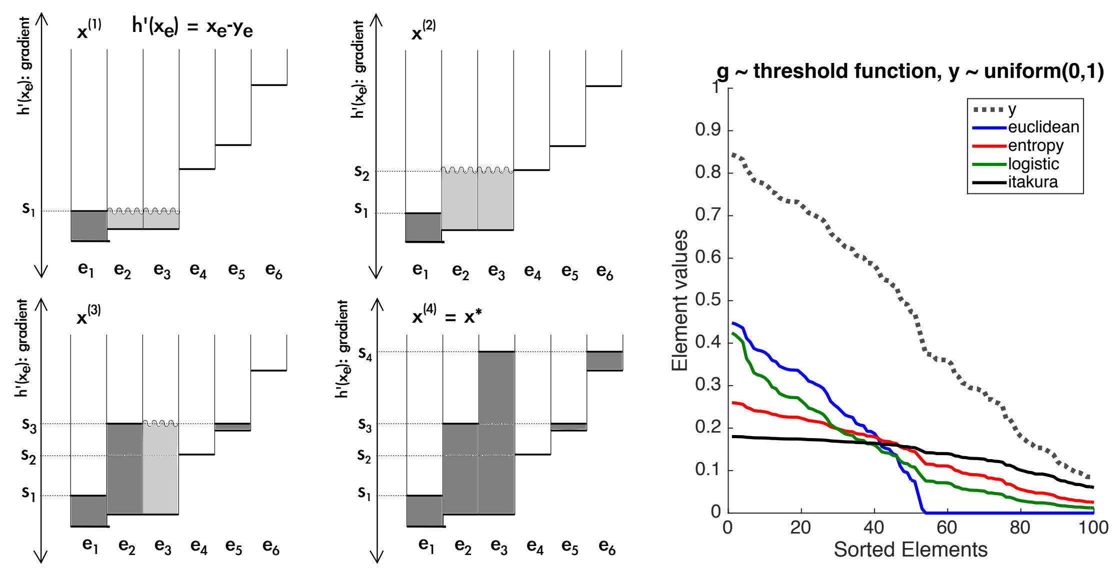

Figure 1: (left) Illustrative water-filling view of the INC-FIX *algorithm for minimizing the squared Euclidean distance $h(x) = \frac{1}{2}\|x - y\|^2$. We start with $x = 0$ and at any time in the algorithm $h'(x_e) = x_e - y_e$. Squiggly lines depict how the point moves in time as the gradient value increases. First, $\{e_1\}$ becomes tight ($x(e_1) = f(\{e_1\})$), next iteration $\{e_1, e_4\}$ are tight, then the tight set grows to $\{e_1, e_2, e_4, e_5\}$ and finally the optimum is reached: $x^{(4)} = x^*$ when all the elements become tight and the gradient is $\nabla h(x^{(4)}) = (s_1, s_3, s_4, s_2, s_3, s_4)^T$. (right) A vector y (generated from a uniform distribution) is projected under different divergences (Euclidean, entropy, logistic and Itakura-Saito) onto a cardinality-based submodular base polytope (generated using a threshold function g) over a ground set of 100 elements.*

[6] f is monotone if $f(A) \leq f(B)$ for all $A \subseteq B \subseteq E$. For any non-negative submodular function f, we can consider a corresponding monotone submodular function $\bar{f}$ such that $P(f) = P(\bar{f})$ (see e.g., Section 44.4 of [30]), where $P(f)$ is the independent set polytope defined as $P(f) = \{x \in \mathbb{R}_+^{\mathrm{E}} : x(U) \leq f(U) \; \forall \; U \subseteq E\}$.

Theorem 2.1. *Consider any strongly convex separable function $h : \mathcal{D} \to \mathbb{R}$, and monotone submodular function $f : 2^E \to \mathbb{R}$ with $f(\emptyset) = 0$. Assume $P(f) \subseteq \overline{\mathcal{D}}$ and $\nabla h(\mathcal{D}) = \mathbb{R}^E$. For $x^* \in \mathbb{R}^E$, let $F_1, F_2, \ldots, F_k$ be a partition of the ground set E such that $(\nabla h(x^*))_e = c_i$ for all $e \in F_i$ and $c_i < c_j$ for $i < j$. Then, $x^* = \text{argmin}_{z \in B(f)} h(z)$ if and only if x^* lies in the face H_{opt} of $B(f)$ given by $H_{opt} := \{z \in B(f) | \, z(F_1 \cup \ldots \cup F_i) = f(F_1 \cup \ldots \cup F_i) \; \forall \, 1 \leq i \leq k\}$.*

Bregman Projections over Base Polytopes For minimizing the squared Euclidean distance and the KL-divergence, the INC-FIX algorithm reduces to starting from a point inside $P(f)$ and moving along lines[7] $\chi(M)$ and $y \cdot \chi(M)$ respectively, for a certain subset of elements $M \subseteq E$. In each iteration, the size of M reduces by at least one element, leading to termination of the algorithm after at most $|E|$ line searches. Each of these line searches inside $P(f)$ can be performed using the discrete Newton method [4] that requires one parametric submodular function minimization (PSFM) (for e.g. [26]) or $O(|E|)$ submodular function minimizations (SFM) (for e.g. [17]). The running time for INC-FIX is $\tilde{O}(|E|^5\gamma + |E|^6)$ where γ is the time required for one function evaluation, using the algorithm for SFM [17]. This approach works more generally for separable convex functions h whenever the increase in the gradient space corresponds to moving along a line in $P(f)$.

Cardinality-based Submodular Functions A submodular function is cardinality-based if $f(S) = g(|S|)$ $(S \subseteq E)$ for some concave function $g : \mathbb{N} \to \mathbb{R}$. We consider *uniformly separable* mirror maps given by $\omega(x) = \sum_{e \in E} w(x(e))$ for $x \in \mathbb{R}^E$ where $w : \mathcal{D}_w \to \mathbb{R}$ is strongly-convex (Table 2). To minimize the divergence with respect to a point $y \in \mathbb{R}^E$, we give a modification of the INC-FIX algorithm that starts by selecting a total order $\succ$ on the elements in E such that $s \succ t$ whenever $y(s) > y(t)$. We show that each intermediate iterate x in the algorithm satisfies $x(s) \geq x(t)$ whenever $s \succ t$ for all $s, t \in E$. Because of this property, we can check for tight constraints in $P(f)$ efficiently by considering the top k elements of each iterate (as the submodular function is cardinality-based). We give the complete description of this algorithm in Algorithm 2 (in the appendix). We show that the algorithm simplifies to have a running time of $O(|E|^2)$ for any uniformly separable mirror map over cardinality-based submodular base polytopes, assuming an oracle access to solve the following problem:

$$\text{Find } \; \epsilon : \sum_{e \in M} (w')^{-1}(\epsilon + w'(y_e)) = c \tag{1}$$

for a given $c \geq 0$ and $M \subseteq E$. Since w' and $(w')^{-1}$ are increasing functions, binary search can be used to solve this subproblem. For minimizing squared Euclidean distance and KL-divergence, this subproblem can be solved exactly in constant time. In Figure 1, we consider a vector $y \in [0, 1]^{100}$ generated from the uniform distribution that is sorted so that $y(e_1) \geq y(e_2) \ldots \geq y(e_{100})$ and project it under different divergences onto the base polytope of a submodular function given by $f(S) = g(|S|)$ where $g(|S|) = \min(0.7|S|, 14)$. We plot the values of 100 elements in the order of decreasing y-values, in Figure 1, and note that the projected points satisfy the same ordering.

Rounding to the base polytope A nice property of the INC-FIX algorithm is that once an element value is fixed, it stays at that value for the rest of the algorithm. In any iteration i of the INC-FIX algorithm ($1 \leq i \leq |E|$), let the iterate be $x^{(i)}$ with the maximum set of tight elements T_i. Then, $x^{(i)}(e) = x^*(e)$ for all $e \in T_i$ where $x^* = \arg\min_{z \in B(f)} h(z)$. This gives a lower bound when minimizing Bregman divergences: $h(x^*) \geq \sum_{e \in T_i} h_e(x_e^{(i)})$ that converges monotonically to the optimal solution value. Further, consider any ordering on the elements of the ground set $E = \{e_1, e_2, \ldots, e_n\}$ such that $T_i = \{e_1, e_2, \ldots, e_{|T_i|}\}$. Then, one can round $x^{(i)}$ to a point x in the base polytope $B(f)$ by simply setting $x(e) = x^{(i)}(e)$ for $e \in T_i$ and $x(e_j) = f(\{e_1, \ldots e_j\}) - f(\{e_1, \ldots, e_{j-1}\})$ for all $j > |T_i|$. For minimizing Bregman divergences, this gives a practical way of determining gap from optimality and terminating the algorithm with an approximate projection.

References

[1] N. Ailon. Improved Bounds for Online Learning Over the Permutahedron and Other Ranking Polytopes. *AISTATS*, 2014.

[7] By moving along a line d, we mean finding $\max \delta$ such that $x + \delta d \in P(f)$ for some given $x \in P(f)$. By $\chi(M)$ we mean the characteristic vector of $M \subseteq E$ and by $y \cdot \chi(M)$ we mean the vector $d(e) = y(e)$ if $e \in M$, $d(e) = 0$ otherwise.

[2] F. Bach. Learning with submodular functions: A convex optimization perspective. *arXiv:1111.6453*, 2011.
[3] A. Beck and M. Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.
[4] W. Dinkelbach. On nonlinear fractional programming. *Management Science*, 13(7):492–498, 1967.
[5] J. Djolonga and A. Krause. From MAP to marginals: Variational inference in bayesian submodular models. *NIPS*, 2014.
[6] J. Edmonds. Submodular functions, matroids, and certain polyhedra. *Combinatorial structures and their applications*, pages 69–87, 1970.
[7] D. Chakrabarty et al. Design is as easy as optimization. *ALP*, pages 477–488, 2006.
[8] D. Suehiro et al. Online prediction under submodular constraints. *ALT*, 2012.
[9] J. Audibert et al. Regret in online combinatorial optimization. *Math. of OR*, 39(1):31–45, 2013.
[10] N. Immorlica et al. Dueling algorithms. *STOC*, 2011.
[11] N. Srebro et al. On the universality of online mirror descent. *NIPS*, 2011.
[12] S. Yasutake et al. Online linear optimization over permutations. *Algorithms and Computation*, 2011.
[13] T. Koo et al. Structured prediction models via the matrix-tree theorem. *EMNLP-CoNLL*, 2007.
[14] V. Syrgkanis et al. Fast convergence of regularized learning in games. *NIPS*, 2015.
[15] W. Krichene et al. Efficient Bregman projections onto the simplex. *CDC*, 2015.
[16] Y. T. Lee et al. A faster cutting plane method & implications for combinatorial and convex optimization. *FOCS*, 2015.
[17] Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.
[18] S. Fujishige. Lexicographically optimal base of a polymatroid with respect to a weight vector. *Mathematics of Operations Research*, 1980.
[19] H. Groenevelt. Two algorithms for maximizing a separable concave function over a polymatroid feasible region. *EJOR*, 54(2):227–236, 1991.
[20] D. P. Helmbold and M. K. Warmuth. Learning permutations with exponential weights. *JMLR*, 2009.
[21] W. M. Koolen, M. K. Warmuth, and J. Kivinen. Hedging Structured Concepts. *COLT*, 2010.
[22] K. Nagano. On convex minimization over base polytopes. *IPCO*, 2007.
[23] K. Nagano. A strongly polynomial algorithm for line search in submodular polyhedra. *Discrete Optimization*, 4(3):349–359, 2007.
[24] K. Nagano and Y. Kawahara. Structured convex optimization under submodular constraints. *UAI*, 2013.
[25] A. Nemirovski. Prox-method with rate of convergence O(1/t) for variational inequalities with Lipschitz continuous monotone operators. *SIAM Journal on Optimization*, 2004.
[26] A. Nemirovski and D. B. Yudin. Problem complexity and method efficiency in optimization. 1983.
[27] A. Rakhlin and K. Sridharan. Optimization, learning, and games with predictable sequences. *NIPS*, 2013.
[28] A. Schrijver. *Combinatorial optimization: polyhedra and efficiency*. Springer, 2003.
[29] M. K. Warmuth and D. Kuzmin. Randomized PCA algorithms with regret bounds that are logarithmic in the dimension. In *NIPS*, 2006.

A Convex Minimization on Base Polytopes of Polymatroids

Technical Description of the INC-FIX **algorithm:** The complete description of the INC-FIX algorithm is given in Algorithm 1. We refer to the initial starting point as $x^{(0)}$. The algorithm constructs a sequence of points $x^{(0)}, x^{(1)}, \ldots, x^{(k)} = x^*$ in $P(f)$. At the beginning of iteration i, the set of *non-fixed* elements whose value can potentially be increased without violating any constraint is referred to as N_{i-1}. The iterate $x^{(i)}$ is obtained by *increasing* the value of minimum weight elements of $x^{(i-1)}$ in N_{i-1} weighted by $(\nabla h(x))_e$ such that the resulting point stays in $P(f)$. Iteration i of the main loop ends when some non-fixed element becomes tight and we *fix* the value on these elements by updating N_i. We continue until all the elements are fixed, i.e., $N_i = \emptyset$. We denote by $T(x)$ the maximal set of tight elements[8] in x (which is unique by submodularity of f).

Technical details for the Card-Inc-Fix algorithm: The algorithm starts by selecting a total order $\succ$ on the elements in E such that $s \succ t$ whenever $y(s) > y(t)$. We can show that each increase step in the INC-FIX algorithm results in iterates $x^{(0)}, x^{(1)}, \ldots$ such that $x^{(i)}(s) \geq x^{(i)}(t)$ whenever $s \succ t$ for all $s, t \in E$, for each iteration i. Since the value of $x^{(i)}$ follows the ordering $\succ$, we can check for tight constraints by considering the top k elements of each iterate (as the submodular function is cardinality-based). Furthermore, the maximal tight set $T(x)$ is simply the largest $|T(x)|$ elements of x, simplifying computations further. We give the complete description of the CARD-INC-FIX

[8]A set $S \subseteq E$ is tight if $x(S) = f(S)$.

Algorithm 1: INC-FIX

input : $f : 2^E \to \mathbb{R}$, $h = \sum_{e \in E} h_e$, and input $x^{(0)}$
output : $x^* = \arg\min_{z \in B(f)} \sum_e h_e(z(e))$
$N_0 = E, i = 0$;
repeat // Main loop starts
  $i \leftarrow i + 1, x = x^{(i-1)}$
  $M = \text{argmin}_{e \in N_{i-1}} \nabla(h(x))_e$ // M is the minimum weight elements of x
  while $T(x) \cap M = \emptyset$ **do** // Inner loop starts
    $\epsilon_1 = \max\{\delta : (\nabla h)^{-1}(\nabla h(x) + \delta \chi(M)) \in P(f)\}$ // Maximum possible increase in gradient while staying in P(f)
    $\epsilon_2 = \min_{e \in N_{i-1} \setminus M} (\nabla h(x))_e - \min_{e \in N_{i-1}} (\nabla h(x))_e$ // Gap to 2nd highest gradient
    $x \leftarrow (\nabla h)^{-1}(\nabla h(x) + \min(\epsilon_1, \epsilon_2)\chi(M))$ // Increase till M changes or new tight constraint
    $M = \text{argmin}_{e \in N_{i-1}} (\nabla h(x))_e$ // Update set M of minimum weight elements
  end
  $x^{(i)} = x, M_i = M$ // Bookkeeping
  $N_i = N_{i-1} \setminus (M_i \cap T(x^{(i)}))$ // Fix tight elements in M_i
until $N_i = \emptyset$; // Until no element can be increased
Return $x^* = x^{(i)}$.

algorithm in Algorithm 2. Let $F_w(\epsilon, y)$ be the value an element must be raised to such that the gradient of the divergence with respect to y is ϵ, i.e. $F_w(\epsilon, y) = (w')^{-1}(\epsilon + w'(y))$.

Algorithm 2: CARD-FIX

1 **input** : $f : f(S) = g(|S|)$ for $S \subseteq E$, $w(\cdot) : \mathcal{D}_w \to \mathbb{R}$, $\omega(x) = \sum_e w(x(e))$, $y \in \mathbb{R}^E$, and input $x^{(0)}$
2 **output** : $x^* = \text{argmin}_{z \in B(f)} D_\omega(z, y)$
3 $i = 0$
4 Set $\succ$ s.t. $s \succ t$ whenever $y(s) > y(t)$ $(\forall s, t \in E)$, let $E = \{e_1 \succ e_2 \ldots \succ e_m\}$ $(m = |E|)$
5 **repeat**
6   $i \leftarrow i + 1, x = x^{(i-1)}$
7   For $k \in \{|T(x)| + 1, \ldots, m\}$, set $\epsilon_k : \sum_{e = e_{|T(x)|+1}}^{e_k} F_w(\epsilon_k, y_e) = g(k) - g(|T(x)|)$
8   $\epsilon^{(i)} = \min_{|T(x)|+1 \le k \le m} \epsilon_k$
9   $x^{(i)}(e) \leftarrow \begin{cases} x(e) & \text{for } e \in T(x), \\ \max\{F_w(\epsilon^{(i)}, y_e), x(e)\} & \text{for } e \in E \setminus T(x) \end{cases}$
10 **until** $T(x) = E$;
11 Return $x^* = x^{(i)}$.

We now state the main theorem about the correctness of the CARD-INC-FIX algorithm.

Theorem A.1. *Consider a submodular function f cardinality-based, mirror map $\omega(x) = \sum_{e \in E} w(x(e))$, and $w : \mathcal{D}_w \to \mathbb{R}$ strongly convex. Let $\mathcal{D}_w$ be a convex open set in $\mathbb{R}$, $P(f) \subseteq \overline{\mathcal{D}}$, $B(f) \cap \mathcal{D} \neq \emptyset$, $w'(\mathcal{D}_w) = \mathbb{R}$ such that there exists a valid starting point. Then, the output of* CARD-INC-FIX *algorithm is $x^* = \text{argmin}_{z \in B(f)} D_\omega(z, y)$ in running time $O(|E|^2)$.*

The CARD-INC-FIX algorithm can be seen to be a simplified version to INC-FIX that uses the properties of cardinality-based submodular functions. Note that since F_w is an increasing functions of ϵ_k, binary search or Newton's method can be used to solve for ϵ_k in the algorithm. The algorithm simplifies to have a running time of $O(|E|^2)$ for any uniformly separable mirror map over cardinality-based submodular base polytopes, assuming an oracle access to solve for step (7). We refer to the full version of the paper for complete proofs and details.