Near-Optimal Design of Experiments via Regret Minimization

Zeyuan Allen-Zhu * 1 Yuanzhi Li * 2 Aarti Singh * 3 Yining Wang * 3

Abstract
We consider computationally tractable methods for the experimental design problem, where \( k \) out of \( n \) design points of dimension \( p \) are selected so that certain optimality criteria are approximately satisfied. Our algorithm finds a \((1 + \varepsilon)\)-approximate optimal design when \( k \) is a linear function of \( p \); in contrast, existing results require \( k \) to be super-linear in \( p \). Our algorithm also handles all popular optimality criteria, while existing ones only handle one or two such criteria. Numerical results on synthetic and real-world design problems verify the practical effectiveness of the proposed algorithm.

1. Introduction

Experimental design is an important problem in statistics and machine learning research (Pukelsheim, 2006). Consider a linear regression model

\[
y = X\beta_0 + w,
\]

where \( X \in \mathbb{R}^{n \times p} \) is a pool of \( n \) design points, \( y \) is the response vector, \( \beta_0 \) is a \( p \)-dimensional unknown regression model and \( w \) is a vector of i.i.d. noise variables satisfying \( Ew_i = 0 \) and \( Ew_i^2 < \infty \). The experimental design problem is to select a small subset of rows (i.e., design points) \( X_S \) from the design pool \( X \) so that the statistical power of estimating \( \beta_0 \) is maximized from noisy response \( y_S \) on the selected designs \( X_S \).

As an example, consider a material synthesis application where \( p \) is the number of variables (e.g., temperature, pressure, duration) that are hypothesized to affect the quality of the synthesized material and \( n \) is the total number of combinations of different parameters of experimental conditions. As experiments are expensive and time-consuming, one wishes to select \( k \ll n \) experimental settings from \( X \) that are the most statistically efficient for establishing a model that connects experimental parameters with synthesized material quality, \( y \). The experimental design problem is also related to many machine learning tasks, such as linear bandits (Deshpande & Montanari, 2012; Huang et al., 2016), diversity sampling (Kulesza & Taskar, 2012) and active learning (Ma et al., 2013; Chaudhuri et al., 2015; Hazan & Karnin, 2015; Balcan & Long, 2013; Wang & Singh, 2016).

Since statistical efficiency can be measured in various ways, there exist a number of optimality criteria to guide the selection of experiments. We review some optimality criteria in Sec. 2 and interested readers are referred to Sec. 6 of (Pukelsheim, 2006) for a comprehensive review.

Typically, an optimality criterion is a function \( f: \mathbb{S}_k^+ \to \mathbb{R} \) that maps from the \( p \)-dimensional positive definite cone to a real number. The experimental design problem can then be formulated as a combinatorial optimization problem:

\[
S^*(k) = \arg \min_{S \in \mathbb{S}(n,k)} f(X_S^\top X_S),
\]

where \( S \) is either a set or a multi-set of size \( k \), and \( X_S \in \mathbb{R}^{k \times p} \) is formed by stacking the rows of \( X \) that are in \( S \). The constraint set \( \mathbb{S}(n,k) \) is defined as follows:

1. **With replacement**: \( \mathbb{S}_1(n,k) = \{ S \text{ multi-set : } S \subseteq [n], |S| \leq k \} \). Under this setting, \( X_S \) may contain duplicate rows of the design pool \( X \).

2. **Without replacement**: \( \mathbb{S}_2(n,k) = \{ S \text{ standard set : } S \subseteq [n], |S| \leq k \} \). Under this setting, \( X_S \) only contains distinct rows of the design pool \( X \).

The “with replacement” setting is classical in statistics literature, where the multiple measurements in \( y \) with respect to the same design point lead to different values with statistically independent noise. The “without replacement” setting, on the other hand, is more relevant in machine learning applications, because labels are not likely to change if the same data point (e.g., the same image) is considered twice. Finally, it is worth pointing out that the “with replacement” setting is easier, because it can be reduced (in polynomial time) to the “without replacement” setting by replicating each row of \( X \) for \( k \) times.

For many popular choices of \( f \), the exact optimization

---

*Author names listed in alphabetic order. 1 Microsoft Research, Redmond, USA 2 Princeton University, Princeton, USA 3 Carnegie Mellon University, Pittsburgh, USA. Correspondence to: Yining Wang <yiningwa@cs.cmu.edu>.

Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017. Copyright 2017 by the author(s).
Near-optimal design of experiments via regret minimization

Table 1. Comparison with existing results on computationally efficient experimental design. An algorithm produces a subset $\hat{S}$ of size $k$, and the approximation ratio is defined as $f(X_\hat{S}^T X_\hat{S})/\min_{S\subseteq S_k(n,r)} f(X_S^T X_S)$ for $r \leq k$. $T_{OPT}$ denotes the time complexity for solving the continuous convex optimization problem in Eq. (6).

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Replacement?</th>
<th>Deterministic?</th>
<th>Time complexity</th>
<th>Constraints†</th>
<th>Criteria</th>
<th>Approx. ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pipage rounding</td>
<td>Yes</td>
<td>Yes</td>
<td>$T_{OPT} + O(n^2p^2)$</td>
<td>$k = r \geq p$</td>
<td>D,T</td>
<td>$(1 - 1/e)^{-1}$</td>
</tr>
<tr>
<td>(Bouchou et al., 2010)</td>
<td>Yes</td>
<td>No</td>
<td>$T_{OPT} + O(n)$</td>
<td>$k = r \geq p$</td>
<td>D</td>
<td>$(n/k)^{1/p}$</td>
</tr>
<tr>
<td>(Avron &amp; Boutsidis, 2013)</td>
<td>Yes</td>
<td>Yes</td>
<td>$T_{OPT} + O(n^2p^2)$</td>
<td>$k = r \geq p$</td>
<td>A</td>
<td>$\frac{n}{k + 1}$</td>
</tr>
<tr>
<td>(Wang et al., 2016)</td>
<td>No</td>
<td>No</td>
<td>$T_{OPT} + O(nk)$</td>
<td>$k = \Omega(r)$, $r = \Omega(p^2/\varepsilon^2)$</td>
<td>A,V</td>
<td>$1 + \varepsilon$</td>
</tr>
<tr>
<td>This paper — Eq. (3)</td>
<td>No</td>
<td>Yes</td>
<td>$T_{OPT} + O(pk^3)$</td>
<td>$k = r = \Omega(p^2/\varepsilon^2)$</td>
<td>A,D,T,E,V,G</td>
<td>$1 + \varepsilon$</td>
</tr>
<tr>
<td>This paper — Eq. (4)</td>
<td>Yes</td>
<td>Yes</td>
<td>$T_{OPT} + O(pk^3)$</td>
<td>$k = r &gt; 2p$</td>
<td>A,D,T,E,V,G</td>
<td>$O(1)$</td>
</tr>
<tr>
<td>This paper — Eq. (5)</td>
<td>Yes</td>
<td>Yes</td>
<td>$T_{OPT} + O(pk^3)$</td>
<td>$k = \Omega(r), r \geq p/\varepsilon^2$</td>
<td>A,D,T,E,V,G</td>
<td>$1 + \varepsilon$</td>
</tr>
</tbody>
</table>

† In $O(\cdot)$ and $\Omega(\cdot)$ we hide logarithmic dependency over $n$, $p$, and $k$.

problem in Eq. (2) is NP-hard (Çivril & Magdon-Ismail, 2009; Černý & Hladík, 2012). In this paper, we propose a computationally tractable algorithm that approximately computes Eq. (2) for a wide range of optimality criteria, and under very weak conditions on $n$, $k$, and $p$.

Below is our main theorem:

**Theorem 1.1.** Suppose $b \in \{1, 2\}$, $n > k > p$ and let $f : S^+_p \rightarrow \mathbb{R}$ be a regular optimality criterion (cf. Definition 2.1). There exists a polynomial-time algorithm that outputs $\hat{S} \in S_k(n,k)$ for any input matrix $X \in \mathbb{R}^{n \times p}$ with full column rank, and $\hat{S}$ satisfies the following:

1. For $b = 1$ (with replacement), there exists an absolute constant $C_0 \leq 32$ such that, for any $\varepsilon \in (0, 1)$, if $k \geq C_0 p/\varepsilon^2$ then
   \[
   f(X_\hat{S}^T X_\hat{S}) \leq (1 + \varepsilon) \cdot \min_{S \subseteq S_k(n,k)} f(X_S^T X_S). \tag{3}
   \]

2. For $b = 2$ (without replacement) and any $\xi > 2$, there exists constant $C_1(\xi) > 0$ depending only on $\xi$ such that, if $k \geq \xi p$ then
   \[
   f(X_\hat{S}^T X_\hat{S}) \leq C_1(\xi) \cdot \min_{S \subseteq S_2(n,k)} f(X_S^T X_S). \tag{4}
   \]
   Moreover, for $\xi \geq 4$ we have $C_1(\xi) \leq 32$.

3. For $b = 2$ (without replacement) and any $\varepsilon \in (0, 1/2)$, if $k, r$ satisfy $k \geq 4(1 + 7\varepsilon)r$ and $r \geq p/\varepsilon^2$, then
   \[
   f(X_\hat{S}^T X_\hat{S}) \leq (1 + \varepsilon) \cdot \min_{S \subseteq S_2(n,r)} f(X_S^T X_S). \tag{5}
   \]

We interpret the significance of Theorem 1.1 as follows.

- Under a very mild condition of $k > 2p$, our polynomial-time algorithm finds a set $\hat{S} \subset [n]$ of size $k$, with objective value $f(X_\hat{S}^T X_\hat{S})$ being at most $O(1)$ a constant times the optimum. See Eq. (4).

- If replacement ($b = 1$) or over-sampling ($k > r$) is allowed, the approximation ratio can be tightened to $1 + \varepsilon$ for arbitrarily small $\varepsilon > 0$. See Eq. (3) and (5).

- In all of the three cases, we only require $k$ to grow linearly in $p$. Recall that $k \geq p$ is necessary to ensure the singularity of $X_\hat{S}^T X_\hat{S}$. In contrast, no polynomial-time algorithm has achieved $O(1)$ approximation in the regime $k = O(p)$ for non-submodular optimality criteria (e.g., $A$- and $V$-optimality) under the without replacement setting.

- Our algorithm works for any regular optimality criterion. To the best of our knowledge, no known polynomial-time algorithm can achieve a $(1 + \varepsilon)$ approximation for the $D$- and $T$-optimality criteria, or even an $O(1)$ approximation for the $E$- and $G$-optimality criteria. See Table 1 for a comparison.

The key idea behind our proof of Theorem 1.1 is a regret minimization characterization of the least eigenvalue of positive semidefinite (PSD) matrices. Similar ideas were developed in (Allen-Zhu et al., 2015; Silva et al., 2016) to construct efficient algorithms for linear-sized graph sparsifiers. In this paper we adopt the regret minimization framework and present novel potential function analysis for the specific application of experimental design.

### 1.1. Notations

$S^+_p$ is the positive definite cone of $p \times p$ matrices: a $p \times p$ symmetric matrix $A$ belongs to $S^+_p$ if and only if $v^T A v > 0$ for all $v \in \mathbb{R}^p \setminus \{0\}$. For symmetric matrices $A$ and $B$, we write $A \preceq B$ if $v^T (A - B) v \geq 0$ for all $v \in \mathbb{R}^p$. The inner product $(A, B)$ is defined as $(A, B) = \text{tr}(B^T A) = \sum_{i,j=1}^p A_{ij} B_{ij}$. We use $\|A\|_2 = \sup_{v \in \mathbb{R}^p \setminus \{0\}} \|A v\|_2/\|v\|_2$ to denote the spectral norm, and $\|A\|_F = \sqrt{\sum_{i,j=1}^p A_{ij}^2}$ to denote
Near-optimal design of experiments via regret minimization

the Frobenius norm of $A$. For $A \succeq 0$, we write $B = A^{1/2}$ as the unique $B \succeq 0$ that satisfies $B^2 = A$. For a design pool $X \in \mathbb{R}^{n \times p}$, we use $x_i \in \mathbb{R}^p$ to denote the $i$-th row of $X$. We use $\sigma_{\min}(A)$ for the least (smallest) singular value of a PSD matrix $A$.

1.2. Related work

Experimental design is an old topic in statistics research (Pukelsheim, 2006; Fedorov, 1972). Computationally efficient experimental design algorithms (with provable guarantees) are, however, a less studied field. In the case of submodular optimality criteria (e.g., D- and T-optimality), the classical pipage rounding method (Ageev & Sviridenko, 2004; Horel et al., 2014; Ravi et al., 2016) combined with semi-definite programming results in computationally efficient algorithms that enjoy a constant approximation ratio. Bouhtou et al. (2010) improves the approximation ratio when $k$ is very close to $n$. Deshpande & Rademacher (2010); Li et al. (2017) considered polynomial-time algorithms for sampling from a D-optimality criterion. These algorithms are not applicable to non-submodular criteria, such as A-, V-, E- or G-optimality.

For the particular A-optimality criterion, (Avron & Boutsidis, 2013) proposed a greedy algorithm with an approximation ratio of $O(n/k)$ with respect to $f(X^T X)$. It was shown that in the worst case $\min_{|S| \leq k} f(X_S X_S^T) \approx O(n/k) \cdot f(X^T X)$ and hence the bound is tight. However, for general design pool $\min_{|S| \leq k} f(X_S X_S^T)$ could be far smaller than $O(n/k) \cdot f(X^T X)$, making the theoretical results powerless in such scenarios. Wang et al. (2016) considered a variant of the greedy method and showed an approximation ratio quadratic in design dimension $p$ and independent of pool size $n$.

Wang et al. (2016) derived algorithms based on effective resistance sampling (Spielman & Srivastava, 2011) that attain $(1 + \varepsilon)$ approximation ratio if $k = \Omega(p \log p / \varepsilon^2)$ and repetitions of design points are allowed. The algorithm fundamentally relies on the capability of ‘re-weighting’ (repeating) design points and cannot be adapted to the more general ‘without replacement’ setting. Naive sampling based methods were considered in (Wang et al., 2016; Chaudhuri et al., 2015; Dhillon et al., 2013), which also achieve $(1+\varepsilon)$ approximation but requires the subset size $k$ to be much larger than the condition number of $X$.

Another however different topic is low-rank matrix column subset selection and CUR approximation, which seeks column subset $C$ and row subset $R$ such that $\|X - CC^T X\|_F$ and/or $\|X - CUR\|_F$ are minimized (Drineas et al., 2008; Boutsidis & Woodruff, 2014; Wang & Singh, 2015b; Drineas & Mahoney, 2005; Wang & Zhang, 2013; Wang & Singh, 2015a). These problems are unsupervised in nature and do not in general correspond to statistical properties under supervised regression settings. Pilanci & Wainwright (2016); Raskutti & Mahoney (2014); Woodruff (2014) considered fast methods for solving ordinary least squares (OLS) problems. They are computationally oriented and typically require knowledge of the full response vector $y$, which is different from the experimental design problem.

2. Regular criteria and continuous relaxation

We start with the definition of regular optimality criteria:

**Definition 2.1** (Regular criteria). An optimality criterion $f : \mathbb{S}_p^+ \rightarrow \mathbb{R}$ is regular if it satisfies the following properties:

1. Convexity: $f(\lambda A + (1 - \lambda)B) \leq \lambda f(A) + (1 - \lambda) f(B)$ for all $\lambda \in [0, 1]$ and $A, B \in \mathbb{S}_p^+$;
2. Monotonicity: If $A \preceq B$ then $f(A) \geq f(B)$;
3. Reciprocal multiplicity: $f(tA) = t^{-1} f(A)$ for all $t > 0$ and $A \in \mathbb{S}_p^+$.

Almost all optimality criteria used in the experimental design literature are regular. Below we list a few popular examples; their statistical implications can be found in (Pukelsheim, 2006):

- A-optimality (Average): $f_A(\Sigma) = \frac{1}{p} \text{tr}(\Sigma^{-1})$;
- D-optimality (Determinant): $f_D(\Sigma) = (\det |\Sigma|)^{-\frac{1}{p}}$;
- T-optimality (Trace): $f_T(\Sigma) = p/\text{tr}(\Sigma)$;
- E-optimality (Eigenvalue): $f_E(\Sigma) = \|\Sigma^{-1}\|_2$;
- V-optimality ( Variance): $f_V(\Sigma) = \frac{1}{p} \text{tr}(\Sigma) (\Sigma^{-1} X^T X)$;
- G-optimality: $f_G(\Sigma) = \max \text{diag}(X \Sigma^{-1} X^T)$.

The (A-, D-, T-, V-) criteria concern estimates of regression coefficients and the (V-, G-) criteria are about in-sample predictions. All criteria listed above are regular. Note that for D-optimality the proxy function $g_D(\Sigma) = - \log \det(\Sigma)$ is considered to satisfy the convexity property. In addition, by the standard arithmetic inequality we have that $f_T \leq f_D \leq f_A \leq f_E$ and that $f_V \leq f_G$.

Although exact optimization of the combinatorial problem Eq. (2) is intractable, it is nevertheless easy to solve a continuous relaxation of Eq. (2) given the convexity property in Definition 2.1. We consider the following continuous optimization problem:

\[
\pi^*(b) = \arg \min_{\pi = (\pi_1, \ldots, \pi_n)} f \left( \sum_{i=1}^{n} \pi_i x_i x_i^T \right),
\]

s.t. $\pi \geq 0$, $\|\pi\|_1 \leq r$, $\|b = 2\cdot \|\pi\|_\infty \leq 1$.

\[1\]This property could be relaxed to allow a proxy function $g : \mathbb{S}_p^+ \rightarrow \mathbb{R}$ being convex, where $g(A) \leq g(B) \Leftrightarrow f(A) \leq f(B)$. 

The $\|\pi\|_1 \leq r$ constraint makes sure only $r$ rows of $X$ are “selected”, where $r \leq k$ is a parameter that controls the degree of oversampling. The $0 \leq \pi_i \leq 1$ constraint enforces that each row of $X$ is “selected” at most once and is only applicable to the without replacement setting ($b = 2$). Eq. (6) is a relaxation of the original combinatorial problem Eq. (2), which we formalize below:

**Fact 2.1.** For $b \in \{1, 2\}$ we have $f(\sum_{i=1}^n \pi_i^*(b) x_i x_i^\top) \leq \min_{s \in \mathbb{S}(n, r)} f(X_S X_S^\top) f(X_S X_S^\top)$

In addition, because of the monotonicity property of $f$ the sum constraint must bind:

**Fact 2.2.** For $b \in \{1, 2\}$ it holds that $\sum_{i=1}^n \pi_i^*(b) = r$.

Proofs of Facts 2.1 and 2.2 are straightforward and are placed in the supplementary material.

Both the objective function and the constraint set in Eq. (6) are convex, and hence it can be efficiently solved to global optimality by conventional convex optimization algorithms. In particular, for differentiable $f$ we suggest the following projected gradient descent (PGD) procedure:

$$\pi^{(t+1)} = P_C \left( \pi^{(t)} - \gamma_t \nabla f(\pi^{(t)}) \right),$$

where $P_C(x) = \arg \min_{y \in C} \|x - y\|_2$ is the projection operator onto the feasible set $C = \{ x \in \mathbb{R}^n : \pi \geq 0, \|\pi\|_1 \leq r, \|\pi\|_\infty \leq 1 \}$ and $\gamma_t$ is a sequence of step sizes typically chosen by backtracking line search. When $f$ is not differentiable everywhere, projected subgradient descent could be used with either constant or diminishing step sizes. We defer detailed gradient computations to the supplementary material. It was shown in (Wang et al., 2016; Su et al., 2012) that the projection operator $P_C(x)$ could be efficiently computed up to precision $\delta = O(n \log(||x||_\infty/\delta))$ operations.

### 3. Sparsification via regret minimization

The optimal solution $\pi^*$ of Eq. (6) does not naturally lead to a valid approximation of the combinatorial problem in Eq. (2), because the number of non-zero components in $\pi^*$ may far exceed $k$. The primary focus of this section is to design efficient algorithms that sparsify the optimal solution $\pi^*$ into $s \in [k]^n$ (with replacement) or $s \in \{0, 1\}^n$ (without replacement), while at the same time bounding the increase in the objective.

Due to the monotonicity and reciprocal multiplicity properties of $f$, it suffices to find a sparsifier $s$ that satisfies

$$\sum_{i=1}^n s_i x_i x_i^\top \geq \tau \cdot \sum_{i=1}^n \pi_i^* x_i x_i^\top$$

for some constant $\tau \in (0, 1)$. By Definition 2.1, Eq. (8) immediately implies $f(\sum_{i=1}^n s_i x_i x_i^\top) \leq \tau^{-1} f(\sum_{i=1}^n \pi_i^* x_i x_i^\top)$. The key idea behind our algorithm is a regret-minimization interpretation of the least eigenvalue of a positive definite matrix, which arises from recent progress in the spectral graph sparsification literature (Silva et al., 2016; Allen-Zhu et al., 2015).

In the rest of this section, we adopt the notation that $\Pi = \text{diag}(\pi^*)$ and $S = \text{diag}(s)$, both being $n \times n$ non-negative diagonal matrices. We also use $I$ to denote the identity matrix, whose dimension should be clear from the context.

#### 3.1. The whitening trick

Consider the linear transform $x_i \mapsto (\Pi_x x_i ^{\top})^{1/2} x_i =: \tilde{x}_i$. It is easy to verify that $\sum_{i=1}^n \pi_i^* x_i x_i^\top = I$. Such a transformation is usually referred to as whitening, because the sample covariance of the transformed data is the identity matrix. Define $W = \sum_{i=1}^n s_i \tilde{x}_i \tilde{x}_i^\top$. We then have the following:

**Proposition 3.1.** For $\tau > 0$, $W \succeq \tau I$ if and only if $(\sum_{i=1}^n s_i \tilde{x}_i \tilde{x}_i^\top) \succeq \tau (\sum_{i=1}^n \pi_i^* x_i x_i^\top)$.

**Proof.** The proposition holds because $W \succeq \tau I$ if and only if $(\Pi X^\top)^{1/2} W (\Pi X^\top)^{1/2} \succeq \tau \Pi X^\top$, and that $(\Pi X^\top)^{1/2} W (\Pi X^\top)^{1/2} = \Pi X^\top$.

Proposition 3.1 shows that, without loss of generality, we may assume $\sum_{i=1}^n \pi_i^* x_i x_i^\top = \Pi X^\top$ is XIXI. The question of proving $W = \Pi X^\top \succeq \tau I$ is then reduced to bounding the smallest eigenvalue of $W$.

Recall that $W$ can be written as a sum of rank-1 PSD matrices $W = \sum_{i=1}^k F_i$, where $F_i = x_i x_i^\top$ for some $i \in [n]$. In the next section we give a novel characterization of the least eigenvalue of $W$ from a regret minimization perspective. The problem of lower bounding the least eigenvalue of $W$ can then be reduced to bounding the regret of a particular Follow-The-Regularized-Leader (FTRL) algorithm, which is much easier task as FTRL admits closed-form solutions.

#### 3.2. Smallest eigenvalue as regret minimization

We first review the concept of regret minimization in a classical linear bandit setting. Let $\Delta_p = \{ A \in \mathbb{R}^{p \times p} : A \succeq 0, \text{tr}(A) = 1 \}$ be an action space that consists of positive semi-definite matrices of dimension $p$ and unit trace norm. Consider the linear bandit problem, which operates in $k$ iterations. At iteration $t$, the player chooses an action $A_t \in \Delta_p$; afterwards, a “reference” action $F_t \succeq 0$ is observed and the loss $\langle F_t, A_t \rangle$ is incurred. The objective of the player is to minimize his/her regret:

$$R(\{A_t\}_{t=1}^k) := \sum_{t=1}^k \langle F_t, A_t \rangle - \inf_{U \in \Delta_p} \sum_{t=1}^k \langle F_t, U \rangle,$$
which is the “excess loss” of \( \{A_t\}_{k=1}^k \) compared to the single optimal action \( U \in \Delta_p \) in hindsight, knowing all the reference actions \( \{F_t\}_{t=1}^k \). A popular algorithm for regret minimization is Follow-The-Regularized-Leader (FTRL), also known to be equivalent to Mirror Descent (MD) (McMahan, 2011), which solves for

\[
A_t = \arg \min_{A \in \Delta_p} \left\{ w(A) + \alpha \sum_{t=1}^{k-1} \langle F_t, A \rangle \right\}.
\]

Here \( w(A) \) is a regularization term and \( \alpha > 0 \) is a parameter that balances model fitting and regularization. For the proof of our purpose we adopt the \( \ell_{1/2} \)-regularizer \( w(A) = -2r \langle A^{1/2} \rangle \) introduced in (Allen-Zhu et al., 2015), which leads to the closed-form solution

\[
A_t = \left( c_1 I + \alpha \sum_{t=1}^{k-1} F_t \right)^{-2},
\]

where \( c_1 \in \mathbb{R} \) is the unique constant that ensures \( A_t \in \Delta_p \). The following lemma from (Allen-Zhu et al., 2015) bounds the regret of FTRL using the particular \( \ell_{1/2} \)-regularizer:

**Lemma 3.1** (Theorem 3.2 of (Allen-Zhu et al., 2015), specialized to \( \ell_{1/2} \)-regularization). Suppose \( \alpha > 0 \), \( \text{rank}(F_t) = 1 \) and let \( \{A_t\}_{k=1}^k \) be FTRL solutions defined in Eq. (10). If \( \alpha \langle F_t, A_t^{1/2} \rangle > -1 \) for all \( t \), then

\[
R(\{A_t\}_{t=1}^k) := \sum_{t=1}^{k} \langle F_t, A_t \rangle - \inf_{U \in \Delta_p} \sum_{t=1}^{k} \langle F_t, U \rangle 
\leq \alpha \sum_{t=1}^{k} \frac{\langle F_t, A_t \rangle \langle F_t, A_t^{1/2} \rangle}{1 + \alpha \langle F_t, A_t^{1/2} \rangle} + 2 \sqrt{p}.\]

Now consider each \( F_t = x_i x_i^\top \) to be the outer product of a design point selected from the design pool \( X \). One remarkable consequence of Lemma 3.1 is that, in order to lower bound the smallest eigenvalue of \( \sum_{t=1}^{k} F_t \), which by definition is \( \inf_{U \in \Delta_p} \langle \sum_{t=1}^{k} F_t, U \rangle \), it suffices to lower bound \( \sum_{t=1}^{k} \langle F_t, A_t \rangle \). Because \( A_t \) admits closed-form expression in Eq. (10), choosing a sequence of \( \{F_t\}_{t=1}^k \) with large \( \sum_{t=1}^{k} \langle F_t, A_t \rangle \) becomes a much more manageable analytical task, which we shall formalize in the next section.

### 3.3. Proof of Theorem 1.1

Re-organizing terms in Lemma 3.1 we obtain

\[
\inf_{U \in \Delta_p} \sum_{t=1}^{k} \langle F_t, U \rangle \geq \sum_{t=1}^{k} \frac{\langle F_t, A_t \rangle}{1 + \alpha \langle F_t, A_t^{1/2} \rangle} - 2 \sqrt{p}.\]

The \( k \) near-optimal design points are selected in a sequential manner. Let \( S_t \in \mathcal{S}_p(n, t) \) be the set of selected design points at or prior to iteration \( t \) (\( A_0 = \emptyset \)), and define \( F_t = x_i x_i^\top \), where \( i_t \) is the design point selected at iteration \( t \). Define also \( A_t = \sum_{t=1}^{k} F_t = \sum_{i \in S_t} x_i x_i^\top \).

We first consider the with replacement setting \( b = 1 \).

**Lemma 3.2.** Suppose \( \sum_{i=1}^{n} \pi^*_i x_i x_i^\top = I \) where \( \pi^*_i \geq 0 \) and \( \sum_{i=1}^{n} \pi_i^* = r \). Then for \( 1 \leq t \leq k \) we have that

\[
\max_{i \in [n]} \frac{(x_i x_i^\top, A_t)}{1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle} \geq \frac{1}{r + \sqrt{p}}.
\]

**Proof.** Recall that \( \text{tr}(A_t) = 1 \) and \( \sum_{i=1}^{n} \pi^*_i x_i x_i^\top = I \). Subsequently, \( \sum_{i=1}^{n} \langle x_i x_i^\top, A_t \rangle = 1 \). On the other hand, we have that \( \sum_{i=1}^{n} \pi_i^* (1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle) \leq \sum_{i=1}^{n} \pi_i^* + \alpha \cdot \text{tr}(A_t^{1/2}) \leq r + \alpha \cdot \text{tr}(A_t^{1/2}) \leq r + \sqrt{p} \). Here \((a)\) is due to the optimization constraint that \( ||x_i^*||_1 \leq r \), and \((b)\) is because \( \text{tr}(A_t^{1/2}) = ||\sigma(A_t^{1/2})||_2 \leq \sqrt{p} \cdot ||\sigma(A_t)||_2 \leq \sqrt{p} \cdot ||A_t||_F \leq \sqrt{p} \cdot ||A_t||_F \), where \( \sigma(t) \) is the vector of all eigenvalues of a PSD matrix. Combining both inequalities we have that \( \max_{i \in [n]} \frac{(x_i x_i^\top, A_t)}{1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle} \leq \sum_{i=1}^{n} \pi_i^* (1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle) \leq \sum_{i=1}^{n} \pi_i^* (1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle) \).\]

Let \( i_t = \arg \max_{i \in [n]} \frac{(x_i x_i^\top, A_t)}{1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle} \) be the design point selected at iteration \( t \). Combining Eq. (11) and Lemma 3.2,

\[
A_k = \sum_{i \in S_k} x_i x_i^\top \geq \frac{k}{r + \alpha \sqrt{p}} - 2 \sqrt{p} I.\]

To prove Eq. (3), set \( \alpha = 8 \sqrt{p} \). Because \( k = r \geq C_0 \alpha^2 \), we have that \( \frac{k}{r + \alpha \sqrt{p}} \geq \frac{1}{1 + \varepsilon / C_0} \). With \( C_0 = 32 \) the right-hand side is lower bounded by \( 1 - \varepsilon / 2 \). Eq. (3) is thus proved because \( (1 - \varepsilon / 2)^2 \leq 1 - \varepsilon \).

We next consider the without replacement setting \( b = 2 \).

**Lemma 3.3.** Fix arbitrary \( \beta \in (0, 1) \) and suppose \( \sum_{i=1}^{n} \pi^*_i x_i x_i^\top = I \) where \( \pi^*_i \in [0, \beta] \) and \( \sum_{i=1}^{n} \pi_i^* = r \). Then for all \( 1 \leq t \leq k \),

\[
\max_{i \notin S_{t-1}} \frac{(x_i x_i^\top, A_t)}{1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle} \leq \frac{1 - \beta \sigma_{\min}(A_{t-1}) - \sqrt{p} / \alpha}{r + \alpha \sqrt{p}}.
\]

**Proof.** On one hand, we have \( \sum_{i \notin S_{t-1}} \pi_i^* (x_i x_i^\top, A_t) \geq \langle A_t, I - \beta A_{t-1} \rangle \geq 1 - \text{tr} \left[ \left( \alpha A_{t-1} - c_t I \right)^{-2} \beta A_{t-1} \right] \geq 1 + \frac{\beta c_t}{\alpha} \cdot \frac{2}{\alpha} \cdot \text{tr} \left[ \left( \alpha A_{t-1} - c_t I \right)^{-2} \right] \geq 1 + \frac{\beta c_t}{\alpha} - \frac{2}{\alpha} \cdot \frac{1}{r + \alpha \sqrt{p}}.\]

Here \((a)\) is due to \( \sum_{i=1}^{n} \pi^*_i x_i x_i^\top = I \) and \( \pi_i^* \in [0, \beta] \); \((b)\) is due to \( \langle A_t, I \rangle = \text{tr}(A_t) = 1 \) and \((c)\) is proven in the proof of Lemma 3.2. Because \( \alpha A_{t-1} - c_t I \geq 0 \), we conclude that \( c_t \geq -\beta \sigma_{\min}(A_{t-1}) \) and therefore \( \sum_{i \notin S_{t-1}} \pi_i^* (x_i x_i^\top, A_t) \geq 1 - \beta \sigma_{\min}(A_{t-1}) - \sqrt{p} / \alpha \). On the other hand, \( \sum_{i \notin S_{t-1}} \pi_i^* (1 + \alpha \langle x_i x_i^\top, A_t^{1/2} \rangle) \leq ...
Near-optimal design of experiments via regret minimization

Subsequently, \( \max_{i \in A_{t-1}} \frac{(x_i, \beta_i, \Lambda_i)}{1 + \alpha(x_i, \beta_i, \Lambda_i)} \geq 1 - \frac{\beta \sigma_{\min}(A_{t-1})}{(r + \alpha \sqrt{p})/\alpha} \).

\[
\sum_{i \in A_{t-1}} \pi_i^*(x_i, \beta_i, \Lambda_i) \geq 1 - \frac{\beta \sigma_{\min}(A_{t-1})}{(r + \alpha \sqrt{p})/\alpha}.
\]

Let \( i_t = \arg \max_{i \in A_{t-1}} \frac{(x_i, \beta_i, \Lambda_i)}{1 + \alpha(x_i, \beta_i, \Lambda_i)} \). Combining Eq. (11) and Lemma 3.3 with \( \beta = 1 \), we have that

\[
\Lambda_k \geq \left( \frac{k - 1 + \frac{1}{2} - \sqrt{p}/\alpha}{r + \alpha \sqrt{p}} - 2 \frac{\sqrt{p}}{\alpha} \right) I, \tag{13}
\]

where \( \kappa_t := \sigma_{\min}(\Lambda_t) \). We are now ready to prove Eqs. (4,5) in Theorem 1.1.

**Proof of Eq. (4).** Note that

\[
\Lambda_k \geq \sup_{u \geq 0} \min \left\{ u, \frac{1 - \frac{u}{(r + \alpha \sqrt{p})/\alpha} \cdot \frac{k - 1}{2}}{1 + \alpha \sqrt{p}} \right\} I. \tag{14}
\]

Eq. (14) can be proved by a case analysis: if \( u \leq \kappa_t \) for some \( 1 \leq t \leq k \) then \( \sigma_{\min}(\Lambda_k) \geq \sigma_{\min}(\Lambda_{t-1}) \geq u \); otherwise \( 1 - \kappa_t - \sqrt{p}/\alpha \geq 1 - u - \sqrt{p}/\alpha \) for all \( 1 \leq t \leq k \). Suppose \( k = r = g_0 \) for some \( \xi > 2 \), and let \( \alpha = \nu \sqrt{p}, u = 1 - 2(\xi - 3)/(2 + 2r/\xi) \), where \( \nu > 1 \) is some parameter to be specified later. Eq. (14) then yields \( \Lambda_k \geq (1 - 2(\xi - 3)/\nu)/(2 + 2r/\xi) I \). Because \( \xi > 2 \), it is possible to select \( \nu > 0 \) such that \( C_1(\xi)^{-1} = (1 - 2(\xi - 3)/(2 + 2r/\xi))^{-1} > 0 \). Finally, for \( \xi \geq 4 \) and \( \nu = 8 \) we have \( C_1(\xi)^{-1} \geq 1/32 \). Eq. (4) is then proved. \( \Box \)

**Proof of Eq. (5).** Let \( \beta \in (0, 1) \) be a parameter to be specified later, and define \( \Sigma^*_\beta := \sum_{i \geq \beta} \pi_i^*(x_i, x_i^T) \) and \( \Sigma^* := I - \Sigma^*_\beta = \sum_{i \leq \beta} \pi_i^*(x_i, x_i^T) \). Let \( \tilde{S} \) be constructed such that it includes all points in \( \Sigma^*_\beta \) and \( \Sigma^* \). Define \( \alpha = 2(1 - r/\xi), \beta := \sum_{i \geq \beta} \pi_i^*, k := k - k', \tilde{r} := r - \sqrt{p}/\alpha \), \( r \leq r - \sqrt{p}/\alpha \), \( \alpha \sqrt{p} = \tilde{r} - 2p/\xi \). Let \( \Lambda = \sum_{i \in \tilde{S}} x_i x_i^T \) be the sample covariance of the selected subset. By the definition of \( \tilde{S} \) and Lemma 3.3, together with the whitening trick (Sec. 3.1) on \( \Sigma^*_\beta \), we have

\[
\Lambda \geq \Sigma^* + \sup_{u \geq 0} \min \left\{ u, (1 - \beta - u - \varepsilon/2) \right\} \tilde{r}/\tilde{r} - \varepsilon \}
\]

where the second line holds because \( \Sigma^* + \Sigma^* = I \) and \( u \leq 1 \). Now set \( \beta = 0.5 \) and note that \( k' \leq r/\sqrt{p} \leq 2r' \) by definition of \( S^* \). Subsequently, \( r \geq p/\varepsilon^2 \) and \( k \geq 4(1 + \varepsilon) \) for \( \varepsilon \in (0, 1/2) \) implies that \( \tilde{r} \geq (1 + 2\varepsilon)/(1 - \beta) \), which yields \( u \geq 1 - \varepsilon/2 \) and hence \( f(X, X^\dagger S) \leq (1 + \varepsilon)f(X, X^\dagger S^*) \). Eq. (5) is thus proved. \( \Box \)

**Algorithm 1** Near-optimal experimental design

1: **Input:** design pool \( X \in \mathbb{R}^{n \times p} \), budget parameters \( k \geq r \geq p \), algorithmic parameter \( \alpha > 0 \).
2: Solve the convex optimization problem Eq. (6) with parameter \( s_i \).
3: Whiten: \( X \leftarrow X(X^\dagger \text{diag}(\pi^*) X)^{-1/2} \).
4: Initialization: \( A_0 = \emptyset \).
5: for \( t = 1 \) to \( k \) do
6: \( c_t \leftarrow \text{FINDCONSTANT}(\sum_{i \in A_{t-1}} x_i x_i^T, \alpha); \)
7: \( \Lambda_t \leftarrow (c_t I + \sum_{i \in A_{t-1}} x_i x_i^T)^{-2}; \)
8: if \( b = 1 \) then \( \Gamma_t = [n]; \) else \( \Gamma_t = [n] \setminus A_{t-1}; \)
9: \( i_t \leftarrow \arg \max_{i \in \Gamma_t} \frac{1}{1 + \alpha(x_i, \beta_i, \Lambda_i)} \);\( \Lambda_t = A_{t-1} \cup \{i_t\}; \)
11: end for
12: **Output:** \( \hat{S} = \Lambda_k \).

**Algorithm 2** FINDCONSTANT(\( Z, \alpha \))

1: **Initialization:** \( c_t = -\sigma_{\min}(Z), c_u = \sqrt{p}; \) \( \varepsilon = 10^{-9}; \)
2: while \( |c_t - c_u| > \varepsilon \) do
3: \( \bar{c} \leftarrow (c_t + c_u)/2; \)
4: if \( \|\pi^*(Z - \bar{c} I)\|_2 \leq 1 \) then \( c_t \leftarrow \bar{c}; \) else \( c_u \leftarrow \bar{c}; \)
5: end while
6: **Output:** \( c = (c_t + c_u)/2 \).

Our proof of Theorem 1.1 is constructive and yields a computationally efficient iterative algorithm which finds subset \( \hat{S} = \hat{S}_b(n, k) \) that satisfies the approximation results in Theorem 1.1. In Algorithm 1 we give a pseudocode description of the algorithm, which makes use of a binary search routine (Algorithm 2) that finds the unique constant \( c_t \) for which \( \text{tr}(\Lambda_t) = \text{tr}(|c_t I + \sum_{i \in A_{t-1}} x_i x_i^T|^{-2}) = 1 \). Note that for Eq. (5) to be valid, it is necessary to run Algorithm 2 on the remaining set of \( \pi^* \) after including all points \( x_i \) with \( \pi_i^* \geq 1/2 \) in \( \hat{S} \).

4. Extension to generalized linear models

The experimental algorithm presented in this paper could be easily extended beyond the linear regression model. For this purpose we consider the Generalized Linear Model (GLM), which assumes that

\[
y \mid x \sim \text{known distribution and } \beta_0 \text{ is an unknown } p \text{-dimensional regression model. Examples include the logistic regression model } p(y = 1 \mid x) = \frac{\exp(x^\dagger \beta_0)}{1 + \exp(x^\dagger \beta_0)} \text{ and the Poisson count model } p(y_i = y \mid x) = \exp(y^\dagger \beta_0 - e^{-x^\dagger \beta_0});\]

and many others.

Let \( S \in S_b(n, k) \) be the set of selected design points from \( X \). Under the classical statistics regime,
Table 2. Simulation results on synthetic data of size $n = 1000$ and $k = 50$. Uniform sampling and weighted sampling are run for 50 independent trials and the median objective is reported. “Inf” means the sample covariance $X_S X_S$ does not belong to $S_p^k$.

<table>
<thead>
<tr>
<th></th>
<th>$k = 2p = 100$</th>
<th>$k = 3p = 150$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$f_A$</td>
<td>$f_D$</td>
</tr>
<tr>
<td><strong>Uniform Sampling</strong></td>
<td>34.29</td>
<td>7.25</td>
</tr>
<tr>
<td><strong>Weighted Sampling</strong></td>
<td>23.42</td>
<td>4.57</td>
</tr>
<tr>
<td><strong>Fedorov’s Exchange</strong></td>
<td>23.17</td>
<td>5.52</td>
</tr>
<tr>
<td><strong>Algorithm 1</strong></td>
<td>4.6</td>
<td>26.444</td>
</tr>
<tr>
<td><strong>Algorithm 1</strong></td>
<td>12.55</td>
<td>4.72</td>
</tr>
</tbody>
</table>

5. Numerical results

We compare the proposed method with several baseline methods on both synthetic and real-world data sets. We only consider the harder “without replacement” setting, where each row of $X$ can be selected at most once.

5.1. Methods and their implementation

We compare our algorithm with three simple heuristic methods that apply to all optimality criteria:

1. **Uniform sampling**: $\hat{S}$ is sampled uniformly at random without replacement from the design pool $X$.

2. **Weighted sampling**: first the optimal solution $\pi^*$ of Eq. (6) is computed with $r = k$; afterwards, $\hat{S}$ is sampled without replacement according to the distribution specified by $\pi^*/k$. Recall that (Wang et al., 2016) proved that weighted sampling works when $k$ is sufficiently large compared to $p$ (cf. Table 1).

3. **Fedorov’s exchange** (Miller & Nguyen, 1994): the algorithm starts with a random subset $S_0 \in S_n(n,k)$ and iteratively exchanges two coordinates $i \in S_0, j \notin S_0$ such that the objective is minimized after the exchange. The algorithm terminates if no such exchange can reduce the objective, or $T$ iterations are reached.

All algorithms are implemented in MATLAB, except for the Fedorov’s exchange algorithm, which is implemented in C due to efficiency concerns. We also apply the Sherman-Morrison formula \((A + \lambda uu^T)^{-1} = A^{-1} + \frac{\lambda uu^T}{1 + \lambda u^TA^{-1}u}\) and the matrix determinant lemma \(\det(A + \lambda uu^T) = \det A + \lambda \text{Tr}(Auu^T)\).
Table 3. Results on the Minnesota wind speed dataset ($n = 2642, p = 15, k = 30$). MSE is defined as $\sqrt{\frac{1}{n}\|y - V\beta\|^2}$.

<table>
<thead>
<tr>
<th></th>
<th>$f_V$</th>
<th>MSE</th>
<th>$f_G$</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Sampling</td>
<td>94.1</td>
<td>1.10</td>
<td>3093</td>
<td>1.34</td>
</tr>
<tr>
<td>Weighted Sampling</td>
<td>21.4</td>
<td>0.89</td>
<td>2451</td>
<td>1.13</td>
</tr>
<tr>
<td>Fedorov’s Exchange</td>
<td>10.0</td>
<td>0.86</td>
<td>29.2</td>
<td>0.78</td>
</tr>
<tr>
<td>(Running time /secs)</td>
<td>15</td>
<td>-</td>
<td>1857</td>
<td>-</td>
</tr>
<tr>
<td>Full-sample OLS</td>
<td>-</td>
<td>0.55</td>
<td>-</td>
<td>0.55</td>
</tr>
</tbody>
</table>

$(I + \lambda u^T A^{-1} u^T) \det(A)$ to accelerate computations of rank-1 updates of matrix inverse and determinant. For uniform sampling and weighted sampling, we report the median objective of 50 independent trials. We only report the objective for one trial of Fedorov’s exchange method due to time constraints. The maximum number of iterations $T$ for Fedorov’s exchange is set at $T = 100$. We always set $k = r$ in the optimization problem Eq. (6), and details of solving Eq. (6) are placed in the appendix. In Algorithm 1 we set $\alpha = 10$; our simulations suggest that the algorithm is not sensitive to $\alpha$.

5.2. Synthetic data

We synthesize a $1000 \times 50$ design pool $X$ as follows:

$$X = \begin{bmatrix} X_A & 0_{500 \times 25} \\ 0_{500 \times 25} & X_B \end{bmatrix}.$$  

$X_A$ is a $500 \times 25$ random Gaussian matrix, re-scaled so that the eigenvalues of $X_A^T X_A$ satisfy a quadratic decay: $\sigma_j(X_A^T X_A) \propto j^{-2}$; $X_B$ is a $500 \times 25$ Gaussian matrix with i.i.d. standard Normal variables. Both $X_A$ and $X_B$ have comparable Frobenius norm.

In Table 2 we report results on all 6 optimality criteria ($f_A, f_D, f_T, f_E, f_V, f_G$) for $k \in \{2p, 3p, 5p, 10p\}$. We also report the running time (measured in seconds) of Algorithm 1 and the Fedorov’s exchange algorithm. The other two sampling based algorithms are very efficient and always terminate within one second. We observe that our algorithm has the best performance for $f_E$ and $f_G$, while still achieving comparable results for the other optimality criteria. It is also robust when $k$ is small compared to $p$, while sampling based methods occasionally produce designs that are not even full rank. Finally, Algorithm 1 is computationally efficient and terminates within seconds for all settings.

5.3. The Minnesota wind speed dataset

The Minnesota wind dataset collects wind speed information across $n = 2642$ locations in Minnesota, USA for a period of 24 months (for the purpose of this experiment, we only use wind speed data for one month). The 2642 locations are connected with 3304 bi-directional roads, which form an $n \times n$ sparse unweighted undirected graph $G$. Let $L = \text{diag}(d)^T G$ be the $n \times n$ Laplacian of $G$, where $d$ is a vector of node degrees, and let $V \in \mathbb{R}^{n \times p}$ be an orthonormal eigenbasis corresponding to the smallest $p$ eigenvalues of $L$. (Chen et al., 2015) shows that the relatively smooth wind speed signal $y \in \mathbb{R}^n$ can be well approximated by using only $p = 15$ graph Laplacian basis.

In Table 3 we compare the mean-square error (MSE) for prediction on the full design pool $V$: $\text{MSE} = \sqrt{\frac{1}{n} \|y - V\beta\|^2}$. Because the objective is prediction based, we only consider the two prediction related criteria: $f_V(\Sigma) = \text{tr}(V\Sigma^{-1}V^T)$ and $f_G(\Sigma) = \max \text{diag}(V\Sigma^{-1}V^T)$. The subset size $k$ is set as $k = 2p = 30$, which is much smaller than $n = 2642$. We observe that Algorithm 1 consistently outperforms the other heuristic methods, and is so efficient that its running time is negligible. It is also interesting that by using $k = 30$ samples Algorithm 1 already achieves an MSE that is comparable to the OLS on the entire $n = 2642$ design pool.

6. Concluding remarks and open questions

We proposed a computationally efficient algorithm that approximately computes optimal solutions for the experimental design problem, with near-optimal requirement on $k$ (i.e., the number of experiments to choose). In particular, we obtained a constant approximation under the very weak condition $k \geq 2p$, and a $(1 + \epsilon)$ approximation if replacement or over-sampling is allowed. Our algorithm works for all regular optimality criteria.

An important open question is to achieve $(1 + \epsilon)$ relative approximation ratio under the “proper sampling” regime $k = r$, or the “slight over-sampling” regime $k = (1 + \delta)r$, for the without replacement model. It was shown in (Wang et al., 2016) that a simple greedy method achieves $(1 + \epsilon)$ approximation ratio for A- and V-optimality provided that $k = \Omega(p^2/\epsilon)$. Whether such analysis can be extended to other optimality criteria and whether the $p^2$ term can be further reduced to a near linear function of $p$ remain open.

Another practical question is to develop fast-converging optimization methods for the continuous problem in Eq. (6), especially for criteria that are not differentiable such as the E- and G-optimality, where subgradient methods have very slow convergence rate.

Acknowledgement This work is supported by NSF grants CAREER IIS-1252412 and CCF-1563918. We thank Adams Wei Yu for providing an efficient implementation of the projection step, and other useful discussions.
References


Near-optimal design of experiments via regret minimization


