Supermodular Approximation of Norms and Applications

Thomas Kesselheim
Institute of Computer Science,
University of Bonn,
Bonn, Germany
thomas.kesselheim@uni-bonn.de

Marco Molinaro
Microsoft Research and PUC-Rio,
Rio de Janeiro, Brazil
mmolinaro@microsoft.com

Sahil Singla
School of Computer Science,
Georgia Tech,
Atlanta, USA
ssingla@gatech.edu

ABSTRACT

Many classical problems in theoretical computer science involve norms, even if implicitly; for example, both XOS functions and downward-closed sets are equivalent to some norms. The last decade has seen a lot of interest in designing algorithms beyond the standard ℓ_p norms $\|\cdot\|_p$. Despite notable advancements, many existing methods remain tailored to specific problems, leaving a broader applicability to general norms less understood. This paper investigates the intrinsic properties of ℓ_p norms that facilitate their widespread use and seeks to abstract these qualities to a more general setting.

We identify *supermodularity*—often reserved for combinatorial set functions and characterized by monotone gradients—as a defining feature beneficial for $\|\cdot\|_p^p$. We introduce the notion of p-supermodularity for norms, asserting that a norm is p-supermodular if its p^{th} power function exhibits supermodularity. The association of supermodularity with norms offers a new lens through which to view and construct algorithms.

Our work demonstrates that for a large class of problems p-supermodularity is a sufficient criterion for developing good algorithms. This is either by reframing existing algorithms for problems like Online Load-Balancing and Bandits with Knapsacks through a supermodular lens, or by introducing novel analyses for problems such as Online Covering, Online Packing, and Stochastic Probing. Moreover, we prove that every symmetric norm can be approximated by a p-supermodular norm. Together, these recover and extend several existing results, and support p-supermodularity as a unified theoretical framework for optimization challenges centered around norm-related problems.

CCS CONCEPTS

• **Theory of computation → Online algorithms**; *Approximation algorithms analysis.*

KEYWORDS

Online Algorithms, Set Cover, Packing LPs, Orlicz norms, Stochastic Probing, Bandits with Knapsacks, Load Balancing

ACM Reference Format:
Thomas Kesselheim, Marco Molinaro, and Sahil Singla. 2024. Supermodular Approximation of Norms and Applications. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC '24), June 24–28, 2024, Vancouver, BC, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3618260.3649734

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
STOC '24, June 24–28, 2024, Vancouver, BC, Canada
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0383-6/24/06
https://doi.org/10.1145/3618260.3649734

1 INTRODUCTION

Many classical problems in theoretical computer science are framed in terms of optimizing norm objectives. For instance, Load-Balancing involves minimizing the maximum machine load, which is an ℓ_∞ objective, while Set Cover aims at minimizing the ℓ_1 objective, or the number of selected sets. However, contemporary applications, such as energy-efficient scheduling [2], network routing [24], paging [39], and budget allocation [1], demand algorithms that are capable of handling more complex objectives. Norms also underline other seemingly unrelated concepts in computer science, such as XOS functions from algorithmic game theory (both are max of linear functions) and downward-closed constraints from combinatorial optimization (the downward-closed set corresponds to the unit ball of the norm); these connections are further discussed in Section 1.4.

Hence, ongoing efforts have focused on designing good algorithms for general norm objectives. Notably, the last decade has seen a lot of progress in this direction for the class of *symmetric norms*—those invariant to coordinate permutations. Examples include ℓ_p norms, Top-k norm, and Orlicz norms. They offer rich possibilities, e.g., enabling the simultaneous capture of multiple symmetric norm objectives, as their maximum is also a symmetric norm. We have seen the fruit of this in algorithms for a range of applications like Load-Balancing [17, 18], Stochastic Probing [45], Bandits with Knapsacks [35], clustering [17, 18], nearest-neighbor search [5, 6], and linear regression [4, 48].

Despite the above progress, our understanding of applying algorithms beyond ℓ_p norms remains incomplete. For instance, while [9] (where 3 independent papers were merged) provide an algorithm for Online Cover with ℓ_p norms, which was extended to sum of ℓ_p norms in [44], the extension to general symmetric norms is unresolved. Indeed, [44] posed as an open question whether good Online Cover algorithms exist for more general norms. Other less understood applications with norms include Online Packing [14] and Stochastic Probing [28].

A notable limitation of current techniques extending beyond ℓ_p norms is that they are often ad-hoc. Our aim is to create a unified framework that provides a better understanding of norms in this context, simplifies proofs, and enhances generalizability.

> *What properties of ℓ_p norms make them amenable to various applications? Can we reduce the problem of designing good algorithms for general norms to ℓ_p norms?*

A common approach taken when working with ℓ_p norms is to instead work with the function $\|x\|_p^p = \sum_i x_i^p$. This function has several nice properties, e.g., it is separable and convex. We want

to understand its fundamental properties that suffice for many applications, hoping that this would allow us to define similar nice functions beyond ℓ_p norms.

We identify Supermodularity, characterized by monotone gradients, as a particularly valuable property of $\|x\|_p^p$. This may sound intriguing because Supermodularity is typically associated with combinatorial set functions and not a priori norms. This is perhaps because all norms, except for scalings of ℓ_1, are *not* Supermodular. We therefore propose that a norm $\|\cdot\|$ is p-Supermodular if $\|\cdot\|^p$ exhibits Supermodularity.

We show that for a large class of problems involving norms or equivalent objects, p-Supermodularity suffices to design good algorithms. This is either by reframing existing algorithms for problems like Online Load-Balancing [35] and Bandits with Knapsacks [32, 36] through a Supermodular lens or by introducing novel analyses for problems such as Online Covering [9], Online Packing [14], and Stochastic Probing [28, 45].

Moreover, we demonstrate that p-Supermodular approximations of norms are possible for large classes of norms, especially for all symmetric norms. Our approach paves the path for a unified approach to algorithm design involving norms and for obtaining guarantees that only depend polylogarithmically on the number of dimensions n. In particular, it can bypass the limitations of ubiquitous approaches like the use of "concentration + union bound" or Multiplicative Weights Update, that typically cannot give bounds depending only on the ambient dimension (they usually depend on the number of linear inequalities/constraints that define the norm/set); we expand on this a bit later.

1.1 p-Supermodularity and a Quick Application

Throughout the paper, we only deal with non-negative vectors, i.e., $x \in \mathbb{R}_+^n$, and monotone norms, namely those where $\|x\| \geq \|y\|$ if $x \geq y$.

We now give the central definition, p-Supermodularity: a monotone norm $\|\cdot\|$ is p-Supermodular if its p-th power $\|\cdot\|^p$ has increasing marginal gains (a.k.a. supermodularity).

Definition 1.1 (p-Supermodularity). A monotone norm $\|\cdot\|$ is p-Supermodular for $p \geq 1$ if for all $u, v, w \in \mathbb{R}_+^n$,

$$\|u+v+w\|^p - \|u+v\|^p \geq \|u+w\|^p - \|u\|^p.$$

As an example, ℓ_p norms are p-Supermodular (follows from convexity of x^p). It may not be immediately clear, but the larger the p, the weaker this condition is and easier to satisfy (but the guarantees of the algorithm also become weaker as p grows). In Section 2.1 we present an in-depth discussion of p-Supermodularity, including this and other properties, equivalent characterizations, how to create new p-Supermodular norms from old ones, etc.

But to give a quick illustration of why p-Supermodularity is useful, we consider the classic *Online Load-Balancing* problem [8, 10]. In this problem, there are T jobs arriving one-by-one that are to be scheduled on n machines. On arrival, job $t \in [T]$ reveals how much size $p_{ti} \in \mathbb{R}_+$ it takes if executed on machine $i \in [n]$. Given an n-dimensional norm $\|\cdot\|$, the goal is to find an online assignment to minimize the norm of the load vector, i.e., $\|\Lambda_T\|$ where the i-th coordinate of Λ_T is the sum of sizes of the jobs assigned to the i-th machine. The following simple argument shows why p-Supermodularity implies a good algorithm for Online Load-Balancing.

Theorem 1.2. *For Online Load-Balancing problem with a norm objective that is p-*Supermodular*, there is an $O(p)$-competitive algorithm.*

Proof. The algorithm is simple: be greedy with respect to $\|\cdot\|$, i.e., allocate job t to a machine such that the increase in the norm of load vector is the smallest, breaking ties arbitrarily.

For the analysis, let $v_t \in \mathbb{R}_+^n$ be the load vector that the algorithm incurs at time t and $\Lambda_t := v_1 + \ldots + v_t$, and let v_t^* and Λ_t^* be defined analogously for the hindsight optimal solution. Then the cost of the algorithm to the power of p is

$$\begin{aligned}
\|\Lambda_T\|^p &= \sum_t \left(\|\Lambda_t\|^p - \|\Lambda_{t-1}\|^p \right) \\
&\leq \sum_t \left(\|\Lambda_{t-1} + v_t^*\|^p - \|\Lambda_{t-1}\|^p \right) \\
&\leq \sum_t \left(\|\Lambda_T + \Lambda_{t-1}^* + v_t^*\|^p - \|\Lambda_T + \Lambda_{t-1}^*\|^p \right) \\
&= \|\Lambda_T + \Lambda_T^*\|^p - \|\Lambda_T\|^p,
\end{aligned}$$

where the first inequality follows from the greedyness of the algorithm and the second inequality from p-Supermodularity. Rearranging and taking p-th root gives

$$2^{1/p}\|\Lambda_T\| \leq \|\Lambda_T + \Lambda_T^*\| \leq \|\Lambda_T\| + \|\Lambda_T^*\|.$$

Thus, $\|\Lambda_T\| \leq \frac{1}{2^{1/p}-1}\|\Lambda_T^*\| = O(p) \cdot \|\Lambda_T^*\|$ as desired. □

Since ℓ_p norms are p-Supermodular, we obtain $O(p)$-competitive algorithms for Online Load-Balancing with these norms, implying the results of [8, 10].

1.2 p-Supermodular Approximation and our Technique via Orlicz Norms

One difficulty is that many norms (e.g., ℓ_∞) are not p-Supermodular for a reasonable p (e.g., polylogarithmic in the number of dimensions n). Indeed, the greedy algorithm for online load balancing is known to be $\Omega(n)$-competitive for ℓ_∞ [8]. However, in such cases one would like to *approximate* the original norm by a p-Supermodular norm before running the algorithm; e.g., approximate ℓ_∞ by $\ell_{\log n}$.

One of our main contributions is showing that such an approximation exists for large classes of norms. Formally, we say that a norm $|||\cdot|||$ α-approximates another norm $\|\cdot\|$ if

$$\|x\| \leq |||x||| \leq \alpha \cdot \|x\| \qquad \text{for all } x \in \mathbb{R}_+^n.$$

As our first main result (in Section 2), we show that all symmetric norms can be approximated by an $O(\log n)$-Supermodular norm.

Theorem 1.1. *Every monotone symmetric norm $\|\cdot\|$ in n dimensions can be $O(\log n)$-approximated by an $O(\log n)$-*Supermodular *norm.*

Moreover, this approximation can be done efficiently given Ball-Optimization oracle[1] access to the norm $\|\cdot\|$. This result plays

[1]We use the definition in [17], whereby Ball-Optimization oracle allows us to compute $\max_{v:\|v\|\leq 1}\langle x, v\rangle$ for any vector $x \in \mathbb{R}^n$ with a single oracle call.

a crucial role not only in allowing us to rederive many existing results for symmetric norms in a unified way, but also to obtain new results where previously general symmetric norms could not be handled.

We now give a high-level idea of the different steps in the proof of Theorem 1.1.

Reduction to Top-k norms. The reason why general norms are often difficult to work with is that they cannot be easily described. An approach that has been widely successful when dealing with symmetric norms is to instead work with Top-k norms— sum of the largest k coordinates of a non-negative vector. Besides giving a natural way to interpolate between ℓ_1 and ℓ_∞, they actually form a "basis" for all symmetric norms. In particular, it is known that any symmetric norm can be $O(\log n)$-approximated by the max of polynomially many (weighted) Top-k norms (see Lemma 2.15). Leveraging this property, we reduce our problem in that of finding p-Supermodular approximations of Top-k norms.

Our Approach via Orlicz Norms. Even though Top-k norms have a very simple structure, it is still not clear how to design p-Supermodular approximations for them. Not only thinking about p-th power of functions in high dimensional setting is not easy, but there is no constant or "wiggle room" in the definition of p-Supermodularity to absorb errors. Our main idea to overcome this is to instead work with *Orlicz norms* (defined in Section 2.2). These norms are fundamental objects in functional analysis (e.g., see book [29]) and have also found use in statistics and computer science; see for example [4, 48] for their application in regression. Orlicz functions are much easier to work with because they are defined via a 1-dimensional function $\mathbb{R}_+ \to \mathbb{R}_+$.

So our next step is showing that any Top-k norm can be $O(1)$-approximated by an Orlicz norm. This effectively reduce our task of designing a p-Supermodular approximation from an n-dimensional situation to a 1-dimensional situation.

Approximating Orlicz Norms. The last step is showing that every Orlicz norm can be approximated by a p-Supermodular one.

Theorem 1.2. *Every Orlicz norm $\|\cdot\|_G$ in n-dimensions can be $O(1)$-approximated pointwise by a (twice differentiable) $O(\log n)$-*Supermodular *norm.*

As an example, an immediate corollary of this result along with Theorem 1.2 is an $O(\log n)$-competitive algorithm for Online Load-Balancing with an Orlicz norm objective.

Our key handle for approaching Theorem 1.2 is the proof of a sufficient guarantee for an Orlicz norm to be p-Supermodular: the 1-dimensional function G generating it should grow "at most like a polynomial of power p" (Lemma 2.9). Then the construction of the approximation in the theorem proceeds in three steps. First, we simplify the structure of the Orlicz function G by approximating it with a sum of (increasing) "hinge" functions $\tilde{G}(t) := \sum_i \tilde{g}_i(t)$. These hinge function, by definition, have a sharp "kink", hence do not satisfy the requisite growth condition. Thus, the next step is to approximate them by smoother functions $f_i(t)$ that grow at most like power p. The standard smooth approximations of hinge functions (e.g., Hubber loss) do not give the desired approximation properties, so we design an approximation that depends on the relation between the slope and the location of the kink of the hinge function. Finally, we show that the Orlicz norm $\|\cdot\|_F$, generated by the the function $F(t) = \sum_i f_i(t)$, both approximates $\|\cdot\|_G$ and is $O(\log n)$-Supermodular.

Putting these ideas together, gives the desired approximation of every symmetric norm by an $O(\log n)$-Supermodular one.

1.3 Direct Applications of p-Supermodularity

Next, we detail a variety of applications for p-Supermodular functions. Our discussion includes both reinterpretations of existing algorithms through the lens of Supermodularity and the introduction of novel techniques that leverage Supermodularity to address previously intractable problems. In this section, we discuss applications that immediately follow from prior works due to p-Supermodularity.

1.3.1 Online Covering with a Norm Objective. The OnlineCover problem is defined as follows: a norm $f : \mathbb{R}^n \to \mathbb{R}$ is given upfront, and at each round r a new constraint $\langle A^r, x\rangle \geq 1$ arrives (for some non-negative vector $A^r \in \mathbb{R}^n$). The algorithm needs to maintain a non-negative solution $x \in \mathbb{R}^n_+$ that satisfies the constraints $\langle A_1, y\rangle \geq 1, \ldots, \langle A_r, y\rangle \geq 1$ seen thus far, and is only allowed to increase the values of the variables x over the rounds. The goal is to minimize the cost $f(x)$ of the final solution x.

When the cost function f is linear (i.e., the ℓ_1 norm), this corresponds to the classical problem of Online Covering LPs [3, 15], where $O(\log s)$-competitive algorithms are known (s is the maximum row sparsity) [14, 26]. This was first extended to $O(p \log s)$-competitive algorithms when f is the ℓ_p norm [9], and was later extended to sums of ℓ_p norms [44]. [44] posed as an open question whether good online coverage algorithms exist outside of ℓ_p-based norms. The following result, which follows directly by applying the algorithm of [9] to the p-Supermodular approximations of Orlicz and symmetric norms provided by Theorem 1.2 and Theorem 1.1, shows that this is indeed the case.

Corollary 1.3. *In the OnlineCover problem, if the objective can be α-approximated by a p-*Supermodular *norm then there exists an $O(\alpha p \log s)$-competitive algorithm, where s is the maximum row sparsity. Hence, if the objective is an Orlicz norm then this yields $O(\log n \log s)$ competitive ratio, and if the objective is a symmetric norm then this yields $O(\log^2 n \log s)$ competitive ratio.*

Since ℓ_p-norms are p-Supermodular, this extends the result of [9].

1.3.2 Applications via Gradient Stability: Bandits with Knapsacks or Vector Costs. Recently, [35] introduced the notion of gradient stability of norms and showed that it implies good algorithms for online problems such as Online Load-Balancing, Bandits with Vector Costs, and Bandits with Knapsacks. (Gradient stability, however, does not suffice for other applications in this paper, like for Online Covering, Online Packing, Stochastic Probing, and robust algorithms.) In the full version, we show that gradient stability is (strictly) weaker than p-Supermodularity, and hence we can recover all of the results in [35]. Due to Theorem 1.2 for Orlicz norms, this also improves the approximation factors in all these applications from $O(\log^2 n)$ to $O(\log n)$ for Orlicz norms. See the full version for more details.

1.3.3 Robust Algorithms. Supermodularity also has implications for online problem in stochastic, and even better, *robust* input models. Concretely, consider the Online Load-Balancing problem from Section 1.1, but in the Mixed model where the time steps are partitioned (unbeknownst to the algorithm) into an *adversarial* part and a *stochastic* part, where in the latter jobs are generated i.i.d. from an unknown distribution. Such models that interpolate between the pessimism and optimism of the pure worst-case and stochastic models, respectively, have received significant attention in both online algorithms [7, 12, 21, 33, 34, 37, 40–42] and online learning (see [23] and references within).

Consider the (Generalized)[2] Online Load-Balancing in this model, with processing times normalized to be in $[0, 1]$. For the ℓ_p-norm objective, [43] designed an algorithm with cost most $O(1)\cdot\mathrm{OPT}_{Stoch}+O(\min\{p,\log n\})\cdot\mathrm{OPT}_{Adv}+O(\min\{p,\log m\}\, n^{1/p})$, where OPT_{Adv} and OPT_{Stoch} are the hindsight optimal solutions for the items on each part of the input. That is, the algorithm has strong performance on the "easy" part of the instance, while being robust to "unpredictable" jobs. We extend this result beyond ℓ_p-norm objectives, by applying Theorem 1 of [43] and our p-Supermodular approximation for Orlicz norms from Theorem 1.2.

Corollary 1.4. *Consider the (Generalized) Online Load-Balancing problem in the Mixed model with processing times in* $[0, 1]$. *If the objective function can be α-approximated by a p-*Supermodular *norm $\|\cdot\|$, then there is an algorithm with cost at most $O(\alpha)\,\mathrm{OPT}_{Stoch}+O(\alpha p^2)\,\mathrm{OPT}_{Adv}+O(\alpha p\|\mathbf{1}\|)$. For Orlicz norm objective, this becomes $O(1)\,\mathrm{OPT}_{Stoch}+O(\log^2 n)\,\mathrm{OPT}_{Adv}+O(\log n\cdot\|\mathbf{1}\|)$.*

1.4 New Applications using p-Supermodularity

We discuss applications that require additional work but crucially rely on p-Supermodularity. The details can be found in the full version.

1.4.1 Online Covering with Composition of Norms. To illustrate the general applicability of our ideas, in particular going beyond symmetric norms, let us reconsider the OnlineCover problem but now with "composition of norms" objective. This version of the problem is surprisingly general: its offline version captures the fractional setting of other fundamental problems such as Generalized Load-Balancing [20] and Facility Location.

Formally, in OnlineCover with composition of norms, the objective function is defined by a monotone outer norm $\|\cdot\|$ in $\mathbb{R}^k$, monotone inner norms $f_1,\dots,f_k$ in $\mathbb{R}^n$, and subsets of coordinates $S_1,\dots,S_\ell\subseteq[n]$ to allow the inner norms to only depend on a subset of the coordinates, i.e.,

$$\|f_1(y|_{S_1}),\dots,f_k(y|_{S_k})\|,$$

where $y|_{S_\ell}\in\mathbb{R}^{S_\ell}$ is the sub-vector of y with the coordinates indexed by S_ℓ. The objective function is given upfront, but the constraints $\langle A_1,y\rangle\ge 1,\langle A_2,y\rangle\ge 1,\dots,\langle A_m,y\rangle\ge 1$ arrive in rounds, one-by-one, where $A_r\in[0,1]^n$ is the rth row of A. For each round r, the algorithm needs to maintain a non-negative solution $y\in\mathbb{R}^n_+$ that satisfies the constraints $\langle A_1,y\rangle\ge 1,\dots,\langle A_r,y\rangle\ge 1$ seen thus far, and is only allowed to increase the values of the variables y over the rounds. The goal is to minimize the composed norm objective.

Our next theorem shows that good algorithms exist for OnlineCover even with composition of p-Supermodular norms objectives. (Since this composed norm may not be p-Supermodular, Corollary 1.3 does not apply.)

Theorem 1.3. *If the outer norm $\|\cdot\|$ is p'-*Supermodular *and the inner norms f_ℓ's are p-*Supermodular*, then there is an $O(p'\,p\log^2 d\rho\gamma)$-competitive algorithm for OnlineCover, where d is the maximum between the sparsity of the constraints and the size of the coordinate restrictions, namely $d=\max\{\max_r supp(A_r),\ \max_\ell|S_\ell|\}$, $\rho=\max_{r,i:(A_r)_i\neq 0}\frac{1}{(A_r)_i}$, and $\gamma=\max_\ell\frac{\max_{i\in S_\ell}f_\ell(e_i)}{\min_{i\in S_\ell}f_\ell(e_i)}$.*

Unlike Corollary 1.3 that followed from p-Supermodularity immediately, this result needs new ideas to analyze the algorithm. We combine ideas from Fenchel duality used in [9] with breaking up the evolution of the algorithm into phases where the gradients the norm behaves almost p-Supermodular, inspired by [44] in the ℓ_p-case.

1.4.2 Online Packing. The OnlinePacking problem has the form:

$$\max\langle c,x\rangle \qquad \text{s.t.}\quad Ax\le b \text{ and } x\ge 0, \tag{1}$$

where $c\in\mathbb{R}^T, A\in\mathbb{R}^{\#\text{constraints}\times T}$, and $b\in\mathbb{R}^{\#\text{constraints}}$ have all non-negative entries. At the t-th step, we see the value c_t of the item and its vector size $(a_{1,t},\dots,a_{\#\text{constraints},t})$, and have to immediately set x_t (which cannot be changed later). The classic online primal-dual algorithms were designed to address such problems [14, 15], and we know $O(\log(\rho\cdot\#\text{constraints}))$-competitive algorithms, where $\rho=\max_i\frac{\max_t a_{i,t}/c_t}{\min_{t:a_{i,t}>0}a_{i,t}/c_i}$ is the "width" of the instance.

For many packing problems, however, the # constraints is exponential in number of items T, e.g., matroids are given by $\{\sum_{t\in S}x_t\le r(S),\ \forall S\subseteq[T]\}$ where r is the rank function. In such situations, a competitive ratio that depends logarithmically on the number of constraints is not interesting, and we are interested in obtaining competitive ratios that only depend on the intrinsic dimension of the problem.

More formally, we consider the general OnlinePacking problem of the form:

$$\max\langle c,x\rangle \qquad \text{s.t.}\quad Ax\in P \text{ and } x\ge 0, \tag{2}$$

where P is an n-dimensional downward closed set. Again, T items come one-by-one (along with c_t and $(a_{1,t},\dots,a_{m,t})$) and we need to immediately set x_t. Can we obtain polylog(n,T,ρ)-competitive online algorithms? In the stochastic setting of this problem, where items come in a random order (secretary model) or from known distributions (prophet model), Rubinstein [47] obtained $O(\log^2 T)$-competitive algorithms (see also [1]). But in the adversarial online model, despite being a very basic problem, we do not know of good online algorithms beyond very simple P.

We propose the use of p-Supermodularity as a way of tackling this problem. The connection with norms is because there is a 1-1 equivalence between downward closed sets P and monotone norms, given by the gauge function $\|x\|_P:=\inf\{\alpha>0:\frac{x}{\alpha}\in P\}$, where $x\in P\Leftrightarrow\|x\|_P\le 1$. Thus, the packing constraint $Ax\in P$ in (2) is

[2] This is the generalization where there are k "options" for processing each job, and each option incurs possible different loads on multiple machines.

equivalent to $\|Ax\|_P \le 1$. Our next result illustrates the potential of this approach.

Theorem 1.4. *Consider an instance of the problem* OnlinePacking *where the norm associated with the feasible set P admits an α-approximation by a differentiable p-*Supermodular *norm.*

- *If a β-approximation* $\text{OPT} \le \widetilde{\text{OPT}} \le \beta\text{OPT}$ *of* OPT *is known, then there is an algorithm whose expected value is $O(\alpha) \cdot \max\{p, \log \alpha\beta\}$-competitive.*
- *If no approximation of* OPT *is known, then there is an algorithm whose expected value is $O(\alpha)\cdot\max\{p, \log n\rho\}$-competitive, where ρ is an upper bound on the width $\frac{\max_{i,t}(a_{i,t}\cdot\alpha\,\|e_i\|_P/c_t)}{\min_{i,t:a_{i,t}>0}(a_{i,t}\cdot\|e_i\|_P/c_t)}$.*

When $P = \{x \in \mathbb{R}^n : 0 \le x \le b\}$ in (2), the norm $\|\cdot\|_P$ is just ℓ_∞ with rescaled coordinates. Hence, Theorem 1.4 together with $O(\log n)$-Supermodular approximation of ℓ_∞ gives an $O(\log(n\rho))$-competitive algorithm for the setting of (1), which essentially is the same classical guarantee of [14], albeit with a slightly different notion of width ρ. Moreover, if our Conjecture 1.6 about p-Supermodularity of general monotone norms is true then this gives the desired polylog$(n\rho)$-approx for every downward closed P. As a side comment, this result/technique highlights a fact that we were unaware of, even for the classical problem (1), that if an estimate of OPT within poly(n) factors is available, then one can avoid the dependence on any width parameter ρ.

1.4.3 Adaptivity Gaps and Decoupling Inequalities. We show that p-Supermodularity is related to another fundamental concept, namely the power of adaptivity when making decisions under stochastic uncertainty. To illustrate that, we consider the problem of Stochastic Probing (StochProbing), which was introduced as a generalization of stochastic matching [11, 19] and has been greatly studied in the last decade [13, 25, 27, 28, 45].

In this problem, there are n items with unknown non-negative values $X_1, \ldots, X_n$ that were drawn independently from known distributions. Items need to be *probed* for their values to be revealed. There is a downward-closed family $\mathcal{F} \subseteq [n]$ indicating the feasible sets of probes (e.g., if the items correspond to edges in a graph, $\mathcal{F}$ can say that at most k edges incident on a node can be queried). Finally, there is a monotone function $f : \mathbb{R}_+^n \to \mathbb{R}_+$, and the goal is to probe a set $S \in \mathcal{F}$ of elements so as to maximize $\mathbb{E}f(X_S)$, where X_S has coordinate i equal to X_i if $i \in S$ and 0 otherwise (continuing the graph example, $f(x)$ can be the maximum matching with edge values given by x).

The optimal probing strategy is generally *adaptive*, i.e., it probes elements one at a time and may change its decisions based on the observed values. Since adaptive strategies are complicated (can be an exponential-sized decision tree, and probes cannot be performed in parallel), one often resorts to *non-adaptive* strategies that select the probe set S upfront only based on the value distributions. The fundamental question is how much do we lose by making decisions non-adaptively, i.e., if $\textsc{Adapt}(X, \mathcal{F}, f)$ denotes the value of the optimal adaptive strategy and $\textsc{NonAdapt}(X, \mathcal{F}, f)$ denotes the value of the optimal non-adaptive one, then what is the maximum possible *adaptivity gap* $\frac{\textsc{Adapt}(X,\mathcal{F},f)}{\textsc{NonAdapt}(X,\mathcal{F},f)}$ for a class of instances.

For submodular set functions, the adaptivity gap is known to be 2 [13, 28]. For XOS set functions of width w, [28] showed the adaptivity gap is at most $O(\log w)$, where a width-w XOS set function $f : 2^{[n]} \to R_+$ is a max over w linear set functions. The authors conjectured that the adaptivity gap for all XOS set functions should be poly-logarithmic in n, independent of their width. Since a monotone norm is nothing but a max over linear functions (given by the dual-norm unit ball), they form an extension of XOS set functions from the hypercube to all non-negative real vectors. Thus, the generalized conjecture of [28] is the following:

Conjecture 1.5. *The adaptivity gap for stochastic probing with monotone norms is poly*log n.

We prove this conjecture for Supermodular norms.

Theorem 1.5. *For every p-*Supermodular *objective function f,* StochProbing *has adaptivity gap at most $O(p)$.*

This simultaneously recovers the $O(\log w)$ adaptivity gap result of [28] (via Lemma 2.4) and the result of [45] for all monotone symmetric norms (within polylog(n)). Moreover, if our Conjecture 1.6 about Supermodularity of general monotone norms is true, this would settle the full Conjecture 1.5. Importantly, neither the techniques from [28] nor [45] seem able to prove Conjecture 1.5: the former uses a "concentration + union bound" over the linear functions composing f (leading to the expected $O(\log w)$ loss), and the latter showed an $\Omega(\sqrt{n})$ lower bound for non-symmetric functions with their approach.

The proof of Theorem 1.5 is similar to the Load-Balancing application of Section 1.1: we replace one-by-one the actions of the optimal adaptive strategy Adapt by those of the "hallucination-based" non-adaptive strategy that runs Adapt on "hallucinated samples" $\bar{X}_i$'s (but receives value according to the true item values X_i's). However, additional probabilistic arguments are required; in particular, we need to prove a result of the type "$\mathbb{E}\|V_1 + \ldots + V_n\|^p \lesssim \mathbb{E}\|\bar{V}_1 + \ldots + \bar{V}_n\|^p$ implies $\mathbb{E}\|V_1 + \ldots + V_n\| \lesssim p \cdot \mathbb{E}\|\bar{V}_1 + \ldots + \bar{V}_n\|$", where V_i's and $\bar{V}_i$'s will correspond to Adapt and the hallucinating strategy, respectively. We do this via an interpolation idea inspired by Burkholder [16].

In fact, we prove a more general result than Theorem 1.5 that show the connections with probability and geometry of Banach spaces: a decoupling inequality for *tangent sequences* of random variables (see the full version); these have applications from concentration inequalities [46] to Online Learning [22, 49]. Two sequences of random variables $V_1, V_2, \ldots, V_n$ and $\bar{V}_1, \bar{V}_2, \ldots, \bar{V}_n$ are called *tangent* if conditioned up to time $t-1$, V_t and $\bar{V}_t$ have the same distribution. We show that for such tangent sequences in $\mathbb{R}_+^d$, for a p-Supermodular norm $\|\cdot\|$ we have $\mathbb{E}\|V_1 + \ldots + V_n\| \le O(p) \cdot \mathbb{E}\|\bar{V}_1 + \ldots + \bar{V}_n\|$, independent of the number of dimensions. This complements the (stronger) results known for the so-called UMD Banach spaces [31].[3]

[3]We remark that $\mathbb{R}^d$ equipped with the ℓ_1 norm is not a UMD space, while it is a 1-Supermodular norm, making our assumptions, and conclusions, distinct from this literature.

1.5 Our Conjecture and Future Directions

In this work we demonstrate that p-Supermodularity is widely applicable to many problems involving norm objectives (from online to stochastic and from maximization to minimization problems). Our Theorem 1.1 shows that all symmetric norms have an $O(\log n)$-Supermodular approximation. We conjecture that such an approximation should exist for all norms.

Conjecture 1.6. *Any monotone norm in n dimensions can be poly$\log n$-approximated in the positive orthant by a norm that is poly$\log n$-*Supermodular.

If true, this conjecture will significantly push the boundary of what's known. It is akin to the phenomenon of going "beyond the trivial union bound" that appears in multiple settings. For instance, it will positively resolve the adaptivity gap conjecture of [28] for XOS functions where the current best results depend on the number of linear functions, and it will give online packing/covering algorithms that do not depend on the number of constraints but only on the ambient dimension.

Another interesting future direction is to obtain integral solutions for the OnlineCover problem. Similar to the work of [44], our Corollary 1.3 and Theorem 1.3 can only handle the fractional OnlineCover problem. Unlike the classic online set cover (ℓ_1 objective), where randomized rounding suffices to obtain integral solutions, it is easy to show that we cannot round w.r.t. the natural fractional relaxation of the problem since there is a large integrality gap. Hence, a new idea will be required to capture integrality in the objective.

p-Supermodularity is also related to the classic *Online Linear Optimization* (e.g., see book [30]). For the maximization version of the problem, in the full version we show how to obtain total value at least $(1-\varepsilon)\mathrm{OPT} - \frac{p\cdot D}{\varepsilon}$ when a norm associated to the problem is p-Supermodular, where D is "diameter" parameter. In the case of prediction with experts, this recovers the standard $(1-\varepsilon)\mathrm{OPT} - O(\frac{\log d}{\varepsilon})$ bound (d being the number of experts), and generalizes the result of [42] when the player chooses actions on the ℓ_p ball. This gives an intriguing alternative to the standard methods like Online Mirror Descent and Follow the Perturbed Leader. It would be interesting to find further implications of this result, and more broadly p-Supermodularity, in the future.

In the next section we discuss properties of p-Supermodularity and defer the proofs of the applications to the full version.

2 SUPERMODULAR APPROXIMATION OF NORMS

In this section we discuss p-Supermodularity and how many general norms can be approximated by p-Supermodular norms.

2.1 p-Supermodularity and its Basic Properties

p-Supermodularity can be understood in a natural and more workable manner through the first and second derivatives of the norms; this is the approach we use in most of our results. While norms may not be differentiable, using standard smoothing techniques, every p-Supermodular norm can be $(1+\varepsilon)$-approximated by another p-Supermodular norm that is infinitely differentiable everywhere except at the origin; see the full version.

We give equivalent characterizations of p-Supermodular norms via their gradients and Hessians.

Lemma 2.1 (Equivalent characterizations). *For a differentiable norm $\|\cdot\|$, the following are equivalent:*

- (p-Supermodularity): $\|\cdot\|$ *is* p-Supermodular.
- (Gradient property): $\|\cdot\|^p$ *has monotone gradients over the non-negative orthant, i.e., for all $u, v \in \mathbb{R}^n_+$ and $\forall i \in [n]$,*

$$\nabla_i\Big(\|u+v\|^p\Big) \geq \nabla_i\Big(\|u\|^p\Big) \iff \frac{\nabla_i\|u+v\|}{\nabla_i\|u\|} \geq \Big(\frac{\|u\|}{\|u+v\|}\Big)^{p-1}.$$

- (Hessian property): *If $\|\cdot\|$ is twice differentiable, then these are equivalent to: For all $u \in \mathbb{R}^n_+$ and $\forall i, j \in [n]$,*

$$\nabla^2_{i,j}\Big(\|u\|^p\Big) \geq 0 \iff \nabla^2_{i,j}\|u\| \geq -(p-1)\frac{1}{\|u\|}\nabla_i\|u\|\cdot\nabla_j\|u\|.$$

Proof. The first part of the Gradient property follows when we take $\|w\| \to 0$. For the second part, use $\nabla\|u\|^p = p\cdot\|u\|^{p-1}\cdot\nabla\|u\|$.

The first part of the Hessian property follows from monotonicity of gradients. For the second part, use

$$\frac{1}{p}\nabla^2_{i,j}\Big(\|u\|^p\Big) = \|u\|^{p-2}\cdot\Big((p-1)\cdot\nabla_i\|u\|\cdot\nabla_j\|u\| + \|u\|\cdot\nabla^2_{i,j}\|u\|\Big). \quad \square$$

Two immediate implications of the above equivalence are the following:

Corollary 2.2. *A differentiable* p-Supermodular *norm $\|\cdot\|$ is also p'-*Supermodular *for $p' \geq p$.*

Corollary 2.3. *If $\|\cdot\|\colon \mathbb{R}^n \to \mathbb{R}$ is* p-Supermodular *and $A \in \mathbb{R}^{n\times m}_{\geq 0}$ then the norm $|||x||| := \|Ax\|$ is* p-Supermodular.

As mentioned in the introduction, for every $p \geq 1$ the ℓ_p norm is p-Supermodular. This follows, e.g., from the gradient property of p-Supermodular norms. For $p \geq \log n$, the ℓ_p norm is $O(1)$-approximated by $\ell_{\log n}$. So in particular, ℓ_∞ can be $O(1)$-approximated by $(\log n)$-Supermodular norm. We first generalize this fact (ℓ_∞ is max of n inequalities that are each 1-Supermodular).

Lemma 2.4. *If $f_1, f_2, \ldots, f_w$ are differentiable* p-Supermodular *norms, then the norm $x \mapsto \max_i f_i(x)$ can be 2-approximated by a* $\max\{p, \log w\}$-Supermodular *norm.*

Proof. Let $p' = \max\{p, \log w\}$ and consider the norm $|||x||| := (\sum_i f_i(x)^{p'})^{1/p'}$. As $\max_i f_i(x)^{p'} \leq \sum_i f_i(x)^{p'} \leq w\cdot\max_i f_i(x)^{p'}$, we have

$$\begin{aligned}\max_i f_i(x) = (\max_i f_i(x)^{p'})^{1/p'} &\leq |||x||| \\ &\leq (w\cdot\max_i f_i(x)^{p'})^{1/p'} \\ &= w^{1/p'}\max_i f_i(x) \\ &\leq 2\max_i f_i(x).\end{aligned}$$

Furthermore, for all $u, v \in \mathbb{R}^n_+$, we have

$$\nabla|||u+v|||^{p'} = \sum_i f_i(u+v)^{p'} \geq \sum_i \nabla f_i(u)^{p'} = \nabla|||u|||^{p'},$$

since each f_i is p'-Supermodular. $\square$

An implication of this is that any norm in n dimensions can be $O(1)$-approximated by an n-Supermodular norm. This is because we can find a $\frac{1}{4}$-net $\mathcal{N} \subseteq \mathcal{A}$ of the unit ball of the dual norm of size $2^{O(n)}$. Since, $|||x||| := \max_{a\in\mathcal{N}}\langle a, |x|\rangle$ is an $O(1)$ approximation of $\|x\|$ and $\langle a, |x|\rangle$ is a re-weighted ℓ_1 norm, Lemma 2.4 implies that $|||x|||$ is n-Supermodular norm.

Corollary 2.5. *Any monotone norm in n-dimensions can be $O(1)$-approximated by an n-*Supermodular *norm.*

Although p-Supermodular norms have several nice properties, they also exhibit some strange properties. For instance, sum of two p-Supermodular norms can be very far from being p-Supermodular.

Lemma 2.6. *The norm $\|x\| = \|x\|_1 + \|x\|_2$ is not p-*Supermodular *for any $p = o(\sqrt{n})$.*

Proof. Consider some $i \neq j \in [n]$. By Hessian property in Lemma 2.1, for $\|x\|_1 + \|x\|_2$ to be p-Supermodular, we must have

$$\begin{aligned} -\frac{\nabla_i\|x\|_2 \cdot \nabla_j\|x\|_2}{\|x\|_2} &= \nabla^2_{i,j}\|x\| \\ &\geq -(p-1)\frac{\nabla_i\|x\| \cdot \nabla_j\|x\|}{\|x\|} \\ &= -(p-1)\frac{\left(1+\nabla_i\|x\|_2\right)\cdot\left(1+\nabla_j\|x\|_2\right)}{\|x\|_1+\|x\|_2}. \end{aligned}$$

Since $\nabla_i\|x\|_2 = \frac{x_i}{\|x\|_2}$, we can simplify to get

$$\frac{x_i \cdot x_j}{\|x\|_2^3} \leq (p-1)\cdot\frac{\left(\|x\|_2+x_i\right)\cdot\left(\|x\|_2+x_j\right)}{\left(\|x\|_1+\|x\|_2\right)\cdot\|x\|_2^2}.$$

Now consider the vector $x = (\sqrt{n}, \sqrt{n}, 1, 1, \ldots, 1)$, i.e., a vector having the first two coordinates $\sqrt{n}$ and every other coordinate 1. Note that $\|x\|_1 = \Theta(n)$ and $\|x\|_2 = \Theta(\sqrt{n})$. For $i = 1$ and $j = 2$, the last inequality gives

$$\frac{n}{\Theta(n^{3/2})} \leq (p-1)\cdot\frac{\Theta(\sqrt{n})\cdot\Theta(\sqrt{n})}{\Theta(n)\cdot\Theta(\sqrt{n})^2} = \frac{p-1}{\Theta(n)},$$

which is only possible for $p = \Omega(\sqrt{n})$. □

2.2 Orlicz Norms and a Sufficient Condition for p-Supermodularity

The following class of Orlicz functions and Orlicz norms will play a crucial role in all our norm approximations.

Definition 2.7 (Orlicz Function). A continuous function $G : \mathbb{R}_+ \to \mathbb{R}_+$ is called an *Orlicz function* if it is convex, increasing, and satisfies $G(0) = 0$ and $\lim_{t\to\infty} G(t) \to \infty$.

Definition 2.8 (Orlicz Norm). Given an Orlicz function G, the associated *Orlicz norm* is defined by

$$\|x\|_G := \inf\left\{\alpha > 0 : \sum_i G\left(\frac{|x_i|}{\alpha}\right) \leq 1\right\}.$$

Since we only focus on non-negative vectors, we will ignore throughout the absolute value $|x_i|$.

For example, any ℓ_p is an Orlicz norm when we select $G(t) = t^p$. Orlicz norms are fundamental in functional analysis [38], but have also found versatile applications in TCS. For instance, in regression the choice between ℓ_1 and ℓ_2 norms depends on outliers and stability, so an Orlicz norm based on the popular Huber convex loss function is better suited [4, 48]. Later we will show that Orlicz norms can be used to approximate any symmetric norm.

The following lemma is our main tool for working with Orlicz norms. It states that for such a norm to be p-Supermodular, it suffices that its generating function G grows "at most like power p". The key is that this reduces the analysis of the n-dimensional norms $\|\cdot\|_G$ to the analysis of 1-dimensional functions, which is significantly easier.

Lemma 2.9. *Consider a twice differentiable convex function $G : \mathbb{R}_+ \to \mathbb{R}_+$. If G satisfies*

$$G''(t)\cdot t \leq (p-1)\cdot G'(t)\ \forall t \geq 0,$$

*then the Orlicz norm $\|x\|_G$ is $(2p-1)$-*Supermodular.

Notice that the function $G(t) = t^p$ satisfies this condition, at equality. While in this special case the norm $\|\cdot\|_G = \ell_p$ is p-Supermodular, in general we obtain the slightly weaker conclusion of $(2p-1)$-Supermodularity.

The rest of the subsection proves this lemma. The proof will rely on the Hessian property of p-Supermodular norms. First, we observe the following formula for the gradient of the Orlicz norm $\|\cdot\|_G$; this can be found on page 24 of [38], but we repeat the proof for completeness.

Claim 2.1. *If G is differentiable, then the gradient of the Orlicz norm $\|\cdot\|_G$ is given by*

$$\nabla_i\|x\|_G = \frac{G'\left(\frac{x_i}{\|x\|_G}\right)}{\sum_\ell \frac{x_\ell}{\|x\|_G}\cdot G'\left(\frac{x_\ell}{\|x\|_G}\right)}.$$

Proof. Consider $H(x, c) := \sum_\ell G(\frac{x_\ell}{c})$. Since $H(x, \|x\|_G) = 1$ is constant, we get

$$\begin{aligned} 0 &= \frac{\partial}{\partial x_i}H(x, \|x\|_G) \\ &= \frac{1}{\|x\|_G}G'\left(\frac{x_i}{\|x\|_G}\right) - \sum_\ell\left(G'\left(\frac{x_\ell}{\|x\|_G}\right)\cdot\frac{x_\ell}{\|x\|_G^2}\right)\cdot\nabla_i\|x\|_G. \end{aligned}$$ □

To simplify notation, we define the following.

Definition 2.10. Let

$$\tilde{x}_\ell := \frac{x_\ell}{\|x\|} \quad \text{and} \quad \gamma(x) := \sum_\ell \frac{x_\ell}{\|x\|}\cdot G'\left(\frac{x_\ell}{\|x\|}\right).$$

Hence, $\nabla_i\|x\|_G = \frac{G'(\tilde{x}_i)}{\gamma(x)}$.

Differentiating the expression for the gradient $\nabla_i\|x\|_G$ gives a close-form formula for the Hessian of the Orlicz norm. (To be careful with the chain rules, we use brackets; for example $\nabla_j(g(h(x)))$ to denote the gradient of the composed function $g \circ h$, not of just g.)

Claim 2.2. *If G is twice differentiable, then the Hessian of the norm*

$$\nabla^2_{ij}\|x\| = \frac{1}{\gamma(x)}\cdot\nabla_j(G'(\tilde{x}_i)) - \frac{\nabla_i\|x\|}{\gamma(x)}\cdot\sum_\ell\left(\tilde{x}_\ell\cdot\nabla_j(G'(\tilde{x}_\ell))\right). \quad (3)$$

Before proving the claim (which is mostly algebra), we complete the proof of the lemma.

Proof of Lemma 2.9. When $\ell \neq j$ we have $\nabla_j \tilde{x}_\ell = \nabla_j(\frac{x_\ell}{\|x\|_G}) = -\frac{x_\ell \cdot \nabla_j \|x\|_G}{\|x\|_G^2} = -\tilde{x}_\ell \cdot \frac{\nabla_j \|x\|_G}{\|x\|_G}$, and when $\ell = j$ we get an extra $+\frac{1}{\|x\|_G}$ from the product rule. Letting $\mathbf{1}(\ell = j)$ denote the indicator that $\ell = j$, this implies

$$\nabla_j \tilde{x}_\ell \;=\; -\frac{x_\ell \cdot \nabla_j \|x\|}{\|x\|^2} \;+\; \mathbf{1}(\ell = j) \cdot \frac{1}{\|x\|}. \tag{4}$$

Applying this to (3) and using $\nabla_j(G'(\tilde{x}_\ell)) = G''(\tilde{x}_\ell) \cdot \nabla_j \tilde{x}_\ell$, we get

$$\begin{aligned}
\nabla^2_{ij}\|x\| &= -\frac{G''(\tilde{x}_i) \cdot x_i \cdot \nabla_j \|x\|}{\gamma(x) \cdot \|x\|^2} \;+\; \mathbf{1}(i=j) \cdot \frac{G''(\tilde{x}_i)}{\gamma(x) \cdot \|x\|} \\
&- \frac{\nabla_i \|x\|}{\gamma(x)} \cdot \left[-\sum_\ell \left(\tilde{x}_\ell \cdot G''(\tilde{x}_\ell) \cdot \frac{x_\ell \cdot \nabla_j \|x\|}{\|x\|^2} \right) + \frac{\tilde{x}_j \cdot G''(\tilde{x}_j)}{\|x\|} \right] \\
&\geq -\frac{1}{\|x\|} \left[\nabla_i \|x\| \cdot \frac{\tilde{x}_j \cdot G''(\tilde{x}_j)}{\gamma(x)} + \nabla_j \|x\| \cdot \frac{\tilde{x}_i \cdot G''(\tilde{x}_i)}{\gamma(x)} \right],
\end{aligned} \tag{5}$$

where the inequality uses that the missing terms are non-negative for $x \geq 0$.

Moreover, the assumption on G implies that

$$\frac{\tilde{x}_j \cdot G''(\tilde{x}_j)}{\gamma(x)} \;\leq\; (p-1)\frac{G'(\tilde{x}_j)}{\gamma(x)} \;=\; (p-1)\nabla_j \|x\|.$$

Similarly, we get for i that $\frac{\tilde{x}_i \cdot G''(\tilde{x}_i)}{\gamma(x)} \leq (p-1)\nabla_i \|x\|$. Plugging these bounds into (5) gives

$$\nabla^2_{ij}\|x\| \geq -(2p-2)\frac{1}{\|x\|} \nabla_i \|x\| \cdot \nabla_j \|x\|,$$

which proves Lemma 2.9 by Lemma 2.1. □

Finally, we prove the missing claim.

Proof of Claim 2.2. Differentiating w.r.t. x_j gradient $\nabla_i \|x\|_G = \frac{G'(\tilde{x}_i)}{\gamma(x)}$ from Lemma 2.1 gives

$$\begin{aligned}
\nabla^2_{ij}\|x\|_G &= \frac{1}{\gamma(x)} \cdot \nabla_j(G'(\tilde{x}_i)) \;-\; G'(\tilde{x}_i) \cdot \frac{1}{\gamma(x)^2} \cdot \nabla_j \gamma(x) \\
&= \frac{1}{\gamma(x)} \cdot \nabla_j(G'(\tilde{x}_i)) \;-\; \frac{\nabla_i \|x\|_G}{\gamma(x)} \cdot \nabla_j \gamma(x).
\end{aligned} \tag{6}$$

We expand the gradient $\nabla_j \gamma(x)$ of the second term:

$$\nabla_j \gamma(x) = \sum_\ell \nabla_j \left(\tilde{x}_\ell G'(\tilde{x}_\ell) \right) = \sum_\ell \left(\nabla_j \tilde{x}_\ell G'(\tilde{x}_\ell) + \tilde{x}_\ell \nabla_j (G'(\tilde{x}_\ell)) \right).$$

By (4), we have

$$\begin{aligned}
\sum_\ell \nabla_j \tilde{x}_\ell \cdot G'(\tilde{x}_\ell) &= -\sum_\ell \tilde{x}_\ell \cdot \frac{\nabla_j \|x\|_G}{\|x\|_G} \cdot G'(\tilde{x}_\ell) + \frac{1}{\|x\|_G} \cdot G'(\tilde{x}_j) \\
&= -\frac{\nabla_j \|x\|_G}{\|x\|_G} \cdot \gamma(x) + \frac{G'(\tilde{x}_j)}{\|x\|_G} \quad = \quad 0.
\end{aligned}$$

This implies

$$\nabla_j \gamma(x) \;=\; \sum_\ell \tilde{x}_\ell \cdot \nabla_j (G'(\tilde{x}_\ell)),$$

which proves the claim by substitution in (6). □

2.3 Approximation of Orlicz Norms

This section shows that every Orlicz norm can be approximated by an $O(\log n)$-Supermodular norm.

Theorem 1.2. *Every Orlicz norm $\|\cdot\|_G$ in n-dimensions can be $O(1)$-approximated pointwise by a (twice differentiable) $O(\log n)$-*Supermodular *norm.*

Before giving an overview of the proof of the theorem, it will help the discussion to have the following lemma that shows that to approximate an Orlicz norm $\|\cdot\|_G$, it suffices to approximate the corresponding Orlicz function G.

Lemma 2.11. *Suppose $\tilde{G}$ is an Orlicz function satisfying for all t with $G(t) \leq 1$:*

1. *$G(t) \leq \tilde{G}(t)$.*
2. *$\tilde{G}(t/\gamma) \;\leq\; \alpha\, G(t) + \frac{1}{n}$ for some universal constants $\alpha \geq 0$ and $\gamma \geq 1$.*

Then, $\|x\|_G \leq \|x\|_{\tilde{G}} \leq \gamma(\alpha+1)\|x\|_G$.

Proof. The first inequality $G(t) \leq \tilde{G}(t)$ implies that $\|x\|_G \leq \|x\|_{\tilde{G}}$. Moreover, by convexity and $\alpha \geq 0$, we have $\tilde{G}(\frac{t}{\gamma(\alpha+1)}) \leq (1 - \frac{1}{\alpha+1})\tilde{G}(0) + \frac{1}{\alpha+1}\tilde{G}(t/\gamma) = \frac{1}{\alpha+1}\tilde{G}(t/\gamma)$ since $\tilde{G}$ is an Orlicz function. So,

$$\begin{aligned}
\sum_i \tilde{G}\left(\frac{x_i}{\gamma(\alpha+1)\|x\|_G} \right) &\leq \frac{1}{\alpha+1} \sum_i \tilde{G}\left(\frac{x_i}{\gamma \|x\|_G} \right) \\
&\leq \frac{1}{\alpha+1} \sum_i \left[\alpha \cdot G\left(\frac{x_i}{\|x\|_G} \right) + \frac{1}{n} \right] = 1,
\end{aligned}$$

where the last inequality uses $\gamma \geq 1$. By definition of Orlicz norm, this implies $\|x\|_{\tilde{G}} \leq \gamma(\alpha+1)\|x\|_G$. □

Observe that we do not care how the Orlicz function $\tilde{G}$ behaves after $G(t) > 1$, since these values do not matter for Orlicz norm $\|\cdot\|_G$.

Proof Overview of Theorem 1.2. Given the sufficient condition for p-Supermodularity via the growth rate of the Orlicz function from Lemma 2.9 and Lemma 2.11 above, the proof of Theorem 1.2 involves three steps. First, we simplify the structure of the Orlicz function G by approximating it with a sum of (increasing) "hinge" functions $\tilde{G}(t) := \sum_i \tilde{g}_i(t)$ in the interval where $G(t) \leq 1$. These hinge function by definition have a sharp "kink", hence do not satisfy the requisite growth condition. Thus, the next step is to approximate them by smoother functions $f_i(t)$ that grow at most like power p. However, the standard smooth approximations of hinge functions (e.g. Hubber loss) do not give the desired properties, so we use a subtler approximation that depends on the relation between the slope and the location of the kink of the hinge function (this is because the approximation condition required by Lemma 2.11 is mostly multiplicative, while standard approximations focus on additive error). Finally, we show that the Orlicz norm $\|\cdot\|_F$, where $F(t) = \sum_i f_i(t)$, both approximates $\|\cdot\|_G$ and is $O(\log n)$-Supermodular.

Proof of Theorem 1.2. This first claim gives the desired approximation of G by piecewise linear functions with n slopes.

Claim 2.3. *There are $a_1, \dots, a_n, b_1, \dots, b_n \geq 0$ such that $\tilde{G} : \mathbb{R}_+ \to \mathbb{R}_+$ defined by $\tilde{G}(t) = \sum_{i=1}^n \max\{0, a_i t - b_i\}$ fulfills*

$$\|x\|_G \leq \|x\|_{\tilde{G}} \leq 2\|x\|_G, \ \forall x \in \mathbb{R}_+^n.$$

Proof. Since G is an Orlicz function, it is continuous and satisfies $G(0) = 0$ with $\lim_{t\to\infty} G(t) = \infty$. Hence, there are points $t_0 = 0, t_1, t_2, \dots, t_n \in \mathbb{R}_+$ such that $G(t_i) = \frac{i}{n}$. Choose a_i and b_i such that $a_i t_{i-1} - b_i = 0$ and $a_i t_i - b_i = \frac{1}{n} - \sum_{j<i} a_j (t_i - t_{i-1})$. By this definition $\tilde{G}(t_i) = \sum_{i=1}^n \max\{0, a_i t - b_i\} = G(t_i) = \frac{i}{n}$ for all $i = 0, 1, \dots, n$.

We claim that $G(t) \leq \tilde{G}(t) \leq G(t) + \frac{1}{n}$ for all t with $G(t) \in [0, 1]$. The first inequality follows from the convexity of G, and the second inequality follows because for all $t \in [t_i, t_{i+1}]$ we have $\tilde{G}(t) \leq \tilde{G}(t_{i+1}) = \frac{i+1}{n} \leq G(t) + \frac{1}{n}$. Hence, Lemma 2.11 concludes the proof of the claim. □

Next, we will approximate piecewise linear functions $\max\{0, a_i t - b_i\}$ with Orlicz functions. This approximation will depend on whether $b_i \geq 1$ or not.

Definition 2.12. Let H be the set of indices $i \in [n]$ such that $b_i \geq 1$ and $L = [n] \setminus H$ be the other indices. For $p \geq 2(\ln n) + 1$, define

$$F(t) := \sum_{i=1}^n f_i(t), \text{where } f_i(t) = \begin{cases} 2 \cdot (\frac{2a_i}{b_i+1})^p \cdot t^p & , \text{if } i \in H \\ (b_i^p + (a_i t)^p)^{1/p} - b_i & , \text{if } i \in L \end{cases}.$$

The idea behind this construction is the following: first write $\tilde{g}_i(t) := \max\{0, a_i t - b_i\} = \max\{b_i, a_i t\} - b_i$ and notice that $\tilde{G}(t) = \sum_{i=1}^n \tilde{g}_i(t)$. When $b_i \geq 1$, then the points t where $\tilde{g}_i(t)$ equals 0 and 1 (respectively, $\frac{b_i}{a_i}$ and $\frac{b_i+1}{a_i}$) are within a factor of 2, namely $\tilde{g}_i$ fairly sharply jumps from 0 to 1; in this case, we replace it by the sharply increasing function $f_i(t) = (\frac{2a_i}{b_i+1})^p \cdot t^p$. Otherwise, the function $\tilde{g}_i$ does not increase so sharply and we just replace the maximum in $\tilde{g}_i(t) = \max\{b_i, a_i t\} - b_i$ by the ℓ_p norm to obtain f_i. Then to obtain F, we take the sum of the functions f_i.

We first prove that $f_i(t)$ approximates $\tilde{g}_i(t)$ in a suitable way. We will also show that f_i grows at most like power p. (In the following claim, the intuition behind the truncation $\min\{\cdot, 2\}$ is that in definition of the Orlicz norm, the places where the generating function G is bigger than 1 are not important; instead of 2, one can use any value strictly bigger than 1.)

Claim 2.4. *Consider $p \geq 2(\ln n) + 1$. For all $i \in [n]$, we have*

(1) $f_i(t) \geq \min\{\tilde{g}_i(t), 2\}$ *for all* $t \geq 0$.
(2) $f_i(\frac{t}{4}) \leq 2\tilde{g}_i(t) + \frac{1}{n^2}$ *for all* t *with* $\tilde{g}_i(t) \leq 1$.
(3) $f_i''(t) \cdot t \leq (p-1) \cdot f_i'(t)$ *for all* $t \geq 0$.

Proof. We prove these properties separately for the cases $b_i \geq 1$ and $b_i \in [0, 1)$.

Case 1: $b_i \geq 1$, so $f_i(t) = 2\,(\frac{2a_i}{b_i+1})^p \cdot t^p$.

For Item 1, notice that for $t \in [0, \frac{b_i}{a_i}]$ we have $\min\{\tilde{g}(t), 2\} = 0$ and for $t > \frac{b_i}{a_i}$ we have $\min\{\tilde{g}(t), 2\} \leq 2$, by definition. Since $f_i(t) \geq 0$ for $t \in [0, \frac{b_i}{a_i}]$, and for $t \geq \frac{b_i}{a_i}$

$$f_i(t) \geq 2\left(\frac{2b_i}{b_i+1}\right)^p \geq 2,$$

where the last inequality uses $b_i \geq 1$. Thus, we have $f_i(t) \geq \min\{\tilde{g}_i(t), 2\}$ for all $t \geq 0$.

For Item 2, for all $t \in [0, \tilde{g}_i^{-1}(1)]$ (this interval is the same as $[0, \frac{b_i+1}{a_i}]$) we have

$$f_i(t/4) \leq 2 \cdot \left(\frac{2a_i}{b_i+1}\right)^p \cdot \left(\frac{b_i+1}{4a_i}\right)^p = \frac{1}{2^{p-1}} \leq 2\tilde{g}(t) + \frac{1}{n^2}.$$

Item 3 holds with equality. Namely, by taking the second-derivative of $f_i(t)$, we get

$$f_i''(t) \cdot t = p \cdot (p-1) \cdot 2 \cdot \left(\frac{2a_i}{b_i+1}\right)^p \cdot t^{p-1} = (p-1) \cdot f_i'(t).$$

Case 2: $b_i \in [0, 1)$, so $f_i(t) = (b_i^p + (a_i t)^p)^{1/p} - b_i$.

For Item 1, observe that $f_i(t) = (b_i^p + (a_i t)^p)^{1/p} - b_i \geq \max\{b_i, a_i t\} - b_i = \tilde{g}(t)$.

For Item 2, for all $t \in [0, \frac{2b_i}{a_i})$, we have

$$\begin{aligned} f_i(t/4) &\leq \left((b_i)^p + (b_i/2)^p\right)^{1/p} - b_i = b_i\left(1 + \frac{1}{2^p}\right)^{1/p} - b_i \\ &\leq b_i\left(1 + \frac{1}{p2^p}\right) - b_i \\ &\leq \tilde{g}_i(t) + \frac{1}{n^2}, \end{aligned}$$

where the last inequality uses the fact that we are in a case $b_i \leq 1$. On the other hand, when $t \geq \frac{2b_i}{a_i}$, then $b_i \leq \frac{a_i t}{2}$ and so $\tilde{g}_i(t) = \max\{0, a_i t - b_i\} \geq \frac{a_i t}{2}$; at the same time,

$$\begin{aligned} f_i(t/4) \leq \left((a_i t/2)^p + (a_i t/4)^p\right)^{1/p} &= ((1/2)^p + (1/4)^p)^{1/p} \cdot a_i t \\ &\leq a_i t. \end{aligned}$$

Putting these observations together, gives $f_i(t/4) \leq 2\tilde{g}_i(t)$, proving Item 2.

For Item 3, compute the derivatives to get

$$f_i'(t) = \frac{a_i^p t^{p-1}}{(b_i^p + (a_i t)^p)^{1-\frac{1}{p}}} \text{ and}$$

$$f_i''(t) = \frac{(p-1)a_i^p t^{p-2}}{(b_i^p + (a_i t)^p)^{1-\frac{1}{p}}} - (p-1)\frac{a_i^{2p} t^{2(p-1)}}{(b_i^p + (a_i t)^p)^{2-\frac{1}{p}}}.$$

The last term in $f_i''(t)$ is non-positive, and so it follows that $f_i''(t) \cdot t \leq (p-1) \cdot f_i'(t)$. □

Now we use the last claim to prove that $\|\cdot\|_F$ approximates $\|\cdot\|_{\tilde{G}}$.

Claim 2.5. *If $p \geq \log n + 1$, then for every $x \in \mathbb{R}_+^n$ we have $\|x\|_{\tilde{G}} \leq \|x\|_F \leq 12\|x\|_{\tilde{G}}$.*

Proof. First, from Claim 2.4 we get

$$\begin{aligned} F(t) = \sum_{i=1}^n f_i(t) &\overset{Claim\ 2.4}{\geq} \sum_{i=1}^n \min\{2, \tilde{g}_i(t)\} \\ &\geq \min\left\{2, \sum_{i=1}^n \tilde{g}_i(t)\right\} = \min\{2, \tilde{G}(t)\}. \end{aligned}$$

Moreover, for any t with $1 \geq \tilde{G}(t) \geq \tilde{g}_i(t)$, we have from Claim 2.4 that

$$F(t/4) = \sum_{i=1}^{n} f_i(t/4) \overset{Claim\ 2.4}{\leq} \sum_{i=1}^{n} \left(2\tilde{g}_i(t) + \frac{1}{n^2}\right) = 2\tilde{G}(t) + \frac{1}{n}.$$

Now, applying Lemma 2.11 for $\alpha = 2$ and $\gamma = 4$ implies $\|x\|_G \leq \|x\|_{\tilde{G}} \leq 4(2+1)\|x\|_G$. □

Finally, we show that the norm $\|\cdot\|_F$ is $(2p-1)$-Supermodular.

Claim 2.6. *The norm $\|\cdot\|_F$ is $(2p-1)$-Supermodular.*

Proof. Due to Lemma 2.9, it suffices to show that $F''(t) \cdot t \leq (p-1) \cdot F'(t)$ for all $t \geq 0$. We have

$$F''(t) \cdot t = \sum_{i=1}^{n} f_i''(t)t \leq \sum_{i=1}^{n} (p-1) f_i'(t) = (p-1) \cdot F'(t). \quad \square$$

Claims 2.3, 2.5, and 2.6 together give the desired approximation to the Orlicz norm $\|\cdot\|_G$, proving Theorem 1.2. □

2.4 Approximation of Top-k and Symmetric Norms

In this section we will give p-Supermodular norm approximations of Top-k and Symmetric Norms. The strategy is to first construct such an approximation for Top-k norms; general symmetric norms are then handled by writing them as a composition of Top-k norms and applying the p-Supermodular approximation to each term.

Approximation of Top-k norms. Even though the Top-k norms have a simple structure, it is not clear how to approximate them by a p-Supermodular norm directly. Instead, we resort to an intermediate step of expressing a Top-k norm (approximately) as an Orlicz norm.

Theorem 2.7. *For every $k \in [n]$, the Top-k norm $\|\cdot\|_{Top\text{-}k}$ in n-dimensions can be 2-approximated by an Orlicz norm.*

Together with Theorem 1.2 from the previous section, this implies the following.

Corollary 2.13. *For every $k \geq 1$, the Top-k norm $\|\cdot\|_{Top\text{-}k}$ in n-dimensions can be 2-approximated by an $O(\log n)$-Supermodular norm.*

The construction in the proof of Theorem 2.7 is inspired by the embedding of Top-k norms into ℓ_∞ by Andoni et al. [6]. They considered the "Orlicz function" $G(t)$ that is 0 until $t = \frac{1}{k}$ and behaves as the identity afterwards, i.e., $G(t) := t \cdot \mathbf{1}(t \geq \frac{1}{k})$. The rough intuition of why the associated "Orlicz norm" approximately captures the Top-k norm of a vector u is because $\frac{u}{\|u\|_{\text{Top-k}}}$ has $\approx k$ coordinates with value above $\frac{1}{k}$ (the top $\approx k$ coordinates), which are picked up by G and give $\sum_i G(\frac{u_i}{\|u\|_{\text{Top-k}}}) \approx \sum_{i \text{ in top } k} \frac{u_i}{\|u\|_{\text{Top-k}}} \approx 1$; thus, by definition of Orlicz norm, $\|u\|_G \approx \|u\|_{\text{Top-k}}$. However, this function G is not convex due to a jump at $t = 1/k$, so it does not actually give a norm. Convexitfying this function also does not work: the convexified version of G is the identity, which yields the ℓ_1 norm, does not approximate Top-k. Interestingly, a modification of this convexification actually works.

Proof of Theorem 2.7. We define the Orlicz function $G(t) := \max\{0, t - \frac{1}{k}\}$. We show that the norm $\|\cdot\|_G$ generated by this function is a 2-approximation to the Top-k norm.

Upper bound $\|x\|_G \leq \|x\|_{Top\text{-}k}$. By the definition of Orlicz norm, it suffices to show that $\sum_i G(\frac{x_i}{\|x\|_{\text{Top-k}}}) \leq 1$. For that, since there are at most k coordinates having $x_i \geq \frac{\|x\|_{\text{Top-k}}}{k}$, we get

$$\sum_i G\left(\frac{x_i}{\|x\|_{\text{Top-k}}}\right) = \sum_{i:\, x_i \geq \|x\|_{\text{Top-k}}/k} \left(\frac{x_i}{\|x\|_{\text{Top-k}}} - \frac{1}{k}\right) \leq \frac{\|x\|_{\text{Top-k}}}{\|x\|_{\text{Top-k}}} - 1 < 1.$$

Lower bound $\|x\|_G \geq \frac{\|x\|_{Top\text{-}k}}{2}$. By the definition of Orlicz norm, it suffices to show that for any $\alpha < \frac{1}{2}$, we have $\sum_i G(\frac{x_i}{\alpha\|x\|_{\text{Top-k}}}) > 1$. Let T_k denote the set of the k largest coordinates of x. Then,

$$\sum_i G\left(\frac{x_i}{\alpha\|x\|_{\text{Top-k}}}\right) \geq \sum_{i \in T_k} G\left(\frac{x_i}{\alpha\|x\|_{\text{Top-k}}}\right) \geq \sum_{i \in T_k} \left(\frac{x_i}{\alpha\|x\|_{\text{Top-k}}} - \frac{1}{k}\right) = \frac{1}{\alpha} - 1,$$

which is > 1 whenever $\alpha < \frac{1}{2}$. This concludes the proof of Theorem 2.7. □

Given Theorem 2.7, one might wonder whether all symmetric norms can be approximated within a constant factor by Orlicz norms. The following lemma shows that this is impossible.

Lemma 2.14. *There exist symmetric norms that cannot be approximated to within a $O(\log n)^{1-\epsilon}$ factor by an Orlicz norm for any constant $\epsilon > 0$.*

We defer the proof of this observation to the full version.

Approximation of symmetric norms. Although Lemma 2.14 rules out the possibility of approximating any symmetric norm by an Orlicz norm within a constant factor, we show that every symmetric norm can be $O(\log n)$-approximated by an an $O(\log n)$-Supermodular norm.

Theorem 1.1. *Every monotone symmetric norm $\|\cdot\|$ in n dimensions can be $O(\log n)$-approximated by an $O(\log n)$-Supermodular norm.*

As mentioned before, the idea is write a general symmetric norm as composition of Top-k norms and applying the p-Supermodular approximation to each term. More precisely, the following lemma, proved in [35] (and a similar property in [6, 17]), shows that the any monotone symmetric norm can be approximated by Top-k norms.

Lemma 2.15 ([35, Lemma 2.5]). *For any monotone symmetric norm $\|\cdot\|$ in $\mathbb{R}^d$, there are $\log n$ non-negative scalars $c_1, c_2 \ldots, c_{\log n}$ such that the norm*

$$|||x||| := \left\| \left(c_1\|x\|_{Top\text{-}2^1}, \ldots, c_{\log n}\|x\|_{Top\text{-}2^{\log n}}\right) \right\|_\infty \tag{7}$$

satisfies $\|x\| \leq |||x||| \leq 2\log n \cdot \|x\|$.

With the decomposition of monotone symmetric norms into Top-k norms in Lemma 2.15 and the p-Supermodular approximation to the latter in Corollary 2.13, we can now prove that every symmetric norm can be $O(\log n)$-approximated by an $O(\log n)$-Supermodular norm.

Proof of Theorem 1.1. Consider a monotone symmetric norm and its approximation $|||x|||$ given by Lemma 2.15. Let f_k be the p-Supermodular 2-approximation of the Top-k norm as given by Corollary 2.13, where $p = \Theta(\log n)$. We replace in $|||x|||$ the Top-k norms by these approximations, and the outer ℓ_∞-norm by the ℓ_p-norm to obtain the norm

$$g(x) := \left(\sum_{i=1}^{\log n} c_i^p \cdot \left(f_{2^i}(x)\right)^p \right)^{1/p}.$$

By the standard ℓ_p to ℓ_∞ comparison, we that $g(x)$ is a constant approximation to $|||x|||$ since $p = \Theta(\log n)$. Hence, $g(x)$ is an $O(\log n)$-approximation to the original norm $\|x\|$.

Moreover, to see that g is p-Supermodular, consider the gradient of g^p, which is given by

$$\nabla(g(x)^p) = \sum_{i=1}^{\log n} c_i^p \cdot \nabla\left(f_{2^i}(x)^p\right).$$

Since each norm f_j is p-Supermodular and the multipliers c_i are non-negative, $\nabla(g(x)^p)$ is non-decreasing. By the Gradient property in Lemma 2.1, this implies p-Supermodularity. □

We remark that given a Ball-Optimization oracle, we can evaluate at a given point the value and gradient of the approximating norm constructed in Theorem 1.1, up to error ε, in time poly$(\log \frac{1}{\varepsilon}, n)$. This is because the decomposition into Top-k norms from Lemma 2.15 can be found in polytime given this oracle (e.g., see [17, 35]), the Orlicz function of the Orlicz norm approximation of each Top-k can be constructed explicitly, and the value and gradient of this Orlicz norm can be evaluated by binary search on the scaling α in the definition of the Orlicz norm (and Claim 2.1).

ACKNOWLEDGMENTS

The second author was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, and by Bolsa de Produtividade em Pesquisa #312751/2021-4 from CNPq. The third author was supported in part by NSF award CCF-2327010.

REFERENCES

[1] Shipra Agrawal and Nikhil R. Devanur. 2015. Fast Algorithms for Online Stochastic Convex Programming. In *Proceedings of SODA*. 1405–1424.
[2] Susanne Albers. 2010. Energy-efficient algorithms. *Commun. ACM* 53, 5 (2010), 86–96.
[3] Noga Alon, Baruch Awerbuch, Yossi Azar, Niv Buchbinder, and Joseph Naor. 2003. The online set cover problem. In *Proceedings of STOC*. 100–105.
[4] Alexandr Andoni, Chengyu Lin, Ying Sheng, Peilin Zhong, and Ruiqi Zhong. 2018. Subspace Embedding and Linear Regression with Orlicz Norm. In *Proceedings of ICML*, Vol. 80. PMLR, 224–233.
[5] Alexandr Andoni, Assaf Naor, Aleksandar Nikolov, Ilya Razenshteyn, and Erik Waingarten. 2018. Hölder Homeomorphisms and Approximate Nearest Neighbors. In *Proceedings of FOCS*. 159–169. https://doi.org/10.1109/FOCS.2018.00024
[6] Alexandr Andoni, Huy L. Nguyen, Aleksandar Nikolov, Ilya P. Razenshteyn, and Erik Waingarten. 2017. Approximate near neighbors for general symmetric norms. In *Proceedings of STOC*. ACM, 902–913.
[7] C. J. Argue, Anupam Gupta, Marco Molinaro, and Sahil Singla. 2022. Robust Secretary and Prophet Algorithms for Packing Integer Programs. In *Proceedings of SODA*. SIAM, 1273–1297.
[8] James Aspnes, Yossi Azar, Amos Fiat, Serge A. Plotkin, and Orli Waarts. 1997. Online routing of virtual circuits with applications to load balancing and machine scheduling. *J. ACM* 44, 3 (1997), 486–504.
[9] Yossi Azar, Niv Buchbinder, T.-H. Hubert Chan, Shahar Chen, Ilan Reuven Cohen, Anupam Gupta, Zhiyi Huang, Ning Kang, Viswanath Nagarajan, Joseph Naor, and Debmalya Panigrahi. 2016. Online Algorithms for Covering and Packing Problems with Convex Objectives. In *Proceedings of FOCS*. 148–157.
[10] Yossi Azar, Joseph Naor, and Raphael Rom. 1995. The Competitiveness of On-Line Assignments. *J. Algorithms* 18, 2 (1995), 221–237.
[11] Nikhil Bansal, Anupam Gupta, Jian Li, Julián Mestre, Viswanath Nagarajan, and Atri Rudra. 2012. When LP is the cure for your matching woes: Improved bounds for stochastic matchings. *Algorithmica* 63 (2012), 733–762.
[12] Domagoj Bradac, Anupam Gupta, Sahil Singla, and Goran Zuzic. 2020. Robust Algorithms for the Secretary Problem. In *Proceedings of ITCS*, Vol. 151. 32:1–32:26.
[13] Domagoj Bradac, Sahil Singla, and Goran Zuzic. 2019. (Near) Optimal Adaptivity Gaps for Stochastic Multi-Value Probing. In *Proceedings of APPROX/RANDOM*, Vol. 145. 49:1–49:21.
[14] Niv Buchbinder and Joseph Naor. 2009. Online Primal-Dual Algorithms for Covering and Packing. *Math. Oper. Res.* 34, 2 (2009), 270–286.
[15] Niv Buchbinder and Joseph Seffi Naor. 2009. The design of competitive online algorithms via a primal–dual approach. *Foundations and Trends® in Theoretical Computer Science* 3, 2–3 (2009).
[16] D. L. Burkholder. 1979. A Sharp Inequality for Martingale Transforms. *The Annals of Probability* 7, 5 (1979), 858 – 863. https://doi.org/10.1214/aop/1176994944
[17] Deeparnab Chakrabarty and Chaitanya Swamy. 2019. Approximation algorithms for minimum norm and ordered optimization problems. In *Proceedings of the 51st Annual Symposium on Theory of Computing, STOC*. 126–137.
[18] Deeparnab Chakrabarty and Chaitanya Swamy. 2019. Simpler and Better Algorithms for Minimum-Norm Load Balancing. In *Proceedings of European Symposium on Algorithms, ESA*. 27:1–27:12.
[19] Ning Chen, Nicole Immorlica, Anna R Karlin, Mohammad Mahdian, and Atri Rudra. 2009. Approximating matches made in heaven. In *Proceedings of ICALP*. 266–278.
[20] Shichuan Deng, Jian Li, and Yuval Rabani. 2023. Generalized Unrelated Machine Scheduling Problem. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2898–2916.
[21] Hossein Esfandiari, Nitish Korula, and Vahab Mirrokni. 2018. Allocation with Traffic Spikes: Mixing Adversarial and Stochastic Models. *ACM Trans. Econ. Comput.* 6, 3-4 (2018).
[22] Dylan J. Foster, Alexander Rakhlin, and Karthik Sridharan. 2017. ZigZag: A New Approach to Adaptive Online Learning. In *Proceedings of COLT*, Vol. 65. PMLR, 876–924.
[23] Anupam Gupta, Tomer Koren, and Kunal Talwar. 2019. Better Algorithms for Stochastic Bandits with Adversarial Corruptions. In *Proceedings of the Thirty-Second Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 99)*. PMLR, 1562–1578.
[24] Anupam Gupta, Ravishankar Krishnaswamy, and Kirk Pruhs. 2012. Online Primal-Dual for Non-linear Optimization with Applications to Speed Scaling. In *Proceedings of Approximation and Online Algorithms - International Workshop, WAOA*, Vol. 7846. 173–186.
[25] Anupam Gupta and Viswanath Nagarajan. 2013. A Stochastic Probing Problem with Applications. In *Proceedings of IPCO*, Vol. 7801. 205–216.
[26] Anupam Gupta and Viswanath Nagarajan. 2014. Approximating Sparse Covering Integer Programs Online. *Math. Oper. Res.* 39, 4 (2014), 998–1011.
[27] Anupam Gupta, Viswanath Nagarajan, and Sahil Singla. 2016. Algorithms and adaptivity gaps for stochastic probing. In *Proceedings of Symposium on Discrete Algorithms, SODA*. 1731–1747.
[28] Anupam Gupta, Viswanath Nagarajan, and Sahil Singla. 2017. Adaptivity Gaps for Stochastic Probing: Submodular and XOS Functions. In *Proceedings of Symposium on Discrete Algorithms, SODA*. 1688–1702.
[29] Petteri Harjulehto and Peter Hästö. 2019. *Generalized Orlicz Spaces.* Springer.
[30] Elad Hazan. 2016. Introduction to Online Convex Optimization. *Foundations and Trends in Optimization* 2, 3-4 (2016), 157–325.
[31] Tuomas Hytönen, Jan van Neerven, Mark Veraar, and Lutz Weis. 2016. *Analysis in Banach Spaces : Volume I: Martingales and Littlewood-Paley Theory.* Springer International Publishing.
[32] Nicole Immorlica, Karthik Abinav Sankararaman, Robert E. Schapire, and Aleksandrs Slivkins. 2022. Adversarial Bandits with Knapsacks. *Journal of the ACM, JACM* 69, 6 (2022), 40:1–40:47.
[33] Thomas Kesselheim, Robert D. Kleinberg, and Rad Niazadeh. 2015. Secretary Problems with Non-Uniform Arrival Order. In *Proceedings of STOC*. 879–888.
[34] Thomas Kesselheim and Marco Molinaro. 2020. Knapsack Secretary with Bursty Adversary. In *Proceedings of ICALP*, Vol. 168. 72:1–72:15.
[35] Thomas Kesselheim, Marco Molinaro, and Sahil Singla. 2023. Online and Bandit Algorithms Beyond ℓ_p Norms. In *Proceedings of SODA*. SIAM, 1566–1593.

[36] Thomas Kesselheim and Sahil Singla. 2020. Online Learning with Vector Costs and Bandits with Knapsacks. In *Proceedings of Conference on Learning Theory, COLT*. 2286–2305.
[37] Nitish Korula, Vahab Mirrokni, and Morteza Zadimoghaddam. 2015. Online Submodular Welfare Maximization: Greedy Beats 1/2 in Random Order. In *Proceedings of STOC*. 889–898.
[38] Peter Kosmol and Dieter Müller-Wichards. 2011. *Optimization in function spaces: with stability considerations in Orlicz spaces*. Vol. 13. Walter de Gruyter.
[39] Ishai Menache and Mohit Singh. 2015. Online Caching with Convex Costs: Extended Abstract. In *Proceedings of Symposium on Parallelism in Algorithms and Architectures, SPAA*, Guy E. Blelloch and Kunal Agrawal (Eds.). 46–54.
[40] A. Meyerson. 2001. Online Facility Location. In *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science (FOCS '01)*. IEEE Computer Society, Washington, DC, USA, 426–. http://dl.acm.org/citation.cfm?id=874063.875567
[41] Vahab S. Mirrokni, Shayan Oveis Gharan, and Morteza Zadimoghaddam. 2012. Simultaneous Approximations for Adversarial and Stochastic Online Budgeted Allocation. In *Proceedings of SODA*. 1690–1701.
[42] Marco Molinaro. 2017. Online and Random-order Load Balancing Simultaneously. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms* (Barcelona, Spain) *(SODA '17)*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1638–1650. http://dl.acm.org/citation.cfm?id=3039686.3039794
[43] Marco Molinaro. 2021. Robust Algorithms for Online Convex Problems via Primal-Dual. In *Proceedings of SODA*, Dániel Marx (Ed.). SIAM, 2078–2092.
[44] Viswanath Nagarajan and Xiangkun Shen. 2020. Online covering with ℓ_q-norm objectives and applications to network design. *Math. Program.* 184, 1 (2020), 155–182.
[45] Kalen Patton, Matteo Russo, and Sahil Singla. 2023. Submodular Norms with Applications To Online Facility Location and Stochastic Probing. In *Proceedings of APPROX/RANDOM*, Vol. 275. 23:1–23:22.
[46] Victor de la Peña and Evarist Giné. 1999. *Decoupling: From Dependence to Independence*. Springer-Verlag, New York, NY, USA.
[47] Aviad Rubinstein. 2016. Beyond matroids: secretary problem and prophet inequality with general constraints. In *Proceedings of STOC*. 324–332.
[48] Zhao Song, Ruosong Wang, Lin F. Yang, Hongyang Zhang, and Peilin Zhong. 2019. Efficient Symmetric Norm Regression via Linear Sketching. In *Proceedings of NeurIPS*. Article 75, 828–838 pages.
[49] Karthik Sridharan. 2012. Learning From An Optimization Viewpoint. *CoRR* abs/1204.4145 (2012). arXiv:1204.4145 http://arxiv.org/abs/1204.4145

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009