Efficient Solution of Maximum-Entropy Sampling Problems

Kurt M. Anstreicher*

July 10, 2018

Abstract

We consider a new approach for the maximum-entropy sampling problem (MESP) that is based on bounds obtained by maximizing a function of the form $\operatorname{ldet} M(x)$ over linear constraints, where $M(x)$ is linear in the n-vector x. These bounds can be computed very efficiently and are superior to all previously known bounds for MESP on most benchmark test problems. A branch-and-bound algorithm using the new bounds solves challenging instances of MESP to optimality for the first time.

Keywords: Maximum-entropy sampling, convex programming, nonlinear integer programming

1 Introduction

The maximum entropy sampling problem (MESP) is a problem that arises in spatial statistics. The problem was introduced in [17] and applied to the design of environmental monitoring networks in [8, 20]. In a typical application, C is a sample covariance matrix obtained from time-series observations of an environmental variable at n locations, and it is desired to choose s locations from which to collect subsequent data so as to maximize the information obtained. The resulting problem is then

$$\text{MESP}: \qquad z(C,s) := \max\left\{\operatorname{ldet} C[S,S] \;:\; S \subset N,\ |S| = s\right\},$$

where ldet denotes the natural logarithm of the determinant, $N = \{1, \ldots, n\}$, and for subsets $S \subset N$ and $T \subset N$, $C[S,T]$ denotes the submatrix of C having rows indexed by S and columns indexed by T. The use of entropy as a metric for information, together with the assumption that values at the n locations are drawn from a multivariate normal distribution, leads naturally to the problem MESP because $\operatorname{ldet} C[S,S]$ is, up to constants, the entropy of the Gaussian random variables having covariance matrix $C[S,S]$. For survey articles describing the MESP see [14, 15].

The study of exact algorithms for MESP was initiated in [11]. Exact algorithms to compute a maximum-entropy design use the "branch-and-bound" (B&B) framework, for which a key ingredient is the methodology for producing an upper bound on $z(C,s)$. Subsequent

*Department of Management Sciences, University of Iowa, Iowa City, IA, 52242, USA. Email: kurt-anstreicher@uiowa.edu.

nodes in the B&B tree, corresponding to indeces being fixed into or out of S, result in problems of the same form as MESP but with modified data (C', s'). A fast method that can provide a good upper bound on $z(C, s)$ is critical to the success of this approach. The exact algorithm in [11] used a bound based on the eigenvalues of C. A variety of different bounding methods have subsequently been developed and investigated [1, 2, 3, 4, 5, 9, 16], and several of these have been incorprated into complete B&B algorithms. Recent results using the optimized "masked spectral" [1, 5] and BQP bounds [2] are the most promising so far, although both of these bounds involve challenging computational problems posed over an $n \times n$ variable matrix $X \succeq 0$.

In this paper we consider a new bound for the MESP that is based on maximizing a function of the form $\operatorname{ldet} M(x)$ subject to the constraints $0 \leq x_i \leq 1$, $i = 1, \ldots, n$, $\sum_{i=1}^{n} x_i = s$, where solutions of MESP correspond to binary x. This bound has similarities with both the NLP bound [3, 4] and the BQP bound [2]. The NLP bound is based on maximizing a function of the form $\operatorname{ldet} M(x)$ over the same constraints, where $M(x)$ is a nonlinear function of x. For appropriate parameter choices this function $\operatorname{ldet} M(x)$ is provably concave, although $M(x)$ is too complex for $\operatorname{ldet} M(x)$ to be recognized as concave by the cvx convex-programming system [7]. The BQP bound [2] is based on maximizing a function of the form $\operatorname{ldet} M(X)$, where $M(X)$ is linear in the $n \times n$ matrix variable $X \succeq 0$. We refer to the new bound as the "linx" bound because $M(x)$ is linear in x. Validity of the linx bound is based on a simple but previously unexploited determinant identity (Lemma 1). Advantages of the linx bound over the NLP, BQP and other bounds for the MESP include the following:

- The linx bound can be computed more efficiently than the NLP bound because $M(x)$ is linear in x. The linx bound can be computed much more efficiently than the BQP bound because the linx bound is based on a matrix $M(x)$, $x \in \mathbb{R}^n_+$ instead of a matrix $M(X)$ where $X \succeq 0$.

- Because the linx bound includes variables that explicitly correspond to choosing a submatrix of C, like the NLP and BQP bounds, the bound can be directly employed to handle instances of the constrained MESP (CMESP) that contains additional constraints $\sum_{j \in S} a_{ij} \leq b_i$, $i = 1, \ldots, m$ by simply adding the inequality constraints $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$, $i = 1, \ldots, m$. The fact that such linear constraints can be directly incorporated is a considerable advantage over other bounds for CMESP, such as the eigenvalue-based bound for CMESP described in [13].

- Because the linx bound is based on a convex programming problem, like the NLP and BQP bounds, duality information can be used to potentially fix variables to 0/1 values. Such variable-fixing logic is not possible using bounds based on eigenvalues, including the optimized masked spectral bound [1, 5]. The variable-fixing logic associated with the linx bound is much simpler than for the BQP bound, because the logic in the BQP bound must be based on the SDP condition $X - \operatorname{diag}(X)\operatorname{diag}(X)^T \succeq 0$ rather than the explict bounds $0 \leq x \leq e$. Duality information can also be used as the basis for a strong branching strategy in a B&B implementation.

- In general the optimal solution value for MESP is the same as the value for a "complementary problem" that is based on replacing C with C^{-1} and s with $n - s$. Many bounds for the MESP, including the NLP, BQP and masked-spectral bounds, can differ for the original and complementary problems, and therefore in general both bounds must be computed to obtain the best possible result. We prove that for the linx bound, as for the basic eigenvalue bound [11], the bound based on the complementary problem is identical to the bound based on the original problem and therefore consideration of the complementary problem is unnecessary.

In the next section we define the linx bound, establish its validity and show that the bound is the same for the complementary problem. We also describe a dual problem and show how dual variables can be used to potentially fix variables to 0/1 values. The linx bound, like the NLP and BQP bounds, is sensitive to a scaling of the covariance matrix C. We describe an initial choice for the scaling parameter that has performed well computationally as well as a simple procedure that can be used to approximately optimize the scaling parameter. Computational results using benchmark data sets with $n = 63$ and $n = 124$ as well as a new data set with $n = 90$ show that the BQP bound performs extremely well compared to all previously known bounds for MESP, including the masked spectral [1, 5] and BQP [2] bounds. In Section 3 we describe the implementation of a complete B&B algorithm using the BQP-based bounds. This implementation obtains the best computational results to date on the benchmark instances with $n = 63$, as well as the first optimal solutions for the most difficult benchmark instances with $n = 124$.

Notation: I is an identity matrix and e is a vector of all ones, with the dimension implicit for both. For a square matrix X, $X \succeq 0$ denotes that X is symmetric and positive semidefinite and $\operatorname{diag}(X)$ is the vector of diagonal entries of X. For a vector x, $\operatorname{Diag}(x)$ is the diagonal matrix such that $x = \operatorname{diag}(\operatorname{Diag}(x))$. For square matrices X and Y, $X \bullet Y$ denotes the matrix inner product $X \bullet Y = \operatorname{tr}(XY^T)$.

2 The linx bound

The linx bound is based on the following simple determinant identity, which has apparently never been exploited in the context of the MESP.

Lemma 1. *For a subset $S \subset N = \{1, \ldots, n\}$ let $x_i = 1$, $i \in S$ and $x_i = 0$, $i \in N \setminus S$. Then $\operatorname{ldet}(C \operatorname{Diag}(x) C + I - \operatorname{Diag}(x)) = 2 \operatorname{ldet} C_{SS}$.*

Proof. Let $T = N \setminus S$. After a symmetric permutation of indeces we can write

$$C \operatorname{Diag}(x) C = \begin{pmatrix} C_{SS} & C_{ST} \\ C_{ST}^T & C_{TT} \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} C_{SS} & C_{ST} \\ C_{ST}^T & C_{TT} \end{pmatrix} = \begin{pmatrix} C_{SS}^2 & C_{SS}C_{ST} \\ C_{ST}^T C_{SS} & C_{ST}^T C_{ST} \end{pmatrix},$$

and therefore

$$C \operatorname{Diag}(x) C + I - \operatorname{Diag}(x) = \begin{pmatrix} C_{SS}^2 & C_{SS}C_{ST} \\ C_{ST}^T C_{SS} & C_{ST}^T C_{ST} + I \end{pmatrix}.$$

Applying the well-known Schur determinant formula [10, p.21], we then obtain

$$\begin{aligned} \operatorname{ldet}(C \operatorname{Diag}(x) C + I - \operatorname{Diag}(x)) &= 2 \operatorname{ldet} C_{SS} + \operatorname{ldet}\left(C_{ST}^T C_{ST} + I - C_{ST}^T C_{SS} C_{SS}^{-2} C_{SS} C_{ST}\right) \\ &= 2 \operatorname{ldet} C_{SS}. \end{aligned}$$

□

We define the linx bound via the convex programming problem

$$
\begin{aligned}
z_{\text{linx}}(C,s) = \quad & \max \quad \frac{1}{2}\,\text{ldet}\,(C\,\text{Diag}(x)C + I - \text{Diag}(x)) \\
& \text{s.t.} \quad e^T x = s \\
& \qquad\;\; 0 \le x \le e.
\end{aligned}
\tag{1}
$$

Validity of the bound, $z_{\text{linx}}(C,s) \ge z(C,s)$ then follows immediately from Lemma 1, and solving MESP corresponds to finding the optimal binary solution of (1). The convex optimization problem (1) has a dual which also corresponds to a determinant maximization problem with linear constraints. The dual problem can be derived using the general theory for determinant maximization problems described in [19]; note in particular that (1) is structurally similar to the problem of D-optimal design considered in detail in [19]. Alternatively the dual of (1) can be derived via explicit consideration of the Lagrangian as was done for the BQP bound in [2]. In any case the resulting dual problem is

$$
\begin{aligned}
\min \quad & -\frac{1}{2}\,\text{ldet}\, S \\
\text{s.t.} \quad & \text{tr}(S) + su + e^T v = n \\
& \text{diag}(CSC) - \text{diag}(S) \le ue + v \\
& v \ge 0.
\end{aligned}
\tag{2}
$$

In the dual problem (2) the unconstrained variable u is associated with the constraint $e^T x = s$, and the dual variables $v \ge 0$ are associated with the upper bound constraints $x \le e$. The slack variables in the other inequality constraints of (2) are the dual variables associated with the nonnegativity constraints $x \ge 0$. To demonstrate weak duality between (1) and (2), let x be feasible for (1), (u,v,S) be feasible for (2) and $w = ue + v + \text{diag}(S) - \text{diag}(CSC) \ge 0$. The objective gap between the primal and dual solutions, for convenience ignoring the factor $1/2$ in both, is then

$$
\begin{aligned}
-\,\text{ldet}\, S - \text{ldet}\, M(x) &= -\,\text{ldet}\, S - \text{ldet}\,(C\,\text{Diag}(x)C + I - \text{Diag}(x)) \\
&= -\,\text{ldet}\, S^{1/2}(C\,\text{Diag}(x)C + I - \text{Diag}(x))S^{1/2} \\
&\ge -\,\text{diag}(CSC)^T x - \text{tr}(S) + \text{diag}(S)^T x + n \\
&= w^T x - ue^T x - v^T x - \text{tr}(S) + n \\
&= u(s - e^T x) + v^T(e - x) + w^T x + n - us - v^T e - \text{tr}(S) \\
&= v^T(e - x) + w^T x,
\end{aligned}
\tag{3}
$$

where the inequality uses the fact that for any $X \succ 0$, $\text{ldet}\, X \le \text{ldet}\, I + \text{tr}(X - I) = \text{tr}(X) - n$, which follows from the concavity of $\text{ldet}(\cdot)$ and the fact [6, p.75] that

$$
\frac{\partial\, \text{ldet}(X)}{\partial X} = X^{-1}. \tag{4}
$$

Strong duality holds between problems (1) and (2) because both problems satisfy a Slater condition [19].

The weak duality condition (3) can be used to derive variable-fixing logic for MESP when the linx bound is applied. Let $\hat{z}$ be the objective value for a known solution of MESP. Assume that $\bar{x}$ solves (1), and $(\bar{u}, \bar{v}, \bar{w}, \overline{S})$ solves (2), where $\bar{w} = \bar{u} + \bar{v} + \text{diag}(\overline{S}) - \text{diag}(C\overline{S}C)$. Let $z_{\text{linx}} = \frac{1}{2}\,\text{ldet}\, M(\bar{x}) = -\frac{1}{2}\,\text{ldet}\, \overline{S}$. From (3) we know that for any other x feasible in (1),

$$
\text{ldet}\, M(x) \le 2z_{\text{linx}} - \bar{v}^T(e - x) - \bar{w}^T x.
$$

We immediately conclude that if x^* is a binary solution in (1) with objective value greater than $\hat{z}$, then

$$\bar{w}_i \geq 2(z_{\text{linx}} - \hat{z}) \implies x_i^* = 0, \tag{5}$$
$$\bar{v}_i \geq 2(z_{\text{linx}} - \hat{z}) \implies x_i^* = 1. \tag{6}$$

The dual variables $(\bar{v}, \bar{w})$ associated with a solution of (1) can also be used to devise a strong branching strategy in the context of a B&B algorithm for MESP, which is considered in the next section.

As described in the introduction, for any MESP there is a "complementary" problem that has the same solution value. The complementary problem is defined by replacing C with C^{-1} and s with $n-s$. Using the identity

$$\operatorname{ldet} C[S,S] = \operatorname{ldet} C + \operatorname{ldet} C^{-1}[N \setminus S, N \setminus S],$$

we then have the identity

$$z(C,s) = z(C^{-1}, n-s) + \operatorname{ldet} C.$$

As a result, any bound for the complementary problem, adjusted by the constant $\operatorname{ldet} C$, provides a bound for the original problem. Use of the complementary problem is important when employing many bounds for MESP, including the NLP, optimized masked spectral and BQP bounds. In the next lemma we show that consideration of the complementary problem is unnecessary for the linx bound.

Lemma 2. *For any C and s, the value obtained using the linx bound is the same for the original and complementary problems.*

Proof. It is clear that x is a feasible solution for (1) if and only if $\bar{x} = e - x$ is a feasible solution for the corresponding complementary problem. The objective value associated with $\bar{x}$ in the complementary problem, including adjustment by the constant $\operatorname{ldet} C$, is then

$$\begin{aligned}
& \frac{1}{2}\operatorname{ldet}\left(C^{-1}\operatorname{Diag}(\bar{x})C^{-1} + I - \operatorname{Diag}(\bar{x})\right) + \operatorname{ldet} C \\
= \; & \frac{1}{2}\operatorname{ldet}\left(C^{-1}(\operatorname{Diag}(\bar{x}) + C(I - \operatorname{Diag}(\bar{x}))C)C^{-1}\right) + \operatorname{ldet} C \\
= \; & \frac{1}{2}\operatorname{ldet}\left(C(I - \operatorname{Diag}(\bar{x}))C + \operatorname{Diag}(\bar{x})\right) \\
= \; & \frac{1}{2}\operatorname{ldet}\left(C(\operatorname{Diag}(x)C + I - \operatorname{Diag}(x)\right),
\end{aligned}$$

which is exactly the objective value for the solution x in the original problem. Thus for any solution x in the original problem there is a solution $\bar{x}$ for the complementary problem with the same objective value, and vice versa. □

For an instance of MESP with data matrix C, and a positive scalar γ, note that $z(\gamma\, C, s) = z(C,s) + s \ln \gamma$. It follows that when applying any bound for MESP we are free to scale C by a value γ and then adjust the resulting bound by subtracting $s \ln \gamma$. Some bounds for the MESP, including the eigenvalue bound [11], diagonal bound [9], and masked spectral bound [1, 5] are invariant to such a scaling operation. However the linx bound, like the BQP bound [2], and NLP bound [3, 4] is sensitive to the choice of scaling factor. To choose an appropriate scale factor for the linx bound we use a procedure similar to that employed for the BQP bound in [2]. Define the function

$$v(\gamma, X) = \operatorname{ldet}(\gamma C \operatorname{Diag}(x) C + I - \operatorname{Diag}(x)) - s \ln \gamma.$$

Note that $v(\gamma, \cdot)$ corresponds to scaling C by $\sqrt{\gamma}$, so the objective in (1), adjusted for the value of γ, is exactly $v(\gamma, x)/2$. Using (4), we have

$$\frac{\partial}{\partial \gamma} v(\gamma, x) = F(\gamma, x)^{-1} \bullet (C \operatorname{Diag}(x) C) - s/\gamma,$$

where $F(\gamma, x) = \gamma C \operatorname{Diag}(x) C + I - \operatorname{Diag}(x)$. For γ to be a minimizer of $v(\gamma, x)$ we then require that $\gamma F(\gamma, x)^{-1} \bullet (C \operatorname{Diag}(x) C) = s$, which is equivalent to

$$(e - x)^T \operatorname{diag}(F(\gamma, x)^{-1}) = n - s. \tag{7}$$

To use (7) to improve the scale factor γ we use the fact [6, p.64] that

$$\frac{\partial}{\partial \gamma}(e - x)^T \operatorname{diag}(F(\gamma, X)^{-1}) = -(e - x)^T \operatorname{diag}\left(F(\gamma, x)^{-1} C \operatorname{Diag}(x) C F(\gamma, X)^{-1}\right) \tag{8}$$

to make a first-order correction to γ in an attempt to satisfy (7). Note that (8) implies that the left-hand side of (7) is monotonically decreasing in γ, so (7) has a unique solution. However, if γ is changed then (1) must be re-solved with the new scaling factor applied, so x may also change. For the BQP bound [2], excellent results were obtained using an initial scale factor of $\gamma = 1/\operatorname{diag}(C)_{[s]}$, where $\operatorname{diag}(C)_{[s]}$ is the s'th largest component of $\operatorname{diag}(C)$. To use this same value for the linx bound it seems that it might be appropriate to square it due to the form of (1). In the end we consider an initial value of the form $\gamma = 1/\operatorname{diag}(C)_{[s]}^p$, where $1 \le p \le 2$. The choice of p is determined empirically and varies with C; typically we use either $p = 1$, $p = 1.5$ or $p = 2$. We then update the scale factor until (7) is satisfied within a tolerance of 0.25, which typically requires the solution of 3 or 4 problems of the form (1). We observe that the linx bound appears to be more sensitive to the scaling factor than the BQP bound, and sometimes requires large changes to the initial scaling factor so as to approximately satisfy (7). For this reason, when updating the scale factor γ to a new value γ^+ based on (7) and (8) we use a safeguard of the form $1/5 \le \gamma^+/\gamma \le 5$ to prevent excessively large changes on one update.

To evaluate the linx bound (1) we first consider two benchmark data sets that have been repeatedly used in the literature on MESP. The first, having a matrix C of size $n = 63$, is from [8] and the second, having a matrix C of size $n = 124$ was introduced in [9]. Figures 1 and 2 give the objective gaps between several bounds and the best feasible solutions generated by a heuristic [11] which applies greedy and interchange heuristics to the original and complementary problems. In Figure 1, gap values are given for $s = 3, 4, \ldots, 60$, while in Figure 2 they are given for $s = 10, 20, \ldots, 120$. The eigenvalue and linx bounds are the same for the original and complementary problems; for all other bounds the value reported is the better of the two. In Figure 1 the values for the optimized masked spectral bound (MS) are taken from the computational results of [5], while in Figure 2 they are taken from [1]. For the instances with $n = 63$ the linx and BQP bounds are comparable and dominate the other bounds for all but very small or very large s. For problems of this size the CPU time required to solve one instance of the form (1) using the Matlab-based SDPT3 solver [18] on a PC with an Intel i7-6700 CPU running at 3.40 GHz, with 16G of RAM and a 64-bit OS is

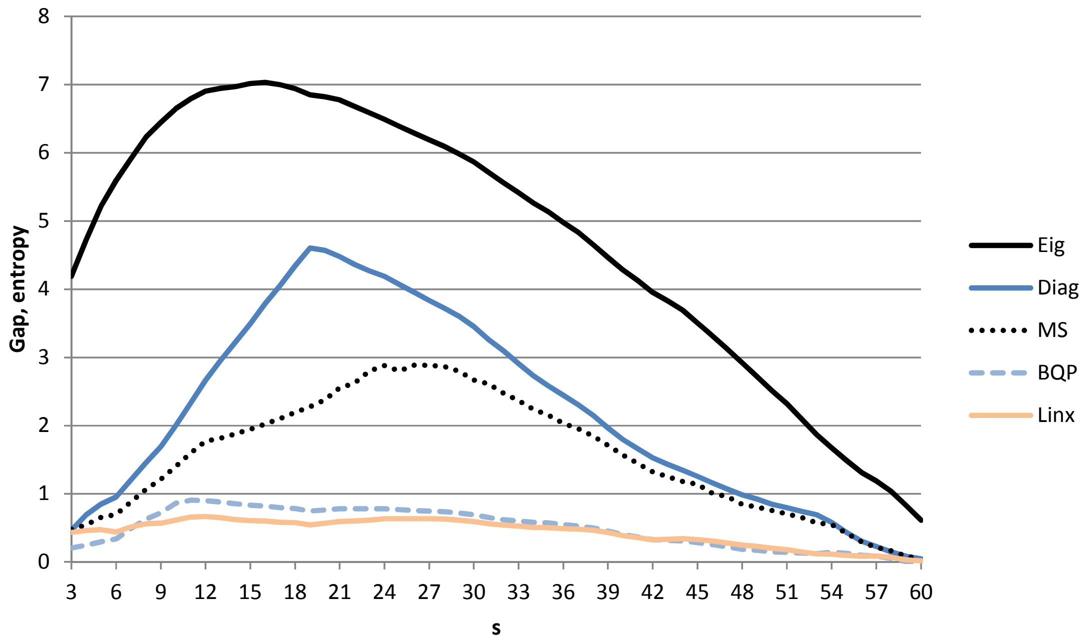

Figure 1: Entropy gaps for bounds on instances with $n = 63$

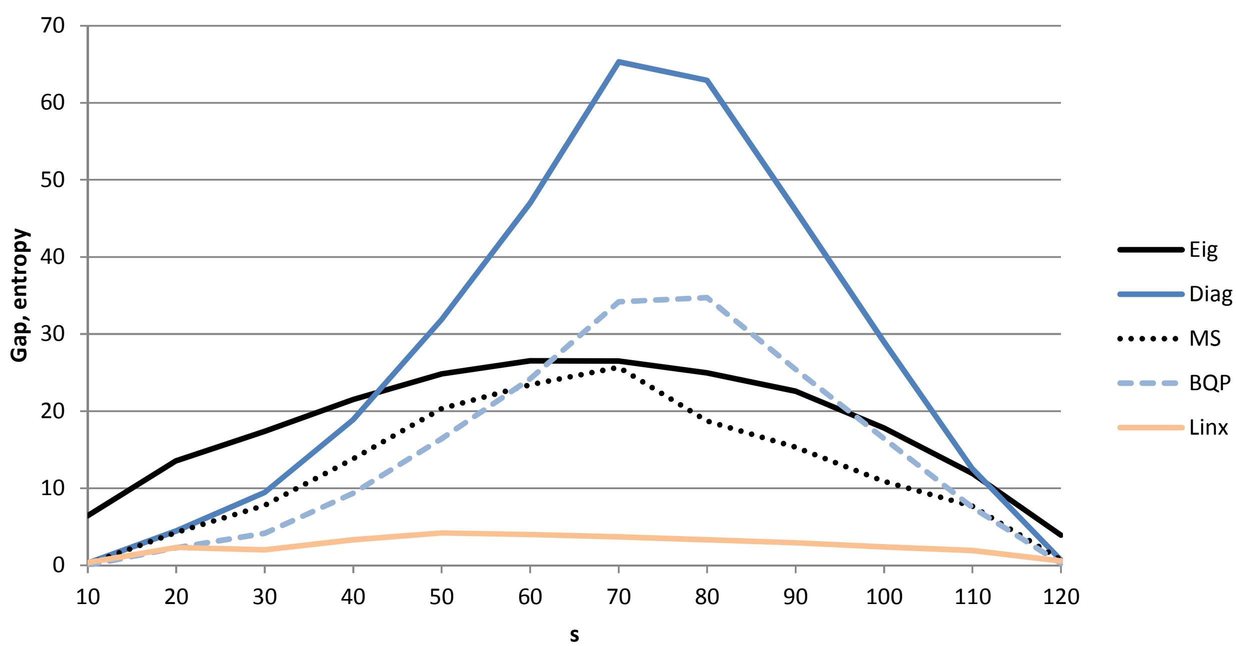

Figure 2: Entropy gaps for bounds on instances with $n = 124$

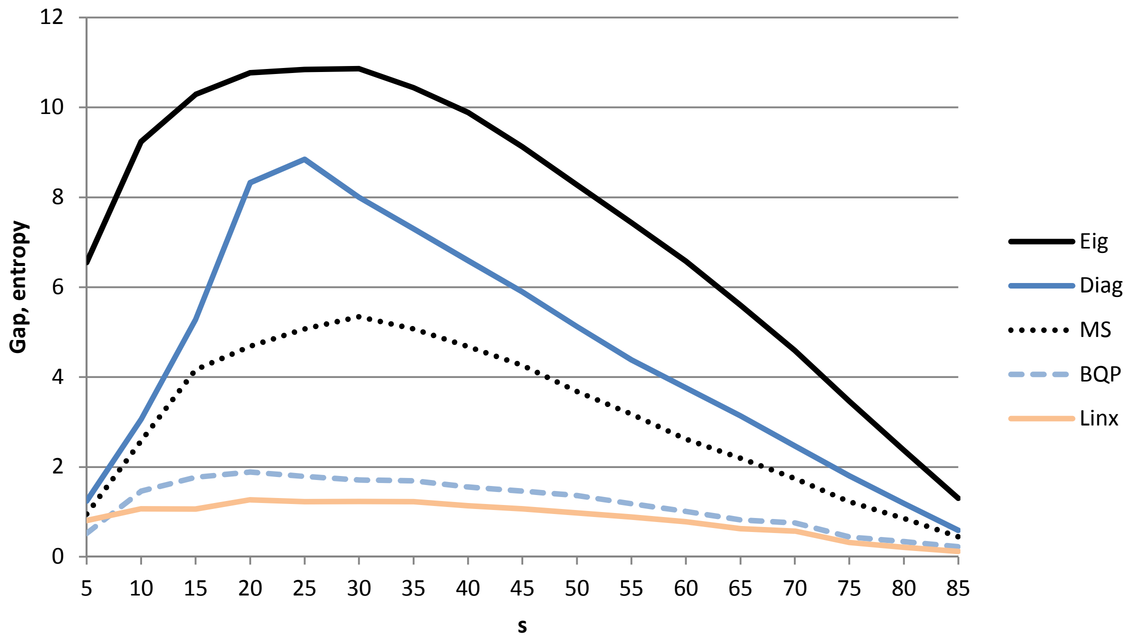

Figure 3: Entropy gaps for bounds on instances with $n = 90$

less than 0.5 seconds, which is approximately 10% of the time required to solve one instance of the corresponding convex optimization problem associated with the BQP bound [2]. This time difference grows with the dimension n; for the problems of size 124, one instance of (1) requires approximately 4 seconds to solve, compared to approximately 100 seconds for the corresponding problem from [2]. For the problems of size $n = 124$ it is clear from Figure 2 that the linx bound is dramatically better than previously known bounds for all but very small or very large values of s.

In addition to the benchmark instances with $n = 63$ and $n = 124$ we considered another data set having a C with $n = 90$ that is based on temperature data from monitoring stations in the Pacific Northwest of the United States [12]. Gaps between several bounds and the heuristic value computed for $s = 5, 10, \ldots, 85$ are shown in Figure 3, which again shows excellent results for the linx bound.

3 Branch and Bound implementation

In this section we consider a Matlab-based B&B implementation that uses the linx bound to solve instances of MESP to optimality. We first describe some important features of our B&B implentation and then give computational results on the same problems considered in the previous section.

Each node in the B&B tree corresponds to a subproblem where some indeces are fixed into or out of S, corresponding to variables x_i being fixed to value 1 or 0, respectively. In MESP, fixing a set of indeces $F \subset N$ into S can be accomplished [11] by forming the reduced problem where C is replaced by the Schur complement

$$C[N', N'] - C[N', F]\, C[F, F]^{-1}\, C[F, N'], \tag{9}$$

where $N' = N \backslash F$, s is replaced with $s' = s - |F|$, and the objective is adjusted by the constant ldet(C_{FF}). Fixing indeces out of S is accomplished by simply deleting the corresponding rows and columns of C.

Our B&B implementation is based on the B&B code using the BQP bounds described in [2, Section 5], and uses the SDPT3 solver [18] to solve instances of (1). Details of the implementation include the following.

1. The B&B tree is initialized with a root node corresponding to the given MESP, and a best-known value (BKV) obtained from a set of heuristics for MESP [11]. We maintain a queue of unfathomed nodes from the tree, which is processed using depth-first search. Each node corresponds to a subproblem with some variables fixed to 0/1 values. When a node is removed from the queue we form the reduced MESP problem obtained by eliminating variables fixed to zero, and fixing variables to one using (9). The resulting problem is an instance of MESP where $s' \leq s$ indeces must be chosen from $n' \leq n$ candidate indeces. If $s' = 1$ or $s' = n' - 1$ then we enumerate the possible solutions. If a solution is found with objective value better than the BKV then the BKV is updated and the set of corresponding indeces saved. In either case the node is discarded.

2. If $1 < s' < n' - 1$ we first check if the bound inherited from the node's parent is less than the BKV. If so (which is possible since the BKV may have been updated since the node was placed on the queue) then the node is discarded. Otherwise we compute a bound by solving a problem of the form (1). At the root node we perform updates to the scale factor γ based on (7) and (8) until (7) is satisfied to a tolerance of 0.25. At all other nodes we compute a bound using the updated scale factor γ inherited from the node's parent. If (7) *not* is satisfied within a tolerance of 0.25 then we compute one update of the scale factor and re-compute the bound; in any case we compute a final update to γ based on (7) and (8) and pass this updated value of γ to child nodes. If the computed bound is less than the BKV then we discard the node. Otherwise we apply the variable-fixing logic from (5) and (6) to fix additional variables to 0/1 values, if possible. Following additional variable-fixing we are left with the problem of choosing s'' indeces from n'' candidate indeces. If $s'' \leq 1$ or $s'' \geq n'' - 1$ then we enumerate the possible solutions, update the BKV if applicable, and discard the node.

3. $1 < s'' < n'' - 1$ then we will replace the node with two children by branching on one index. To choose the branching index we use a strong branching strategy based on the dual variables v and w. Let $\delta_{\max}$ be the maximum of all of the v_i and w_i values for the remaining variables, and let $i_{\max}$ be the index for which this is maximum is attained. Initially we branch on $x_{i_{\max}}$, putting the "easy" node on the queue first and the "hard" node second. If $\delta_{i_{\max}} = w_{i_{\max}}$ then the easy node has $x_{i_{\max}} = 1$ and the hard node has $x_{i_{\max}} = 0$, with the reverse if $\delta_{\max} = v_{i_{\max}}$. Putting the hard node on second and using depth-first search induces an initial "dive" in the B&B tree. This dive eventually produces a feasible solution and typically leads to rapid update of the BKV when the initial BKV is not the optimal value. After the initial dive is complete, resulting in a node being fathomed, we continue to use the strong branching criterion if the value $\delta_{\max}$ is sufficiently large, but switch the order and place the hard node on the queue first and the easy node second. We use simple criteria based on the absolute value of $\delta_{\max}$ as well as the value of $\delta_{\max}$ relative to the current gap to decide if $\delta_{\max}$ is sufficiently large, and if not we simply branch on the variable i with the minimum value of $|x_i - .5|$ in the solution of (1).

Table 1: B&B statistics for $n = 124$ instances

s	Heuristic Value	Optimal Value	Root Gap	Nodes	Time (hours)	Time to Opt (sec)
20	76.999	77.827	2.306	1,239	0.21	181
30	106.674	106.700	2.010	3,413	0.47	147
40	130.162	131.055	3.304	11,202	2.42	167
50	148.661	149.498	4.197	159,302	51.81	1,515
60	163.371	164.012	3.992	171,383	67.01	171
70	172.243	172.528	3.680	140,240	37.93	215
80	174.813	175.091	3.298	42,989	12.71	230
90	171.262	171.262	2.918	21,495	4.82	249
100	162.616	162.865	2.393	5,334	1.15	298
110	147.730	147.933	1.919	1,589	0.40	343

We first consider the instances of MESP with $n = 63$ described in Section 2. These problems were first solved to optimality by a B&B algorithm using the NLP bound [4], and have subsequently been solved using the masked spectral [5] and BQP bounds [2]. For these problems the objective value obtained by the heuristic is either optimal or very close to optimal; the largest difference of approximately 0.04 occurs for the instance with $s = 26$. Our B&B algorithm using the linx bound solves any of these instances to optimality in less than one-half hour on a 3.4 GHz PC, which is approximately 60% of the maximum time required when using the BQP bound [2] on the same machine. Given the difference in solution times for computation of the linx and BQP bounds described in the previous section, one might expect that the speed improvement for the B&B algorithm using the linx bound would be greater. Recall, however, that the root gaps for the linx and BQP bounds are very similar on these instances, as shown in Figure 1, with the gap for the BQP bound lower in some cases. The number of nodes required when using the linx bound is often somewhat higher than when using the BQP bound to solve these problems, and the speed differential between the two bounds will decrease as the dimension of subproblems decreases in the B&B tree.

Most instances using the data matrix C with $n = 124$ have to date been out of reach for exact methods. The most difficult instance previously solved to optimality, with $s = 30$, required 93,652 nodes and 117 hours of computation on a 3.4 GHz PC, using the BQP bound [2]. Using the linx bound, this problem is solved to optimality using only 3,413 nodes and about 30 minutes of CPU time on the same machine, with approximately 75% of the CPU time utilized by the SDPT3 solver. Statistics for the solutions of instances with $s = 20, 30, \ldots, 110$ are given in Table 1. Instances of MESP for a given C are generally most difficult for values of s that are approximately $n/2$. As can be seen from Table 1 the differences between heuristic[1] and optimal values are substantially higher for some of these problems than for the problems with $n = 63$. A notable feature of our B&B algorithm is that the initial dive based on dual information is very effective in rapidly generating the optimal solution. In most cases this occurs shortly after the first node is fathomed at the end of the initial dive, although in some cases (for example the instance with $s = 50$) the optimal solution is found somewhat later in the solution process.

Finally we consider the MESP instances using the matrix C with $n = 90$. Given our success in solving the instances with $n = 124$, and the excellent gaps for the linx bounds

[1] The heuristic value for the instance with $s = 20$ is incorrectly given in Table 1 of [2].

Table 2: B&B statistics for $n = 90$ instances

s	Heuristic Value	Best Value	Root Gap	Final Tol	Nodes	Time (hours)	Time to Best (sec)
10	58.532	58.532	1.068	0.00	8,923	0.58	61
20	111.352	111.482	1.269	0.00	169,339	26.66	54
30	161.515	161.539	1.234	0.25	222,126	46.61	55
40	209.958	209.969	1.132	0.35	215,170	52.04	232
50	257.115	257.160	0.977	0.25	178,143	24.42	52
60	302.975	303.019	0.782	0.15	47,114	3.45	53
70	347.353	347.471	0.575	0.00	5,289	0.29	59
80	389.997	389.997	0.213	0.00	137	0.01	19

shown in Figure 3, we were surprised to find some of these problems to be more difficult to solve to optimality than the corresponding problems with $n = 124$. For example, the instance with $s = 20$ was solved to optimality using 169,339 nodes and 26.66 hours. This is much more effort than required for the instance with $n = 124$, $s = 20$ as reported in Table 1, despite the fact that the root gap and number of possible solutions for the problem with $n = 90$ are both considerably lower. One possibility that could lead to increased complexity to verify optimality in MESP is the situation where a problem has many near-optimal solutions. We believe that this is likely to be the case for the matrix with $n = 90$. The locations of the monitoring stations from which this data was obtained are shown in the map in Figure 4 [12]. It is obvious from the map that there are many locations that are quite close together, which could naturally lead to many near-optimal solutions. Given the difficulty of proving optimality for some of the instances with $n = 90$, we implemented the B&B algorithm allowing for the use of a positive tolerance for fathoming nodes (this same tolerance can also then be applied in the variable-fixing logic). The result using a final tolerance $\tau > 0$ is then that there is no feasible solution to the problem with objective higher than the BKV plus τ. In Table 2 we give results applying the B&B algorithm using the linx bound to instances with $n = 90$. The problems with $s = 10, 20, 70$ and 80 were solved to optimality, while the problems with $s = 30, 40, 50$ and 60 were solved with a tolerance that was initially zero and was then increased to a positive value after one-half hour of CPU time had been used. Based on our experience with other instances of MESP, the initial use of a zero tolerance makes it seem likely that the BKV obtained after one-half hour of computation would be the optimal value. For the instance with $s = 40$ we know that this is not the case. For this problem solutions with the heuristic value of 209.958 and a higher value of 209.965 were found in less than 60 seconds, and a solution with a value of 209.969 was found after less than 4 minutes. However on an earlier attempt to solve the problem to optimality, solutions with values of 209.971 and 209.972 were found much later. The extremely close objective values of these distinct solutions gives further evidence that some of the instances with $n = 90$ suffer from a multiplicity of near-optimal solutions.

Acknowledgement

I am grateful to Jon Lee for many valuable conversations, and to N. Le, W. Wang and J. Zidek for providing the data matrix with $n = 90$ as well as the map of monitoring station locations in Figure 4.

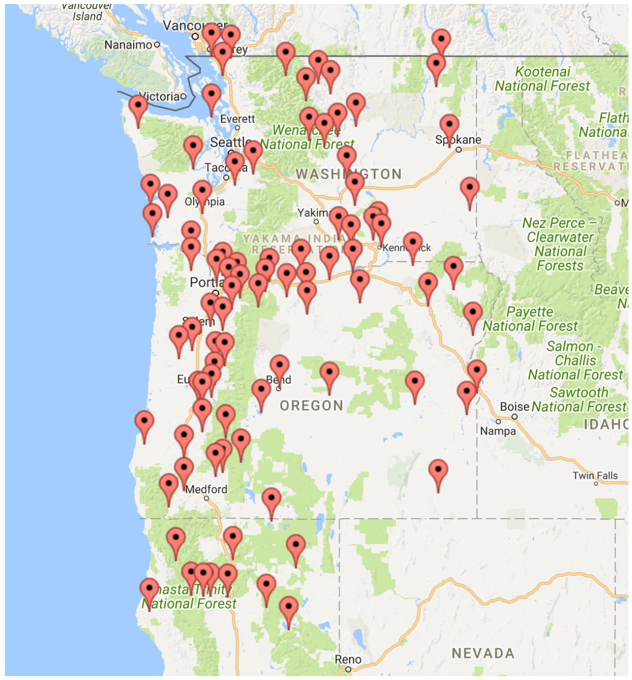

Figure 4: Locations of monitoring stations for matrix with $n = 90$

References

[1] K. Anstreicher and J. Lee. A masked spectral bound for maximum-entropy sampling. In *mODa 7—Advances in model-oriented design and analysis*, Contrib. Statist., pages 1–10. Physica, Heidelberg, 2004.

[2] K. M. Anstreicher. Maximum-entropy sampling and the Boolean quadric polytope. *J. Global Optim.*, to appear, 2018.

[3] K. M. Anstreicher, M. Fampa, J. Lee, and J. Williams. Continuous relaxations for constrained maximum-entropy sampling. In *Integer Programming and Combinatorial Optimization (Vancouver, BC, 1996)*, volume 1084 of *Lecture Notes in Comput. Sci.*, pages 234–248. Springer, Berlin, 1996.

[4] K. M. Anstreicher, M. Fampa, J. Lee, and J. Williams. Using continuous nonlinear relaxations to solve constrained maximum-entropy sampling problems. *Mathematical Programming*, 85(2, Ser. A):221–240, 1999.

[5] S. Burer and J. Lee. Solving maximum-entropy sampling problems using factored masks. *Mathematical Programming Series B*, 109:263–281, 2007.

[6] A. Graham. *Kronecker Products and Matrix Calculus: with Applications.* Ellis Horwood, Chichester, 1981.

[7] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, 2017.

[8] P. Guttorp, N. Le, P. Sampson, and J. Zidek. Using entropy in the redesign of an environmental monitoring network. Technical Report 116, Department of Statistics, The University of British Columbia, 1992.

[9] A. Hoffman, J. Lee, and J. Williams. New upper bounds for maximum-entropy sampling. In *mODa 6—Advances in model-oriented design and analysis (Puchberg/Schneeberg, 2001)*, Contrib. Statist., pages 143–153. Physica, Heidelberg, 2001.

[10] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 1985.

[11] C.-W. Ko, J. Lee, and M. Queyranne. An exact algorithm for maximum entropy sampling. *Operations Research*, 43(4):684–691, 1995.

[12] N. Le, W. Wang, and J. Zidek. Private communication, 2017.

[13] J. Lee. Constrained maximum-entropy sampling. *Operations Research*, 46(5):655–664, 1998.

[14] J. Lee. Semidefinite programming in experimental design. In R. Saigal, H. Wolkowicz, and L. Vandenberghe, editors, *Handbook of Semidefinite Programming*, volume 27 of *International Series in Operations Research and Management Science*, pages 528–532. Kluwer Acad. Publ., Boston, MA, 2000.

[15] J. Lee. Maximum-entropy sampling. In A. H. El-Shaarawi and W. W. Piegorsch, editors, *Encyclopedia of Environmetrics*, volume 3, pages 1229–1234. John Wiley & Sons Inc., 2001.

[16] J. Lee and J. Williams. A linear integer programming bound for maximum-entropy sampling. *Mathematical Programming*, 94(2-3, Ser. B):247–256, 2003.

[17] M. Shewry and H. P. Wynn. Maximum entropy sampling. *Journal of Applied Statistics*, 46:165–170, 1987.

[18] K.-C. Toh, M. J. Todd, and R. H. Tutuncu. On the implementation and usage of SDPT3 - a MATLAB software package for semidefinite-quadratic-linear programming, version 4.0. In M. Anjos and J. Lasserre, editors, *Handbook on Semidefinite, Conic and Polynomial Optimization*, volume 166 of *International Series in Operations Research and Management Science*, pages 715–754. Springer, 2012.

[19] L. Vandenberghe, S. Boyd, and S.-P. Wu. Determinant maximization with linear matrix inequality constraints. *SIAM Journal on Matrix Analysis and Applications*, 19(2):499–533, 1998.

[20] S. Wu and J. V. Zidek. An entropy based review of selected NADP/NTN network sites for 1983-86. *Atmospheric Environment*, 26A:2089–2103, 1992.