Optimal design of experiments for implicit models

Belmiro P.M. Duarte
Instituto Politécnico de Coimbra,
Instituto Superior de Engenharia de Coimbra,
Department of Chemical and Biological Engineering,
Rua Pedro Nunes, Quinta da Nora, 3030−199 Coimbra, Portugal.
and
CIEPQPF, Department of Chemical Engineering,
University of Coimbra,
Rua Sílvio Lima – Pólo II, 3030-790 Coimbra, Portugal.
and
Anthony C. Atkinson
Department of Statistics,
London School of Economics,
London WC2A 2AE, United Kingdom.
and
José F.O. Granjo
CIEPQPF, Department of Chemical Engineering,
University of Coimbra,
Rua Sílvio Lima – Pólo II, 3030-790 Coimbra, Portugal.
and
Nuno M.C. Oliveira
CIEPQPF, Department of Chemical Engineering,
University of Coimbra,
Rua Sílvio Lima – Pólo II, 3030-790 Coimbra, Portugal.

December 1, 2020

Abstract

Explicit models representing the response variables as functions of the control variables are standard in virtually all scientific fields. For these models there is a vast literature on the optimal design of experiments to provide good estimates of the parameters with the use of minimal resources. Contrarily, the optimal design of experiments for implicit models is more complex and has not been systematically addressed. Nevertheless, there are practical examples where the models relating the response variables, the parameters and the factors are implicit or hardly convertible into an explicit form.

We propose a general formulation for developing the theory of the optimal design of experiments (ODoE) for implicit algebraic models to specifically find continuous local designs. The treatment relies on converting the ODoE problem into an optimization problem of the Nonlinear Programming class which includes the construction of the parameter sensitivities and the Cholesky decomposition of the Fisher Information Matrix. The Nonlinear Programming problem generated has multiple local optima, and we use global solvers, combined with an equivalence theorem from the theory of ODoE, to ensure the global optimality of our continuous optimal designs. We consider D– and A–optimality criteria and apply the approach to five examples of practical interest in chemistry and thermodynamics.

Keywords: Model-based optimal designs, Continuous designs, Implicit models, Nonlinear Programming.

1 Motivation

The optimal design of experiments (ODoE) is a well-established and increasingly important subfield of statistics. Running experiments is costly and users want to rein in costs without sacrificing the statistical efficiency of inferences. The literature on the construction of optimal experimental designs for *explicit models* is extensive (Atkinson et al., 2007; Pukelsheim, 1993; Pronzato and Pázman, 2013). In this paper we employ the nomenclature commonly used in systems theory and apply the term *explicit model* to functionals relating the set of control factors, parameters and process states:

$$\boldsymbol{s} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\theta}), \tag{1}$$

where $\boldsymbol{f}(\bullet) \in \mathbb{R}^{n_s}$ is a set of continuously differentiable functions, $\boldsymbol{x} \in \mathbf{X} \subset \mathbb{R}^{n_x}$ is the set of *control factors*, known without error, $\boldsymbol{s} \in \mathbf{S} \subset \mathbb{R}^{n_s}$ the set of *state variables* that fully characterize the process state after the experiment, $\boldsymbol{\theta} \in \mathbf{P} \subset \mathbb{R}^{n_\theta}$ the set of parameters, $\mathbf{X}$, $\mathbf{S}$ and $\mathbf{P}$ are compact domains of factors, states and parameters, respectively, n_s the number of process states, n_x the number of control factors and n_θ the number of parameters to be estimated from the experiment. Let the *responses*, $\boldsymbol{y}$, be a subset of state variables measured in the experiment, i.e. $\boldsymbol{y} \in \mathbf{Y} \subset \mathbb{R}^{n_y} \subseteq \mathbb{R}^{n_s}$ and $n_y (\leq n_s)$ be the number of response variables.

Under the common assumption that responses $\boldsymbol{y}$ are affected by independent and identically distributed (iid) observational error $\boldsymbol{\varepsilon} \in \mathbb{R}^{n_y}$ with zero mean and standard deviation ς_i, $i \in \{1, \cdots, n_y\}$ previously known, then

$$\boldsymbol{y} = G\,\boldsymbol{s} + \boldsymbol{\varepsilon}, \tag{2}$$

where the expected values of the responses are

$$\mathbb{E}(\boldsymbol{y}) = G\,\boldsymbol{s}, \tag{3}$$

with $\mathbb{E}(\bullet)$ standing for the expectation. The domain $\mathbf{Y}$ is also closed and compact, and G is a $n_y \times n_s$ matrix of 0's except for the elements $G_{i,j},\ i \in \{1,\cdots,n_y\},\ j \in \{1,\cdots,n_s\}$, which are 1 if the j^{th} process state is measured and allocated to the i^{th} element of $\boldsymbol{y}$. Here, $\mathbf{P} = \bigotimes_{j=1}^{n_\theta}[\theta_j^{\text{LO}}, \theta_j^{\text{UP}}]$ is the Cartesian domain of parameters, θ_j representing a local value of parameter j and θ_j^{LO} and θ_j^{UP} are the lower and upper values admissible for θ_j. Further, $\boldsymbol{p}$ is a specific vector of parameters and is used to represent the values of $\boldsymbol{\theta}$ used in designing the experiment, i.e. the vector of parameters for which the locally optimal design is to be calculated. Finally, the error in control factors is considered negligible.

Contrarily, systematic approaches for finding optimal experimental designs for *implicit models* are still elusive. Herein, we define implicit models as representations similar to

$$\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) = \mathbf{0}, \tag{4}$$

where $\boldsymbol{g}(\bullet) \in \mathbb{R}^{n_s}$ is also a set of continuously differentiable functions, the responses are obtained from states with (3), and all the other symbology has the same meaning as introduced above. Implicit models, as they were called by Marshall (2003), have other designations, among which are *implicit functional relationships*, see Sachs (1976) and Seber and Wild (2003, Chap. 10), and *implicit regression models* (Reilly and Patino-Lea1, 1981). Implicit models are a generalization of explicit relationships that have not been exploited because of (i) reduced practical interest since, in most applications, the response variables measured are explicitly related to the control factors and parameters; and (ii) the complexity of constructing ODoE which requires the use of numerical procedures to construct the parameter sensitivities embedded in the optimal design problem. Given the knowledge gap detected, our goal is to propose a general approach for calculating continuous locally optimal experimental designs for implicit models of practical interest. Typically these models appear in the characterization of Vapor-Liquid Equilibrium (VLE) or Liquid-Liquid Equilibrium data (Englezos et al., 1990; Gao et al., 2017), VLE thermodynamic

consistency checking (Dohnal and Fenclová, 1985), and potentiometric titration (Hofman and Krzyżanowska, 1986).

Amo-Salas et al. (2016) discuss the optimal design of experiments for implicit models but their conceptualization of implicit models is less general than ours. Their study considers models with separated variables (responses, states and inputs) where the relation between the responses and the explanatory variables is unknown, contrarily to its inverse. Algebraic manipulations allow reformulating the model as explicit, and finding the optimal design on the response variables space for this new functional. Here, we address implicit models of non-separable variables that require numerical methods to determine the states for each $\boldsymbol{x}$ and $\boldsymbol{\theta}$. Dovì et al. (1993) consider errors-in-variable implicit models similar to ours. They assume both explanatory variables and responses are subject to noise, but contrarily to our work, only discuss sequential designs; they propose an optimization-based approach finding one experiment at a time.

In this work we address implicit *algebraic* models similar to (4). The optimization of experimental designs for models described by differential equations usually benefits from additional considerations for a more efficient numerical implementation, such as determination of the derivatives of the response functions, and these problems have been considered separately. Experimental designs for such models have been addressed by several authors aiming to (i) find the optimal set of sampling times to observe the system (Atkinson and Bogacka, 2002); and (ii) find the optimal profile of control actions to maximize the information obtained from a (dynamic) experiment (Franceschini and Macchietto, 2008; Bauer et al., 2000; Körkel et al., 2004; Balsa-Canto et al., 2016; Galvanin and Bezzo, 2018). The approach proposed herein can be directly applied to differential systems when their numerical solution involves the discretization (or parametrization) of the domain of the independent variables since, in both cases, sets of algebraic equations are obtained, representing local approximations of the solution. Examples of methods

in this category are considered by Hoang et al. (2013) and Duarte et al. (2019), where a simultaneous approach based on orthogonal collocation on finite elements is used to parameterize the time domain.

To motivate the following developments, an illustrative example arising from thermodynamics is considered, to establish a more complete background of the problems addressed herein.

1.1 Illustrative example

Vapor-liquid equilibrium (VLE) experiments are repeatedly undertaken to build models which are subsequently shipped with process simulators and modeling systems. VLE models are mostly built from experiments of binary mixtures and are crucial in designing, optimizing and controlling process equipment, and find application in chemical and other industries. Constructing adequate mathematical models from VLE data usually requires intensive experimental work.

Without loss of generality, let us consider that the experiments take place at moderate pressure, the binary mixture is a non-electrolyte solution, and no associative reactions occur in the vapor phase. Under these assumptions, a broad family of Gibbs free energy models is described in literature to represent non-ideal liquid mixtures, one being the Wilson model (Wilson, 1964), presented later in Section 4.2. Accordingly, an implicit model $\boldsymbol{g}(\bullet)$ with two nonlinear algebraic equalities and two state variables is used to model the VLE data, see model (32). The most common experimental setup requires measuring only the molar fraction of a component in the liquid phase; the composition of the vapor phase is estimated from the measurement using the equilibrium relations. Since the fugacities are dependent on the composition of the liquid phase and are strongly nonlinear, the models are formed by implicit nonlinear functions of the control factor (i.e., the pressure), the response (i.e., the molar fraction of one of the components in the liquid phase) and the parameters to be estimated from the experiments, which are commonly called

Binary Interaction Parameters (BIPs). Practically, there is substantial interest in finding optimal experimental designs to characterize VLE so that maximum information is obtained given the resources available. Although $\boldsymbol{g}(\bullet)$ can change with the kind of phase equilibrium or thermodynamic models used, the basic nature of the ODoE problem to be solved is similar.

1.2 Algorithms for finding Optimal Experimental Designs for explicit models

Model-based optimal experimental designs provide maximum information at minimum cost. Our setup is that we have a given implicit parametric model defined on a *compact* design space, a given design criterion and a given budget determining the total number of observations, N. The design problem, commonly designated as *exact* optimal design of experiments, is to determine optimal design points, which are members of the design space that describes the experimental conditions, and the number of replicates at each of these design points, subject to the requirement that they sum to N. Since we consider *continuous* optimal design problems where $N \to +\infty$, our goal is to determine optimal design points and the relative effort (weights) at each of these design points, subject to the requirement that they sum to 1. The previous approximation leads to a P-hard reformulation of the original exact optimal design problem and, consequently, to convex optimization problems (Fedorov and Hackl, 1997). However, its numerical treatment translates into optimization problems with multinomial terms requiring specialized global optimization solvers to assure the global optimum is found.

Over the last decades, algorithms have been developed and continually improved for generating different types of optimal designs for explicit algebraic models. Various numerical algorithms developed for the construction of such designs are based on exchange methods, originally proposed for the D–optimality criterion (Mitchell and Miller Jr., 1970; Wynn, 1970; Fedorov,

1972). The numerical efficiency of these Wynn–Fedorov schemes has been improved by several authors, including Wu (1978); Wu and Wynn (1978); Pronzato (2003); Harman and Pronzato (2007). Some of these algorithms are reviewed, compared and discussed in Meyer and Nachtsheim (1995) and Pronzato (2008), among others. Another approach to finding continuous optimal designs is based on Multiplicative algorithms, which have found broad application due to their simplicity (Mandal et al., 2015). The basic algorithm was proposed by Titterington (1976) and later exploited in Pázman (1986); Fellman (1989); Pukelsheim and Torsney (1991); Torsney and Mandal (2006); Mandal and Torsney (2006); Dette et al. (2008); Torsney and Martín-Martín (2009); Yu (2010c,b). Recently, cocktail algorithms, that rely on both exchange and multiplicative algorithms, have been proposed (Yu, 2010a), and improved (Yang et al., 2013).

Because our models are nonlinear, the optimal design depends on the values of the parameters. We therefore find *locally optimal* designs for specified point prior values of the parameters. In practice, the continuous designs have to be rounded to provide exact designs with integer valued allocations of experimental effort. Rules for such rounding are in Pukelsheim and Rieder (1992).

Mathematical programming algorithms can currently solve complex, high-dimensional optimization problems, especially when these are P-hard. Examples of applications of mathematical programming algorithms for finding continuous optimal designs are Linear Programming (Gaivoronski, 1986; Harman and Jurík, 2008), Second Order Conic Programming (Sagnol, 2011; Sagnol and Harman, 2015), Semidefinite Programming (SDP) (Vandenberghe and Boyd, 1999; Papp, 2012; Duarte and Wong, 2015), Semi Infinite Programming (SIP) (Duarte and Wong, 2014; Duarte et al., 2015), and Nonlinear Programming (NLP) (Chaloner and Larntz, 1989; Molchanov and Zuyev, 2002). Applications based on optimization procedures relying on metaheuristic algorithms are also reported in the literature, see Heredia-Langner et al. (2004) for Genetic Algorithms, Woods (2010) for Simulating Annealing, Chen et al. (2015) for Particle Swarm Optimiza-

tion and Masoudi et al. (2019) for the Imperialist Competitive Algorithm, among others.

The proposed approach is grounded on mathematical programming. Our formulation leads to an optimization problem of the NLP class; since the problem has multiple local optima, a global optimizer is used. The main novelty of the numerical aspects of the proposed formulations is that the equations representing the model and the sensitivity construction are embedded in the optimal design problem as additional constraints. The same holds for matrix algebra operations required for computing D– and A–optimality criteria. This strategy allows us to find optimal designs that satisfy the model equations and guarantees that all the solutions in the convergence process are feasible. The Cholesky decomposition in the optimization problem allows automating the computation of the determinant and trace of the inverse of the Fisher Information Matrix (FIM) within the optimization problem, see Duarte et al. (2020).

To systematize our contribution, the innovative aspects of this paper are:

1. a systematic approach to find optimal experimental designs for nonlinear implicit non-separable algebraic models, a broader class than that of explicit models commonly considered in Model-based Optimal Design of Experiments;

2. mathematical programming-based formulations to find D– and A–optimal designs for this class of models. The optimality of the numerically found designs is checked using an equivalence theorem;

3. the application of the formulations proposed to common models in thermodynamics and chemistry areas.

The paper is organized as follows. Section 2 introduces the background and the notation used to formulate the problem, including the equivalence theorem, as well as the fundamentals of nonlinear programming. Section 3 presents the mathematical programming formulation for

finding D– and A–optimal designs for implicit models. Details of the construction of the Fisher Information Matrix (FIM) are given, which in turn requires the calculation of the sensitivity coefficients.

Section 4 applies these formulations to finding optimal designs. First we consider a simple example to analyze the details of the algorithm, which we subsequently apply to five additional problems in the fields of chemistry, image processing, and thermodynamics. Finally, Section 5 reviews the formulation and offers a summary of the results obtained. Appendix A provides the details of the numerical implementation.

2 Notation and background

This section establishes the nomenclature used in the representation of the models. In §2.1 we present the experimental design problems outlined above. Then, in §2.2, overview the fundamentals of NLP.

2.1 Optimal experimental design

Bold face lowercase letters represent vectors, bold face capital letters continuous domains, blackboard bold capital letters discrete domains and capital letters matrices. Finite sets containing ι elements are compactly represented by $[\![\iota]\!] \equiv \{1, \cdots, \iota\}$. The transpose operation of a matrix is represented by "$^\intercal$" and the trace of matrix by $\mathrm{tr}(\bullet)$.

We recall model (4) and consider a continuous design with K support points at $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K$. The weights at these points are, respectively, $w_1, w_2, \ldots, w_K$ where K is chosen by the user so that $K \geq n_\theta$. To implement the design for a total of N observations, we take roughly $N \times w_k$ observations at $\boldsymbol{x}_k$, $k \in [\![K]\!]$, subject to $N \times w_1 + \cdots + N \times w_K = N$, and each summand is an integer.

For models with n_x control factors, we denote the k^{th} support point by $\boldsymbol{x}_k^\intercal = (x_{k,1}, \ldots, x_{k,n_x})$ and represent the design ξ by K rows $(\boldsymbol{x}_k^\intercal, w_k)$, $k \in [\![K]\!]$ with $\sum_{k=1}^K w_k = 1$. In what is to follow, we let $\Xi \equiv \mathbf{X}^K \times \Sigma$ be the space of feasible K-point designs over $\mathbf{X}$ where Σ is the $K-1$-simplex in the domain of weights $\Sigma = \{w_k : \ w_k \geq 0, \ \forall k \in [\![K]\!], \ \sum_{k=1}^K w_k = 1\}$.

The information resulting from an experimental design is measured by its FIM. The elements of the normalized FIM are the negative expectation of the second order derivatives of the log-likelihood of (4), $\mathscr{L}(\xi, \boldsymbol{\theta})$, with respect to the parameters, given by

$$\mathscr{M}(\xi) = -\mathbb{E}\left[\frac{\partial}{\partial \boldsymbol{\theta}}\left(\frac{\partial \mathscr{L}(\xi)}{\partial \boldsymbol{\theta}^\intercal}\right)\right] = \int_{\xi \in \Xi} M(\boldsymbol{x}) \ \mathrm{d}(\xi) = \sum_{k=1}^K w_k \, M(\boldsymbol{x}_k), \tag{5}$$

where $\mathscr{M}(\xi)$ is the *global* FIM from the design ξ, $M(\boldsymbol{x}_k)$ is the *local* FIM from point $\boldsymbol{x}_k$.

Herein, we focus on the class of design criteria proposed by Kiefer (1974) where each member in the class, indexed by a parameter δ, is positively homogeneous and defined on the set of symmetric $n_\theta \times n_\theta$ semi-positive definite matrices given by

$$\Phi_\delta[\mathscr{M}(\xi)] = \left[\frac{1}{n_\theta} \ \mathrm{tr}(\mathscr{M}(\xi)^\delta)\right]^{1/\delta}. \tag{6}$$

The maximization of Φ_δ for $\delta \neq 0$ is equivalent to minimizing $\mathrm{tr}(\mathscr{M}(\xi)^\delta)$ when $\delta < 0$. Practically, Φ_δ becomes $[\mathrm{tr}(\mathscr{M}(\xi)^{-1})]^{-1}$ for $\delta = -1$, which is A–optimality, and $[\det[\mathscr{M}(\xi)]]^{1/n_\theta}$ when $\delta \to 0$, which is D–optimality. These design criteria are suitable for estimating model parameters as they maximize the FIM in various ways. For the D–optimality criterion the volume of the confidence region of $\boldsymbol{\theta}$ is proportional to $\det[\mathscr{M}^{-1/2}(\xi)]$. Then, maximizing the determinant (or a convenient convex function of the determinant) of the FIM leads to the smallest possible volume. Consequently, the ODoE problem can be cast as an optimization problem. For example, when $\boldsymbol{p}$ is fixed, the locally D– and A–optimal designs are respectively defined by

$$\xi_D = \arg\max_{\xi \in \Xi} \log\{\det[\mathscr{M}(\xi, \boldsymbol{p})]\}, \tag{7}$$

$$\xi_A = \arg\min_{\xi \in \Xi} \mathrm{tr}[\mathscr{M}(\xi, \boldsymbol{p})^{-1}], \tag{8}$$

where the criteria (7-8) are $+\infty$ for designs with singular information matrices. Herein we limit our analysis to D– and A–optimal designs that are the most commonly used in practical applications. Without loss of generality, the formulations proposed in the following sections can easily be extended to other criteria of Kiefer class, such as G– and I–optimality if interest is in prediction rather than parameter estimation.

When the design criterion is convex (which is the case for the above criteria), the global optimality of a design ξ in $\mathbf{X}$ can be verified using an equivalence theorem based on the consideration of the directional derivative of the objective function (Kiefer and Wolfowitz, 1960; Fedorov, 1972; Whittle, 1973; Kiefer, 1974; Silvey, 1980; Pukelsheim, 1993). For instance, if we let $\boldsymbol{\delta}_x$ be the degenerate design at the point $\boldsymbol{x} \in \mathbf{X}$, the equivalence theorems for D– and A–optimality are as follow: (i) ξ_D is D–optimal if and only if

$$\mathrm{tr}\left\{[\mathscr{M}(\xi_D)]^{-1} M(\boldsymbol{\delta}_x)\right\} - n_\theta \leq 0, \quad \forall \boldsymbol{x} \in \mathbf{X}; \tag{9}$$

(ii) ξ_A is globally A–optimal if and only if

$$\mathrm{tr}\left\{[\mathscr{M}(\xi_A)]^{-2} M(\boldsymbol{\delta}_x)\right\} - \mathrm{tr}\left\{[\mathscr{M}(\xi_A)]^{-1}\right\} \leq 0, \quad \forall \boldsymbol{x} \in \mathbf{X}. \tag{10}$$

We call the functions on the left side of the inequalities (9-10) *dispersion functions* and denote them by $\Psi(\boldsymbol{x}|\xi)$. To compare the D–optimal efficiency, an indicator of the information content extracted from two different designs, say ξ_D and ξ_D^{ref}, where the last one is the reference, we use

$$\mathrm{Eff}_D = \left\{ \frac{\det[\mathscr{M}(\xi_D, \boldsymbol{\theta})]}{\det[\mathscr{M}(\xi_D^{\mathrm{ref}}, \boldsymbol{\theta})]} \right\}^{1/n_\theta}, \tag{11}$$

and, similarly, for A–optimality criterion, the efficiency of ξ_A relative to ξ_A^{ref} is defined by

$$\mathrm{Eff}_A = \frac{\mathrm{tr}[\mathscr{M}^{-1}(\xi_A^{\mathrm{ref}}, \boldsymbol{\theta})]}{\mathrm{tr}[\mathscr{M}^{-1}(\xi_A, \boldsymbol{\theta})]}. \tag{12}$$

2.2 Nonlinear Programming

In this section we introduce the fundamentals of NLP which are used to solve the design problems (7-8). Nonlinear Programming seeks to find the global optimum $\boldsymbol{x}$ of a convex or nonconvex nonlinear function $f : \mathbf{X} \mapsto \mathbb{R}$ in a compact domain $\mathbf{X}$ with possibly nonlinear constraints. The general structure of the NLP problems is:

$$\min_{\boldsymbol{x}\in\mathbf{X}} f(\boldsymbol{x}) \tag{13a}$$

$$\text{s.t. } \boldsymbol{g}(\boldsymbol{x}) \leq \mathbf{0} \tag{13b}$$

$$\boldsymbol{h}(\boldsymbol{x}) = \mathbf{0}, \tag{13c}$$

where (13b) represents a set of r_i inequalities, and (13c) represents a set of r_e equality constraints. The functions $f(\boldsymbol{x})$, $\boldsymbol{g}(\boldsymbol{x})$ and $\boldsymbol{h}(\boldsymbol{x})$ are twice differentiable. In our context, the variable $\boldsymbol{x} \in \mathbf{X}$ includes the location of the support points as well as the weights quantifying the relative effort required at each one. By construction $\mathbf{X}$ in (13a) is closed which is what we have for Ξ.

Nested and gradient projection methods are commonly used to solve NLP problems. Some examples are the General Reduced Gradient (GRG) (Drud, 1985, 1994) and the Trust-Region (Coleman and Li, 1994) algorithms. Other methods are Sequential Quadratic Programming (SQP) (Gill et al., 2005) and the Interior-Point (IP) (Byrd et al., 1999). Ruszczyński (2006) provides an overview of NLP algorithms.

3 NLP formulations for optimal design of experiments

In this section we introduce NLP formulations for finding $K-$point, D–, and A–optimal designs for implicit models. In Section 3.1 we introduce the approach for calculating the sensitivity

coefficients and constructing the FIM. Then, in Sections 3.2 and 3.3, we respectively present the formulations for finding D– and A–optimal designs.

3.1 Construction of the FIM

Here we introduce the methodology used for computing the sensitivity coefficients of the response variables in the implicit model (4) and the related FIM.

Applying the chain rule to differentiate the i^{th} equality in (4), $g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) = 0$, with respect to parameter θ_l, $l \in [\![n_\theta]\!]$, yields

$$\sum_{j=1}^{n_s} \left(\frac{\partial g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})}{\partial s_j} \right) \frac{\partial s_j}{\partial \theta_l} + \frac{\partial g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})}{\partial \theta_l} = 0, \quad i \in [\![n_s]\!],\ l \in [\![n_\theta]\!]. \tag{14}$$

Let $\boldsymbol{\alpha}_{i,k} \in \mathbb{R}^{1\times n_s}$ be the vector of derivatives of the i^{th} function $g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})$ with respect to the model states at the k^{th} support point $\boldsymbol{x}_k$, $\boldsymbol{\beta}_{l,k} \in \mathbb{R}^{n_s\times 1}$ be the vector containing the sensitivities of the state variables with respect to parameter l at support point k. Each sensitivity coefficient is denoted by $\sigma_{j,l,k}$, and $\gamma_{i,l,k} \in \mathbb{R}$ is the vector (with a single element) containing the derivative of function i, $g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})$, with respect to parameter l at support point k, i.e.

$$\begin{aligned}
\boldsymbol{\alpha}_{i,k} &= \left(\tfrac{\partial g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})}{\partial s_1}, \ \cdots, \ \tfrac{\partial g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})}{\partial s_j}, \ \cdots, \ \tfrac{\partial g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})}{\partial s_{n_s}} \right),\ i \in [\![n_s]\!],\ \boldsymbol{x} \in \{\boldsymbol{x}_k :\ k \in [\![K]\!]\} \\
\boldsymbol{\beta}_{l,k} &= \left(\tfrac{\partial s_1}{\partial \theta_l}, \ \cdots, \ \tfrac{\partial s_j}{\partial \theta_l}, \ \cdots, \ \tfrac{\partial s_{n_s}}{\partial \theta_l} \right)^{\mathsf{T}},\ l \in [\![n_\theta]\!],\ \boldsymbol{x} \in \{\boldsymbol{x}_k :\ k \in [\![K]\!]\} \\
\gamma_{i,l,k} &= \left(\tfrac{\partial g_i(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})}{\partial \theta_l} \right),\ i \in [\![n_s]\!],\ l \in [\![n_\theta]\!],\ \boldsymbol{x} \in \{\boldsymbol{x}_k :\ k \in [\![K]\!]\}.
\end{aligned}$$

The calculation of the full set of sensitivities for the k^{th} support point $\boldsymbol{x}_k$ requires solving the algebraic equations (AEs) system with respect to sensitivities contained in $\boldsymbol{\beta}_{l,k}$, $l \in [\![n_\theta]\!]$, $k \in [\![K]\!]$, for all the algebraic equations in (4), i.e.,

$$\begin{pmatrix} \boldsymbol{\alpha}_{1,k} & \cdots & \mathbf{0}_{n_s} & \cdots & \mathbf{0}_{n_s} \\ \mathbf{0}_{n_s} & \cdots & \boldsymbol{\alpha}_{1,k} & \cdots & \mathbf{0}_{n_s} \\ \mathbf{0}_{n_s} & \cdots & \mathbf{0}_{n_s} & \cdots & \boldsymbol{\alpha}_{1,k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\alpha}_{i,k} & \cdots & \mathbf{0}_{n_s} & \cdots & \mathbf{0}_{n_s} \\ \mathbf{0}_{n_s} & \cdots & \boldsymbol{\alpha}_{i,k} & \cdots & \mathbf{0}_{n_s} \\ \mathbf{0}_{n_s} & \cdots & \mathbf{0}_{n_s} & \cdots & \boldsymbol{\alpha}_{i,k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\alpha}_{n_s,k} & \cdots & \mathbf{0}_{n_s} & \cdots & \mathbf{0}_{n_s} \\ \mathbf{0}_{n_s} & \cdots & \boldsymbol{\alpha}_{n_s,k} & \cdots & \mathbf{0}_{n_s} \\ \mathbf{0}_{n_s} & \cdots & \mathbf{0}_{n_s} & \cdots & \boldsymbol{\alpha}_{n_s,k} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}_{1,k} \\ \vdots \\ \boldsymbol{\beta}_{l,k} \\ \vdots \\ \boldsymbol{\beta}_{n_\theta,k} \end{pmatrix} + \begin{pmatrix} \gamma_{1,1,k} \\ \vdots \\ \gamma_{1,l,k} \\ \vdots \\ \gamma_{1,n_\theta,k} \\ \vdots \\ \gamma_{n_s,1,k} \\ \vdots \\ \gamma_{n_s,l,k} \\ \vdots \\ \gamma_{n_s,n_\theta,k} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \boldsymbol{x} \in \{\boldsymbol{x}_k : k \in [\![K]\!]\}, \quad (15)$$

where $\mathbf{0}_{n_s}$ is the row vector of zeros of size n_s. The system (15) is compactly represented as

$$A_k \, \boldsymbol{v}_k + \boldsymbol{z}_k = 0, \qquad k \in [\![K]\!], \tag{16}$$

where the vector $\boldsymbol{v}_k$ contains all the sensitivity coefficients $\sigma_{j,l,k},\ j \in [\![n_s]\!],\ l \in [\![n_\theta]\!], k \in [\![K]\!]$, A_k is the matrix of derivatives of the functions with respect to the response variables and $\boldsymbol{z}_k$ is the vector with the derivatives of $\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})$ with respect to the parameters at support point $\boldsymbol{x}_k$. Matrices A_k will not be singular (otherwise the model includes linear dependencies between parameters) but they can be ill conditioned; in those cases we can adopt a previous re-parametrization technique as in Quaglio et al. (2019) or parameter scaling strategy as in Hoang et al. (2013).

The sensitivity coefficients of the set of response variables with respect to the parameters are obtained from $\sigma_{j,l,k},\ j \in [\![n_s]\!],\ l \in [\![n_\theta]\!],\ k \in [\![K]\!]$ by using Equation (3). Afterwards, the local FIMs can then be computed. Without loss of generality, we first consider a system with a single

response variable in each experiment, which is the first of the set of state variables characterizing the system. Specifically, we have $n_y = 1$ and G in (3) is a row vector of 0's with $G_{1,1}$ being 1. Then,

$$M(\boldsymbol{x}_k) = \left(\sigma_{1,1,k}, \quad \cdots, \quad \sigma_{1,n_\theta,k}\right)^{\mathsf{T}} \times \left(\sigma_{1,1,k}, \quad \cdots, \quad \sigma_{1,n_\theta,k}\right), \tag{17}$$

where each element of $M(\boldsymbol{x}_k)$ is represented as $m^{\text{loc}}_{l,\ell,k}$, $l, \ell \in [\![n_\theta]\!]$, $k \in [\![K]\!]$.

When the system has more than one response variable the FIM requires first computing $M(\boldsymbol{x}_k)$ for each one with (17) and then weighting them considering the variance-covariance matrix for the corresponding pair of observations (Draper and Hunter, 1966).

3.2 D-optimal designs

In this section we propose a formulation for finding D–optimal continuous designs on Ξ defined in (7).

To maximize $\log\left(\det[\mathscr{M}(\xi, \boldsymbol{p})]\right)$, we apply the Cholesky decomposition to the global FIM and write

$$\mathscr{M}(\xi, \boldsymbol{p}) = \mathscr{U}^{\mathsf{T}}(\xi, \boldsymbol{p})\, \mathscr{U}(\xi, \boldsymbol{p}), \tag{18}$$

where $\mathscr{U}(\xi, \boldsymbol{p})$ is an upper triangular matrix and has positive diagonal elements $u_{i,i}$ when the FIM is positive definite. It follows that

$$\det(\mathscr{M}(\xi, \boldsymbol{p})) = \prod_{i=1}^{n_\theta} u_{i,i}^2, \tag{19}$$

and $\log[\det(\mathscr{M}(\xi, \boldsymbol{p}))] = 2\sum_{i=1}^{n_\theta} \log(u_{i,i})$. Then, maximizing $\det(\mathscr{M}(\xi, \boldsymbol{p}))$ is equivalent to maximizing the sum of the logarithms of the diagonal elements of $\mathscr{U}(\xi, \boldsymbol{p})$.

Let $m_{i,j}$, $i,\ j \in [\![n_\theta]\!]$ be the $(i,j)^{\text{th}}$ element of the global FIM $\mathscr{M}(\xi, \boldsymbol{p})$ and $u_{i,j}$ the $(i,j)^{\text{th}}$

element of $\mathscr{U}(\xi, \boldsymbol{p})$. The formulation for finding a locally D–optimal continuous design is

$$\max_{\boldsymbol{x},\boldsymbol{w}} \sum_{i=1}^{n_\theta} \log(u_{i,i}) \tag{20a}$$

$$\text{s.t. } \boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x}_k, \boldsymbol{\theta}) = \boldsymbol{0}, \quad k \in [\![K]\!] \tag{20b}$$

$$\text{Equations (3, 15-17)} \tag{20c}$$

$$m_{i,j} = \sum_{k=1}^{K} w_k \, m^{\text{loc}}_{i,j,k}, \quad i,\ j \in [\![n_\theta]\!] \tag{20d}$$

$$m_{i,j} = \sum_{l=1}^{n_\theta} u_{l,i} u_{l,j}, \quad i,\ j \in [\![n_\theta]\!],\ i \le j \tag{20e}$$

$$u_{i,i} \ge \zeta, \quad i \in [\![n_\theta]\!] \tag{20f}$$

$$u_{i,j} = 0, \quad i,\ j \in [\![n_\theta]\!],\ i \ge j+1 \tag{20g}$$

$$m_{i,i} \ge u^2_{i,j}, \quad i, j \in [\![n_\theta]\!] \tag{20h}$$

$$\sum_{k=1}^{K} w_k = 1 \tag{20i}$$

$$\boldsymbol{x} \in \mathbf{X}^K,\ \boldsymbol{w} \in [0,1]^K.$$

Here, ζ is a small positive constant to ensure that the FIM is positive definite. For all examples in §4, $\zeta = 1 \times 10^{-5}$. Equation (20b) is the model, (20c) includes the set of equations used to determine the sensitivity coefficients and the observation equation, equation (20d) follows from (5), (20e) represents the Cholesky decomposition, (20f) guarantees that all diagonal elements of $\mathscr{U}(\xi, \boldsymbol{p})$ are positive and (20g) assures that $\mathscr{U}(\xi, \boldsymbol{p})$ is upper triangular. Equation (20h) is a numerical stability condition imposed on the Cholesky factorization of positive semidefinite matrices (Golub and Van Loan, 2013, Theorem 4.2.8) and the constraint (20i) restricts the sum of weights to 1.

When $\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{0}$ has multiple solutions for each support point the problem can be handled

by incorporating additional knowledge about the model, which is formalized as additional constraints (on $\boldsymbol{y}$, $\boldsymbol{s}$ and $\boldsymbol{x}$) and included in model (20), see §4.1 for an example. If the domain of search is not constrained the optimal design found is not affected in terms of efficiency but may be non-unique.

3.3 A-optimal design

Now, we propose a formulation to determine A–optimal continuous designs modeled by (8). The optimization problem requires inverting $\mathscr{M}^{-1}(\xi,\boldsymbol{p})$ which is potentially a numerically unstable operation when the FIM is ill-conditioned. To avoid the explicit computation of the inverse matrix, we apply the Cholesky decomposition to invert the upper diagonal matrix $\mathscr{U}(\xi,\boldsymbol{p})$ that results from the decomposition of $\mathscr{M}(\xi,\boldsymbol{p})$; the rationale is that inverting an upper triangular matrix obtained by Cholesky factorization is numerically more stable than inverting the original matrix (Du Croz and Higham, 1992). The procedure has three steps that are handled simultaneously within the optimization problem: (i) apply the Cholesky decomposition to the FIM, cf. §3.2; (ii) invert the upper triangular matrix $\mathscr{U}(\xi,\boldsymbol{p})$ using the relation $\mathscr{U}(\xi,\boldsymbol{p})\,\mathscr{U}^{-1}(\xi,\boldsymbol{p}) = I_{n_\theta}$, where I_{n_θ} is the $n_\theta-$dimensional identity matrix; and (iii) compute $\mathscr{M}^{-1}(\xi,\boldsymbol{p})$ via $\mathscr{U}^{-1}(\xi,\boldsymbol{p})$, i.e. $\mathscr{M}^{-1}(\xi,\boldsymbol{p}) = \mathscr{U}^{-1}(\xi,\boldsymbol{p}) \times [\mathscr{U}^{-1}(\xi,\boldsymbol{p})]^\intercal$ (Du Croz and Higham, 1992), and, finally, compute $\mathrm{tr}[\mathscr{M}^{-1}(\xi,\boldsymbol{p})]$.

Let $\overline{m}_{i,j}$ be the $(i,j)^{\text{th}}$ entry of $\mathscr{M}^{-1}(\xi,\boldsymbol{p})$ and $\overline{u}_{i,j}$ be the $(i,j)^{\text{th}}$ entry of $\mathscr{U}^{-1}(\xi,\boldsymbol{p})$ where $i,\ j \in [\![n_\theta]\!]$. By construction, $\mathscr{U}(\xi,\boldsymbol{p})$ is positive definite and invertible if all the diagonal elements are positive. The same holds for $\mathscr{U}^{-1}(\xi,\boldsymbol{p})$. Step (i) is the Cholesky decomposition of the

FIM represented by (20e) and the second step corresponds to inverting $\mathscr{U}(\xi, \boldsymbol{p})$ formulated as:

$$\begin{cases} \sum_{l=1}^{n_\theta} u_{i,l}\, \overline{u}_{l,j} = 1 & \text{if } i = j \\ \sum_{l=1}^{n_\theta} u_{i,l}\, \overline{u}_{l,j} = 0 & \text{if } i \neq j, \end{cases} \tag{21}$$

with step (iii) represented by

$$\overline{m}_{i,j} = \sum_{l=1}^{n_\theta} \overline{u}_{i,l} \overline{u}_{l,j}, \quad i,\ j \in [\![n_\theta]\!],\ i \leq j. \tag{22}$$

A–optimal designs minimize $\mathrm{tr}(\mathscr{M}^{-1}(\xi, \boldsymbol{p}))$ or equivalently, minimize the sum of all $\overline{m}_{i,i},\ i \in [\![n_\theta]\!]$. The complete NLP for computing A–optimal designs is

$$\min_{\boldsymbol{x},\boldsymbol{w}} \sum_{i=1}^{n_\theta} \overline{m}_{i,i} \tag{23a}$$

$$\text{s.t. } \boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x}_k, \boldsymbol{\theta}) = \boldsymbol{0}, \quad k \in [\![K]\!] \tag{23b}$$

$$\text{Equations (3, 15-17)} \tag{23c}$$

$$m_{i,j} = \sum_{k=1}^{K} w_k\, m^{\mathrm{loc}}_{i,j,k}, \quad i,\ j \in [\![n_\theta]\!] \tag{23d}$$

$$m_{i,j} = \sum_{l=1}^{n_\theta} u_{l,i} u_{l,j}, \quad i,\ j \in [\![n_\theta]\!],\ i \leq j \tag{23e}$$

$$\sum_{l=1}^{n_\theta} u_{i,l}\, \overline{u}_{l,j} = 1, \quad i,\ j \in [\![n_\theta]\!],\ i = j \tag{23f}$$

$$\sum_{l=1}^{n_\theta} u_{i,l}\, \overline{u}_{l,j} = 0, \quad i,\ j \in [\![n_\theta]\!],\ i \neq j \tag{23g}$$

$$\overline{m}_{i,j} = \sum_{l=1}^{n_\theta} \overline{u}_{i,l} \overline{u}_{l,j}, \quad i,\ j \in [\![n_\theta]\!],\ i \leq j \tag{23h}$$

$$u_{i,i} \geq \zeta, \quad i \in [\![n_\theta]\!] \tag{23i}$$

$$\overline{u}_{i,i} \geq \zeta, \quad i \in [\![n_\theta]\!] \tag{23j}$$

$$u_{i,j} = 0, \quad i,\ j \in [\![n_\theta]\!],\ i \geq j+1 \tag{23k}$$

$$\overline{u}_{i,j} = 0, \quad i,\ j \in [\![n_\theta]\!],\ i \geq j+1 \tag{23l}$$

$$\overline{m}_{i,j} = \overline{m}_{j,i}, \quad i,\ j \in [\![n_\theta]\!],\ i \leq j-1 \tag{23m}$$

$$m_{i,i} \geq u_{i,j}^2, \quad i,j \in [\![n_\theta]\!] \tag{23n}$$

$$\overline{m}_{i,i} \geq \overline{u}_{i,j}^2, \quad i,j \in [\![n_\theta]\!] \tag{23o}$$

$$\sum_{k=1}^{K} w_k = 1 \tag{23p}$$

$$\boldsymbol{x} \in \mathbf{X}^K,\ \boldsymbol{w} \in [0,1]^K.$$

Equations (23b, 23c, 23d, 23e, 23i, 23k, 23n) and (23p) are similar to those in the D–optimal design formulation. Equations (23f-23g) reflect the relationship (21) and generate $\mathscr{U}^{-1}(\xi, \boldsymbol{p})$, equation (23h) captures the constraint (22) to produce $\mathscr{M}^{-1}(\xi, \boldsymbol{p})$ and equations (23k) and (23l), respectively, impose the lower triangular structure of $\mathscr{U}(\xi, \boldsymbol{p})$ and $\mathscr{U}^{-1}(\xi, \boldsymbol{p})$. Equation (23m) imposes the symmetry on $\mathscr{M}^{-1}(\xi, \boldsymbol{p})$ and equations (23i) and (23j), respectively, ensure that the diagonal elements of $\mathscr{U}(\xi, \boldsymbol{p})$ and $\mathscr{U}^{-1}(\xi, \boldsymbol{p})$ are positive. The conditions (23n) and (23o) are the numerical stability insurance for the Cholesky factorization of $\mathscr{M}(\xi, \boldsymbol{p})$ and $\mathscr{M}^{-1}(\xi, \boldsymbol{p})$, respectively. The symmetry of the FIM and its inverse are guaranteed by (23d) and (23m), respectively.

In §A.1 we detail the implementation aspects related with the numerical approach to solve the optimal design problems. All computations in §4 used an Intel Core i7 machine running a 64 bits Windows 10 operating system with a 2.80 GHz processor.

4 Numerical results

We now report locally D– and A–optimal continuous designs for implicit models with the formulations in §3. The optima reported for each design are for $0.5 \times \log[\det(\mathscr{M}(\xi^*, \boldsymbol{p}))]$ and $\mathrm{tr}[\mathscr{M}^{-1}(\xi^*, \boldsymbol{p})]$, where ξ^* are the D– and A–optimal designs at convergence (note the first is a maximizer and the second a minimizer).

We used related theoretical results to suggest an initial value for the number of support points. Specifically, de la Garza (1954) proved that the D–optimal designs for polynomial regression models are minimally supported, i.e. the number of support points is equal to the number of parameters. The same was proved for specific nonlinear forms such as the logistic model and the family of generalized nonlinear logistic functions, see Ford et al. (1992) and Hedayat et al. (1997), respectively. Since our models are of implicit polynomial or exponential nature, we use these guidelines to set the number of support points in the numerical tests. The confirmation of the optimality of the designs obtained is carried out graphically displaying the dispersion function which in turn validates the equivalence theorems (9) and (10). If the number of support points is incorrect, the plot of the dispersion function will show this. In all our examples our initial value of n_θ was correct.

In Section 4.1 we analyze the details of the implementation with a simple example. In §4.2 five additional examples with practical interest are solved.

4.1 A toy example

Here, we use the model

$$\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x}, \boldsymbol{\theta}) := s_1^2 + 2\, s_1 + \theta_1\, x_1 + \exp(-\theta_2\, x_1) = 0, \quad x_1 \in \mathbf{X} \tag{24}$$

to analyze the construction of the FIM and other implementation aspects. We note the model has a single equation ($n_s = 1$), a single response, i.e. $n_y = n_s = 1$ where $\mathbb{E}(y_1) = s_1$, includes a single control factor ($n_x = 1$) and two parameters ($n_\theta = 2$). Here, ε_1 is the observational error affecting the measurements of y_1 characterized in §1 and the design space is $\mathbf{X} \equiv [0, 1]$.

The problem has two possible solutions for s_1 (and y_1) from each value of x_1 in $\mathbf{X}$, one positive and the other negative. We limit the design to non-negative values of s_1 to avoid multiple optimal designs with the same performance. Typically, the optimal design will be supported by two points of the positive branch of s_1 and two points of the negative branch given the symmetry. Thus, four distinct optimal designs might be found by the global optimization solver which reports only one of them containing two of the support points.

The restriction may be removed without any impact on the optimal design obtained but with a resultant increase in CPU time. The locally optimal design is to be determined for $\boldsymbol{\theta} = (-10,\ 0.1)^\intercal$; this set of parameters was chosen so that the problem allows demonstrating the fundamentals of the formulation. The AEs system representing the sensitivity equation (15) for support point k is

$$\begin{pmatrix} 2\,s_1+2 & 0 \\ 0 & 2\,s_1+2 \end{pmatrix} \begin{pmatrix} \sigma_{1,1,k} \\ \sigma_{1,2,k} \end{pmatrix} + \begin{pmatrix} x_1 \\ -x_1\ \exp(-\theta_2\,x_1) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},\ \boldsymbol{x} \in \{\boldsymbol{x}_k :\ k \in [\![K]\!]\}. \quad (25)$$

The local FIMs are then constructed from coefficients $\sigma_{1,l,k},\ l \in [\![n_\theta]\!],\ k \in [\![K]\!]$, employing (17).

To analyze the accuracy of the approach used to find the sensitivity coefficients, we compare the optimal designs for the implicit representation (24) with those for the explicit form of the model, which can be derived in this case,

$$s_1 = -1 + \sqrt{1 - \theta_1\,x_1 - \exp(-\theta_2\,x_1)}, \quad x_1 \in \mathbf{X}. \quad (26)$$

We notice that the optimal designs for (26) were found with the formulations in §3 with the exception of the sensitivity coefficients which were analytically derived and explicitly included in the NLP problems. Specifically, the local FIMs are computed by

$$M(\boldsymbol{x}_k) = \begin{pmatrix} \frac{-x_1}{2\sqrt{1-\theta_1\ x_1-\exp(-\theta_2\ x_1)}} \\ \frac{x_1\ \exp(-\theta_2\ x_1)}{2\sqrt{1-\theta_1\ x_1-\exp(-\theta_2\ x_1)}} \end{pmatrix} \times \begin{pmatrix} \frac{-x_1}{2\sqrt{1-\theta_1\ x_1-\exp(-\theta_2\ x_1)}} & \frac{x_1\ \exp(-\theta_2\ x_1)}{2\sqrt{1-\theta_1\ x_1-\exp(-\theta_2\ x_1)}} \end{pmatrix},$$
$$\boldsymbol{x} \in \{\boldsymbol{x}_k :\ k \in [\![K]\!]\}. \tag{27}$$

The optimal designs found for both the implicit and explicit model forms are shown in Table 1. The matrices containing the optimal designs have the values of the control factor (x_1) for each support point in the first line followed by the values of the response variables (y_1), and finally, in the last line, the weights (w). We note that (i) the designs obtained for implicit and explicit representations are equal, having the same efficiency; (ii) the second point of the designs coincide with the upper bound of the design space; and (iii) the D–optimal design is uniformly distributed (the weight of both points is $1/n_\theta$). The CPU time required by the explicit model is smaller because the size of the NLP problem solving implicit representations is larger, having more variables (i.e., degrees of freedom) and equality constraints.

We now demonstrate the global optimality of the designs found for model (24) by plotting the dispersion functions defined in (9) and (10) for the D– and A–optimality criteria, respectively. These are shown in Figure 1. We note that the dispersion function is limited from above by 0 and attains its maximum at the support points as required for optimal designs. If the restriction on $s_1 > 0$ is removed, two optimal designs with equal efficiency are produced, both sharing the support points but with distinct values of s_1.

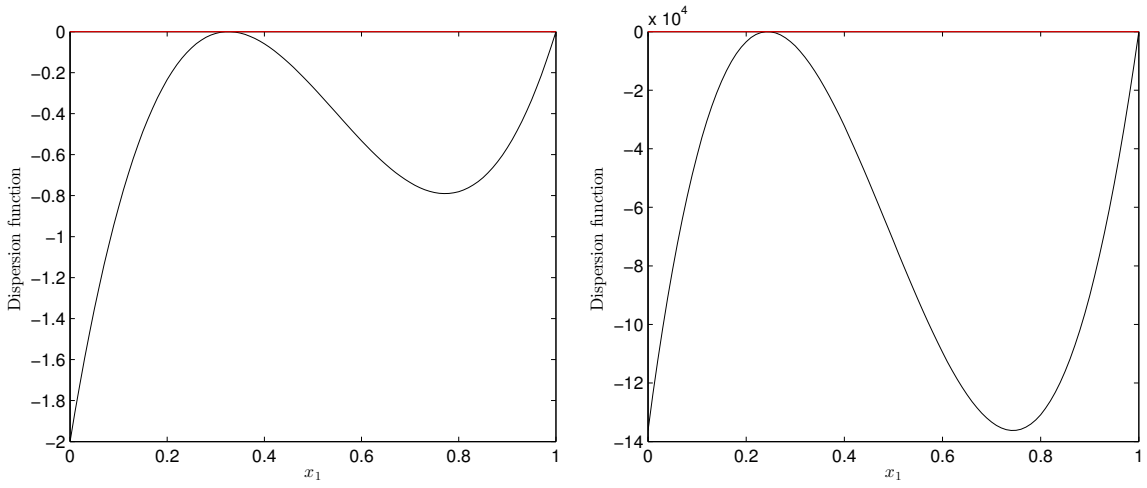
Finally, we compare the performance of the D– and A–optimal designs obtained for (24) using a global solver and a Genetic Algorithm (GA) based solver with the ability to handle constrained

Table 1: Locally optimal continuous designs for implicit (24) and explicit (26) forms of the toy example.

Model	Criterion	Optimal design	Optimum	CPU time (s)
(24)	D-	$\begin{pmatrix} 0.3260 & 1.0000 \\ 0.8143 & 2.1773 \\ 0.5000 & 0.5000 \end{pmatrix}$	-7.7153	6.03
(26)	D-	$\begin{pmatrix} 0.3261 & 1.0000 \\ 0.8147 & 2.1773 \\ 0.5000 & 0.5000 \end{pmatrix}$	-7.7153	5.45
(24)	A-	$\begin{pmatrix} 0.2439 & 1.0000 \\ 0.5693 & 2.1773 \\ 0.6616 & 0.3384 \end{pmatrix}$	1.363×10^5	13.05
(26)	A-	$\begin{pmatrix} 0.2441 & 1.0000 \\ 0.5701 & 2.1773 \\ 0.6618 & 0.3382 \end{pmatrix}$	1.363×10^5	10.78

problems (Conn et al., 1991). The parameters of the GA solver are: (i) a population size of 50 individuals; (ii) a crossover fraction equal to 0.8; (iii) a fraction of individuals kept on the first Pareto front of 0.35; (iv) an absolute tolerance of 1×10^{-6}; and (v) a limit of 10 generations to stop. The optimum obtained with the GA-based solver is -7.8039; consequently, the D–optimal efficiency relative to the design in Table 1 (first line), computed with Eq. (11), is 95.67 %. The CPU time is 869.238 s, two orders of magnitude larger than that in Table 1. This increase is due to the need to solve a system of nonlinear algebraic equations to determine the parameter sensitivities for each individual of the population in the evolutionary procedure.

For the A–optimal design, the optimum obtained with the GA-based solver is 1.5434×10^5, and the A–optimal efficiency relative to the design in Table 1 (third line), obtained via Eq. (12), is 88.31 %. The CPU time is 811.80 s, about 60 times larger than that of the deterministic global optimizer.



(a) (b)

Figure 1: Dispersion functions for: (a) locally D-optimal design; (b) locally A-optimal design for the toy example. Both designs are optimal with $n_\theta = 2$.

Summing up, the comparison shows the advantage, in terms of accuracy and numerical efficiency, of deterministic global optimizers for this kind of problem when such optimizers are combined with a search mechanism that solves the NLP problem from random starting points for the initial values of the experimental variables and the adaption of the region of search to the evolution of the "best" solution found. Therefore, the design problems addressed in §4.2 are solved with a deterministic global optimizer, `OQNLP`.

4.2 Application to realistic examples

We now apply the formulation to five problems from the fields of biomedicine, chemistry and thermodynamics. All examples were chosen to demonstrate that, in practice, implicit models may have application in laboratory experiments. The structure used for tabulating ξ^* is similar

to that employed in the previous example; the first line contains the levels of the control variable, the following n_s the predicted values for the state variables (which include the response), and the last the weight of the corresponding support point. The global optimality of the designs presented in this section was checked plotting the dispersion functions as in §4.1, and they were all demonstrated to be globally optimal. In all the cases, we report the locally optimal designs determined for the set of parameters indicated (or fitted) in the references introducing the respective models.

1. X-ray images of prostheses. The first model considered is the one proposed by Reilly and Patino-Lea1 (1981) for describing the X-ray image of a prosthesis made under conditions such that the X-rays struck the photographic plate in an oblique angle, distorting the image of the spherical ball to an ellipse. We consider that the x-coordinate of the X-rays can be controlled, and that the y-coordinate is related to it by

$$\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := \theta_3\,(s_1-\theta_1)^2 + 2\,\theta_4\,(s_1-\theta_1)\,(x_1-\theta_2) + \theta_5\,(x_1-\theta_2)^2 - 1 = 0, \tag{28}$$

where $n_s = 1$, $n_x = 1$, $n_\theta = 5$ and $\boldsymbol{\theta} = (\theta_1,\ \theta_2,\ \theta_3,\ \theta_4,\ \theta_5)^\intercal$. Here, x_1 is the x-coordinate chosen for taking the image and s_1 is the y-coordinate, with the response variable modeled as $\mathbb{E}(y_1) = s_1$, so $n_y = 1$. The locally optimal designs are to be obtained for $\boldsymbol{\theta} = (-0.99938,\ -2.93105,\ 0.08757,\ 0.01623,\ 0.07975)^\intercal$ and are formed by the set of values of $\boldsymbol{x}$ chosen to maximize the amount of information obtained from measuring y_1. Here, $x_1 \in \mathbf{X} \equiv [-6.0, 0.5]$, and we restrict s_1 so that the values measured (y_1) are in the upper arc of the distorted ellipse which requires the following constraint on the design

$$s_1 \geq -0.1833\,x_1 - 1.5435, \tag{29}$$

to be added to NLP problems (20) and (23). We note the observations are symmetric with respect to the plane (29), and an equivalent design can be obtained if the lower arc is considered, the

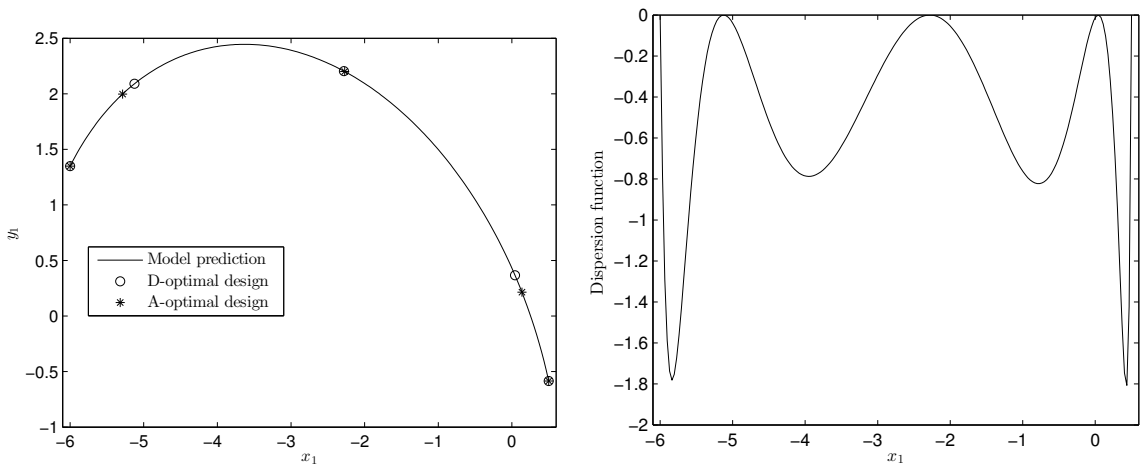
only difference being the symmetry of the x-coordinates. The absence of the constraint produces multiple optimal designs combining points from both arcs, one of them arbitrarily chosen by the solver.

Table 2 presents the optimal designs found and we notice that the D–optimal design is again uniform. Figure 2(a) shows the predicted values of y_1 in the design interval of interest and the support points for both D– and A–optimal designs. The support points of the two designs are close to each other but the weights are substantially different. Figure 2(b) shows the dispersion function for the D–optimal design presented in the first line of Table 2 and proves the design with five support points is D–optimal. Similar checks were made for our remaining examples, but we do not report them here.

Table 2: Locally optimal continuous designs for prosthesis image, model (28), $\boldsymbol{\theta} = (-0.99938,\ -2.93105,\ 0.08757,\ 0.01623,\ 0.07975)^\intercal$, $\mathbf{X} \equiv [-6.0, 0.5]$.

Model	Criterion	Optimal design	Optimum	CPU time (s)
(28)	D-	$\begin{pmatrix} -6.0000 & -5.1250 & -2.2819 & 0.0418 & 0.5000 \\ 1.3486 & 2.0908 & 2.2045 & 0.3665 & -0.5851 \\ 0.2000 & 0.2000 & 0.2000 & 0.2000 & 0.2000 \end{pmatrix}$	4.8998	98.52
(28)	A-	$\begin{pmatrix} -6.0000 & -5.2875 & -2.2706 & 0.1360 & 0.5000 \\ 1.3486 & 1.9973 & 2.2005 & 0.2143 & -0.5851 \\ 0.1268 & 0.2643 & 0.2380 & 0.2571 & 0.1138 \end{pmatrix}$	461.2786	356.13

2. Compressibility of helium at 273.15 K**.** The second model was proposed by Britt and Luecke (1973) to describe the compressibility of helium at 273.15 K. The experimental data were obtained by the Burnett method where the pressure of contained gas at constant temperature is measured before and after expansion into a larger volume. Here, we consider that the pressure



(a) (b)

Figure 2: Prosthesis image, model (28): (a) predicted values and optimal design points; (b) dispersion function for the locally D-optimal design.

before compression is controlled and the pressure after expansion measured. The model is

$$\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := (\theta_1 - \theta_3)\, x_1\, s_1 + (\theta_2\, s_1 - \theta_3\, x_1)\, x_1\, s_1 + x_1 - \theta_3\, s_1 = 0, \tag{30}$$

so that $n_s = 1$, $n_x = 1$ and $n_\theta = 3$, where $\boldsymbol{\theta} = (\theta_1,\ \theta_2,\ \theta_3)^\intercal$. In (30), x_1 is the pressure of the system before expansion (expressed in atm) and s_1 the pressure after expansion (also in atm). We consider that the pressure after expansion is measured (i.e., is the response of the system) and is denoted as y_1 where $\mathbb{E}(y_1) = s_1$ with n_y being set to 1. The locally optimal designs are to be obtained for $\boldsymbol{\theta} = (11.9517622,\ 113.9619475,\ 1.5648810)^\intercal$, and we consider $\mathbf{X} \equiv [20.0, 700.0]$ with the response constrained to $\mathbf{Y} \equiv [0.1, 10.0]$.

The optimal designs found are listed in Table 3 and we note the first support point is constrained by the lower bound of y_1 while the last one is constrained by the upper bound of the control factor. Similarly, the D–optimal design is uniform, and only one support point of the

designs is different.

Table 3: Locally optimal continuous designs for compressibility of helium, model (30); $\boldsymbol{\theta} = (11.9517622,\ 113.9619475,\ 1.5648810)^\intercal$, $\mathbf{X} \equiv [20.0, 700.0]$.

Model	Criterion	Optimal design	Optimum	CPU time (s)
(30)	D-	$\begin{pmatrix} 20.2609 & 72.4753 & 700.0000 \\ 0.1000 & 0.8944 & 9.5201 \\ 0.3333 & 0.3333 & 0.3333 \end{pmatrix}$	-9.8645	13.05
(30)	A-	$\begin{pmatrix} 20.2609 & 58.5581 & 700.0000 \\ 0.1000 & 0.7007 & 9.5201 \\ 0.0991 & 0.8517 & 0.0492 \end{pmatrix}$	3.625×10^6	99.42

3. Shockley's equation for solar cells. The third case is the one-diode model adopted for modeling solar cells. The model includes a photo-generated current source in parallel with a diode, a shunt resistance and a series of resistances modeling the power losses. In practice, the model for different solar cells is fitted through experiments where the voltage (V) is changed and the intensity of the current (I) in the circuit is measured (Silvestre, 2018). The diode is modeled by the Shockley (1949) equation, and the I–V behavior of the solar cell is given by the implicit model

$$\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := s_1 + d_1 \left\{ \exp\left[\frac{d_3}{\theta_2} \left(x_1 + \theta_3\, s_1 \right) \right] - 1 \right\} + d_2 \left(x_1 + \theta_3\, s_1 \right) - \theta_1 = 0, \tag{31}$$

where s_1 is the current intensity, x_1 is the voltage, d_1 and d_2 are parameters of the cell which are estimated from other experiments and d_3 is a constant equal to $q/(k_B\, T)$ where q stands for the charge of the electron, k_B for the Boltzmann constant and T for the absolute temperature of the experiment. For this system we have $n_s = 1$, $n_x = 1$ and $n_\theta = 3$ where $\boldsymbol{\theta} = (\theta_1,\ \theta_2,\ \theta_3)^\intercal$; θ_1 is the photo-generated current, θ_2 is the ideality factor of the diode and θ_3 is the power loss. The

current intensity is the response variable and is represented by y_1 where $\mathbb{E}(y_1) = s_1$ with n_y being set to 1.

Here, we consider the photovoltaic module reported by Sabadus et al. (2017) where $d_1 = 1.243 \times 10^{-7}$, $d_2 = 2.7894 \times 10^{-4}$, and $d_3 = 38.921\,758$ to determine the locally optimal designs for $\boldsymbol{\theta} = (0.146,\ 1.791,\ 0.038)^\intercal$. The design space considered in our study was the one suggested by the authors, i.e. $\mathbf{X} \equiv [0.0,\ 0.65]$ and $\mathbf{Y} \equiv [0.0,\ 0.15]$.

The optimal designs found are listed in Table 4 and we note the first support point is constrained by the lower bound of the design space. The support points of both designs are similar but the weights are quite different.

Table 4: Locally optimal continuous designs for solar cells, model (31), $\boldsymbol{\theta} = (0.146,\ 1.791,\ 0.038)^\intercal$, $\mathbf{X} \equiv [0.0,\ 0.65]$.

Model	Criterion	Optimal design	Optimum	CPU time (s)
(31)	D-	$\begin{pmatrix} 0.0000 & 0.5993 & 0.6349 \\ 0.1460 & 0.0854 & 0.0215 \\ 0.3333 & 0.3333 & 0.3333 \end{pmatrix}$	−4.2972	26.98
(31)	A-	$\begin{pmatrix} 0.0000 & 0.5992 & 0.6349 \\ 0.1460 & 0.0855 & 0.0215 \\ 0.2588 & 0.4950 & 0.2462 \end{pmatrix}$	619.6032	151.80

4. Vapor-Liquid Equilibrium (VLE) characterization. This model describes VLE data obtained experimentally and follows from the assumptions presented in Section 1.1, see Englezos et al. (1990) for consistency analysis. We consider a common isothermal experimental setup where the pressure of the system is to be changed (and controlled) and the composition of component 1 in the mixture is measured in the liquid (L) phase after the equilibrium has been reached.

The model $\boldsymbol{g}(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta})$ is formed by the set of equations

$$g_1(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := s_2\, x_1 - s_1\, \gamma_1(s_1)\, P_1^s(T) = 0 \tag{32a}$$

$$g_2(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := (1-s_2)\, x_1 - (1-s_1)\, \gamma_2(s_1)\, P_2^s(T) = 0 \tag{32b}$$

$$P_i^s(T) = 10^{A_i - B_i/(T+C_i)}, \quad i \in \{1,2\} \tag{32c}$$

$$\log(\gamma_1(s_1)) = (1-s_1)\, \left(\frac{\theta_1}{s_1 + \theta_1\, (1-s_1)} - \frac{\theta_2}{\theta_2\, s_1 + 1 - s_1} \right) - \\ - \log(s_1 + \theta_1\, (1-s_1)) \tag{32d}$$

$$\log(\gamma_2(s_1)) = -s_1\, \left(\frac{\theta_1}{s_1 + \theta_1\, (1-s_1)} - \frac{\theta_2}{\theta_2\, s_1 + 1 - s_1} \right) - \\ - \log(1 - s_1 + \theta_2\, s_1) \tag{32e}$$

where $P_i^s(T)$ stands for the saturation pressure of component i in the mixture ($i \in \{1,2\}$), estimated employing the Antoine equation (32c) and expressed in mm Hg. The state variables are the concentration of component 1 in the L phase, s_1, and in the Vapor (V) phase, s_2. The activity coefficients of both components in the mixture (32d-32e) derived from the Wilson model (Wilson, 1964) are $\gamma_i(s_1)$, $i \in \{1,2\}$. Equations (32a-32b) are the equilibrium relations, T is the temperature expressed in degrees K, A_i, B_i and C_i are Antoine constants and x_1 is the overall pressure at which the measurements are taken, also expressed in mm Hg. The system response is the concentration of component 1 in the L phase, designated as y_1. Here, $n_s = 2$, $n_y = 1$, $n_x = 1$, $n_\theta = 2$ where $\boldsymbol{\theta} = (\theta_1,\ \theta_2)^\intercal$ are the binary interaction parameters between the components in the mixture, and $\mathbb{E}(y_1) = s_1$. It is straightforward to generalize our design procedure to consider measuring the composition of 1 in V phase (s_2), so obtaining two measures per experiment.

For demonstration we consider the binary system formed by methanol (MET) - component (1) - and water (WAT) - component (2). The Antoine constants for both components are in Table 5; we take $T = 312.91$ K. The locally optimal design is found for the binary interaction

parameters between the species $\boldsymbol{\theta} = (0.4139,\ 1.0354)^{\intercal}$. The design region is $\mathbf{X} \equiv [20.0, 760.0]$ and $\mathbf{Y} \equiv [0, 1]$. The optimal design consists of the optimal values of x_1 (pressure) at which the concentration of MET in the L phase is measured after equilibrium is attained, y_1.

Table 5: Antoine constants (Poling et al., 2001) for vapor-liquid equilibrium.

Component	A	B	C
MET (1)	8.08097	1582.27	-34.450
WAT (2)	8.07131	1730.63	-39.724

The optimal designs obtained are shown in Table 6. The designs include the value of x_1 in the first row, the values predicted for s_1 and s_2 (in the second and third rows), and the value of w for each support point.

Table 6: Locally optimal continuous designs for vapor-liquid equilibrium, model (32), $\boldsymbol{\theta} = (0.4139,\ 1.0354)^{\intercal}$, $\mathbf{X} \equiv [20.0, 760.0]$.

Model	Criterion	Optimal design	Optimum	CPU time (s)
(32)	D-	$\begin{pmatrix} 98.2731 & 206.4020 \\ 0.1081 & 0.7067 \\ 0.4985 & 0.8917 \\ 0.5000 & 0.5000 \end{pmatrix}$	-4.2106	4.36
(32)	A-	$\begin{pmatrix} 108.6705 & 196.8287 \\ 0.1436 & 0.6412 \\ 0.5604 & 0.8672 \\ 0.6037 & 0.3963 \end{pmatrix}$	444.0183	12.34

5. Redox reaction titration. In the first three examples in this section we have one state variable, i.e. $n_s = 1$. The fourth example has $n_s = 2$. In our final example we show how the method works for $n_s = 3$ (in all examples $n_x = n_y = 1$). The model represents the titration of redox

reactions where changes in cell potential are used to follow the titration process as a function of the volume of titrant added (Morales, 2002). Here, s_1 is the concentration of oxidant, s_2 the concentration of reductant, s_3 the potential of the cell and the control factor is the volume of reductant added to initial volume of oxidant, x_1. The parameters to be estimated are the equilibrium constant of the redox reaction, θ_1, and the standard cell potential, θ_2. Thus, $\boldsymbol{\theta} = (\theta_1,\ \theta_2)^\intercal$; the response variable is the cell potential, i.e. $\mathbb{E}(y_1) = s_3$. The optimal experimental design is sought by choosing the volumes of titrant at which the system is to be observed so that the amount of information is maximum. The model is:

$$g_1(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := s_1 - \left(\frac{d_1\ d_2 - d_3\ x_1}{d_1 + x_1} + \frac{s_2^2}{\theta_1\ s_1}\right) = 0 \tag{33a}$$

$$g_2(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := s_2 - \left(\frac{d_3\ x_1}{d_1 + x_1} - \frac{s_2^2}{\theta_1\ s_1}\right) = 0 \tag{33b}$$

$$g_3(\boldsymbol{s}|\boldsymbol{x},\boldsymbol{\theta}) := s_3 - \left[\theta_2 + d_4\ \log\left(\frac{s_2}{s_1}\right)\right] = 0, \tag{33c}$$

where d_1 is the initial volume of oxidant, here assumed 50 mL, d_2 is the concentration of oxidant in the solution, $0.1\,\mathrm{mol\,L^{-1}}$, d_3 is the concentration of redox specie in the titrant, also considered $0.1\,\mathrm{mol\,L^{-1}}$, and d_4 is the ratio of $R\ T/F$ where R is the ideal gas constant, T is the absolute temperature and F is the Faraday constant; d_4 at 25 °C is 0.059 V. The optimal design is to be found for $\boldsymbol{\theta} = (0.079,\ 0.700)^\intercal$. The values of constants $d_i,\ i \in [\![4]\!]$, as well as those of the parameters for constructing the locally optimal design are from Morales (2002). The design space is $\mathbf{X} \equiv [1 \times 10^{-2}, 50.0]$ as we limit it to non-zero values to avoid numerical indeterminacies, and impose the positivity of s_1, s_2 and s_3, consistent with the physics of the system. Summing up, this problem is characterized by $n_s = 3$, $n_y = 1$, $n_x = 1$ and $n_\theta = 2$.

The optimal designs obtained are shown in Table 7. The designs include the value of x_1 in the first row, then the values predicted for s_1, s_2 and s_3, and the value of w for each support point. Again, the D–optimal design is uniform, unlike the A–optimal design.

Table 7: Locally optimal continuous designs for redox reaction titration, model (33), $\boldsymbol{\theta} = (0.079,\ 0.700)^{\intercal}$, $\mathbf{X} \equiv [1 \times 10^{-2}, 50.0]$.

Model	Criterion	Optimal design	Optimum	CPU time (s)
(33)	D-	$\begin{pmatrix} 0.0100 & 50.0000 \\ 0.0999 & 0.0390 \\ 2 \times 10^{-5} & 0.0110 \\ 0.1973 & 0.6251 \\ 0.5000 & 0.5000 \end{pmatrix}$	-1.6830	3.48
(33)	A-	$\begin{pmatrix} 0.0100 & 50.0000 \\ 0.0999 & 0.0390 \\ 2 \times 10^{-5} & 0.0110 \\ 0.1973 & 0.6251 \\ 0.5163 & 0.4837 \end{pmatrix}$	30.947	8.86

5 Conclusions

In this paper we have considered the problem of finding continuous optimal experimental designs for algebraic implicit models, and have provided a systematic formulation based on nonlinear programming for their calculation. As far as we know, this is the first paper to address this class of models, which appear in several scientific areas. Our formulation addresses the D– and A–optimality criteria and includes: (i) the generation of the sensitivity coefficients; and (ii) the Cholesky decomposition of the FIM. The first step requires solving the sensitivity equations derived from the chain rule of differentiation, and the second allows optimizing convex functions of the FIM, such as the determinant and the trace. The resulting optimization problem can have multiple local optima, so we use a global optimizer to ensure a global optimum is attained. We have tested our formulation on five examples of practical interest where implicit regression methods were previously applied to data analysis but not to design.

An interesting feature of all D–optimal designs we have found is that the number of support

points is n_θ. This is usually the case for explicit models, especially those that are nonlinear in the parameters. A consequence is that the designs are then uniform, that is, the weight at each support point is $1/n_\theta$. Some comments and a proof are given by Pronzato and Pázman (2013, p.141). Since the result depends solely on the structure of determinants of square matrices, it is not surprising to find that it also holds here for implicit models. However, it cannot be assumed that the result holds for any other model, or even for one of the models exemplified here if the parameter values are changed.

A topic to explore in the future is the extension of the formulations introduced in the present study to exact optimal designs using the ideas of Duarte et al. (2020). This problem has practical interest, examples being the VLE or Liquid-Liquid Equilibrium characterization of mixtures using thermodynamic models. Another limitation of the proposed formulation is that it is focused on locally optimal designs. However, very often the knowledge of the initial values of the parameters is poor. Then, Bayesian or minimax designs which are less inefficient than locally optimal designs for other parameter combinations should be considered. If sequential experimentation is possible, sequential designs, with cycles of optimal experimental design, experimentation and re-estimation of parameters rapidly overcome incorrect prior values for parameters. Box and Hunter (1965) give an example for an algebraic model and designs for VLE characterization provide an instance for implicit algebraic equation models. In some cases of strong uncertainty about prior values of parameters, it may be easier for scientists to predict responses for sufficiently many sets of factor combinations to permit parameter estimation. Bobis and Andersen (1970) describes a sequential design procedure that includes downweighting of the predicted responses as experimental observations become available.

References

Amo-Salas, M., A. Jimenez-Alcázar, and J. López-Fidalgo (2016). Optimal designs for implicit models. In J. Kunert, C. H. Miller, and A. C. Atkinson (Eds.), *mODa 11 - Advances in Model-Oriented Design and Analysis: Proceedings of the 11th International Workshop in Model-Oriented Design and Analysis*, Cham, pp. 11–18. Springer Publishing Company, Incorporated.

Atkinson, A. C. and B. Bogacka (2002). Compound and other optimum designs for systems of nonlinear differential equations arising in chemical kinetics. *Chemometrics and Intelligent Laboratory Systems 61*(1), 17 – 33.

Atkinson, A. C., A. N. Donev, and R. D. Tobias (2007). *Optimum Experimental Designs, with SAS*. Oxford: Oxford University Press.

Balsa-Canto, E., D. Henriques, A. Gábor, and J. R. Banga (2016). AMIGO2, a toolbox for dynamic modeling, optimization and control in systems biology. *Bioinformatics 32*(21), 3357–3359.

Bauer, I., H. G. Bock, S. Körkel, and J. P. Schlöder (2000). Numerical methods for optimum experimental design in DAE systems. *Journal of Computational and Applied Mathematics 120*(1), 1 – 25.

Bobis, A. H. and M. Andersen (1970). An approach for economic discrimination between alternative chemical syntheses. *Technometrics 12*, 439–455.

Box, G. E. P. and W. G. Hunter (1965). Sequential design of experiments for nonlinear models. In *Proceedings IBM Scientific Computing Symposium: Statistics*, pp. 113–137. New York: IBM.

Britt, H. I. and R. H. Luecke (1973). The estimation of parameters in nonlinear, implicit models. *Technometrics 15*(2), 233–247.

Byrd, R. H., M. E. Hribar, and J. Nocedal (1999). An interior point algorithm for large-scale nonlinear programming. *SIAM J. on Optimization 9*(4), 877–900.

Chaloner, K. and K. Larntz (1989). Optimal Bayesian design applied to logistic regression experiments. *Journal of Statistical Planning and Inference 59*, 191–208.

Chen, R.-B., S.-P. Chang, W. Wang, H.-C. Tung, and W. K. Wong (2015). Minimax optimal designs via particle swarm optimization methods. *Statistics and Computing 25*(5), 975–988.

Coleman, T. F. and Y. Li (1994). On the convergence of reflective Newton methods for large-scale nonlinear minimization subject to bounds. *Mathematical Programming 67*(2), 189–224.

Conn, A. R., N. I. M. Gould, and P. L. Toint (1991). A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM J. Numer. Anal. 28*(2), 545–572.

de la Garza, A. (1954). Spacing of information in polynomial regression. *Ann. Math. Statist. 25*(1), 123–130.

Dette, H., A. Pepelyshev, and A. A. Zhigljavsky (2008). Improving updating rules in multiplicative algorithms for computing $D-$optimal designs. *Computational Statistics & Data Analysis 53*(2), 312–320.

Dohnal, V. and D. Fenclová (1985). A new procedure for consistency testing of binary vapour–liquid equilibrium data. *Fluid Phase Equilibria 21*(3), 211 – 235.

Dovì, V. G., A. P. Reverberi, and L. Maga (1993). Optimal design of sequential experiments for error-in-variables models. *Computers & Chemical Engineering 17*(1), 111 – 115.

Draper, N. R. and W. G. Hunter (1966, December). Design of experiments for parameter estimation in multiresponse situations. *Biometrika 53*(3/4), 525–533.

Drud, A. (1985). `CONOPT`: A GRG code for large sparse dynamic nonlinear optimization problems. *Mathematical Programming 31*, 153–191.

Drud, A. (1994). `CONOPT` - A large–scale GRG code. *ORSA Journal on Computing 6*(2), 207–216.

Du Croz, J. and N. Higham (1992). Stability of methods for matrix inversion. *IMA J. Numer. Anal. 12*, 1–19.

Duarte, B. P. M., A. C. Atkinson, J. F. O. Granjo, and N. M. C. Oliveira (2019). Calculating D-optimal designs for compartmental models with a Michaelis–Menten elimination rate. *Journal of Process Control 83*, 88–101.

Duarte, B. P. M., J. F. O. Granjo, and W. K. Wong (2020). Optimal exact designs of experiments via Mixed Integer Nonlinear Programming. *Statistics and Computing 223*, 505–560.

Duarte, B. P. M. and W. K. Wong (2014). A semi-infinite programming based algorithm for finding minimax optimal designs for nonlinear models. *Statistics and Computing 24*(6), 1063–1080.

Duarte, B. P. M. and W. K. Wong (2015). Finding Bayesian optimal designs for nonlinear models: A semidefinite programming-based approach. *International Statistical Review 83*(2), 239–262.

Duarte, B. P. M., W. K. Wong, and A. C. Atkinson (2015). A semi-infinite programming based algorithm for determining $T-$optimum designs for model discrimination. *Journal of Multivariate Analysis 135*, 11 – 24.

Englezos, P., N. Kalogerakis, and P. Bishnoi (1990). Simultaneous regression of binary VLE and VLLE data. *Fluid Phase Equilibria 61*(1), 1 – 15.

Fedorov, V. V. (1972). *Theory of Optimal Experiments*. Academic Press.

Fedorov, V. V. and P. Hackl (1997). *Model-oriented Design of Experiments* (1st. ed.). New York: Springer.

Fellman, J. (1989). An empirical study of a class of iterative searches for optimal designs. *J. Statist. Plann. Inference 21*, 85–92.

Ford, I., B. Torsney, and C. F. J. Wu (1992). The use of a canonical form in the construction of locally optimal designs for non-linear problems. *Journal of the Royal Statistical Society. Series B (Methodological) 54*(2), 569–583.

Franceschini, G. and S. Macchietto (2008). Model-based design of experiments: State of the art. *Chemical Engineering Science 63*, 4846–4872.

Gaivoronski, A. (1986). Linearization methods for optimization of functionals which depend on probability measures. In A. Prékopa and R. J.-B. Wets (Eds.), *Stochastic Programming 84 Part II*, Volume 28 of *Mathematical Programming Studies*, pp. 157–181. Springer Berlin Heidelberg.

Galvanin, F. and F. Bezzo (2018). Chapter 3: Advanced techniques for the optimal design of

experiments in pharmacokinetics. In *Quantitative Systems Pharmacology - Models and Model-Based Systems with Applications*, pp. 65–83. Elsevier.

GAMS Development Corporation (2013). *GAMS - A User's Guide, GAMS Release 24.2.1*. Washington, DC, USA: GAMS Development Corporation.

Gao, J., D. Guan, D. Xu, L. Zhang, and Z. Zhang (2017). Measurement and modeling of liquid–liquid equilibrium for the systems vinyl acetate + acetic acid/ethanol + water at 298.15 and 308.15 K. *Journal of Chemical & Engineering Data 62*(4), 1240–1246.

Gill, P. E., W. Murray, and M. A. Saunders (2005). `SNOPT`: An SQP algorithm for large-scale constrained optimization. *SIAM Rev. 47*(1), 99–131.

Golub, G. and C. Van Loan (2013). *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press.

Harman, R. and T. Jurík (2008). Computing $c-$optimal experimental designs using the Simplex method of linear programming. *Comput. Stat. Data Anal. 53*(2), 247–254.

Harman, R. and L. Pronzato (2007). Improvements on removing non-optimal support points in $D-$optimum design algorithms. *Statistics and Probability Letters 77*, 90–94.

Hedayat, A. S., B. Yan, and J. M. Pezzuto (1997). Modeling and identifying optimum designs for fitting dose-response curves based on raw optical density data. *Journal of the American Statistical Association 92*(439), 1132–1140.

Heredia-Langner, A., D. C. Montgomery, W. M. Carlyle, and C. M. Borror (2004). Model-robust optimal designs: A Genetic Algorithm approach. *Journal of Quality Technology 36*, 263–279.

Hoang, M. D., T. Barz, V. A. Merchan, L. T. Biegler, and H. Arellano-Garcia (2013). Simultaneous solution approach to model-based experimental design. *AIChE Journal 59*(11), 4169–4183.

Hofman, T. and M. Krzyżanowska (1986). Determination of stability constants of complexes from the titration curve by the maximum likelihood principle. *Talanta 33*(10), 851 – 855.

Kiefer, J. (1974). General equivalence theory for optimum design (approximate theory). *Annals of Statistics 2*, 849–879.

Kiefer, J. and J. Wolfowitz (1960). The equivalence of two extremum problem. *Canadian Journal of Mathematics 12*, 363–366.

Körkel, S., E. Kostina, H. G. Bock, and J. P. Schlöder (2004). Numerical methods for optimal control problems in design of robust optimal experiments for nonlinear dynamic processes. *Optimization Methods and Software 19*(3-4), 327–338.

Mandal, A., W. K. Wong, and Y. Yu (2015). Algorithmic searches for optimal designs. In *Handbook of Design and Analysis of Experiments*, Chapter 21, pp. 755–786. New York, NY: CRC Press.

Mandal, S. and B. Torsney (2006). Construction of optimal designs using a clustering approach. *Journal of Statistical Planning and Inference 136*, 1120–1134.

Marshall, S. L. (2003). Generalized least-squares parameter estimation from multiequation implicit models. *AIChE Journal 49*(10), 2577–2594.

Masoudi, E., H. Holling, B. P. M. Duarte, and W. K. Wong (2019). A metaheuristic adaptive

cubature based algorithm to find Bayesian optimal designs for nonlinear models. *Journal of Computational and Graphical Statistics 0*(0), 1–16.

Meyer, R. K. and C. J. Nachtsheim (1995). The coordinate-exchange algorithm for constructing exact optimal experimental designs. *Technometrics 37*, 60–69.

Mitchell, T. J. and F. L. Miller Jr. (1970). Use of design repair to construct designs for special linear models. Technical Report 130-131, Oak Ridge National Laboratory.

Molchanov, I. and S. Zuyev (2002). Steepest descent algorithm in a space of measures. *Statistics and Computing 12*, 115–123.

Morales, D. A. (2002). Mathematical modeling of titration curves. *Journal of Chemometrics 16*(5), 247–260.

Papp, D. (2012). Optimal designs for rational function regression. *Journal of the American Statistical Association 107*, 400–411.

Pázman, A. (1986). *Foundations of Optimum Experimental Design*. Mathematics and its Applications. Springer Netherlands.

Poling, B. E., J. M. Prausnitz, and J. P. O'Connel (2001). *The Properties of Gases and Liquids* (5th. ed.). New York: McGraw-Hill.

Pronzato, L. (2003). Removing non-optimal support points in $D-$optimum design algorithms. *Statistics & Probability Letters 63*(3), 223–228.

Pronzato, L. (2008). Optimal experimental design and some related control problems. *Automatica 44*, 303–325.

Pronzato, L. and A. Pázman (2013). *Design of Experiments in Nonlinear Models*. New York: Springer.

Pukelsheim, F. (1993). *Optimal Design of Experiments*. Philadelphia: SIAM.

Pukelsheim, F. and S. Rieder (1992). Efficient rounding of approximate designs. *Biometrika 79*, 763–770.

Pukelsheim, F. and B. Torsney (1991). Optimal weights for experimental designs on linearly independent support points. *Ann. Statist. 19*(3), 1614–1625.

Quaglio, M., C. Waldron, A. Pankajakshan, E. Cao, A. Gavriilidis, E. S. Fraga, and F. Galvanin (2019). An online reparametrisation approach for robust parameter estimation in automated model identification platforms. *Computers & Chemical Engineering 124*, 270 – 284.

Reilly, P. M. and H. Patino-Lea1 (1981). A Bayesian study of the error-in-variables model. *Technometrics 23*(3), 221–231.

Ruszczyński, A. P. (2006). *Nonlinear Optimization*. Number vol. 13 in Nonlinear optimization. Princeton University Press.

Sabadus, A., V. Mihailetchi, and M. Paulescu (2017). Parameters extraction for the one-diode model of a solar cell. *AIP Conference Proceedings 1916*(1), 040005.

Sachs, W. H. (1976). Implicit multifunctional nonlinear regression analysis. *Technometrics 18*(2), 161–173.

Sagnol, G. (2011). Computing optimal designs of multiresponse experiments reduces to second-order cone programming. *Journal of Statistical Planning and Inference 141*(5), 1684–1708.

Sagnol, G. and R. Harman (2015). Computing exact $D-$optimal designs by mixed integer second order cone programming. *Annals of Statistics 43*(5), 2198–2224.

Seber, G. and C. Wild (2003). *Nonlinear Regression*. New York: John Wiley & Sons.

Shockley, W. (1949). The theory of p-n junctions in semiconductors and p-n junction transistors. *Bell System Technical Journal 28*(3), 435–489.

Silvestre, S. (2018). Chapter 7 - strategies for fault detection and diagnosis of pv systems. In I. Yahyaoui (Ed.), *Advances in Renewable Energies and Power Technologies*, pp. 231 – 255. Elsevier.

Silvey, S. D. (1980). *Optimal Design*. London: Chapman & Hall.

Titterington, D. M. (1976). Algorithms for computing $D-$optimal design on finite design spaces. In *Proc. of the 1976 Conf. on Information Science and Systems*, Volume 3, pp. 213–216. John Hopkins Univ., Baltimore, MD.

Torsney, B. and S. Mandal (2006). Two classes of multiplicative algorithms for constructing optimizing distributions. *Computational Statistics & Data Analysis 51*(3), 1591–1601.

Torsney, B. and R. Martín-Martín (2009). Multiplicative algorithms for computing optimum designs. *Journal of Statistical Planning and Inference 139*(12), 3947 – 3961.

Ugray, Z., L. Lasdon, J. Plummer, F. Glover, J. Kelly, and R. Martí (2005). A multistart scatter search heuristic for smooth NLP and MINLP problems. In *Metaheuristic Optimization via Memory and Evolution*, pp. 25–51. Springer.

Vandenberghe, L. and S. Boyd (1999). Applications of semidefinite programming. *Applied Numerical Mathematics 29*, 283–299.

Whittle, P. (1973). Some general points in the theory of optimal experimental design. *Journal of the Royal Statistical Society, Ser. B 35*, 123–130.

Wilson, G. M. (1964). Vapor-liquid equilibrium. XI. A new expression for the excess free energy of mixing. *Journal of the American Chemical Society 86*(2), 127–130.

Woods, D. C. (2010). Robust designs for binary data: applications of simulated annealing. *Journal of Statistical Computation and Simulation 80*(1), 29–41.

Wu, C.-F. (1978). Some algorithmic aspects of the theory of optimal designs. *Ann. Statist. 6*(6), 1286–1301.

Wu, C.-F. and H. P. Wynn (1978). The convergence of general step-length algorithms for regular optimum design criteria. *Ann. Statist. 6*(6), 1273–1285.

Wynn, H. P. (1970, 10). The sequential generation of $D-$optimum experimental designs. *Ann. Math. Statist. 41*(5), 1655–1664.

Yang, M., S. Biedermann, and E. Tang (2013). On optimal designs for nonlinear models: A general and efficient algorithm. *Journal of the American Statistical Association 108*(504), 1411–1420.

Yu, Y. (2010a). $D-$optimal designs via a cocktail algorithm. *Statistics and Computing 21*(4), 475–481.

Yu, Y. (2010b). Monotonic convergence of a general algorithm for computing optimal designs. *Ann. Statist. 38*(3), 1593–1606.

Yu, Y. (2010c). Strict monotonicity and convergence rate of Titterington's algorithm for computing $D-$optimal designs. *Computational Statistics & Data Analysis 54*, 1419–1425.

Appendix A

A.1 Implementation aspects

The formulations (20) and (23) are coded in the GAMS environment (GAMS Development Corporation, 2013). GAMS is a general modeling system that supports mathematical programming applications in several areas. Upon execution, the code describing the mathematical program is automatically compiled, symbolically transcribed into a set of numerical structures, and all information regarding the gradient and matrix Hessian are generated using the automatic differentiation tool and made available to the solver. We provide a sample of such a code in the Supplementary Material.

The ODoE problems apart, the convexity properties are rather challenging. The calculations require matrix algebra operations embedded in the optimization problems which in turn produce problems with multinomial terms and variables of different scales, that may lead to multiple local optima. To ensure the global optimum is attained we use a global optimizer. Specifically, to determine the optimal design we used a multistart heuristic algorithm-based solver, OQNLP. The algorithm calls an NLP solver from multiple starting points, keeps all the feasible solutions found, and picks the best as the optimal solution of the problem (Ugray et al., 2005). The starting points are computed with a random sampling driver that uses independent normal random variables for initializing each decision variable. Contrarily to deterministic global optimization solvers, OQNLP does not certify that the final solution is a global optimum, but it has been successfully tested on a large set of global optimization problems. To build the initial sampling points the variables need to be bounded, which is what we have since the design space and the region of plausible values are all compact by assumption. The NLP solver called by OQNLP is CONOPT, which in turn uses the Generalized Reduced Gradient (GRG) algorithm (Drud, 1985).

The maximum number of starting points allowed is set to 3000 and the procedure terminates when 100 consecutive NLP solver calls result in an improvement less than 1×10^{-4}. The absolute and relative tolerances of the solver were set equal to 1×10^{-5} and 1×10^{-6}, respectively, with the absolute tolerance being equal to ζ which is the minimum value allowed for the diagonal entries in the FIM or its inverse so that they are positive definite.