The Annals of Statistics
2015, Vol. 43, No. 5, 2198–2224
DOI: 10.1214/15-AOS1339

COMPUTING EXACT D-OPTIMAL DESIGNS BY MIXED INTEGER SECOND-ORDER CONE PROGRAMMING

BY GUILLAUME SAGNOL AND RADOSLAV HARMAN[1]

Zuse Institut Berlin and Comenius University, Bratislava

Let the design of an experiment be represented by an s-dimensional vector $\mathbf{w}$ of weights with nonnegative components. Let the quality of $\mathbf{w}$ for the estimation of the parameters of the statistical model be measured by the criterion of D-optimality, defined as the mth root of the determinant of the information matrix $M(\mathbf{w}) = \sum_{i=1}^{s} w_i A_i A_i^T$, where $A_i, i = 1, \dots, s$ are known matrices with m rows.

In this paper, we show that the criterion of D-optimality is second-order cone representable. As a result, the method of second-order cone programming can be used to compute an approximate D-optimal design with any system of linear constraints on the vector of weights. More importantly, the proposed characterization allows us to compute an *exact* D-optimal design, which is possible thanks to high-quality branch-and-cut solvers specialized to solve mixed integer second-order cone programming problems. Our results extend to the case of the criterion of D_K-optimality, which measures the quality of $\mathbf{w}$ for the estimation of a linear parameter subsystem defined by a full-rank coefficient matrix K.

We prove that some other widely used criteria are also second-order cone representable, for instance, the criteria of A-, A_K-, G- and I-optimality.

We present several numerical examples demonstrating the efficiency and general applicability of the proposed method. We show that in many cases the mixed integer second-order cone programming approach allows us to find a provably optimal exact design, while the standard heuristics systematically miss the optimum.

1. Introduction. Consider an optimal experimental design problem of the form

$$\max_{\mathbf{w} \in \mathcal{W}} \Phi\left(\sum_{i=1}^{s} w_i A_i A_i^T\right), \tag{1.1}$$

where Φ is a criterion mapping the space $\mathbb{S}_m^+$ of $m \times m$ positive semidefinite matrices over the set $\mathbb{R}_+ := [0, \infty)$. In (1.1), $A_i \in \mathbb{R}^{m \times \ell_i}$, $i = 1, \dots, s$ are known matrices, and $\mathcal{W}$ is a compact subset of $\mathbb{R}_+^s$ representing the set of all permissible designs.

Received December 2013; revised January 2015.

[1]Supported by the VEGA 1/0163/13 grant of the Slovak Scientific Grant Agency.

MSC2010 subject classifications. Primary 62K05; secondary 65K05.

Key words and phrases. Optimal experimental design, exact optimal designs, second-order cone programming, mixed integer programming, D-criterion.

Problem (1.1) arises in linear regression models with a design space $\mathcal{X} \equiv [s] := \{1, \dots, s\}$, independent trials and a vector $\boldsymbol{\theta} \in \mathbb{R}^m$ of unknown parameters, provided that the trial in the ith design point results in an ℓ_i-dimensional response $\mathbf{y}_i$, satisfying $E(\mathbf{y}_i) = A_i^T \boldsymbol{\theta}$ and $\mathrm{Var}(\mathbf{y}_i) = \sigma^2 \mathbf{I}_{\ell_i}$, where $\mathbf{I}_k$ is the $k \times k$-identity matrix. For a design $\mathbf{w} \in \mathcal{W}$, the *moment* matrix $M(\mathbf{w}) := \sum_{i=1}^s w_i A_i A_i^T$ represents the total information gained from the design $\mathbf{w}$.

When the criterion Φ satisfies certain properties, problem (1.1) can be interpreted as selecting the weights w_i that yield the *most accurate estimation* of $\boldsymbol{\theta}$. In this paper, we mainly focus on the D-optimal problem, where the criterion Φ is set to

$$\Phi_D : M \to (\det M)^{1/m}. \tag{1.2}$$

In the case of Gaussian measurement error, this corresponds to the problem of minimizing the volume of the standard confidence ellipsoid for the best linear unbiased estimator (BLUE) $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$.

More generally, if the experimenter is interested in the estimation of the parameter subsystem $\boldsymbol{\vartheta} = K^T \boldsymbol{\theta}$, where K is an $m \times k$ matrix ($k \leq m$) of full column rank [$\mathrm{rank}(K) = k$], a relevant criterion is D_K-optimality, obtained when the D-criterion is applied to the information matrix $C_K(M)$ for the linear parametric subsystem given by the coefficient matrix K, defined by (Section 3.2 in [31])

$$C_K(M) = \min_{\substack{L \in \mathbb{R}^{k \times m} \\ LK = \mathbf{I}_k}}{}_{\preceq} \; LML^T.$$

Here the minimum is taken with respect to Löwner ordering, over all left inverses L of K. This information matrix is equal to $(K^T M^- K)^{-1}$ if the estimability condition holds (range $K \subseteq$ range M); otherwise $C_K(M)$ is a singular matrix, so

$$\Phi_{D|K} : M \to \begin{cases} (\det K^T M^- K)^{-1/k}, & \text{if } \operatorname{range} K \subseteq \operatorname{range} M; \\ 0, & \text{otherwise.} \end{cases} \tag{1.3}$$

In the previous formula M^- denotes a generalized inverse of M, that is, a matrix satisfying $MM^-M = M$. Although M^- is not unique in general, the definition of $\Phi_{D|K}$ is consistent. Indeed, the matrix $K^T M^- K$ does not depend on the choice of the generalized inverse M^- if the columns of K are included in the range of M; cf. Pukelsheim [31]. Note that if $k = 1$, that is, if the matrix $K = \mathbf{c}$ is a nonzero vector, then the criterion $\Phi_{D|K}$ is equivalent to the criterion of $\mathbf{c}$-optimality.

Other optimality criteria, such as A, A_K, G and I-optimality, are also discussed in the Appendix.

In the standard form of the problem, $\mathcal{W}$ is the probability simplex

$$\mathcal{W}_\Delta := \left\{ \mathbf{w} \in \mathbb{R}_+^s : \sum_{i=1}^s w_i = 1 \right\},$$

and the design $\mathbf{w}$ is a weight vector indicating the proportions of trials in the individual design points. This problem, called the *optimal approximate design problem* in the literature, is in fact a relaxation of a much more difficult and more fundamental discrete optimization problem: the *optimal exact design problem of size* N, where $\mathcal{W}$ takes the form

$$\mathcal{W}_N := \left\{ \frac{\mathbf{n}}{N} : \mathbf{n} \in \mathbb{N}_0^s, \sum_{i=1}^{s} n_i = N \right\}.$$

Here, the experiment consists of N trials, and if $\mathbf{w} \in \mathcal{W}_N$, then $n_i = N w_i$ indicates the number of trials in the design point i. (In the above definition, $\mathbb{N}_0$ denotes the set of all nonnegative integers, i.e., $0 \in \mathbb{N}_0$.) Note that the constraint $\mathbf{w} \in \mathcal{W}_\Delta$ is obtained from $\mathbf{w} \in \mathcal{W}_N$ by relaxing the integer constraints on $N w_i$.

Many different approaches have been proposed to solve problems of type (1.1). However, most methods are specialized and work only if the feasibility set $\mathcal{W}$ is the probability simplex $\mathcal{W}_\Delta$ or the standard discrete simplex $\mathcal{W}_N$. In the former case (approximate optimal design, $\mathcal{W} = \mathcal{W}_\Delta$), the traditional methods are the Fedorov–Wynn type vertex-direction algorithms [13, 45], and the multiplicative algorithms [39, 41, 47, 48], eventually combined with adaptive changes of the finite grid $\mathcal{X}$ [19, 30, 46]. In the latter case (exact optimal design, $\mathcal{W} = \mathcal{W}_N$), the classical methods are heuristics such as exchange algorithms [3, 13, 27], rounding methods [32] and metaheuristics such as simulated annealing [15] or genetic algorithms [20]. For some small to medium size models, branch-and-bound methods [43] have been used to compute provably optimal solutions.

In many practical situations, however, more complicated constraints are imposed on the design [9], and there is a need for more general algorithms. For example, assume that the experimental region can be partitioned as $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2$, and that 40% (resp., 60%) of the trials should be chosen in $\mathcal{X}_1$ (resp., $\mathcal{X}_2$); that is, the constraint $\mathbf{w} \in \mathcal{W}_\Delta$ is replaced by

$$\mathbf{w} \in \mathcal{W} := \left\{ \mathbf{w} \in \mathbb{R}_+^s : \sum_{i \in \mathcal{X}_1} w_i = 0.4, \sum_{i \in \mathcal{X}_2} w_i = 0.6 \right\}.$$

This is an example of a *stratified design* [16], which is a generalization of the well-known *marginally constrained design* [10]. Other examples of relevant design domains $\mathcal{W}$ defined by a set of linear inequalities are discussed in [42]. For example, it is possible to consider a case in which a total budget is allocated, and the design points are associated to possibly unequal costs $c_1, \dots, c_s$. It is also possible to consider decreasing costs when trials of specific design points are grouped, or to avoid designs that are concentrated on a small number of design points.

For some special linear constraints, the approximate D-optimal design problem can be solved by modifications of the vertex-direction and the multiplicative algorithms (see, e.g., [9, 16, 26]), but the convergence of these methods is usually slow. Recently, modern mathematical programming algorithms [12, 14, 18, 25, 28, 34, 36, 42] have been gaining in popularity. The idea is to reformulate the optimal design problem under a canonical form that specialized solvers can handle, such as *maxdet programs* (MAXDET), *semidefinite programs* (SDP) or *second-order cone programs* (SOCP).

Reformulating an optimal design problem as an SOCP or an SDP is useful in many regards. First, it allows one to use modern software to compute an optimal solution efficiently. Second, the available interior point methods are known to return an ε-optimal solution in polynomial time with respect to the size of the instance and $\log \frac{1}{\varepsilon}$ because a self-concordant barrier exists for these problems; cf. [6]. Third, mathematical programming methods are general in the sense that they are not restricted to the use of *special linear constraints*. Nevertheless, the inclusion of general linear constraints within mathematical programming characterizations is not completely straightforward. For instance, we show in Section 2 that the SOCP formulation of [34] for the standard approximate D-optimal design problem (over $\mathcal{W}_\Delta$) does not yield a valid SOCP formulation of the constrained D-optimal design problem when the constraint $\mathbf{w} \in \mathcal{W}_\Delta$ is replaced by $\mathbf{w} \in \mathcal{W}$.

The main result of this paper is proved in Section 4 and states that the determinant criterion is *SOC-representable*. More precisely, it is possible to express that $(t, \mathbf{w})$ belongs to the hypograph of $\mathbf{w} \to \Phi_D(M(\mathbf{w}))$, that is, $t^m \leq \det M(\mathbf{w})$, as a set of second-order cone inequalities. Consequently, we obtain an alternative SOCP formulation for D-optimality, which remains valid for any weight domain $\mathcal{W}$ that can be expressed by SOC inequalities; see Section 3.

In the Appendix, we prove that other widely used criteria, such as A, G or I-optimality are also SOC-representable. We have summarized the SOCP formulations of constrained D-, A- and G-optimality in Table 1.

Before this paper, the state of the art method for solving optimal design problems with arbitrary linear constraints was the MAXDET formulation of Vandenberghe, Boyd and Wu [42], which is in fact reformulated as an SDP by most interfaces, such as YALMIP [24] or PICOS [35], by using the construction described in [5]. Having an SOCP instead of an SDP formulation has two main advantages. The first is purely computational: it is well known that the computational effort per iteration required by the interior point methods to solve an SOCP is much less than that required to solve an SDP; cf. [1]. When the parameter $\boldsymbol{\theta}$ is of large dimension m, or when the number of candidate support points s is large, the SOCP can improve the computational time by one or two orders of magnitude (compared to MAXDET), as was already evidenced in [34] for D-optimality over the probability simplex $\mathcal{W}_\Delta$.

The second and probably more important benefit of SOCP formulations (compared to SDP) is that specialized solvers can handle SOCP problems with integer variables, while there is currently no reliable solver to handle SDPs with integer variables. Indeed, much progress has been made recently in the development of algorithms for second-order cone programming, when some of the variables are constrained in the integral domain (MISOCP: mixed integer second-order cone

TABLE 1

SOCP formulation of the D_K, A_K and G-optimal design problems over a compact weight region $\mathcal{W} \subseteq \mathbb{R}^s_+$. In the above, K represents a given $m \times k$ matrix of full column rank. The particular case $k = 1$ (where $\mathbf{c} = K$ is a column vector) gives SOCP formulations for the $\mathbf{c}$-optimal design problem, and the case $K = \mathbf{I}_m$ yields the standard D and A-optimality problems. The variables Z_i, Y_i ($i \in [s]$) are of size $\ell_i \times k$, the variables H_i^j ($i \in [s]$, $j \in [s]$) are of size $\ell_j \times \ell_i$, J is of size $k \times k$, the weight vector is $\mathbf{w} \in \mathcal{W}$ and the variables t_{ij} ($i \in [s]$, $j \in [k]$), u_i^j ($i \in [s]$, $j \in [s]$), μ_i ($i \in [s]$) and ρ are scalar

$$\max_{\mathbf{w}\in\mathcal{W}} \Phi_{D|K}\big(M(\mathbf{w})\big) = \max_{\mathbf{w}, Z_i, t_{ij}, J} \prod_{j=1}^{k} (J_{j,j})^{1/k}$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{i\in[s]} A_i Z_i = KJ, \\
& J \text{ is lower triangular}, \\
& \|Z_i \mathbf{e}_j\|^2 \le t_{ij} w_i \qquad \big(i \in [s],\, j \in [k]\big), \\
& \sum_{i=1}^{s} t_{ij} \le J_{j,j} \qquad \big(j \in [k]\big), \\
& t_{ij} \ge 0 \qquad \big(i \in [s],\, j \in [k]\big), \\
& \mathbf{w} \in \mathcal{W},
\end{aligned}$$

$$\max_{\mathbf{w}\in\mathcal{W}} \Phi_{A|K}\big(M(\mathbf{w})\big) = \max_{\mathbf{w}, Y_i, \mu_i} \sum_{i\in[s]} \mu_i$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{i\in[s]} A_i Y_i = \Big(\sum_{i\in[s]} \mu_i\Big) K, \\
& \|Y_i\|_F^2 \le \mu_i w_i \qquad \big(i \in [s]\big), \\
& \mu_i \ge 0 \qquad \big(i \in [s]\big), \\
& \mathbf{w} \in \mathcal{W},
\end{aligned}$$

$$\max_{\mathbf{w}\in\mathcal{W}} \Phi_{G}\big(M(\mathbf{w})\big) = \max_{\mathbf{w}, H_i^j, u_i^j, \rho} \rho$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{j\in[s]} A_j H_i^j = \Big(\sum_{j\in[s]} u_i^j\Big) A_i \qquad \big(i \in [s]\big), \\
& \|H_i^j\|_F^2 \le w_j u_i^j \qquad \big(i \in [s],\, j \in [s]\big), \\
& u_i^j \ge 0 \qquad \big(i \in [s],\, j \in [s]\big), \\
& \rho \le \sum_{j\in[s]} u_i^j \qquad \big(i \in [s]\big), \\
& \mathbf{w} \in \mathcal{W}.
\end{aligned}$$

programming). Thus the SOCP formulation of D-optimality presented in this article, unlike the existing SOCP and SDP formulations, allows us to use those specialized codes to solve exact design problems. Indeed, our formulation is valid for *any compact weight domain* $\mathcal{W}$, so in particular it is valid for the set $\mathcal{W}_N$ of exact designs of size N, and more generally for any polyhedron intersected with a lattice of integer points. Compared to the raw branch-and-bound method for computing exact designs proposed by Welch [43], the MISOCP approach is not only easier to implement, but also much more efficient. The reason is that specialized solvers such as CPLEX [21] or MOSEK [2] rely on branch-and-cut algorithms with sophisticated branching heuristics, and they use *cut inequalities* to separate noninteger solutions.

In Section 5, we demonstrate the general applicability of the proposed approach, incorporating illustrative examples taken from two application areas of the theory of optimal experimental designs. The following key aspects of the MISOCP approach will be emphasized:

(1) the ability to handle any system of linear constraints on the weights;
(2) the ability to compute exact-optimal designs with a proof of optimality;
(3) the ability to rapidly identify a near exact-optimal design for applications where the computing time must remain short, while giving a lower bound on its efficiency; moreover this bound is usually much better than the standard bound obtained from the approximate optimal design.

In particular, our algorithm can compute constrained exact optimal designs, a feature out of reach of the standard computing methods, although some authors have proposed heuristics to handle some special cases such as cost constraints [40, 44]. A notable exception is the recent DQ-optimality approach of Harman and Filová [17], which is a heuristic based on integer quadratic programming (IQP) that can handle the general case of linearly constrained exact designs. However, for some specific D-optimum design problems, the IQP approach leads to very inefficient designs; cf. Section 4 in [17].

In practice, the MISOCP solvers take an input tolerance parameter $\varepsilon > 0$, and the computation stops when a design $\mathbf{w}^*$ is found, with a guarantee that no design $\mathbf{w}$ with value $\Phi(M(\mathbf{w})) \geq (1+\varepsilon)\Phi(M(\mathbf{w}^*))$ exists. In some cases such as D-optimal block designs, there is a positive value of $\varepsilon > 0$ for which the returned design is verifiably optimal; see Section 5. Otherwise we can set $\varepsilon > 0$ to a small constant (i.e., a tolerance allowing a reasonable computation time), so the design found with the MISOCP approach will have an efficiency guarantee of $(1+\varepsilon)^{-1} \geq 1-\varepsilon$, which is usually a much better efficiency bound than the one based on the comparison with the approximate optimal design. In many situations, the solver is further able to terminate with an *optimality status*, which means that the branch and bound tree has been completely trimmed and constitutes a proof of optimality. Moreover, it often produces better designs than the standard heuristics (also in cases when perfect optimality is not guaranteed).

2. Former SOCP formulation of D-optimality. A *second-order cone program* (SOCP) is an optimization problem where a linear function $\mathbf{f}^T\mathbf{x}$ must be maximized, among the vectors $\mathbf{x}$ belonging to a set S-defined by second-order cone inequalities, that is,

$$S = \{\mathbf{x} \in \mathbb{R}^n : \forall i = 1, \dots, N_c, \|G_i\mathbf{x} + \mathbf{h}_i\| \leq \mathbf{c}_i^T\mathbf{x} + d_i\}$$

for some $G_i, \mathbf{h}_i, \mathbf{c}_i, d_i$ of appropriate dimensions. Optimization problems of this class can be solved efficiently to the desired precision using interior point techniques; see [6].

We first recall the result from [34] about D-optimality, rewritten with the notation of the present article. Note that $\|Z\|_F := \sqrt{\operatorname{trace} ZZ^T}$ denotes the Frobenius norm of the matrix Z, which also corresponds to the Euclidean norm of the vectorization of Z: $\|Z\|_F = \|\operatorname{vec}(Z)\|$. In the following formulation, the restriction to lower triangular matrices is just a compact notation for the set of linear constraints that appears in [34]:

PROPOSITION 2.1 (Former SOCP for D-optimality [34]). *Let* $(Z_1, \dots, Z_s, L, \mathbf{w})$ *be optimal for the following SOCP*:

$$\begin{aligned}
&\max_{\substack{Z_i \in \mathbb{R}^{\ell_i \times m} \\ L \in \mathbb{R}^{m\times m} \\ \mathbf{w} \in \mathbb{R}^s_+}} \left(\prod_{k=1}^m L_{k,k}\right)^{1/m} \\
&\quad \text{s.t.} \quad \sum_{i=1}^s A_i Z_i = L, \\
&\qquad\qquad L \text{ is lower triangular}, \\
&\qquad\qquad \|Z_i\|_F \leq \sqrt{m} w_i \qquad \forall i \in [s], \\
&\qquad\qquad \mathbf{w} \in \mathcal{W}_\Delta.
\end{aligned} \tag{2.1}$$

Then $\Phi_D(M(\mathbf{w})) = \det^{1/m} M(\mathbf{w}) = (\prod_k L_{k,k})^{2/m}$, *and* $\mathbf{w} \in \mathcal{W}_\Delta$ *is optimal for the standard approximate D-optimal design problem.*

If we want to solve a D-optimal design problem over another design region $\mathcal{W}$, it is very tempting to replace the last constraint in problem (2.1) by $\mathbf{w} \in \mathcal{W}$. However, this approach fails. Consider, for example, the following experimental design problem with three regression vectors in a two-dimensional space: $A_1 = [1, 0]^T$, $A_2 = [-\frac{1}{2}, \frac{\sqrt{3}}{2}]^T$, $A_3 = [-\frac{1}{2}, -\frac{\sqrt{3}}{2}]^T$. For reasons of symmetry, it is clear that the approximate D-optimal design (over $\mathcal{W}_\Delta$) is $w_1 = w_2 = w_3 = \frac{1}{3}$, and this is indeed the vector $\mathbf{w}$ returned by problem (2.1). Define now $\mathcal{W} := \{\mathbf{w} \in \mathbb{R}^3_+ : \sum_{i=1}^3 w_i = 1, w_1 \geq w_2 + 0.25\}$. The optimal design over $\mathcal{W}$ is $\mathbf{w}^* = [0.4583, 0.2083, 0.3333]$, but solving problem (2.1) with the additional constraint $w_1 \geq w_2 + 0.25$ yields the design $\mathbf{w} = [0.4482, 0.1982, 0.3536]$, which is suboptimal.

It can be proved that any optimal pair of variables $(\mathbf{w}^*, L^*)$ for problem (2.1) satisfies $M(\mathbf{w}^*) = (L^*)(L^*)^T$; that is, L^* is a Cholesky factor of the optimal information matrix. However, this relation is only true for optimality over the unit simplex $\mathcal{W}_\Delta$, which is a consequence of a generalization of Elfving's theorem; cf. [34]. In the present article, we give an alternative SOCP formulation of the D-optimal problem, which remains valid for any compact weight domain $\mathcal{W}$. The main idea of our new formulation is that the Cholesky factorization of a matrix HH^T can be computed by solving an SOCP that mimics the Gram–Schmidt orthogonalization process of the rows of H. Moreover, our new SOCP handles the more general case of D_K-optimality. To derive our result, we use the notion of *SOC-representability*, which we next present.

3. SOC-representability. In this section, we briefly review some basic notions about second-order cone representability. The following definition was introduced by Ben-Tal and Nemirovski [5]:

DEFINITION 3.1 (SOC-representability of a set). A convex set $S \subseteq \mathbb{R}^n$ is said to be *second-order cone representable*, abbreviated *SOC-representable*, if S is the projection of a set in a higher-dimensional space that can be described by a set of second-order cone inequalities. More precisely, S is SOC-representable if and only if there exist $G_i \in \mathbb{R}^{n_i \times (n+m)}, \mathbf{h}_i \in \mathbb{R}^{n_i}, \mathbf{c}_i \in \mathbb{R}^{n+m}, d_i \in \mathbb{R}$ $(i = 1, \dots, N_c)$, such that

$$\mathbf{x} \in S \quad \Longleftrightarrow \quad \exists \mathbf{y} \in \mathbb{R}^m : \forall i = 1, \dots, N_c, \qquad \left\| G_i \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + \mathbf{h}_i \right\| \leq \mathbf{c}_i^T \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + d_i.$$

An important example of an SOC-representable set is the following:

LEMMA 3.2 (Rotated second-order cone inequalities). *The set*

$$S = \{(\mathbf{x}, t, u) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} : \|\mathbf{x}\|^2 \leq tu, t \geq 0, u \geq 0\} \subseteq \mathbb{R}^{n+2}$$

is SOC-representable. In fact, it is easy to see that

$$S = \left\{(\mathbf{x}, t, u) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} : \left\| \begin{matrix} 2\mathbf{x} \\ t - u \end{matrix} \right\| \leq t + u \right\}.$$

The notion of SOC-representability is also defined for functions:

DEFINITION 3.3 (SOC-representability of a function). A convex (resp., concave) function $f : S \subseteq \mathbb{R}^n \mapsto \mathbb{R}$ is said to be SOC-representable if and only if the epigraph of f, $\{(t, \mathbf{x}) : f(\mathbf{x}) \leq t\}$ [resp., the hypograph $\{(t, \mathbf{x}) : t \leq f(\mathbf{x})\}$], is SOC-representable.

It follows immediately from these two definitions that the problem of maximizing a concave SOC-representable function (or minimizing a convex one) over an SOC-representable set can be cast as an SOCP. It is also easy to verify that sets defined by linear equalities (i.e., polyhedrons) are SOC-representable, that intersections of SOC-representable sets are SOC-representable and that the (pointwise) minimum of concave SOC-representable functions is still concave and SOC-representable.

We next give another example which is of major importance for this article: the geometric mean of n nonnegative variables is SOC-representable.

LEMMA 3.4 (SOC-representability of a geometric mean [5]). *Let $n \geq 1$ be an integer. The function f mapping $\mathbf{x} \in \mathbb{R}_+^n$ to $\prod_{i=1}^n x_i^{1/n}$ is SOC-representable.*

For construction of the SOC representation of f, see [23] or [1]. In what follows, we show the case $n = 5$. For all $t \in \mathbb{R}_+$, $\mathbf{x} \in \mathbb{R}_+^5$, we have

$$t^5 \leq x_1x_2x_3x_4x_5 \quad \Longleftrightarrow \quad t^8 \leq x_1x_2x_3x_4x_5t^3$$
$$\Longleftrightarrow \quad \exists \mathbf{u} \in \mathbb{R}_+^5 : \begin{cases} u_1^2 \leq x_1x_2, & u_4^2 \leq u_1u_2, \\ u_2^2 \leq x_3x_4, & u_5^2 \leq u_3t, \\ u_3^2 \leq x_5t, & t^2 \leq u_4u_5, \end{cases}$$

and each of these inequalities can be transformed to a standard second-order cone inequality by Lemma 3.2.

4. SOC-representability of the *D*-criterion. The key to SOC representation of the D-criterion is a Cholesky decomposition of the moment matrix, as given by the following lemma. Note that the lemma is general in the sense that it does not require the estimability conditions to be satisfied.

LEMMA 4.1. *Let H be an $m \times n$ matrix ($m \leq n$), and let K be an $m \times k$ matrix ($k \leq m$) of full column rank. If $k = m$, let $U = K$, and if $k < m$, let U be a nonsingular matrix of the form $[V, K]$, where $V \in \mathbb{R}^{m\times(m-k)}$. Then there exists a QR-decomposition of $H^T U^{-T} = \tilde{Q}\tilde{R}$ where $\tilde{Q}$ is an orthogonal $n \times n$ matrix and $\tilde{R}$ is an upper triangular $n \times m$ matrix, satisfying $\tilde{R}_{ii} \geq 0$ for all $i \in [m]$ and*

$$\tilde{R}_{ii} = 0 \qquad \textit{implies } \tilde{R}_{i1} = \cdots = \tilde{R}_{im} = 0 \qquad \textit{for all } i \in [m]. \tag{4.1}$$

Let L_*^T be the $k \times k$ upper triangular sub-block of $\tilde{R}$ with elements $(L_*^T)_{ij} = \tilde{R}_{m-k+i,m-k+j}$ for all $i, j \in [k]$. Then $C_K(HH^T) = L_*L_*^T$; that is, $L_*L_*^T$ is a Cholesky factorization of the information matrix for the linear parametric system given by the coefficient matrix K, corresponding to the moment matrix HH^T.

PROOF. It is simple to show that a QR decomposition satisfying (4.1) can be obtained from any QR-decomposition $H^T U^{-T} = \bar{Q}\bar{R}$, using an appropriate sequence of Givens rotations and row permutations applied on $\bar{R}$.

Consider the decomposition $H^T U^{-T} = \tilde{Q}\tilde{R}$ satisfying (4.1). Assume that $k < m < n$. Partition the orthogonal matrix $\tilde{Q}$ and the upper triangular matrix $\tilde{R}$ as follows:

$$\tilde{Q} = \begin{array}{cccc} \overset{m-k}{\longleftrightarrow} & \overset{k}{\longleftrightarrow} & \overset{n-m}{\longleftrightarrow} & \\ [Q_1 & Q_* & Q_2] & \updownarrow n, \end{array} \qquad \tilde{R} = \begin{array}{cc} \begin{array}{cc} \overset{m-k}{\longleftrightarrow} & \overset{k}{\longleftrightarrow} \end{array} & \\ \begin{bmatrix} L_1^T & B \\ 0 & L_*^T \\ 0 & 0 \end{bmatrix} & \begin{array}{l} \updownarrow m-k \\ \updownarrow k \\ \updownarrow n-m \end{array} \end{array} \tag{4.2}$$

where the block sizes are indicated on the border of the matrices. Let $U^{-1} = [Z^T, X^T]^T$, where X is a $k \times m$ matrix. Note that $[Z^T, X^T]^T K = U^{-1}K = [0, \mathbf{I}_k]^T$, which implies $XK = \mathbf{I}_k$; that is, X is a left inverse of K. Define $Y = \mathbf{I}_m - KX$. By a direct calculation, we obtain $XH = B^T Q_1^T + L_* Q_*^T$ and $YH = H - KXH = [V, K]\tilde{R}^T \tilde{Q}^T - K(B^T Q_1^T + L_* Q_*^T) = VL_1 Q_1^T$. Therefore, using the orthogonality of $\tilde{Q}$, that is, $Q_1^T Q_1 = \mathbf{I}_{m-k}$, $Q_*^T Q_* = \mathbf{I}_k$, $Q_1^T Q_* = 0$ and a representation of C_K given by [31], Section 3.2, we have

$$\begin{aligned} C_K(HH^T) &= XHH^T X^T - XHH^T Y^T (YHH^T Y^T)^- YHH^T X^T \\ &= B^T B + L_* L_*^T - B^T \underbrace{L_1^T V^T (VL_1 L_1^T V^T)^- VL_1}_{P} B, \end{aligned} \tag{4.3}$$

where P is the orthogonal projector on range$(L_1^T V^T)$. Note that (4.1) implies range$(B) \subseteq$ range(L_1^T), and rank$(V) = m - k$ gives range$(L_1^T) =$ range$(L_1^T V^T)$. That is, $PB = B$, and from (4.3) we obtain the required result $C_K(HH^T) = L_* L_*^T$.

If $k = m$ or $n = m$, the lemma can be proved in a completely analogous way, treating the matrices Q_1, L_1, B (if and only if $k = m$) and Q_2 (if and only if $m = n$) as *empty*. □

The next theorem shows that the blocks Q_* and L_* from decomposition (4.2) can be computed by solving an optimization problem over an SOC-representable set.

THEOREM 4.2. *Let H be an $m \times n$ matrix ($m \leq n$), let K be an $m \times k$ matrix ($k \leq m$) of full column rank and let L_* be optimal for the following problem*:

$$\begin{aligned} \max_{\substack{Q \in \mathbb{R}^{n \times k} \\ L \in \mathbb{R}^{k \times k}}} \quad & \det L \\ \text{s.t.} \quad & L \text{ is lower triangular}, \\ & HQ = KL, \\ & \|Q\mathbf{e}_j\| \leq 1 \qquad \big(j \in [k]\big). \end{aligned} \tag{4.4}$$

Then $\Phi_{D|K}(HH^T) = (\det(L_*))^{2/k}$.

PROOF. Consider the QR decomposition $H^T U^{-T} = \tilde{Q}\tilde{R}$ from the statement of Lemma 4.1, and the block partition (4.2). We will show that the blocks Q_* and L_* form an optimal solution to the problem from the theorem.

First, L_* is clearly lower triangular, and using direct block multiplication together with $Q_*^T Q_* = \mathbf{I}_k$, we can verify that $Q_*^T H^T = L_*^T K^T$, that is, $HQ_* = KL_*$. Second, Q_* has columns of unit length, which implies $\|Q_*\mathbf{e}_j\| = 1$ for all $j \in [k]$. Therefore, Q_*, L_* are feasible. From Lemma 4.1, we know that $C_K(HH^T) = L_* L_*^T$, that is, $(\det(L_*))^{2/k} = \Phi_{D|K}(HH^T)$. To complete the proof of the theorem, we only need to show that any feasible L satisfies $(\det(L))^{2/k} \leq \Phi_{D|K}(HH^T)$.

Let Q, L be a feasible pair of matrices. As in the proof of Lemma 4.1, let $U = [V, K]$ be an invertible matrix, and let $U^{-1} = [Z^T, X^T]^T$, where X is a $k \times m$ matrix. Obviously, $U^{-1}H = [C^T, D^T]^T$, where $C = ZH$ and $D = XH$, and $[C^T, D^T]^T Q = U^{-1}HQ = U^{-1}KL = [0, \mathbf{I}_k]^T L = [0, L^T]^T$, which implies $CQ = 0$ and $DQ = L$. Define the projector $P = \mathbf{I}_n - C^T(CC^T)^- C$, that is, $P^2 = P$, and then observe that $CQ = 0$ entails $PQ = Q$. From the previous equalities and the Cauchy–Schwarz inequality for determinants [e.g., [38], formula 12.5(c)], we have

$$\det(LL^T) = (\det(DQ))^2 = (\det(DPQ))^2 \leq \det(DPD^T)\det(Q^T Q). \tag{4.5}$$

The Hadamard determinant inequality (e.g., [38], formula 12.27) and the feasibility of Q give

$$\det(Q^T Q) \leq \prod_{i=1}^{k} (Q^T Q)_{ii} = \prod_{i=1}^{k} \|Q\mathbf{e}_i\|^2 \leq 1. \tag{4.6}$$

Combining (4.5) and (4.6), we obtain $\det(LL^T) \leq \det(DPD^T)$, and the proof will be complete, once we prove $DPD^T = C_K(HH^T)$.

Note that $\mathbf{I}_m = UU^{-1} = [V, K][Z^T, X^T]^T = VZ + KX$, that is, $Y := \mathbf{I}_m - KX = VZ$. Moreover, $\operatorname{rank}(V) = m - k$ implies $\operatorname{range}(H^T Y^T) = \operatorname{range}(H^T Z^T \times V^T) = \operatorname{range}(H^T Z^T)$; that is, the orthogonal projectors $H^T Y^T (YHH^T Y^T)^- HY$ and $H^T Z^T (ZHH^T Z^T)^- HZ$ coincide. Consequently, using [31], Section 3.2, we have

$$\begin{aligned} C_K(HH^T) &= XHH^T X^T - XHH^T Y^T (YHH^T Y^T)^- YHH^T X^T \\ &= XHH^T X^T - XHH^T Z^T (ZHH^T Z^T)^- ZHH^T Z^T \\ &= DD^T - DC^T (CC^T)^- CD^T = DPD^T. \end{aligned}$$ □

We next apply Theorem 4.2 to the matrix $H = [\sqrt{w_1}A_1, \ldots, \sqrt{w_s}A_s]$. This will allow us to express $\Phi_{D|K}(M(\mathbf{w}))$ as the optimal value of an SOCP. Moreover, we make a change of variables which transforms the optimization problem into an SOCP where $\mathbf{w}$ may play the role of a variable.

THEOREM 4.3. *Let K be an $m \times k$ matrix ($k \leq m$) of full column rank. For all nonnegative weight vectors $\mathbf{w} \in \mathbb{R}_+^s$, denote by $OPT(\mathbf{w})$ the optimal value of the following optimization problem, where the optimization variables are $t_{ij} \in \mathbb{R}_+$ ($\forall i \in [s], \forall j \in [k]$), $Z_i \in \mathbb{R}^{\ell_i \times k}$ ($\forall i \in [s]$) and $J \in \mathbb{R}^{k \times k}$:*

$$\max_{Z_i, t_{ij}, J} \left(\prod_{j=1}^{k} J_{j,j} \right)^{1/k} \tag{4.7a}$$

$$s.t. \quad \sum_{i=1}^{s} A_i Z_i = KJ, \tag{4.7b}$$

$$J \text{ is lower triangular}, \tag{4.7c}$$

$$\|Z_i \mathbf{e}_j\|^2 \leq t_{ij} w_i \qquad \big(i \in [s], j \in [k]\big), \tag{4.7d}$$

$$\sum_{i=1}^{s} t_{ij} \leq J_{j,j} \qquad \big(j \in [k]\big). \tag{4.7e}$$

Then we have

$$OPT(\mathbf{w}) = \Phi_{D|K}\big(M(\mathbf{w})\big).$$

PROOF. Let $\mathbf{w} \in \mathbb{R}_+^s$, and define $H := [\sqrt{w_1} A_1, \ldots, \sqrt{w_s} A_s]$. We are going to show that every feasible solution to problem (4.7a)–(4.7e) yields a feasible solution for problem (4.4) in which $J_{j,j} = L_{j,j}^2$ for all $j \in [k]$, and vice versa. Hence the optimal value of problem (4.7a)–(4.7e) is

$$OPT(\mathbf{w}) = (\det J)^{1/k} = (\det L)^{2/k} = \Phi_{D|K}\big(HH^T\big) = \Phi_{D|K}\big(M(\mathbf{w})\big),$$

from which the conclusion follows.

Consider a feasible solution (Z_i, t_{ij}, J) to problem (4.7a)–(4.7e). We denote by $\mathbf{z}_{ij}$ the jth column of Z_i: $\mathbf{z}_{ij} := Z_i \mathbf{e}_j$. We now make the following change of variables: denote by Q_i the matrix whose jth column is $\mathbf{q}_{ij}$, where

$$\mathbf{q}_{ij} = \begin{cases} \dfrac{\mathbf{z}_{ij}}{\sqrt{w_i}\sqrt{J_{j,j}}}, & \text{if } w_i > 0 \text{ and } J_{j,j} > 0; \\ \mathbf{0}, & \text{otherwise,} \end{cases}$$

and define Q as the vertical concatenation of the Q_i: $Q = [Q_1^T, \ldots, Q_s^T]^T$. Let $j \in [k]$. If $J_{j,j} = 0$, then $\mathbf{q}_{ij} = \mathbf{0}$ for all i, so $\|Q\mathbf{e}_j\|^2 = \sum_i \|\mathbf{q}_{ij}\|^2 = 0 \leq 1$. Otherwise ($J_{j,j} > 0$), constraint (4.7d) together with the nonnegativity of t_{ij} implies $\|\mathbf{q}_{ij}\|^2 \leq \frac{t_{ij}}{J_{j,j}}$, and by constraint (4.7e), we must have

$$\|Q\mathbf{e}_j\|^2 = \sum_i \|\mathbf{q}_{ij}\|^2 \leq \sum_i \frac{t_{ij}}{J_{j,j}} \leq 1.$$

Observe that constraints (4.7d) and (4.7e) also imply that $\mathbf{z}_{ij} = \mathbf{0}$ whenever $w_i = 0$ or $J_{j,j} = 0$, so that for all $i \in [s]$, $j \in [k]$, we can write $\mathbf{z}_{ij} = \sqrt{w_i}\sqrt{J_{j,j}}\mathbf{q}_{ij}$. Now, we define the matrix L column-wise as follows:

$$\forall j \in [k], \qquad L\mathbf{e}_j := \begin{cases} \dfrac{J\mathbf{e}_j}{\sqrt{J_{j,j}}}, & \text{if } J_{j,j} > 0; \\ \mathbf{0}, & \text{otherwise.} \end{cases}$$

Note that L is lower triangular [because so is J; see (4.7c)]. We can now prove that $HQ = KL$, which we do column-wise. If $J_{j,j} = 0$, then we know that $Q\mathbf{e}_j = \mathbf{0}$, so the jth columns of HQ and KL are zero. If $J_{j,j} > 0$, then using (4.7b) we have

$$KL\mathbf{e}_j = \frac{KJ\mathbf{e}_j}{\sqrt{J_{j,j}}} = \frac{\sum_i A_i \mathbf{z}_{ij}}{\sqrt{J_{j,j}}} = \sum_i \sqrt{w_i} A_i \mathbf{q}_{ij} = HQ\mathbf{e}_j.$$

Hence the proposed change of variables transforms a feasible solution (Z, t_{ij}, J) to problem (4.7a)–(4.7e) into a feasible pair (Q, L) for problem (4.4), with the property $J_{j,j} = L_{j,j}^2$ for all $j \in [k]$.

Conversely, let (Q, L) be feasible for problem (4.4), where H has been set to $[\sqrt{w_1}A_1, \dots, \sqrt{w_s}A_s]$. For $i \in [s]$, define Z_i as the matrix of size $\ell_i \times k$ whose jth column is $\mathbf{z}_{ij} = \sqrt{w_i} L_{j,j} \mathbf{q}_{ij}$, and J as the lower triangular matrix whose jth column is $J\mathbf{e}_j = L_{j,j} L\mathbf{e}_j$. We have $\sum_i A_i Z_i = KJ$, which can be verified column-wise as follows:

$$KJ\mathbf{e}_j = L_{j,j} KL\mathbf{e}_j = L_{j,j} HQ\mathbf{e}_j = L_{j,j} \sum_i \sqrt{w_i} A_i \mathbf{q}_{ij} = \sum_i A_i \mathbf{z}_{ij} = \sum_i A_i Z_i \mathbf{e}_j.$$

Define further $t_{ij} = L_{j,j}^2 \|\mathbf{q}_{ij}\|^2$, so that constraints (4.7d) and (4.7e) hold. This shows that (Z_i, t_{ij}, J) is feasible, with $J_{j,j} = L_{j,j}^2$ for all $j \in [k]$, and the proof is complete. □

COROLLARY 4.4 (SOC-representability of $\Phi_{D|K}$). *For any $m \times k$ matrix K of rank k, the function $\mathbf{w} \to \Phi_{D|K}(M(\mathbf{w}))$ is SOC-representable.*

PROOF. Problem (4.7a)–(4.7e) can be reformulated as an SOCP, because by Lemmas 3.4 and 3.2 the geometric mean in (4.7a) and inequalities of the form $\|Z_i\mathbf{e}_j\|^2 \le t_{ij} w_i$ are SOC-representable. Hence the optimal value of (4.7a)–(4.7e), $\mathbf{w} \to OPT(\mathbf{w})$, is SOC-representable, and we know from Theorem 4.3 that $OPT(\mathbf{w}) = \Phi_{D|K}(M(\mathbf{w}))$. □

COROLLARY 4.5 [(MI)SOCP formulation of the D-optimal design problem]. *If the set $\mathcal{W}$ is SOC-representable (in particular, if $\mathcal{W}$ is defined by a set of linear inequalities), then the constrained D_K-optimal design problem (1.1) can be cast as an SOCP. If $\mathcal{W}$ is the intersection of an SOC-representable set with the integer lattice $\mathbb{Z}^s$, then the exact D_K-optimal design problem over $\mathcal{W}$ can be cast as an MISOCP.*

For $K = \mathbf{I}_m$, Corollaries 4.4 and 4.5 cover the case of the standard D-optimality. The (MI)SOCP formulation of problem (1.1) for D_K-optimality ($\Phi = \Phi_{D|K}$) is summarized in Table 1, together with formulations for the other criteria presented in the Appendix. Finally, we note that the SOCP formulation of the optimal design problem with constraints on the weights has consequences in terms of complexity, which we next present.

Complexity of computing constrained approximate D_K-optimal designs. Recall that s denotes the number of candidate support points, and $k \leq m$ denotes the number of features that we wish to estimate. (The full rank coefficient matrix K is in $\mathbb{R}^{m\times k}$.) Assume for simplicity that $\ell_i = \ell$ for all $i \in [s]$, that the set of design weights $\mathcal{W}$ is defined by a set of n inequalities and that k is a power of 2, so that the geometric mean can be represented by k inequalities and k auxiliary variables; cf. Lemma 3.4 or [36] for more details. Then the SOCP formulation for D_K-optimality of Table 1 contains:

- $s + s\ell k + sk + \frac{1}{2}k(k+1) + k$ variables,
- $mk + k + n$ linear (in)equalities,
- k SOC inequalities of size 2 and ks SOC inequalities of size $\ell + 1$.

The number of iterations required by the interior point methods (IPM) to compute an ε-approximate solution depends only on the number q of second-order cones. Indeed it is shown in [5] that the IPM finds an ε-approximate solution after at most $\sqrt{q}\,O(\log\frac{1}{\varepsilon})$ iterations, which is $\sqrt{k(s+1)}\,O(\log\frac{1}{\varepsilon})$ iterations in our setting. However, it is well known that this bound is overconservative, and in practice the IPM always returns an excellent solution after 10 to 40 iterations, almost independently of the problem size. In other words, the critical point is the algorithmic complexity of one iteration. Again, a result of [5] (Section 4.6.2) allows us to bound the number of algorithmic operations for one iteration in $O(ks\ell((ks\ell)^2 + (mk+n)^2))$, which is $O((ks\ell)^3)$ if m and n are not too large. But it is well known that this bound is very conservative, too. In fact, the bottleneck of one iteration is the resolution of a linear system of the form $B\boldsymbol{\delta} = \boldsymbol{\beta}$, where B is a $O(ks\ell) \times O(ks\ell)$ symmetric positive semidefinite matrix. In practice, for SOCPs the matrix B has a "*diagonal + sparse low rank*" structure, which allows for an efficient computation of the Newton direction $\boldsymbol{\delta}$ [1].

5. Examples. In this section, we will present numerical results for several examples taken from various application areas of the theory of optimal designs. With these examples, we aim to demonstrate the general applicability of the (MI)SOCP technique for the computation of exact or approximate D-optimal designs.

Our computations were conducted on a PC with a 4-core processor at 3 GHz. We used MOSEK [2] to solve the approximate optimal design problems and CPLEX [21] for the exact optimal design problems (with integer constraints). The solvers were interfaced through the Python package PICOS [35], which allows users to pass (MI)SOCP models to different solvers in a simple fashion. We refer the reader to the example section of the PICOS documentation for a practical implementation of the (MI)SOCP approach for optimal design problems.

It is common to compare several designs against each other by using the metric of D-efficiency, which is defined as

$$\mathrm{eff}_D(\mathbf{w}) = \frac{\Phi_D(M(\mathbf{w}))}{\Phi_D(M(\mathbf{w}^*))} = \left(\frac{\det M(\mathbf{w})}{\det M(\mathbf{w}^*)}\right)^{1/m},$$

where $\mathbf{w}^*$ is a reference design, such that $M(\mathbf{w}^*)$ is nonsingular. Unless stated otherwise, we always give D-efficiencies relative to the optimal design; that is, $\mathbf{w}^*$ is a solution to problem (1.1).

Block designs with blocks of size two. An important category of models studied in the experimental design literature is the class of *block designs*. Here the effect of t treatments should be compared, but their effects can only be measured inside a number b of *blocks*, each inducing a block effect on the measurements. The optimal design problem entails choosing which treatments should be tested together in each block. We refer the reader to Bailey and Cameron [4] for a comprehensive review on the combinatorics of block designs.

In the case where the blocks are of size two, that is, the treatments can be tested pairwise against each other, a design can be represented by a vector $\mathbf{w} = [w_{1,2}, w_{1,3}, \dots, w_{1,t}, \dots, w_{t-1,t}]$ of size $s = \binom{t}{2}$. For $i < j$, $w_{i,j}$ indicates the number of blocks where treatments i and j are tested simultaneously. The observation matrix associated with the block (i, j) can be chosen as the column vector of dimension $m = (t - 1)$,

$$A_{i,j} = P(\mathbf{e}_i - \mathbf{e}_j), \tag{5.1}$$

where $\mathbf{e}_i$ denotes the ith unit vector in the canonical basis of $\mathbb{R}^t$ and P is the matrix that transforms a t-dimensional vector $\mathbf{v}$ to the vector obtained by keeping the first $(t - 1)$ coordinates of $\mathbf{v}$.

The problem of D-optimality has a nice graph theoretic interpretation: let $\mathbf{w} \in \mathbb{N}_0^s$ be a feasible block design, and denote by G the graph with t vertices and an edge of multiplicity $w_{i,j}$ for every pair of nodes (i, j). (If $w_{i,j} = 0$, then there is no edge from i to j.) This graph is called the *concurrence graph* of the design. We have $M(\mathbf{w}) = PL(\mathbf{w})P^T$, where $L(\mathbf{w}) := \sum_{i,j} w_{i,j}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^T \in \mathbb{R}^{t\times t}$ is the Laplacian of G. In other words, $M(\mathbf{w})$ is the submatrix of the Laplacian of G obtained by removing its last row and last column. So by Kirchhoff's theorem the determinant of $M(\mathbf{w})$ is the number of spanning trees of G. In other words, the exact D-optimal designs of size N correspond to the graphs with t nodes and N edges that have a maximum number of spanning trees.

REMARK 5.1. There is an alternative parametrization of block designs with blocks of size two; see [17]. Define the observation matrices by

$$A'_{i,j} = U^T(\mathbf{e}_i - \mathbf{e}_j), \tag{5.2}$$

where the columns of $U \in \mathbb{R}^{t\times(t-1)}$ form an orthonormal basis of $\operatorname{Ker} \mathbf{1}$ ($\mathbf{1}$ is the vector with all components equal to 1); that is, the $t \times t$-matrix $[U, \frac{1}{\sqrt{t}}\mathbf{1}]$ is orthogonal. It can be seen that the $t-1$ eigenvalues of $M'(\mathbf{w}) = \sum_{i,j} w_{i,j} A'_{i,j} A'^T_{i,j} = U^T L(\mathbf{w}) U$ coincide with the $t-1$ largest eigenvalues of $L(\mathbf{w})$, and the smallest eigenvalue of $L(\mathbf{w})$ is 0. So the set of D-optimal designs for observation models (5.1) and (5.2) coincide. In our experiments, we have used the former model (5.1) because it involves sparse information matrices and yields more efficient computations. However, note that for some other criteria depending on the eigenvalues of the information matrix, the model given by (5.2) should be used.

To illustrate the new capability of the MISOCP approach, we computed designs of $N = 15$ blocks on $t = 10$ treatments by imposing different types of constraints on the replication numbers (i.e., the numbers of times that each treatment is tested). Such constraints can be easily expressed by linear (in)equalities. For example, a design $\mathbf{w}$ has treatment j replicated r_j times if and only if

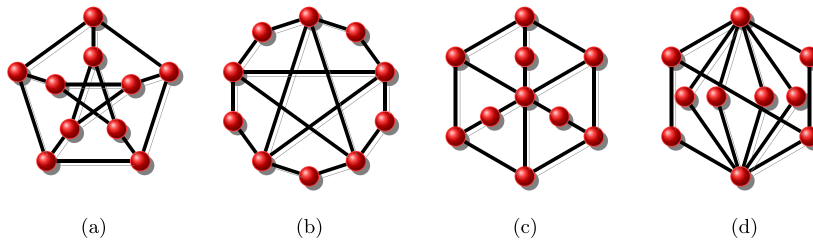
$$\sum_{i=1}^{j-1} w_{i,j} + \sum_{i=j+1}^{t} w_{j,i} = r_j.$$

The concurrence graphs of these constrained optimal designs are displayed in Figure 1. Note that these constrained exact optimal designs cannot be computed by any of the standard methods.

Mixed integer optimization solvers rely on sophisticated branch-and-cut algorithms. After each iteration, the value $L = \Phi_D(M(\hat{\mathbf{w}}))$ of the best solution $\hat{\mathbf{w}}$ found so far is compared to an upper bound U provided by a series of continuous relaxation of the problem, and the *gap* defined by $\delta = \frac{U-L}{L}$ is displayed. Note that δ can directly be interpreted as a guarantee on the D-efficiency of $\hat{\mathbf{w}}$, namely $\mathrm{eff}_D(\hat{\mathbf{w}}) \geq (1+\delta)^{-1}$. The following remark shows that for block designs, the current best solution is actually proved to be exact D-optimal as soon as the gap δ reaches a small tolerance parameter $\varepsilon > 0$.

REMARK 5.2. Let $T_{\mathbf{w}}$ denote the number of spanning trees of the concurrence graph G corresponding to an exact design $\mathbf{w}$, and T^* denote the maximal number of spanning trees for a particular block design problem. By using the fact that $T_{\mathbf{w}} = \det M(\mathbf{w})$ is an integer, it can be seen that a tolerance parameter of

$$\varepsilon = \left(1 + \frac{1}{T^*}\right)^{1/m} - 1 \simeq \frac{1}{mT^*}$$



Design	(a)	(b)	(c)	(d)
CPU (s)	9.07	4.9	13.8	5.7
Lower bound on eff_D (initial)	90.15%	92.56%	91.36%	91.27%
Lower bound on eff_D (10 min)	100.0%	96.56%	98.04%	100.0%

FIG. 1. *Concurrence graphs of the D-optimal designs of $N = 15$ blocks on $t = 10$ treatments, among the class of* 2*-block designs that* (a) *are equireplicate*; (b) *have half of the treatments replicated* 2 *times, and the other half replicated* 4 *times*; (c) *have one treatment replicated at least* 6 *times*; (d) *have two treatments replicated at least* 6 *times. For each case, the table gives the time required by the MISOCP solver to find the optimal design*; *the* (*initial*) *lower bound on the D-efficiency of the optimal design, compared to the constrained approximate design*; *the lower bound on the D-efficiency of the optimal design that is guaranteed after* 10 *min of computing time.*

ensures that the design $\mathbf{w}^*$ returned by the MISOCP approach is (perfectly) optimal. We have used this value of ε in our numerical experiments. When the value of T^* is unknown, note that an upper bound can be used (e.g., the bound $T^* \leq \frac{1}{t}(\frac{2N}{t-1})^{t-1}$ given by the optimal design $\mathbf{w} = [\frac{N}{s}, \dots, \frac{N}{s}]^T$ for the relaxed problem without integer constraints).

To achieve a faster convergence, a few variables can be set equal to 0 or 1 in order to break the symmetry of the problem. For example, if we search for a D-optimal design in a class of exact designs with at least one treatment replicated exactly 4 times, we can assume without loss of generality that treatment 1 has replication number 4, so $w_{1,2} = w_{1,3} = w_{1,4} = w_{1,5} = 1$ and $w_{1,i} = 0$ for all $i \in \{6, \dots, t\}$.

The table in Figure 1 gives information on the computing time required by CPLEX. In all four situations, the optimal design was found in the first seconds of computation. However, note that the time required to obtain a certificate of optimality can be much longer [a few minutes for cases (a) and (d), and as much as 3 hours for case (b)]. However, the bound on the D-efficiency provided by the MISOCP solver after a few minutes is already much better than the standard bound of D-efficiency relative to the (constrained) approximate optimal design.

This example also demonstrates that sometimes we can use independent theoretical results to add some linear constraints to the original optimum design problem that can greatly improve the computational efficiency. Indeed, it has been conjectured that every optimal block design with blocks of size two is (almost) equireplicate for $t-1 \leq N \leq \binom{t}{2}$. The conjecture is known to hold for $t \leq 11$ [8] and for all pairs (t, N) such that $N \geq \binom{t}{2} - t + 2$ [29]. The MISOCP solver required 333.7 s to obtain a certificate of optimality of the design plotted in Figure 1(a) in the class of equireplicate designs. In contrast, several hours of computation are required if we omit the constraints on the replication numbers in the MISOCP formulation.

More computational results for optimal block designs can be found in an earlier version of this manuscript that is available on the web [37]. In particular, we show that even for the case of standard (unconstrained) exact design problems ($\mathcal{W} = \mathcal{W}_N$), the MISOCP approach sometimes outperforms state-of-the-art algorithms such as the KL-exchange procedure [3]. The manuscript [37] also presents numerical results on other criteria, such as A-optimality and G-optimality.

Locally D-optimal design in a study of chemical kinetics. Another classical field of application of the theory of optimal experimental designs is the study of chemical kinetics. Here, the goal is to select the points in time at which a chemical reaction should be observed, to estimate the kinetic parameters $\boldsymbol{\theta} \in \mathbb{R}^m$ of the reaction (rates, orders, etc.). The measurements at time t are of the form $\mathbf{y}_t = \boldsymbol{\eta}_t(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}_t$, where $\boldsymbol{\eta}_t(\boldsymbol{\theta}) = [\eta_t^1, \dots, \eta_t^k]^T$ is the vector of the concentrations of k reactants at time t and $\boldsymbol{\varepsilon}_t$ is a random error. The kinetic models are usually given as a set of differential equations, which can be solved numerically to find the concentrations $\boldsymbol{\eta}_t(\boldsymbol{\theta})$ over time. Unlike the linear model described in the introduction of this paper, in chemical kinetics the expected measurements $\mathbb{E}[\mathbf{y}_t] = \boldsymbol{\eta}_t(\boldsymbol{\theta})$ at time t depend nonlinearly on the vector $\boldsymbol{\theta}$ of unknown parameters of the reaction. So a classical approach is to search for a *locally optimal design* using a prior estimate $\boldsymbol{\theta}_0$ of the parameter, that is, a design which would be optimal if the true value of the parameters was $\boldsymbol{\theta}_0$. To do this, the observation equations are linearized around $\boldsymbol{\theta}_0$, so in practice we replace the observation matrix A_t of each individual trial at time t by its *sensitivity* at $\boldsymbol{\theta}_0$, which is defined as

$$F_t := \frac{\partial \boldsymbol{\eta}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} = \left.\begin{pmatrix} \frac{\partial \eta_t^1}{\partial \theta_1} & \cdots & \frac{\partial \eta_t^k}{\partial \theta_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial \eta_t^1}{\partial \theta_m} & \cdots & \frac{\partial \eta_t^k}{\partial \theta_m} \end{pmatrix}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \in \mathbb{R}^{m \times k}.$$

A classical example is presented in [3], the study of two consecutive reactions

$$A \xrightarrow{\theta_1} B \xrightarrow{\theta_2} C.$$

The chemical reactions are assumed to be of order θ_3 and θ_4, respectively, so the concentrations of the reactants are determined by the differential equations

$$\begin{aligned}\frac{d[A]}{dt} &= -\theta_1[A]^{\theta_3},\\ \frac{d[B]}{dt} &= \theta_1[A]^{\theta_3} - \theta_2[B]^{\theta_4},\\ \frac{d[C]}{dt} &= \theta_2[B]^{\theta_4},\end{aligned} \tag{5.3}$$

together with the initial condition $([A],[B],[C])|_{t=0} = (1,0,0)$. These equations can be differentiated with respect to $\theta_1,\dots,\theta_4$, which yields another set of differential equations that determines the elements $\frac{\partial \eta_t^j}{\partial \theta_i}$ of the sensitivity matrices.

We now assume that measurements can be performed at each $t \in \mathcal{X} = \{0.2, 0.4, \dots, 19.8, 20\}$, where the time is expressed in seconds, and that the observed quantities are the concentrations of the reactants A and C, that is, $k = 2$ and $\boldsymbol{\eta}_t^T = ([A](t),[C](t))$. We have solved numerically the differential equations governing the entries of $(F_t)_{t\in\mathcal{X}}$ for $\boldsymbol{\theta}_0 := [1, 0.5, 1, 2]^T$. These sensitivities are plotted in Figure 2.

We used the MISOCP method to compute the exact D-optimal design of size $N = 5$ for this problem (for the prior estimate $\boldsymbol{\theta}_0$). The optimum consists in taking 1 measurement at $t = 0.8$, 3 measurements at $t = 2.8$ and 1 measurement at $t = 16.6$. In comparison, the exchange algorithm (using the same settings as described for the block designs, with $N^R = 100$) found a design with 1 measurement for each $t \in \{0.8, 3.4, 17.4\}$ and 2 measurements at $t = 2.6$. This design is of course very close to the optimum (its D-efficiency is 98.42%), but we note that the *true* optimum could not be identified by the exchange algorithm, even with a very large number of tries. We ran the exchange procedure $N^R = 5000$ times which took 100 s and returned a design of D-efficiency 99.42%, while the MISOCP found a provable optimal design after 25 s (CPLEX returned the status `MIP_OPTIMAL`).

We plotted these designs in Figure 3 together with the concentrations of the reactants over time when we assume $\boldsymbol{\theta} = \boldsymbol{\theta}_0$. In the figure, we have also plotted other designs which can be of interest to practitioners. For example, it might be natural to search designs where at most 1 measurement is taken at a given point in time. The exchange algorithm can also be adapted to the case of binary designs (by rejecting candidate points that already support the design during the exchange procedure). It returned a design of D-efficiency 98.97%. The last case we have considered is the following: assume that the experimenter must wait at least one second after a measurement before performing another measurement. This constraint can be modeled as a set of inequalities that can be added into the MISOCP

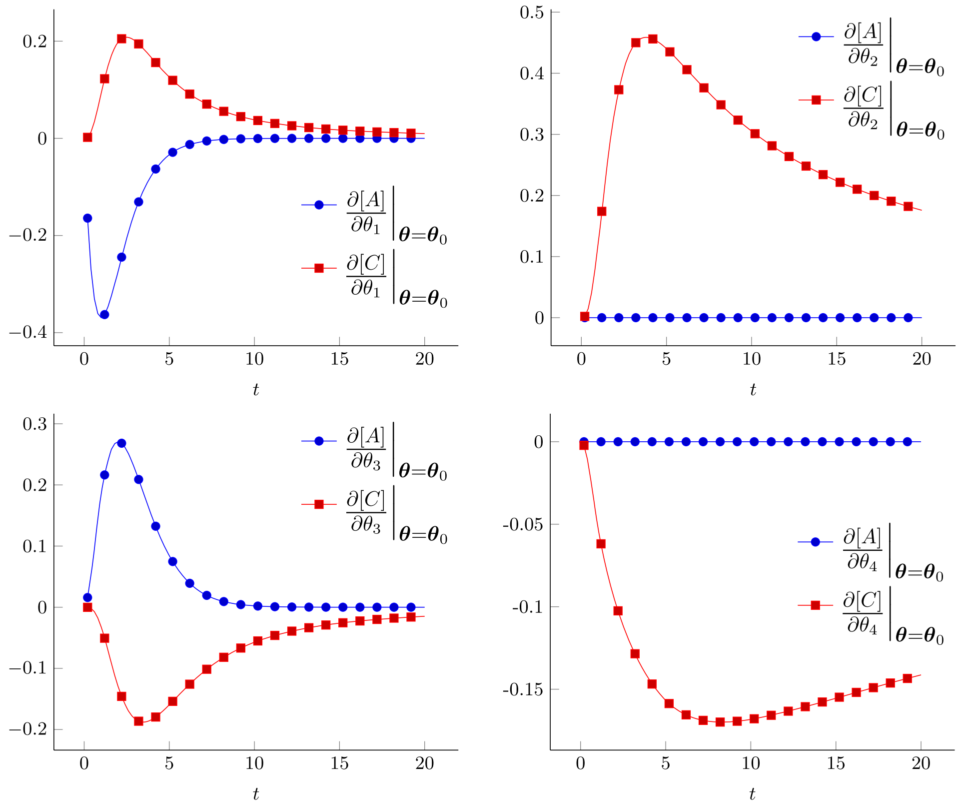


FIG. 2. *Measurement sensitivities (entries of F_t) plotted against time for* $\boldsymbol{\theta}_0 = [1, 0.5, 1, 2]^T$.

formulation,

$$\begin{aligned}
&\{w_{0.2} + w_{0.4} + w_{0.6} + w_{0.8} + w_{1.0} \leq 1,\\
&\ w_{0.4} + w_{0.6} + w_{0.8} + w_{1.0} + w_{1.2} \leq 1, \dots,\\
&\ w_{19.2} + w_{19.4} + w_{19.6} + w_{19.8} + w_{20.0} \leq 1\}.
\end{aligned}$$

This model was solved in 42 s with CPLEX, and the corresponding optimal design is depicted on the last row of Figure 3. We do not know of any other algorithm that can handle this type of exact design problem with several linear constraints.

APPENDIX: OTHER OPTIMALITY CRITERIA

A.1. A_K-optimality. Another widely used criterion in optimal design is A-optimality, which is defined by

$$\Phi_A : M \to \begin{cases} (\operatorname{trace} M^{-1})^{-1}, & \text{if } M \text{ is nonsingular;}\\ 0, & \text{otherwise.} \end{cases}$$

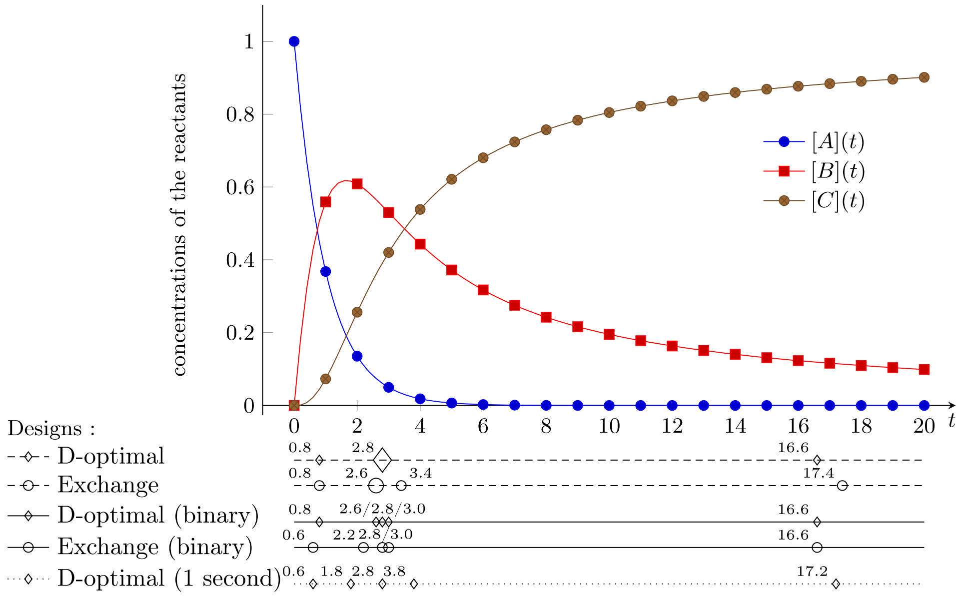


FIG. 3. *Concentration of the reactants against time (determined by solving equation (5.3), assuming* $\boldsymbol{\theta} = \boldsymbol{\theta}_0 = [1, 0.5, 1, 2]^T$). *Several designs are represented below the graph. The marks indicate the time at which the measurements should be performed, and the size of the marks indicate the number of measurements at a given point in time. Binary means that the design space is restricted to designs having at most one measurement for each* $t \in \mathcal{X}$, *and* 1 *second means that at least* 1 *second must separate* 2 *measurements.*

More generally, it is possible to use the criterion of A_K-optimality if the experimenter is interested in the estimation of the parameter subsystem $\boldsymbol{\vartheta} = K^T\boldsymbol{\theta}$,

$$\Phi_{A|K} : M \to \begin{cases} (\operatorname{trace} K^T M^- K)^{-1}, & \text{if } \operatorname{range} K \subseteq \operatorname{range} M; \\ 0, & \text{otherwise.} \end{cases}$$

Here M^- denotes a generalized inverse of M; see the discussion following equation (1.3) in the Introduction. Note that $\Phi_{A|K}$ coincides with Φ_A if $K = \mathbf{I}_m$, and $\Phi_{A|K}$ reduces to the criterion of $\mathbf{c}$-optimality when $K = \mathbf{c} \neq \mathbf{0}$ is a column vector.

The following lemma was already used in [34], under a slightly different form. In fact, this lemma is a consequence of the Gauss–Markov theorem, which states that the variance–covariance matrix of the best linear unbiased estimator of $K^T\boldsymbol{\theta}$ is proportional to $K^T M(\mathbf{w})^- K$ (e.g., Pukelsheim [31]).

LEMMA A.1. *Let* K *be an* $(m \times k)$*-matrix, and let* $\mathbf{w} \in \mathbb{R}_+^s$ *be a vector of design weights, such that the estimability condition* $\operatorname{range} K \subseteq \operatorname{range} M(\mathbf{w})$ *is sat-*

isfied. Define $I := \{i \in [s] : w_i > 0\}$. *Then*

$$\text{(A.1)}\qquad \begin{aligned} \operatorname{trace} K^T M(\mathbf{w})^- K = \min_{(Z_i)_{i\in I}} & \sum_{i\in I} \frac{\|Z_i\|_F^2}{w_i} \\ \text{s.t.} \quad & \sum_{i\in I} A_i Z_i = K, \end{aligned}$$

where the variables Z_i $(i \in I)$ *are of size* $\ell_i \times k$.

After some changes of variable, we obtain an SOC representation of $\Phi_{A|K}$:

PROPOSITION A.2. *Let* K *be an* $(m \times k)$-*matrix, and let* $\mathbf{w} \in \mathbb{R}_+^s$ *be a vector of design weights. Then*

$$\text{(A.2)}\qquad \begin{aligned} \Phi_{A|K}\big(M(\mathbf{w})\big) = \max_{\boldsymbol{\mu}\in\mathbb{R}_+^s, Y_i\in\mathbb{R}^{\ell_i\times k}} & \sum_{i\in[s]} \mu_i \\ \text{s.t.} \quad & \sum_{i\in[s]} A_i Y_i = \Big(\sum_{i\in[s]} \mu_i\Big) K, \\ & \forall i \in [s], \|Y_i\|_F^2 \le w_i \mu_i. \end{aligned}$$

PROOF. We first handle the case where the estimability condition is not satisfied. In this situation, we have $\Phi_{A|K}(M(\mathbf{w})) = 0$, and we will see that the first constraint of problem (A.2) can only be satisfied if $\sum_{i=1}^s \mu_i = 0$. Note that the second constraint of problem (A.2) implies $Y_i = 0$ for all $i \notin I$. Hence every column of the matrix $\sum_{i=1}^s A_i Y_i$ must be in the set $\operatorname{range}[\sqrt{w_1}A_1, \dots, \sqrt{w_s}A_s] = \operatorname{range} M(\mathbf{w})$. Thus if (at least) one column of K is not included in the range of $M(\mathbf{w})$, then we must have $\sum_i \mu_i = 0$.

Now, assume that the estimability condition $\operatorname{range} K \subseteq \operatorname{range} M(\mathbf{w})$ holds, so that

$$\Phi_{A|K}\big(M(\mathbf{w})\big) = \big(\operatorname{trace} K^T M(\mathbf{w})^- K\big)^{-1} > 0.$$

Let Z_i $(\forall i \in I)$ be optimal matrices for the problem on the right-hand side of (A.1). Then for all $i \in I$, define $\lambda_i := \operatorname{trace} w_i^{-1} Z_i^T Z_i = w_i^{-1}\|Z_i\|_F^2$, $\mu_i := (\sum_j \lambda_j)^{-2}\lambda_i$, and $Y_i := (\sum_j \lambda_j)^{-1} Z_i$ [note that $\sum_{i\in I} \lambda_i = \operatorname{trace} K^T M(\mathbf{w})^- K > 0$], and for $i \in [s] \setminus I$ let $\mu_i := 0$ and $Y_i := 0 \in \mathbb{R}^{\ell_i\times k}$. We have $\sum_{i\in[s]} \mu_i = (\sum_{i\in I} \lambda_i)^{-1} = \Phi_{A|K}(M(\mathbf{w}))$, and by construction the variables μ_i and Y_i satisfy the constraints of problem (A.2).

Conversely, let μ_i and Y_i be feasible variables for problem (A.2). If $\sum_i \mu_i = 0$, then we have $\sum_i \mu_i < \Phi_{A|K}(M(\mathbf{w}))$. Otherwise, define $Z_i := (\sum_{i\in[s]} \mu_i)^{-1} Y_i$, so that the variables Z_i $(i \in I)$ are feasible for the problem on the right-hand side of (A.1). Hence

$$\begin{aligned}\operatorname{trace} K^T M(\mathbf{w})^- K &\leq \sum_{i \in I} \frac{1}{w_i} \|Z_i\|_F^2 = \sum_{i \in I} \frac{1}{w_i (\sum_{i \in [s]} \mu_i)^2} \|Y_i\|_F^2 \\ &\leq \sum_{i \in I} \frac{w_i \mu_i}{w_i (\sum_{i \in [s]} \mu_i)^2} \leq \frac{1}{\sum_{i \in [s]} \mu_i}.\end{aligned}$$

Finally, we obtain the desired inequality by taking the inverse

$$\sum_{i \in [s]} \mu_i \leq \Phi_{A|K}\big(M(\mathbf{w})\big).$$

This completes the proof of the proposition. □

COROLLARY A.3. *Let K be an $m \times k$ matrix. The function $\mathbf{w} \mapsto \Phi_{A|K}(M(\mathbf{w}))$ is SOC-representable.*

The reformulation of problem (1.1) for the criterion $\Phi = \Phi_{A|K}$ as an (MI)SOCP is indicated in Table 1.

REMARK A.4 (The case of $\mathbf{c}$-optimality). The case of $\mathbf{c}$-optimality arises as a special case of both A_K and D_K-optimality when the matrix $K = \mathbf{c} \neq \mathbf{0}$ is a column vector ($k = 1$). The two SOCP formulations (for $\Phi_{A|\mathbf{c}}$ and $\Phi_{D|\mathbf{c}}$- in Table 1) are equivalent, which can be verified by the change of variables $Y_i = J_{1,1}^{-1} Z_i$, $\mu_i = J_{1,1}^{-2} t_{i1}$. (Note that here the matrix J is of size 1×1, i.e., a scalar.)

We next show how Proposition A.2 can be used to obtain an SOC representation of G and I-optimality.

A.2. *G*-optimality. A criterion closely related to D-optimality is the criterion of G-optimality,

$$\Phi_G : M \to \Big(\max_{i \in [s]} \operatorname{trace} A_i^T M^- A_i\Big)^{-1} = \min_{i \in [s]} \Phi_{A|A_i}(M),$$

where the equality holds if we use the convention $\operatorname{trace} K^T M^- K := +\infty$ for all matrices M that do not satisfy the estimability condition ($\operatorname{range} K \not\subseteq \operatorname{range} M$). In the common case of single-response experiments for linear models, the matrices A_i are column vectors, and the scalar $\sigma^2 A_i^T M(\mathbf{w})^- A_i$ represents the variance of the prediction $\hat{\mathbf{y}}_i = A_i^T \hat{\boldsymbol{\theta}}$. Hence G-optimal designs minimize the maximum variance of the predicted values $\hat{\mathbf{y}}_1, \dots, \hat{\mathbf{y}}_s$.

The G and D-optimality criteria are related to each other by the celebrated equivalence theorem of Kiefer and Wolfowitz [22], which was generalized to the case of multivariate regression ($\ell_i > 1$) by Fedorov in 1972 [13]. An important consequence of this theorem is that D- and G-optimal designs coincide when the weight domain $\mathcal{W}$ is the standard probability simplex $\mathcal{W}_\Delta$. However, exact G-optimal designs do not necessarily coincide with their D-optimal counterparts. In a recent article [33], the Brent minimization algorithm was proposed to compute near exact G-optimal factorial designs. But in general, we do not know any standard algorithm for the computation of exact G-optimal designs or G-optimal designs over arbitrary weight domains $\mathcal{W}$ that are defined by a set of linear inequalities.

We know from Corollary A.3 that the concave functions $f_i : \mathbf{w} \to \Phi_{A|A_i}(M(\mathbf{w}))$ are SOC-representable, and hence their minimum is also concave and SOC-representable. An (MI)SOCP formulation of problem (1.1) for the criterion $\Phi = \Phi_G$ is indicated in Table 1. For the case where the weight domain $\mathcal{W}$ is the probability simplex $\mathcal{W}_\Delta$, it gives a new alternative SOCP formulation for D-optimality. Note, however, that in this situation, the SOCP formulation (2.1) for D-optimality from [34] is usually more compact (i.e., it involves fewer variables and fewer constraints) than the G-optimality SOCP of Table 1.

A.3. *I*-optimality. Another widely used criterion is the one of I-optimality (or V-optimality). Here, the criterion is the inverse of the average of the variances of the predicted values $\hat{\mathbf{y}}_1, \dots, \hat{\mathbf{y}}_s$:

$$\Phi_I : M \to \left(\frac{1}{s} \sum_{i\in[s]} \operatorname{trace} A_i^T M^- A_i\right)^{-1}.$$

In fact, this criterion coincides with the $\Phi_{A|K}$ criterion, by setting K to any matrix of full column rank satisfying $KK^T = \frac{1}{s}\sum_{i=1}^s A_i A_i^T$; see, for example, Section 9.8 in [31]. Hence Φ_I-optimal designs can be computed by SOCP. Note that there is also a weighted version of I-optimality, which can be reduced to an A_K-optimal design problem in the same manner.

A.4. Bayesian optimal designs for nonlinear models. For nonlinear models, the information matrix of a design $\mathbf{w}$ depends on the value of the unknown parameter $\boldsymbol{\theta}$ [we denote it by $M(\mathbf{w}, \boldsymbol{\theta})$]; see, for example, [7]. One way to handle this challenging cyclic problem is to search a design $\mathbf{w}$ maximizing the expected value $\Phi_\pi(\mathbf{w})$ of the criterion Φ with respect to some prior π,

$$\Phi_\pi(\mathbf{w}) := \int_{\boldsymbol{\theta}\in\mathbb{R}^m} \Phi\big(M(\mathbf{w}, \boldsymbol{\theta})\big)\,\pi(d\boldsymbol{\theta}).$$

Another alternative, known as *standardized Bayesian design*, is to search for a design maximizing the expected efficiency

$$\phi_\pi(\mathbf{w}) := \int_{\boldsymbol{\theta}\in\mathbb{R}^m} \frac{\Phi(M(\mathbf{w}, \boldsymbol{\theta}))}{\max_{\boldsymbol{\omega}\in\mathcal{W}} \Phi(M(\boldsymbol{\omega}, \boldsymbol{\theta}))}\,\pi(d\boldsymbol{\theta}).$$

In a recent article, Duarte and Wong approximated such integrals by finite sums using Gaussian quadrature formulas [11], in order to obtain SDP formulations of Bayesian optimal design problems. By using the same technique, we immediately see that the Bayesian versions Φ_π and ϕ_π of a SOC-representable criterion Φ are also SOC-representable (modulo the approximation of the integral by a finite sum). This offers the possibility of computing (constrained) exact Bayesian designs by using MISOCP solvers.

Finally, we point out that the standard Bayesian versions of the D- and A-criteria have forms that slightly differ from the formulas given above, and which have other statistical interpretations (see [7] for more details),

$$\Phi_{D,\pi}(\mathbf{w}) := \int_{\boldsymbol{\theta}\in\mathbb{R}^m} \operatorname{logdet} M(\mathbf{w},\boldsymbol{\theta})\pi(d\boldsymbol{\theta}),$$

$$\Phi_{A,\pi}(\mathbf{w}) := -\int_{\boldsymbol{\theta}\in\mathbb{R}^m} \operatorname{trace} M(\mathbf{w},\boldsymbol{\theta})^{-1}\pi(d\boldsymbol{\theta}).$$

Bayesian optimality with respect to the above criteria can also be formulated as an (MI)SOCP, by combining the techniques used in the present paper with those of [11].

REFERENCES

[1] ALIZADEH, F. and GOLDFARB, D. (2003). Second-order cone programming. *Math. Program.* **95** 3–51. MR1971381

[2] ANDERSEN, E. D., JENSEN, B., JENSEN, J., SANDVIK, R. and WORSØE, U. (2009). MOSEK Version 6. Technical Report TR–2009–3, MOSEK.

[3] ATKINSON, A. C. and DONEV, A. N. (1992). *Optimum Experimental Designs* **8**. Oxford Univ. Press, Oxford.

[4] BAILEY, R. A. and CAMERON, P. J. (2009). Combinatorics of optimal designs. In *Surveys in Combinatorics* 2009. *London Mathematical Society Lecture Note Series* **365** 19–73. Cambridge Univ. Press, Cambridge. MR2588537

[5] BEN-TAL, A. and NEMIROVSKI, A. (1987). *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications* **2**. SIAM, Philadelphia.

[6] BOYD, S. and VANDENBERGHE, L. (2004). *Convex Optimization*. Cambridge Univ. Press, Cambridge. MR2061575

[7] CHALONER, K. and VERDINELLI, I. (1995). Bayesian experimental design: A review. *Statist. Sci.* **10** 273–304. MR1390519

[8] CHEN, A. and ESFAHANIAN, A. H. (2005). A demography of t-optimal (n,m) graphs where $n <= 12$. In *AMCS* 121–127. CSREA Press, Las Vegas, NV.

[9] COOK, D. and FEDOROV, V. (1995). Constrained optimization of experimental design. *Statistics* **26** 129–178. MR1318209

[10] COOK, R. D. and THIBODEAU, L. A. (1980). Marginally restricted D-optimal designs. *J. Amer. Statist. Assoc.* **75** 366–371. MR0577367

[11] DUARTE, B. P. M. and WONG, W. K. (2015). Finding Bayesian optimal designs for nonlinear models: A semidefinite programming-based approach. *Int. Stat. Rev.* To appear. DOI:10.1111/insr.12073.

[12] FEDOROV, V. and LEE, J. (2000). Design of experiments in statistics. In *Handbook of Semidefinite Programming* (H. Wolkowicz, R. Saigal and L. Vandenberghe, eds.). Kluwer, Dordrecht.

[13] FEDOROV, V. V. (1972). *Theory of Optimal Experiments*. Academic Press, New York. MR0403103

[14] FILOVÁ, L., TRNOVSKÁ, M. and HARMAN, R. (2012). Computing maximin efficient experimental designs using the methods of semidefinite programming. *Metrika* **75** 709–719. MR2946655

[15] HAINES, L. M. (1987). The application of the annealing algorithm to the construction of exact optimal designs for linear-regression models. *Technometrics* **29** 439–447.

[16] HARMAN, R. (2014). Multiplicative methods for computing D-optimal stratified designs of experiments. *J. Statist. Plann. Inference* **146** 82–94. MR3132481

[17] HARMAN, R. and FILOVÁ, L. (2014). Computing efficient exact designs of experiments using integer quadratic programming. *Comput. Statist. Data Anal.* **71** 1159–1167. MR3132036

[18] HARMAN, R. and JURÍK, T. (2008). Computing c-optimal experimental designs using the simplex method of linear programming. *Comput. Statist. Data Anal.* **53** 247–254. MR2649082

[19] HARMAN, R. and PRONZATO, L. (2007). Improvements on removing nonoptimal support points in D-optimum design algorithms. *Statist. Probab. Lett.* **77** 90–94. MR2339022

[20] HEREDIA-LANGNER, A., CARLYLE, W. M., MONTGOMERY, D. C., BORROR, C. M. and RUNGER, G. C. (2003). Genetic algorithms for the construction of D-optimal designs. *J. Qual. Technol.* **35** 28–46.

[21] IBM ILOG (2009). IBM ILOG CPLEX V12.1. User's manual for CPLEX. Technical report, International Business Machines Corporation.

[22] KIEFER, J. and WOLFOWITZ, J. (1960). The equivalence of two extremum problems. *Canad. J. Math.* **12** 363–366. MR0117842

[23] LOBO, M. S., VANDENBERGHE, L., BOYD, S. and LEBRET, H. (1998). Applications of second-order cone programming. *Linear Algebra Appl.* **284** 193–228. MR1655138

[24] LÖFBERG, J. (2004). YALMIP: A toolbox for modeling and optimization in Matlab. In 2004 *IEEE International Symposium on Computer Aided Control Systems Design* 284–289. IEEE, New York.

[25] LU, Z. and PONG, T. K. (2013). Computing optimal experimental designs via interior point method. *SIAM J. Matrix Anal. Appl.* **34** 1556–1580. MR3131482

[26] MARTÍN-MARTÍN, R., TORSNEY, B. and LÓPEZ-FIDALGO, J. (2007). Construction of marginally and conditionally restricted designs using multiplicative algorithms. *Comput. Statist. Data Anal.* **51** 5547–5561. MR2407659

[27] MITCHELL, T. J. (1974). An algorithm for the construction of "D-optimal" experimental designs. *Technometrics* **16** 203–210. MR0386181

[28] PAPP, D. (2012). Optimal designs for rational function regression. *J. Amer. Statist. Assoc.* **107** 400–411. MR2949369

[29] PETINGI, L., BOESCH, F. and SUFFEL, C. (1998). On the characterization of graphs with maximum number of spanning trees. *Discrete Math.* **179** 155–166. MR1489080

[30] PRONZATO, L. and ZHIGLJAVSKY, A. (2014). Algorithmic construction of optimal designs on compact sets for concave and differentiable criteria. *J. Statist. Plann. Inference* **154** 141–155.

[31] PUKELSHEIM, F. (1993). *Optimal Design of Experiments*. Wiley, New York. MR1211416

[32] PUKELSHEIM, F. and RIEDER, S. (1992). Efficient rounding of approximate designs. *Biometrika* **79** 763–770. MR1209476

[33] RODRIGUEZ, M., JONES, B., BORROR, C. M. and MONTGOMERY, D. C. (2010). Generating and assessing exact G-optimal designs. *J. Qual. Technol.* **42** 3–20.

[34] SAGNOL, G. (2011). Computing optimal designs of multiresponse experiments reduces to second-order cone programming. *J. Statist. Plann. Inference* **141** 1684–1708. MR2763200

[35] SAGNOL, G. (2012). PICOS, a Python interface to conic optimization solvers. Technical Report No. 12–48, ZIB, http://picos.zib.de.

[36] SAGNOL, G. (2013). On the semidefinite representation of real functions applied to symmetric matrices. *Linear Algebra Appl*. **439** 2829–2843. MR3116395

[37] SAGNOL, G. and HARMAN, R. (2013). Computing exact D-optimal designs by mixed integer second order cone programming. Preprint. Available at arXiv:1307.4953v2.

[38] SEBER, G. A. F. (2008). *A Matrix Handbook for Statisticians*. Wiley, Hoboken, NJ. MR2365265

[39] SILVEY, S. D., TITTERINGTON, D. M. and TORSNEY, B. (1978). An algorithm for optimal designs on a finite design space. *Comm*. *Statist*. *Theory Methods* **7** 1379–1389.

[40] TACK, L. and VANDEBROEK, M. (2004). Budget constrained run orders in optimum design. *J*. *Statist*. *Plann*. *Inference* **124** 231–249. MR2066237

[41] TITTERINGTON, D. M. (1976). Algorithms for computing D-optimal design on finite design spaces. In *Proceedings of the* 1976 *Conf. on Information Science and Systems* 213–216. Dept. of Electronic Engineering, John Hopkins Univ., Baltimore, MD.

[42] VANDENBERGHE, L., BOYD, S. and WU, S.-P. (1998). Determinant maximization with linear matrix inequality constraints. *SIAM J. Matrix Anal*. *Appl*. **19** 499–533. MR1614078

[43] WELCH, W. J. (1982). Branch-and-bound search for experimental designs based on D-optimality and other criteria. *Technometrics* **24** 41–48. MR0653110

[44] WRIGHT, S. E., SIGAL, B. M. and BAILER, A. J. (2010). Workweek optimization of experimental designs: Exact designs for variable sampling costs. *J. Agric*. *Biol*. *Environ*. *Stat*. **15** 491–509. MR2788636

[45] WYNN, H. P. (1970). The sequential generation of D-optimum experimental designs. *Ann*. *Math*. *Statist*. **41** 1655–1664. MR0267704

[46] YANG, M., BIEDERMANN, S. and TANG, E. (2013). On optimal designs for nonlinear models: A general and efficient algorithm. *J. Amer. Statist*. *Assoc*. **108** 1411–1420. MR3174717

[47] YU, Y. (2010). Monotonic convergence of a general algorithm for computing optimal designs. *Ann*. *Statist*. **38** 1593–1606. MR2662353

[48] YU, Y. (2011). D-optimal designs via a cocktail algorithm. *Stat*. *Comput*. **21** 475–481. MR2826686

DEPARTMENT OPTIMIZATION
ZUSE INSTITUT BERLIN
TAKUSTR. 7
14195 BERLIN
GERMANY
E-MAIL: sagnol@zib.de

FACULTY OF MATHEMATICS,
PHYSICS AND INFORMATICS
COMENIUS UNIVERSITY
MLYNSKÁ DOLINA
84248 BRATISLAVA
SLOVAKIA
E-MAIL: harman@fmph.uniba.sk