Computers and Chemical Engineering 73 (2015) 116–127

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

A combined first-principles and data-driven approach to model building

Alison Cozad[a], Nikolaos V. Sahinidis[a,b,*], David C. Miller[b]

[a] Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, United States
[b] U.S. Department of Energy, National Energy Technology Laboratory, Pittsburgh, PA 15236, United States

ARTICLE INFO

Article history:
Received 11 June 2014
Received in revised form 15 October 2014
Accepted 28 November 2014
Available online 8 December 2014

Keywords:
Regression
Surrogate models
Semi-infinite programming

ABSTRACT

We address a central theme of empirical model building: the incorporation of first-principles information in a data-driven model-building process. By enabling modelers to leverage all available information, regression models can be constructed using measured data along with theory-driven knowledge of response variable bounds, thermodynamic limitations, boundary conditions, and other aspects of system knowledge.

We expand the inclusion of regression constraints beyond intra-parameter relationships to relationships between combinations of predictors and response variables. Since the functional form of these constraints is more intuitive, they can be used to reveal hidden relationships between regression parameters that are not directly available to the modeler. First, we describe classes of *a priori* modeling constraints. Next, we propose a semi-infinite programming approach for the incorporation of these novel constraints. Finally, we detail several application areas and provide extensive computational results.

1. Introduction

Often, modelers must decide between (a) utilizing first-principles models, intuition, *etc.* or (b) constructing surrogate models using empirical data. We propose a combination of these two techniques that augments empirical modeling with first-principles information, intuition, and other *a priori* system characterization techniques to build accurate, physically realizable models. By doing this, we leverage the synergistic effects of empirical data, first-principles derivation, and intuition. Observed data points are often sampled at a premium, incurring costs associated with computational time, raw materials, and/or other resources. Frequently, additional insights provided by system knowledge, intuition, or the application of first-principles analysis are available without additional computational, financial, or other costly resource requirements. Knowledge of a less empirical nature, including limits on the response variables; known relationships between response and predictor variables; and relationships among responses, can be applied in conjunction with experimental data. For example, ensuring the nonnegativity of a modeled geometric length, enforcing a sum-to-one constraint on modeled chemical fractional compositions, and ensuring that derivative bounds obey thermodynamic principles are all practical applications of beneficial nonempirical insights.

We aim to build regression models (U):

$$\text{(U)} \quad \min_{\beta \in \mathcal{A}} \quad g(\beta; x_1, x_2, \ldots, x_N, z_1, z_2, \ldots, z_N)$$

that determine m regression parameters (coefficients) β that minimize a given loss function g (*e.g.*, squared error, regularized error, or an information criterion) over a set of original regression constraints $\mathcal{A}$ based on data points (x_i, z_i), $i = 1 \ldots N$. For conciseness, we will refer to $g(\beta; x_1, x_2, \ldots, x_N, z_1, z_2, \ldots, z_N)$ as $g(\beta)$.

To formally introduce insightful nonempirical information, we would like to enforce the following constraint on a regression problem:

$$\Omega(\mathcal{X}) := \left\{\beta \in \mathbb{R}^m : f\left[x, \hat{z}(x;\beta)\right] \leq 0, \quad x \in \mathcal{X}\right\} \tag{1}$$

where function f is a constraint in the space of the predictor(s) x and modeled response(s) $\hat{z}$, and $\mathcal{X}$ is a nonempty subset of $\mathbb{R}^n$. Eq. (1) can be used to reduce the feasible region $\mathcal{A}$ for any general regression analysis formulation: linear least squares, nonlinear least squares, regularized regression, best subset methods, and other characterization techniques. In fact, these constraints can be used alongside current gray-box or semi-physical modeling techniques (Nelles,

[*] Corresponding author at: Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, United States. Tel.: +1 412 2683338; fax: +1 412 268 7139.
E-mail address: sahinidis@cmu.edu (N.V. Sahinidis).

http://dx.doi.org/10.1016/j.compchemeng.2014.11.010
0098-1354/© 2014 Elsevier Ltd. All rights reserved.

2001; Pearson and Pottmann, 2000), where a balance between first principles knowledge and empirical data is desirable – typically, where model structure is chosen from system knowledge and parameters are selected to match sampled data.

By incorporating system knowledge beyond sampled data, we refine the feasible domain as the intersection of $\mathcal{A}$ and Ω and solve problem (C):

$$\text{(C)} \quad \min_{\beta \in \mathcal{A} \cap \Omega(\mathcal{X})} \quad g(\beta)$$

where Ω is defined over the domain $x \in \mathcal{X}$ while the original regression problem (U) exists in the space $\beta \in \mathcal{A}$.

Rao (1965) and Bard (1974) were the first to use *a priori* parameter relationships in regression through simple equality constraints. Recently, the use of such relationships has expanded to include inequality constraints in the space of the regression parameters, a case that arises more naturally in practice (Knopov and Korkhin, 2011). Inequality relationships between regression parameters have been applied to both linear and nonlinear least squares problems in the fields of statistics (Judge and Takayama, 1966; Liew, 1976), economics (Thompson, 1982; Rezk, 1976), and engineering (Gibbons and McDonald, 1999). Most notably, Korkhin has investigated the properties of simple parameter restrictions (Korkhin, 1985, 2002, 2005), nonlinear parameter restrictions (Korkhin, 1998, 1999), and, more recently, the formulation of inequality constraints with deterministic and stochastic right-hand sides (Korkhin, 2013).

Previous work employs *a priori* knowledge to reveal relationships between subsets of regression parameters that serve to restrict their range. To the best of our knowledge, there has been no investigation into the enforcement of *a priori* information that directly relates predictors to regressors. We aim to use these novel relationships between predictors and regressors to restrict the feasible region in the original problem space.

Since previous applications of constrained regression have been restricted to the parameter space β of the regression problem, these techniques are inherently specific to the functional form of the response. For example, consider a quadratic response model, $\hat{z}(x) = \beta_0 + \beta_1 x + \beta_2 x^2$, and the *a priori* insight $\beta_1 \geq \beta_2$. If an exponential function, $\hat{z}(x) = \beta_0 + \beta_1 \exp(x)$, produces a more favorable fit, there is no standard way to translate constraints from the quadratic to the exponential model. On the other hand, enforcing a lower bound on the response, $\hat{z}(x) \geq 0 \quad \forall x$, rather than the β-space, produces a constraint that is independent of the model's functional form. Additionally, system insight in the x-domain may be more intuitive and readily available than knowledge of a unique and contrived regression model's functional form.

A complication arises from the realization that Eq. (1) is valid for the full problem space and, therefore, needs to be enforced for every point $x \in \mathcal{X}$, *i.e.*, at infinitely many points. Semi-infinite programming (SIP) problems are optimization models that have finitely many variables and infinitely many constraints (Reemtsen and Rückmann, 1998). These problem formulations are common in the fields of approximation theory, optimal control, and eigenvalue computations, among others. In each case, one or more parametric constraints result in one constraint for each value of an optimization parameter (in this case x) that varies within its given domain (Hettich and Kortanek, 1993; Reemtsen and Rückmann, 1998).

The first significant work on SIP, by John (1948), provides necessary and sufficient conditions for the solution to a semi-infinite program. Initially, SIP research focused on linear and convex nonlinear semi-infinite programming (Hettich and Kortanek, 1993; Reemtsen and Rückmann, 1998; Goberna and López, 2002). More recently, advances in global optimization, including BARON (Tawarmalani and Sahinidis, 2005), have made the solution of general nonconvex SIP problems more tractable (Chang and Sahinidis, 2011). In problem (C), the objective is often convex, as is the case for linear least squares regression. However, the feasible region – as we show in Section 4 – is generally nonlinear and nonconvex. The key to solving an SIP problem, independent of the solution method, is the optimization of $\max_{x \in \mathcal{X}} \; f\left[x, \hat{z}(x;\beta)\right]$ to locate the maximum violation. This subproblem is significant because $\beta \in \Omega(\mathcal{X})$ if and only if $\max_{x \in \mathcal{X}} f\left[x, \hat{z}(x;\beta)\right] \leq 0$ (Reemtsen and Görner, 1998).

To assess the benefits afforded by augmenting standard regression problems (U) with *a priori* information as in (C), we utilize the test platform ALAMO (Cozad et al., 2014). ALAMO is a software package designed to generate models that are as accurate and as simple as possible. This combination of accuracy and simplicity is well-suited for regression.

The remainder of the paper is organized as follows. In Section 2, we outline the modeling and sampling methods of the ALAMO test platform. We propose a methodology to solve problem (C) in its most general form in Section 3. In Section 4, we detail classes of applications of domain-constrained regression using our solution strategy: restricting individual and multiple responses, constraining response model derivatives, and expanding or contracting the enforcement domain. In Section 5, we demonstrate the efficacy of our approach using numerical examples. Next, in Section 6, we present extensive computational results demonstrating the effectiveness of the proposed methods. Finally, we offer conclusions in Section 7.

2. ALAMO

ALAMO is a learning software that identifies simple, accurate surrogate models using a minimal set of sample points from blackbox emulators such as experiments, simulations, and legacy code. ALAMO initially builds a low-complexity surrogate model using a best subset technique that leverages a mixed-integer programming formulation to consider a large number of potential functional forms. The model is subsequently tested, exploited, and improved through the use of derivative-free optimization solvers that adaptively sample new simulation or experimental points. For more information about ALAMO, see Cozad et al. (2014).

In this section, we detail relevant ALAMO model-building methods as applied to parametric regression. The functional form of a regression model is assumed to be unknown to ALAMO. Instead, ALAMO poses a simple set of basis functions, *e.g.*, x, x^2, $1/x$, $\log(x)$, and a constant term. Once a set of potential basis functions is collected, ALAMO attempts to construct the lowest complexity function that accurately models sample data. To do this, a mixed-integer quadratic program (MIQP) is solved to select basis functions for increasing model complexity. In a solution of the MIQP, the simple basis functions, $X_j(x), j \in \mathcal{B}$, are active when the corresponding binary variable $y_j = 1$ and inactive when $y_j = 0$. The size of the model, specified by a parameter T corresponding to the number of active binary variables, is increased until a goodness-of-fit measure, such as the corrected Akaike Information Criterion (Hurvich and Tsai, 1993), worsens with an increase in model size. As an example, using the list of basis functions given above, the MIQP is as follows:

$$\begin{aligned}
\text{(M)} \quad \min \quad & g(\beta) = \sum_{i=1}^{N} \left(z_i - \left[\beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 \frac{1}{x} + \beta_4 \log(x) \right] \right)^2 \\
\text{s.t.} \quad & \beta_j^{\text{lo}} y_j \leq \beta_j \leq \beta_j^{\text{up}} y_j \qquad j = 0, \ldots, 4 \\
& y_0 + y_1 + y_2 + y_3 + y_4 = T \\
& y_j \in \{0, 1\} \qquad j = 0, \ldots, 4
\end{aligned}$$

While a typical basis set is often far larger, this simple example illustrates the form of the objective $g(\beta)$ and original constraint set $\beta \in \mathcal{A}$ before intersection with new *a priori* constraints $\Omega(\mathcal{X})$.

Once a model has been identified, it is improved systematically in ALAMO through the use of an adaptive sampling technique that adds new simulation or experimental points to the training set. New sample points are selected to maximize model inconsistency in the original design space $x \in \mathcal{X}$, as defined by box constraints on x, using derivative-free optimization methods (Rios and Sahinidis, 2013).

3. Proposed methodology

In this section, we outline a numerical solution method for the semi-infinite constrained regression problem (C) and demonstrate key algorithmic insights using an illustrative example. Problem (C) is solved by adapting and applying a standard, two-phase general SIP method. For details on this generalized SIP approach, see Reemtsen and Görner (1998).

In Phase I, we solve a relaxation of (C), where the parametric constraint is enforced over a finite subset $x \in \mathcal{X}^l \subset \mathcal{X}$:

$$\text{(PI)} \quad \min_{\beta \in \mathcal{A} \cap \Omega(\mathcal{X}^l)} \quad g(\beta)$$

By solving (PI), we find an approximation of the regression parameters β^l over the relaxed feasible region defined as

$$\Omega(\mathcal{X}^l) := \{\beta \in \mathbb{R}^m : f[x, \hat{z}(x; \beta)] \le 0, \quad x \in \mathcal{X}^l\} \tag{2}$$

During Phase II, we solve the maximum violation problem:

$$\text{(PII)} \quad \max_{x \in \mathcal{X}} \quad f(x, \hat{z}(x; \beta^l))$$

After (PI) is solved, its solution point β^l is used to solve (PII). If the solution x^l to (PII) satisfies $f(x^l) \le 0$, the method terminates; otherwise, an updated feasible set $\mathcal{X}^l = \mathcal{X}^l \cup x^l$ is used to repeat Phase I. In general, (PI) will preserve the convexity of the original regression problem (U), though several specific exceptions to this observation are listed in Section 4. For most linear regression problems, the feasible region $\Omega(\mathcal{X}^l)$ is linear. (PII), however, is generally nonconvex and necessitates the use of a global optimization solver.

Each phase will increase in complexity with an increase of problem size. The Phase I problem increases in complexity as the number of basis functions increases – increased flexibility in the functional form of the model imparts greater resource requirements during model selection. The Phase II problem is solved in the space of the original problem variables; as such, the Phase II problem becomes more resource intensive with an increase in the number of predictor variables, x. In the past, the global solution of large nonconvex optimization problems such as (PII) was intractable. Modern global optimization solvers, however, are capable of handling several thousands of variables (Sahinidis, 2014).

˜Zaković and Rustem (2002) solve (PII) to find (a) the maximum violation and (b) any feasible violation. In both cases, they employ a global search strategy using a multistart local optimization approach. For (b), the complete solution of (PII) is required before feasibility to the original problem (C) can be guaranteed. In their findings, (a) requires fewer iterations, while (b) results in a significant reduction of computational effort. Consequently, the authors indicate that (b) is the superior solution method. We propose a method that is similar to (b), but we employ a branch-and-reduce optimization solver – instead of a multi-start local search – to guarantee rigorous global optimality. Moreover, we seek to locate several isolated feasible solutions to (PII) during Phase II with the aim of reducing the total number of constrained regression iterations.

We begin the algorithm with either an empty feasible set $\mathcal{X}^0 = \emptyset$ or some nonempty set of feasible points, for example those selected by a design of experiments (*e.g.*, random sampling, Latin-hypercube sampling (McKay et al., 1979), or factorial design (Simpson et al., 2001)). Next, we solve (PI) to find an initial approximation for β^0. Using the built-in functionality of the global optimizer BARON, we use $\beta^l = \beta^0$ to locate up to n_{viol} isolated feasible points of the following problem:

$$\begin{aligned} (\text{PII}_{\text{feas}}) \quad & \max_{x \in \mathcal{X}} \quad f(x, \hat{z}(x; \beta^l)) \\ & \text{s.t.} \quad f(x, \hat{z}(x; \beta^l)) - \epsilon_{\text{viol}} \ge 0 \end{aligned}$$

Often, feasible solutions for continuous optimization problems are located sufficiently close enough as to be nearly identical. We ensure that our feasible points are not redundant by selecting isolated feasible solutions such that $\|x_i^0 - x_{i'}^0\|_\infty \ge \epsilon_{\text{isol}}$ for every pair of points i and i' (Sahinidis, 2014). By solving (PII_{feas}) using an objective function that reflects the magnitude of violation, we enable BARON to locate a set of isolated feasible points with comparatively large, ranked violations. We also exclude points with $f(x, \hat{z}(x; \beta^l)) = 0$ that are feasible to the original problem (C) by using a small number ϵ_{viol} to ensure a strict violation.

After updating $\mathcal{X}^{l+1} = \mathcal{X}^l \cup x^l$, (PI) is solved again with an updated feasible region $\Omega(\mathcal{X}^{l+1})$. We proceed until it can be shown that the current iteration's parameters, β^l, are both optimal and feasible for (C). This is true if and only if β^l is the solution to (PI) and $\max_{x \in \mathcal{X}} f(x, \hat{z}(x; \beta^l)) \le 0$. An outline of the proposed semi-infinite constrained regression algorithm is included in Algorithm 1.

Algorithm 1. Solve semi-infinite constrained regression problem

```
Given a training set of dependent and independent data points $[x_{id}, z_{ik}]$ for $i = 1, \ldots, N$, $d = 1, \ldots, n$, $k = 1, \ldots, m$; requested feasible points $n_{\text{viol}}$; and relevant tolerance values

Initialize $l = 0$ and $\mathcal{X}^0$ as $\emptyset$ or by using a design of experiments
while ($f(x^l) > \epsilon$) or ($l < 1$) do
    $\beta^l \leftarrow$ solve (PI)
    $x^l, f(x^l) \leftarrow$ find $n_{\text{viol}}$ isolated feasible points for ($\text{PII}_{\text{feas}}$)
    if $x^l == \emptyset$ then
        $\beta^l$ is both feasible and optimal to (C)
        break
    else
        Update feasible set, $\mathcal{X}^{l+1} \leftarrow \mathcal{X}^l \cup x^l$
    end if
    $l \leftarrow l + 1$
end while
```

3.1. Illustrative example

To detail the steps involved in the proposed technique, we construct a model for $z = x^5$ using a fixed functional form $\hat{z}(x) = \beta_1 x + \beta_2 x^3$ over $0 \le x \le 1$. The true function is nonnegative over this region; therefore, we pose the following constrained regression problem:

$$\begin{aligned} \min_{\beta_1, \beta_2} \quad & \sum_{i=1}^{4} (z_i - [\beta_1 x + \beta_2 x^3])^2 \\ \text{s.t.} \quad & \beta_1 x + \beta_2 x^3 \ge 0 \qquad x \in [0, 1] \end{aligned}$$

where model parameters β_1^l and β_2^l are selected to minimize model error over four data points while ensuring nonnegativity in the response model. An ordinary least squares regression objective for a regression model with a fixed functional form is used to illustrate this example more concisely. The discretized Phase I problem:

$$\begin{aligned} \min_{\beta_1, \beta_2} \quad & \sum_{i=1}^{4} (z_i - [\beta_1 x + \beta_2 x^3])^2 \\ \text{s.t.} \quad & \beta_1 x + \beta_2 x^3 \ge 0 \qquad x \in \mathcal{X}^l \end{aligned}$$

Table 1
Surrogate models and errors for the illustrative example.

Surrogate models		Training error, 10^{-3}	Test error, 10^{-3}
Constrained regression			
Phase I (l = 1)	$\hat{z}(x) = -0.308\,x + 1.30\,x^3$	19.1	0.172
Phase I (l = 2)	$\hat{z}(x) = -0.588\,x + 1.02\,x^3$	34.4	0.0303
Final	$\hat{z}(x) = 0.000\,x + 0.954\,x^3$	40.2	0.00353
Ordinary least squares regression			
Final	$\hat{z}(x) = -0.308\,x + 1.30\,x^3$	19.1	0.172

is formulated by enforcing the parametric constraint for discrete values of x in the feasible set $\mathcal{X}^l$ for constrained regression iteration l. We use the following Phase II problem:

$$\begin{aligned} \min \quad & \beta_1^l\, x + \beta_2^l\, x^3 \\ \text{s.t.} \quad & \beta_1^l\, x + \beta_2^l\, x^3 \le 0 - \epsilon \\ & 0 \le x \le 1 \end{aligned}$$

to maximize the nonnegativity violation, where feasible solutions to this problem x^l are added to the feasible set before the Phase I problem is resolved. If the maximum violation problem is infeasible for iteration l, the model parameters β_1^l and β_2^l are feasible and optimal to the semi-infinite problem.

Table 1 shows the regression models from each Phase I solution. Since the initial feasible set $\mathcal{X}^l$ is empty, the Phase I model for l = 1 matches the ordinary least squares, *i.e.*, the unconstrained regression model. The training and test errors are included in Table 1. Here, the training error is calculated using the four training points and the test error is calculated using 1000 evenly distributed sample points. For all results, unless stated otherwise, all errors are calculated using the root mean squared error between a given data set and a model. During each iteration, the newly imposed constraints worsen the objective or training error by restricting the feasible space. Despite this degradation of the training data objective function, the added constraints result in improved test error during each iteration. This is because the squared error objective is an approximation of the true error.

Here, we introduce an error comparison metric representing the error factor calculated for method m using the following equation:

$$EF_m = \frac{\text{RMSE}_m}{\text{RMSE}_{\text{best}}} \qquad (3)$$

where RMSE_m is the root mean squared error for the method of interest and $\text{RMSE}_{\text{best}}$ is the error of the best solution for a given modeling problem. Therefore, the best modeling method has an EF = 1 while larger values quantify the inferior solutions exhibited by other methods. The training error factors for the unconstrained and the constrained problem of the above example are 1 and 2.11, respectively. This shows that the unconstrained problem is more accurate on the training data set. In contrast, the test error factors for the unconstrained and constrained problems are 48.9 and 1, which shows that the constrained model is far more effective at predicting the values of unknown samples.

For each Phase II iteration, we add two points to the feasible set. These points are listed in Table 2. By adding multiple feasible points per round, we avoid unnecessary Phase I solutions during trial model generation. This is significant because the solution of Phase I is often resource intensive.

To illustrate the form of the Phase I and Phase II problems, an instance of each problem corresponding to iteration 2 is included below.

Table 2
Feasible set for illustrative example.

Constrained regression iteration, l	Point added to feasible set, x^l	Modeled value, $\hat{z}(x^l)$
1	0.240	−0.0560
	0.281	−0.0578[a]
2	0.120	−0.0053
	0.138	−0.0054[a]
3	Bounds guranteed	

[a] Optimal solution.

Phase I (l = 2):

$$\begin{aligned} \min_{\beta_1,\beta_2} \quad & \sum_{i=1}^{4} (z_i - [\beta_1\, x + \beta_2\, x^3])^2 \\ \text{s.t.} \quad & 0.240\,\beta_1 + 0.0138\,\beta_2 \ge 0 \\ & 0.281\,\beta_1 + 0.0223\,\beta_2 \ge 0 \end{aligned}$$

Phase II (l = 2):

$$\begin{aligned} \min_{x} \quad & -0.588\,x + 1.02\,x^3 \\ \text{s.t.} \quad & -0.588\,x + 1.02\,x^3 \le 0 - \epsilon \\ & 0 \le x \le 1 \end{aligned}$$

Increasing the size of the feasible set allows β_1^l and β_2^l to move closer to the feasible space defined by $\beta_1\, x + \beta_2\, x^3 \ge 0$. This feasible region as well as the constrained parameter solutions are depicted in Fig. 1 to illustrate discretized solution improvement with respect to feasibility of the semi-infinite problem. The true function, ordinary least squares regression model, and constrained regression model are plotted with the training data and feasible set in Fig. 2.

By constraining the regression model in the space of the predictor and response variables using freely available *a priori* information, we are able to infer constraining relationships in the β-space that generate physically realizable models that better

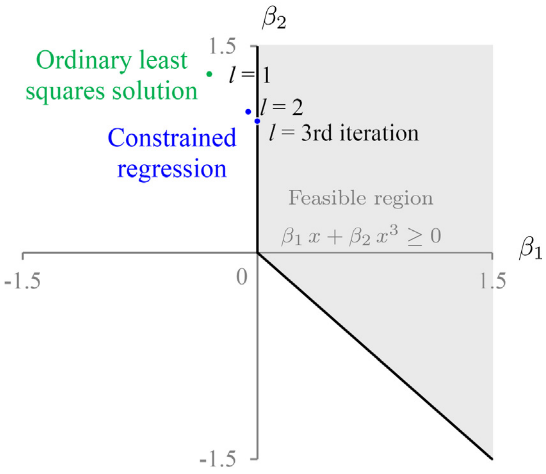

Fig. 1. Feasible region of the constrained regression problem in β-space.

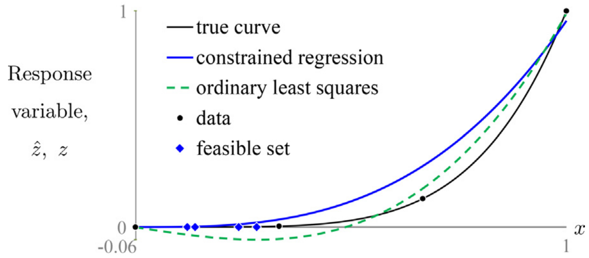

Fig. 2. Model results and true function for $z = x^5$.

predict the response surface. In practice, the constrained regression solution from this simple example would be further refined by re-optimization after subsequent model selection steps.

4. Classes of constrained regression in the *x*- and *z*-domains

In this section, we describe several classes of problems with structures that benefit naturally from constraining the original problem space. Initially, we consider constraining individual responses – a direct application of the methods discussed in the previous section. Next, we extend the proposed methods to enforce relationships among several response variables. Finally, we discuss two specialized applications: restricting response derivatives and an expansion and contraction of the enforcement domain.

4.1. Restricting individual responses

Constraints placed independently on individual response variables result in the semi-infinite feasible region:

$$\Omega(\mathcal{X}) := \{\beta \in \mathbb{R}^m : a\,\hat{z}(x;\beta) + h(x) \le 0, \quad x \in \mathcal{X}\} \tag{4}$$

where h is a function of the predictors x and the coefficient $a \in \mathbb{R}$ effectively scales the response model. We classify constraints of this form by the order of $h(x)$. Zero-order constraints can be used to enforce upper and lower limits on $\hat{z}$; while higher-order constraints can be used to enforce complex restrictions on the feasible region of each response model. As in Eq. (4), the discretized Phase I constraints for this problem are linear in β for linear regression. In this section, we restrict our investigation to constraints that are linear in $\hat{z}$.

4.1.1. Zero-order restrictions

We begin by discussing an implementation of the proposed methodology on a restriction of the response model *via* upper and lower bounds. The most common use of this type of bounding is the enforcement of nonnegativity. Examples of nonnegative response variables include flow rates, absolute pressures, and geometric dimensions, among others. In addition, many other response variables have natural lower and/or upper bounds. For example, compositions and probability distributions range from 0 to 1, while logistic functions have asymptotic limits.

To enforce these *a priori* limits, we impose $\hat{z}(x;\beta) - z^{\text{up}} \le 0$ for the upper bound, z^{up}, and $z^{\text{lo}} - \hat{z}(x;\beta) \le 0$ for the lower bound, z^{lo}. Computational results for combinations of upper and lower bounds are included in Section 6. The application of intuitive bounds on response variables results in the following Phase I and Phase II problem formulations for both lower and upper bounds.

$$(\text{PI}^{\text{bnd}}) \quad \begin{aligned} \min_{\beta \in \mathcal{A}} \quad & g(\beta) & \\ \text{s.t.} \quad & \hat{z}(x;\beta) \le z^{\text{up}} & x \in \mathcal{X}^l \\ & \hat{z}(x;\beta) \ge z^{\text{lo}} & x \in \mathcal{X}^l \end{aligned}$$

$$(\text{PII}^{\text{lobnd}}_{\text{feas}}) \quad \begin{aligned} \min_{x} \quad & \hat{z}(x;\beta^l) \\ \text{s.t.} \quad & \hat{z}(x;\beta^l) \le z^{\text{lo}} - \epsilon_{\text{viol}} \\ & x \in [x^{\text{lo}}, x^{\text{up}}] \end{aligned}$$

$$(\text{PII}^{\text{upbnd}}_{\text{feas}}) \quad \begin{aligned} \max_{x} \quad & \hat{z}(x;\beta^l) \\ \text{s.t.} \quad & \hat{z}(x;\beta^l) \ge z^{\text{up}} + \epsilon_{\text{viol}} \\ & x \in [x^{\text{lo}}, x^{\text{up}}] \end{aligned}$$

where $\mathcal{X} = [x^{\text{lo}}, x^{\text{up}}]$ defines the problem space of the original predictors for any given upper and lower bounds on x. Example 1 in Section 5 provides a numerical demonstration of the enforcement of constraints of this type.

4.1.2. Nonzero-order restrictions

If nonconstant restrictions on a response variable are known, they can be applied in a similar manner. These constraints rely more heavily upon the modeler's system knowledge than simple bounds, but they can lead to increased modeling accuracy and robustness. Linear and nonlinear constraints of this form may come from initial and boundary conditions, mass and energy balances, and problem limits.

Often, knowledge exists for a system that is simpler than the system of interest. Often, that simple system represents a limit or worst-case scenario. In these cases, the simple system can be used to bound the model. For example, for a heat transfer system we can bound heat duty using information from a Carnot engine. If a simple model is available for a reactive system with no byproducts, that model can be used as a lower bound for certain concentrations. As long as the simple system represents a theoretical limit and not an approximation, it can be used to bound the output.

4.2. Restricting multiple responses

Additional relationships between response variables, z_k, $k = 1, 2, \ldots, n_{\text{resp}}$, may also be available to a modeler. One example of such a relationship is a mass or energy balance involving inlet and outlet flows where two or more flows are response models. Another example is the modeling of discretized state variables, such as a fluid velocity profile in a tube, that results in an intrinsic order of modeled variable values of the form: $\hat{z}_{k'} \le \hat{z}_{k>k'}$. The feasible region for a simultaneous restriction of multiple response variables can be described as

$$\Omega(\mathcal{X}) := \{\beta \in \mathbb{R}^m : d(\hat{z}(x;\beta)) + h(x) \le 0, \quad x \in \mathcal{X}\} \tag{5}$$

where d provides a function for the relationship among all responses $k = 1, 2, \ldots, n_{\text{resp}}$.

In the general case, the solution of this problem requires the simultaneous solution of all response models. As a result, this formulation is not compatible with ALAMO, which relies on efficiencies gained by independent treatment of model output variables. In the following subsections, we demonstrate a solution method for the general case and describe an adaptation of this method for the ALAMO framework.

4.2.1. *General case*

As mentioned above, a solution for the general case involves the regression of all response models simultaneously using ($\mathrm{PI}^{\mathrm{mult}}$):

$$(\mathrm{PI}^{\mathrm{mult}}) \qquad \min_{\beta \in \mathcal{A} \cap \Omega(\mathcal{X}^l)} \qquad g^{\mathrm{mult}}(\beta_k)$$

where g^{mult} is the objective of the simultaneous regression problem. The algorithm is shown in Fig. 3.

The fitting objective for the weighted linear least squares regression, using weighting factors, w_K, results in the following formulation and fits each output variable simultaneously:

$$g^{\mathrm{mult}}(\beta_k) = \sum_{k=1}^{n_{\mathrm{resp}}} w_k \sum_{i=1}^{N} \left(z_{ki} - \sum_{j \in \mathcal{B}} \beta_{kj} X_{ij} \right)^2 \qquad (6)$$

4.2.2. *Adaptation for ALAMO*

The ALAMO package does not require fixed functional forms for response variables. As a result, we can use ALAMO to enforce Eq. (5) with a more flexible functional form. Using the ALAMO framework, we solve for each response $z_{k'}$ independently, using the previous iteration's, $l-1$, surrogate models for $z_{k \neq k'}$. The resulting Phase I and Phase II problems are as follows:

$$\begin{aligned} (\mathrm{PI}^{\mathrm{mult}}_{k'}) \quad & \min_{\beta_{k'} \in \mathcal{A}_{k'}} && g(\beta_{k'}) \\ & \text{s.t.} && d(\hat{z}_{k'}(x;\beta_{k'}), \hat{z}_{k \neq k'}(x;\beta_k^{l-1})) + h(x) \leq 0 \quad x \in \mathcal{X}^l \end{aligned}$$

$$\begin{aligned} (\mathrm{PII}^{\mathrm{mult}}_{\mathrm{feas}}) \quad & \max_{x} && d(\hat{z}_k(x;\beta_k^l)) + h(x) \\ & \text{s.t.} && x \in [x^{\mathrm{lo}}, x^{\mathrm{up}}] \end{aligned}$$

where the forms of g_k and $\mathcal{A}_k$ are given by (A) for response variables $k = 1, 2, \ldots, n_{\mathrm{resp}}$. In this case, Phase I requires the solution of ($\mathrm{PI}^{\mathrm{mult}}_{k'}$) for each k. After the regression of each individual response variable, it is possible that $\beta_k^l \notin \Omega(\mathcal{X}^l)$ for $k = 1, 2, \ldots, n_{\mathrm{resp}}$, since ($\mathrm{PI}^{\mathrm{mult}}_{k'}$) is solved using previous models for responses $k \neq k'$. To ensure the feasibility of the resulting model combination after solving ($\mathrm{PI}^{\mathrm{mult}}_{k'}$) for each k, we fix the functional form of each response and solve ($\mathrm{PII}^{\mathrm{mult}}_{\mathrm{feas}}$) using linear least squares regression, as for the general case, using the objective from Eq. (6). This approach is outlined in Fig. 4.

Using the approach in Fig. 4, we ensure that $\beta_k^l \in \Omega(\mathcal{X}^l)$ at the end of Phase I and supply the resulting feasible solution to Phase II.

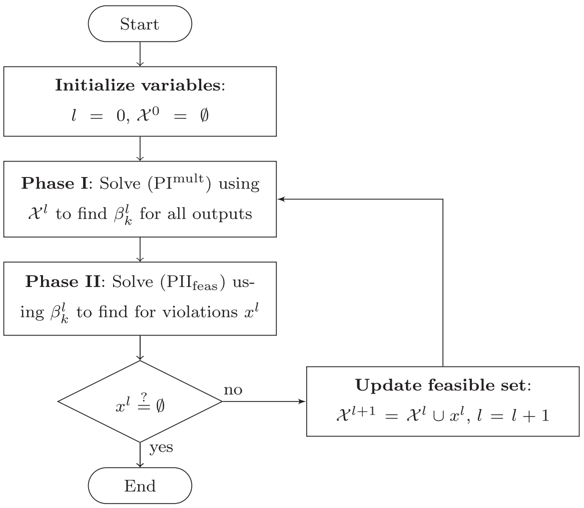
Start
Initialize variables:
$l = 0, \mathcal{X}^0 = \emptyset$
Phase I: Solve ($\mathrm{PI}^{\mathrm{mult}}$) using
$\mathcal{X}^l$ to find β_k^l for all outputs
Phase II: Solve ($\mathrm{PII}_{\mathrm{feas}}$) using β_k^l to find for violations x^l
$x^l \overset{?}{=} \emptyset$
no
Update feasible set:
$\mathcal{X}^{l+1} = \mathcal{X}^l \cup x^l, l = l+1$
yes
End

Fig. 3. Restricting multiple responses for the general case.

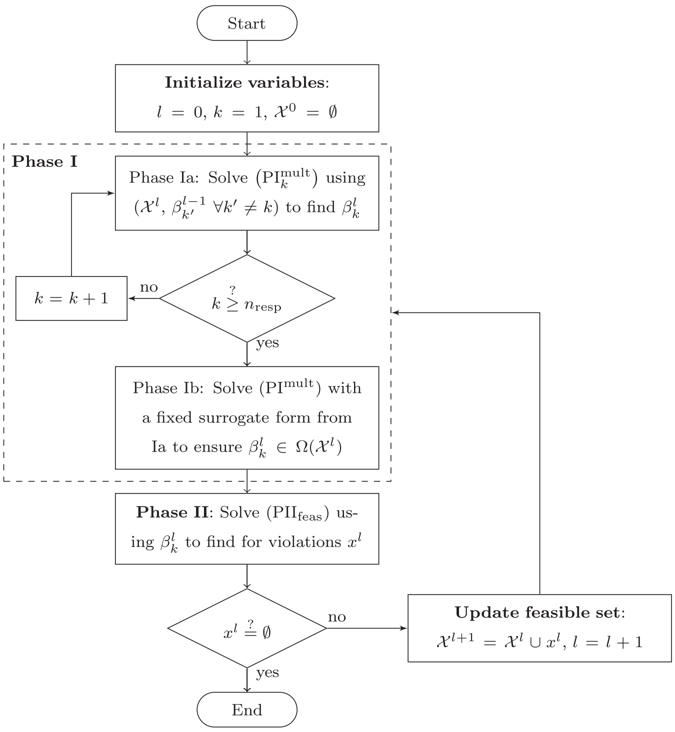
Start
Initialize variables:
$l = 0, k = 1, \mathcal{X}^0 = \emptyset$
Phase I
Phase Ia: Solve ($\mathrm{PI}^{\mathrm{mult}}_k$) using ($\mathcal{X}^l$, $\beta_{k'}^{l-1}$ $\forall k' \neq k$) to find β_k^l
$k = k+1$
no
$k \overset{?}{\geq} n_{\mathrm{resp}}$
yes
Phase Ib: Solve ($\mathrm{PI}^{\mathrm{mult}}$) with a fixed surrogate form from Ia to ensure $\beta_k^l \in \Omega(\mathcal{X}^l)$
Phase II: Solve ($\mathrm{PII}_{\mathrm{feas}}$) using β_k^l to find for violations x^l
$x^l \overset{?}{=} \emptyset$
no
Update feasible set:
$\mathcal{X}^{l+1} = \mathcal{X}^l \cup x^l, l = l+1$
yes
End

Fig. 4. Extending restricting multiple responses to ALAMO.

4.3. *Restricting response derivatives*

Frequently, it is useful to impose restrictions using pre-existing derivative or partial derivative information during the formulation of an empirical model. Although constraints of this type require more knowledge of the underlying system, they may result in an advantageous combination of first-principles theory and empirical data. For problems with *a priori* restrictions on derivatives, regression models must be once- or twice-differentiable, depending on the type of enforced constraints. Restrictions on the feasible regions of the first and second derivatives of each response model, $\hat{z}$, with respect to the predictors, x, have the following functional form.

$$\Omega(\mathcal{X}) := \{\beta \in \mathbb{R}^m : a^{\top} \nabla_x \hat{z} + h(x) \leq 0, \quad x \in \mathcal{X}\} \qquad (7)$$

$$\Omega(\mathcal{X}) := \{\beta \in \mathbb{R}^m : a^{\top} \nabla_x^2 \hat{z} + h(x) \leq 0, \quad x \in \mathcal{X}\} \qquad (8)$$

4.3.1. *First derivative restrictions*

Derivative constraints may represent the most elegant combination of empirical data, first principles, experience, and intuition. One ubiquitous example is the monotonicity of a response variable with respect to one or more predictors. Cumulative distributions, the entropy of an enclosed system, and gas pressure with respect to temperature under ideal or near ideal conditions are all examples of monotonic relationships. Derivative restrictions may also result from imposing initial or boundary conditions, as shown in Section 4.4, where we include an example demonstrating the imposition of an upper bound on the magnitude of the gradient with the aim of model smoothing and the reduction of over-fitting.

4.3.2. *Second derivative restrictions*

Enforcing bounds on the curvature of a surrogate model can be somewhat more complicated. For some modeling applications, however, it can be extremely useful. Consider, for example, the preservation of convexity or concavity of a regression model; if the data are sampled from an underlying convex distribution, it is desirable for the resulting model to be convex.

Enforcing derivative restrictions is particularly enticing when the resulting metamodels will be used in an optimization framework. To enforce such a condition, the modeler must have *a priori* knowledge of the convexity or concavity of the underlying data set. Additionally, the regression model, $\hat{z}(x)$, must be twice differentiable to enforce convexity. This type of restriction is enforced by requiring the Hessian matrix of partial second derivatives $H(\beta, x)$ to be positive-definite. This requirement can, in turn, be enforced by restricting the determinant to be nonnegative:

$$\Omega(\mathcal{X}) := \{\beta \in \mathbb{R}^m : \det(H(\beta, x)) \geq 0, \quad x \in \mathcal{X}\} \tag{9}$$

Consider, for example, that we desire to ensure the convexity of $\hat{z}(x_1, x_2) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2^2 + \beta_4 x_1^2 x_2$ with the following Hessian matrix.

$$H(\beta, x_1, x_2) = \begin{pmatrix} 2\,\beta_4 x_2 & 2\,\beta_3 x_2 + 2\,\beta_4 x_1 \\ 2\,\beta_3 x_2 + 2\,\beta_4 x_1 & 2\,\beta_3 x_1 \end{pmatrix}$$

The enforcement of convexity on the feasible region of the regression problem results from a restriction of the determinant of H to the nonnegative space. This constraint is nonlinear and nonconvex in the β space. Using a least squares regression objective, enforcing convexity would transform a quadratic problem (QP) into a quadratically constrained quadratic problem (QCQP):

$$\Omega(\mathcal{X}) := \left\{\beta \in \mathbb{R}^5 : -\beta_3\,\beta_4 x_1 x_2 - \beta_4^2 x_1^2 - \beta_3^2 x_2^2 \geq 0, \quad x \in \mathcal{X}\right\} \tag{10}$$

In Example 2 in Section 5, we explore the enforcement of numerical properties on a surrogate model; specifically, we enforce a nonnegativity constraint on the second derivative of a one-dimensional problem to ensure the generation of a convex surrogate model.

4.4. Enforcement domain expansion and contraction

In the previous subsections, we primarily restrict the response variables over the range of the original predictor variable; yet, as previously mentioned, it is possible to extend this range to include an *expected extrapolation range* or to contract this domain to enforce conditions over a subset of $\mathcal{X}$ (*e.g.*, boundaries, initial values, or a smaller dimensional space).

4.4.1. Safe extrapolation

By expanding the enforcement domain, we aim to provide a *safe extrapolation*. While, in general, robust prediction techniques avoid extrapolation, engineers and scientists regularly use extrapolation to forecast results beyond an initial sample space (Montgomery et al., 2012). Extrapolation, *i.e.*, using the regression model to predict beyond the range of the original data, increases the likelihood of prediction error (Montgomery et al., 2012). For these reasons, we propose the use of constrained regression over an expanded set $x \in \mathcal{X}_{\text{extrap}}$ with the aim of performing safe extrapolation. Safe extrapolation can be used in conjunction with any of the problem classes described in Section 4 to improve extrapolation accuracy and, often, accuracy within the original problem domain itself. Safe extrapolation is demonstrated using Examples 1 and 2 as well as computational studies in the two following sections.

4.4.2. Boundary and initial conditions

Enforcing boundary conditions while modeling empirical data often leads to a more physically consistent regression model. Boundary conditions are imposed on ordinary or partial differential equations and are categorized into three types: Dirichlet, Neumann, and Robin. Often, these conditions are enforced in a reduced dimensional space. For example, initial conditions provide specifications on the time domain, $t = 0$, without restricting the space domain. For these problems, we reduce the enforcement domain by one or more dimensions x_i to $\mathcal{X} \cap \{x_i = x_i^*\}$.

Dirichlet boundary conditions specify the value of the solution, z, at a fixed location, x_i^*, of the x-domain. Neumann boundary conditions specify the value of the gradient at a fixed location in at least one dimension, x_i^*. Finally, Robin boundary conditions specify a linear combination of function values and derivatives at a fixed location in the domain. The imposition of Dirichlet boundary conditions results in a feasible region that is similar to that provided by Eq. (4), while Neumann and Robin conditions result in the form described by Eqs. (7) and (8).

Standard boundary conditions require parametric equality constraints. Though they impose beneficial restrictions on the feasible region, enforcing these constraints may impede the convergence of the optimization solver. Instead, we permit an ϵ-tolerance on the slack of the equality as follows for $a, b \in \mathbb{R}^n$.

$$\Omega(\mathcal{X}) := \left\{ \beta \in \mathbb{R}^m : \begin{array}{l} \hat{z}(x;\beta) + a^\top \nabla_x \hat{z} + b^\top \nabla_x^2 \hat{z} - h(x) \leq \epsilon_{\text{viol}} \\ \hat{z}(x;\beta) + a^\top \nabla_x \hat{z} + b^\top \nabla_x^2 \hat{z} - h(x) \geq -\epsilon_{\text{viol}} \end{array} \quad x \in \mathcal{X} \cap \{x_i = x_i^*\} \right\}$$

Constraints such as these permit a modeler to use both high fidelity, simulation data and first-principles boundary conditions.

5. Illustrative examples

Here, we demonstrate the application of several classes of x- and z-domain constraints to regression problems using an implementation embedded in the ALAMO software.

5.1. Example 1: bound constrained batch reactor

Modeling chemical reactions using statistical or polynomial fitting functions often results in composition profiles that are not physically realizable. In Example 1, we examine a batch reactor and two first-order reactions in series:

$$A \xrightarrow{k_a} B \xrightarrow{k_b} C$$

where we model the concentration of component B, $[B]$, as a function of batch time $t \in [0.6, 10]$ with kinetic constants $(k_a, k_b) = (0.473, 1.44)$. Boundary conditions at $t = 0$ are available in the form of initial reactor loading concentrations: $[A]_0 = 1$, $[B]_0 = 0$, and $[C]_0 = 0$. In addition, we have *a priori* knowledge that the concentration of component B must be nonnegative and less than the initial concentration of A. In this example, we impose these *a priori* bounds using constrained regression and analyze the quality of the resulting model.

For model generation, we use a training set of 10 data points from a Latin hypercube design of experiments and a test set of 1000 random data points. We allow for potential functional terms including exponentials, logarithms, and several power functions: ±0.5, ±1, ±2, ±3, ±4. In Section 6, we present computational results for a set of similar randomly generated reaction problems.

In Table 3, we compare models generated with and without the application of constrained regression. Both methods result in five-term models. However, the unconstrained method had a test error 7.25 times greater than that of the constrained regression method. In fact, every constrained regression model, from one to five terms, exhibits a test error that is smaller than that of the final five-term unconstrained model.

A maximum of five points are added to the feasible set during each constrained regression iteration. Table 4 includes the data points introduced at each iteration and the corresponding model values for each point in the feasible set. In all cases, we start with an empty feasible set and expand only during the solution of the two-term model. Afterwards, the augmented feasible set is enforced during the solution of the next Phase I problem.

Table 3
Successive models for $[\hat{B}](t)$.

Terms	Test error	Model, $[\hat{B}](t)$
Unconstrained models		
1	0.0418	$0.261/t$
2	0.248	$1.38/t^2 - 1.38/t^3$
3	0.124	$-0.255/\sqrt{t} + 0.818/t - 0.470/t^3$
4	0.793	$-0.298/t + 4.02/t^2 - 7.91/t^4 + 4.67/t^4$
5	0.0856	$0.339 \log t + 2.31/\sqrt{t} - 0.911/t^2 + 0.318/t^4 - 1.494$
Constrained models		
1	0.0418	$0.261/t$
2[a]	–	$1.38/t^2 - 1.38/t^3$
2[a]	–	$-0.102 \log t + 0.219$
2	0.0245	$-0.0325 \log t + 0.236/\sqrt{t}$
3	0.0101	$-0.0609\, t + 0.363\, t^2 + 0.263$
4	0.0171	$-1.13 \log t - 0.805/t + 1.04\sqrt{t} - 0.0604\, t$
5	0.0118	$-0.175 \log t + 0.885/\sqrt{t} - 0.711/t + 0.00395\, t^2 - 0.000197\, t^3$

[a] Unsuccessful trial model.

Table 4
Feasible set for constraining the regression model for $[B]$.

Point added to feasible set, t^l		Modeled value, $[\hat{B}](t^l)$
Iteration $l = 1$	0.6076	−2.3759
	0.6022	−2.4750
	0.6014	−2.4889
	0.6006	−2.5039
	0.6000	−2.5155
Iteration $l = 2$	9.9468	−0.0144
	9.9596	−0.0145
	9.9712	−0.0146
	9.9889	−0.0148
	10.0000	−0.0149

Points added during the solution of the two-term model.

The addition of upper and lower limits on response variables significantly enhances model quality without the requirement of additional sampling. Next, we expand the enforcement domain of these bounds to promote safe extrapolation. Using the same data set and available basis functions, we model the concentration of B over $x \in [0.6, 10]$ while enforcing bounds $[B] \in [0, 1]$ over the expected extrapolation range $x \in [0.1, 11]$.

The resulting one- to five-term models are included in Table 5 along with corresponding test errors. In this table, asterisks denote unsuccessful trial models that require refinement using constrained regression. Test errors for unconstrained, constrained, and extended domain enforcement are provided in Table 6. By extending the enforcement domain, we reduce the error by a factor of 55.2 compared to the unconstrained model and 7.61 compared to the original constrained model while maintaining a constant model complexity of five terms.

Fig. 5 compares the true concentration, sampled data, and models corresponding to each of the three techniques over both the original problem domain and extended enforcement domain. By extending the enforcement domain using constrained regression, we generate a model that exhibits better prediction over both the extended domain and the original problem domain. In Table 6, extrapolation errors for 500 additional validation points from each expansion domain, $t \in [0.1, 0.6]$ and $t \in [10, 11]$, are compared with test error over the original problem space.

As we increase the regression restriction on this example problem, inverse bases functions are deactivated in favor for terms that are do not diverge at small values of t. This is evident for both the original constrained models in Table 3 and the extended domain constrained models in Table 5. Constraining the model in the x- and z-spaces causes the solver to select functional terms and parameter values (*i.e.*, restrict β values) without *a priori* knowledge of relationships among terms and the β space.

5.2. Example 2: enforcing convexity

In Example 2, we model data sampled from $z = x^2 - 0.4x + 0.04 + \epsilon$ over $x \in [-1, 1]$, where ϵ is sampled from a uniform random distribution, $\epsilon \in [-0.25, 0.25]$ using a regression model of the form $\hat{z}(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6 + \beta_7 \exp(x)$. Since the underlying distribution is convex, we enforce the generation of a convex surrogate model. Beyond the increase in model accuracy, there are many cases for which the imposition of favorable numerical properties onto a surrogate model may be beneficial. For example, if the surrogate model generated in this example is used for subsequent optimization, preserving convexity will improve performance of the resulting problem.

To ensure the preservation of convexity, we enforce a nonnegativity bound on the second derivative of the surrogate model: $\hat{z}''(x) = 2\beta_2 + 6\beta_3 x + 12\,\beta_4^2 + 20\,\beta_5 x^3 + 30\beta_6 x^4 + \beta_7 \exp(x)$. This results in the following Phase I constraints and Phase II subproblem.

Table 5
Successive models for $[\hat{B}](t)$ with bounds enforced over $x \in [0.1, 11]$.

Terms	Test error	Model, $[\hat{B}](t)$
Extended domain constrained models		
1[a]	–	$0.261/t$
1	0.0378	$0.156/\sqrt{t}$
2[a]	–	$1.39/t^2 - 1.38/t^3$
2[a]	–	$-0.102 \log t + 0.219$
2	0.0282	$0.350/\sqrt{t} - 0.105$
3	0.0100	$-0.0609\, t + 0.00363\, t^2 + 0.263$
4	0.0044	$0.456\sqrt{t} - 0.281\, t + 0.0208\, t^2 - 0.000705\, t^3$
5[a]	–	$-1.29 \log t - 1.03/t + 0.0799/t^2 + 1.20\sqrt{t} - 0.0719\, t$
5	0.00155	$3.30 \times 10^{-6} \exp t + 0.458\sqrt{t} - 0.280\, t + 0.0180\, t^2 - 5.16 \times 10^{-5}\, t^4$

[a] Unsuccessful trial model.

Table 6
Test error over the original and extended domains for each regression method.

	Test error	
	Original domain $t \in [0.6, 10]$	Extended domain $t \in [0.1, 0.6]$ and $t \in [10, 11]$
Unconstrained regression model	0.0856	0.951
Original domain constrained	0.0118	0.0788
Extended domain constrained	0.00155	0.00717

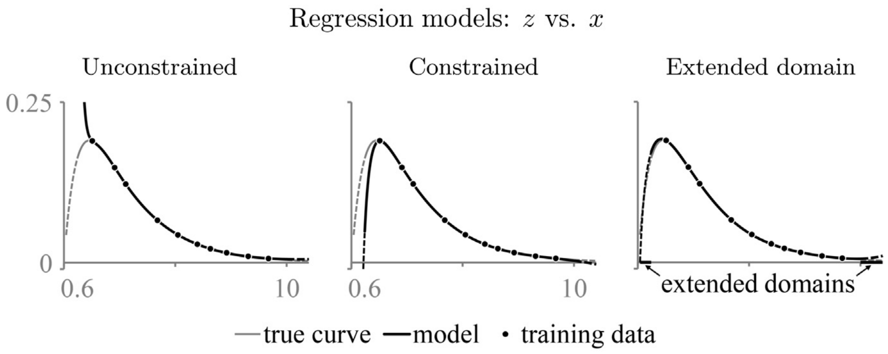

Fig. 5. Regression models for Example 1.

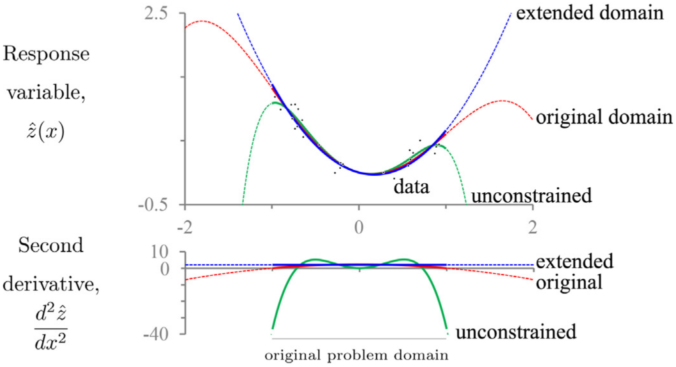

Fig. 6. Convexity enforcement example.

Phase I:

$$2\beta_2 + 6\beta_3\,x + 12\,\beta_4\,x^2 + 20\,\beta_5\,x^3 + 30\beta_6\,x^4 + \beta_7\exp(x) \geq 0 \quad x \in \mathcal{X}^l$$

Phase II:

$$\begin{aligned} \min_{x\in[-1,1]} \quad & f = 2\beta_2^l + 6\beta_3^l\,x + 12\,\beta_4^l\,x2 + 20\,\beta_5^l\,x^3 + 30\beta_6^l\,x^4 \\ & \qquad +\beta_7^l\exp(x) \\ \text{s.t.} \quad & f \leq -\epsilon \end{aligned}$$

We begin by solving an unconstrained fitting problem. Next, we compare this solution to the constrained case with convexity enforcement. To enhance the solution further, we investigate extending the enforcement domain. For this example, the training set consists of 25 points randomly sampled over $x \in [-1, 1]$ and the models are validated using a set of 200 randomly generated points.

The unconstrained solution has a highly nonconvex region near the edges of the x-domain. Fig. 6 provides a graphical representation of surrogate model convexity using plots of the resulting models and second derivatives. Corresponding algebraic models resulting from unconstrained regression, constrained regression, and extended domain constrained regression are provided in Table 7 alongside training and test root mean squared errors. As seen in this table, enforcing convexity over the original sampling domain reduces the test error by 24%. This reduction occurs despite an increase in the training error of 10%. In both cases, a three-term model is selected. The solution is further improved by extending the constraint enforcement to $x \in [-10, 10]$. This extension results in a functional form that more closely matches the true function and simplifies the model without a significant increase in test error.

5.3. Example 3: four-dimensional logistic curve

In Example 3, we examine a four-dimensional model using data sampled from a logistic curve with the functional form:

$$z = \frac{1}{4}\left(\frac{1}{1+e^{-5(x_1-6)}} + \frac{1}{1+e^{-5(x_2-4)}} + \frac{1}{1+e^{-2(x_3-6)}} + \frac{1}{1+e^{-5(x_4-4)}}\right)$$

Systems with logistic behavior are common to engineering, scientific, and natural systems. This logistic curve can range from zero to one. Similar to Example 1, we use limits on z to ensure that the surrogate model matches both the physical system and the empirical data. In Section 6, we present computational results on a larger test set including one-dimensional logistic curves.

We model z using randomly sampled data over $x_i \in [1, 10]$ for $i = 1, 2, 3, 4$. The pool of potential functional forms has 65 terms including both univariate and bivariate basis functions. The univariate terms include exponential, logarithmic, constant, and polynomial powers ($\pm\frac{1}{2}, \pm 1, \pm 2, \pm 3$). The bivariate terms have the form $(x_i x_{i'})^\alpha$ for all six combinations of i and i' for $i \neq i'$ where $\alpha = \{-2, -1, 1, 2\}$. To demonstrate the effectiveness of the proposed techniques over multiple problem sizes, surrogate models are trained using small and large data sets of 10 and 25 instances, respectively.

Similar to Example 1, the bounding constraints added to the Phase I problem are linear, resulting in an MIQP. A maximum of five isolated violation points are identified in each Phase II iteration. Table 8 contains the models identified using unconstrained and constrained subset regression for each data set size.

For both data sets, the unconstrained regression method results in a significantly larger model. In particular, there is a five-term reduction in model complexity when limits on $\hat{z}(x)$ are enforced for the small data set. The unconstrained model over this data set includes an exponential term for each x variable. By all appearances, these exponential terms match the functional form of z; however, this model has a minimum value of -1.35 and maximum value of

Table 7
Models and errors found with unconstrained and enforced convexity approaches.

Terms	Unconstrained regression models	Convexity enforced regression models	Extended domain enforcement
1	$\hat{z} = 1.097\,x^2$	No change	No change
2	$-0.363\,x + 1.023\,x^2$	No change	$-0.363\,x + 1.023\,x^2$
3	$-0.356\,x + 1.31\,x^2 - 0.581\,x^6$	$-0.361\,x + 1.15\,x^2 - 0.193\,x^4$	–
Root mean squared errors			
Training	0.117	0.129	0.137
Test	0.104	0.0791	0.0796

Table 8
Models found for Example 3.

Method	Terms	Model, $\hat{z}(x_1, x_2, x_3, x_4)$
Small training set		
Unconstrained	7	$-9.9 \times 10^{-5}\ \exp(x_1) + 4.9 \times 10^{-4}\ \exp(x_2) - 5.2 \times 10^{-5}\ \exp(x_3) + 1.4 \times 10^{-4}\ \exp(x_4) + 0.11\, x_3^{-3} + 2.2 \times 10^{-3}\, x_1^3 - 7.6 \times 10^{-5}\, (x_1 x_4)^2$
Constrained	2	$0.20\sqrt{x_2} + 0.0034\, x_1 x_4$
Large training set		
Unconstrained	4	$0.26 \log x_2 + 9.3 \times 10^{-4}\, x_1^3 - 9.6 \times 10^{-3}\, x_1 x_2 + 6.4 \times 10^{-3}\, x_2 x_4$
Constrained	2	$0.19\sqrt{x_2} + 0.0041\, x_1 x_4$

4.62, over the original domain, which far exceeds the physical limits of [0, 1]. Moreover, an increase in data set size vastly alters the functional form of the resulting unconstrained regression model. In fact, only one term is retained and terms dependent on x_3 are eliminated. In comparison, the realized range of the four-term model that is trained on the large data is reduced to [−0.005, 1.22]. The ranges of the constrained regression models generated using the small and large data sets are [0.203, 0.972] and [0.189, 0.990]. For constrained regression, the two-term model constructed with enforced limits on $\hat{z}$ is robust to the changes in data set size. Additionally, the constrained regression models are not a function of x_3. This is expected because the reduced coefficient on x_3 in the logistic exponential term.

In addition to the training set used to build each regression model, we compare model prediction error based on three test validation sets: (a) original problem domain, $x \in [1, 10]$, (b) lower extended problem domain, $x \in [0.5, 1]$, and (c) upper extended problem domain, $x \in [10, 11]$. These three validation sets are generated using 1000 random sample points over the original domain and 500 points over the upper and lower extended domains. A comparison of resulting root mean squared errors is provided in Table 9. Enforcing *a priori* limits on the response variable leads to improved error factors of 4.11 and 1.15 for models trained on the small and large data sets, respectively. This suggests that adding theoretical limits has the greatest impact when little initial data is available – when the large relative weight of the bound information can fill gaps in the empirical data.

Enforcing *a priori* limits on the four-dimensional response model, $\hat{z}$, leads to more accurate, parsimonious models that obey asymptotic response limits while remaining robust to changes in the training data set.

6. Computational experiments

In this section, we demonstrate efficacy of the proposed methods through a computational study and compare the accuracy of constraining the x- and z-spaces in contrast to unconstrained regression. For each instance, we use a fixed data set from which we generate a regression model three ways:

1. Unconstrained: subset regression using ALAMO.
2. Constrained: constrained subset regression over the original problem domain using ALAMO.
3. Extended domain constrained: constrained subset regression over an extended or expected extrapolation problem domain using ALAMO.

The test set includes three types of underlying functions: chemical reactor models, normal distributions, and logistic problems that have upper and/or lower bounds. Details of each category are provided in Table 10. In total, the test set contains 120 problems. Ten functional forms from each category are generated using random parameter values sampled from a uniform distribution over the variable ranges specified in Table 10. Using a Latin hypercube design of experiments, four distinct data sets per functional form are generated and tested. The potential functional forms for each regression model include exponentials, logarithms, and several powers: ±0.5, ±1, ±2, ±3, ±4.

Specifications for Methods 2 and 3 are summarized in Table 11. The logistic and batch reactor test instances are bounded above and below while the normal distribution is bounded below.

To compare the three methods, we use the error factor, defined by Eq. (3), as a model quality metric. The cumulative distribution of error factors for the three methods is plotted in Fig. 7. Here, four factors are considered:

1. Training error: error between the model and the points used to train the model.
2. Original domain test error: error between the model and 1000 randomly sampled points over the original problem domain.
3. Lower extended domain test error: error between the model and 500 randomly sampled points over the domain extension to smaller values of the predictor variables.
4. Upper extended domain test error: error between the model and 500 randomly sampled points over the domain extension to larger values of the predictor variables.

The plots in Fig. 7 depict a race-to-the-top metric. Therefore, curves that climb higher for smaller values of the error factor signify more accurate regression modeling. Since the primary purpose of these regression models is an accurate representation of true functional form, we first consider the test accuracy over the original sample space. In 48% of the problems, constrained regression is either the most accurate or equal to the most accurate over this domain and continues to be particularly dominant over the unconstrained regression models. Yet, the unconstrained regression models have significantly smaller training error, with errors at least as low as the smallest training error in 87% of the problems, meaning that the model is over fit since it cannot make use of the additional domain knowledge available in the constrained regressions cases. In the final 13% of the test set, constrained regression generates more complex models than unconstrained regression. This superior prediction capability further motivates the addition of response variable bounds to improve model accuracy over a simple empirical data modeling technique.

By extending the bound enforcement domain as described in Table 11, the test error over both the lower and upper extrapolation range is improved to allow for safer model extrapolation. The extended domain constrained models demonstrate the highest accuracy for 58the lower and upper bounds, respectively. These extended domain methods are dominant over unconstrained models and are superior to constraining only the original problem space. In particular, there is benefit in the lower extrapolation of these extended constraints over the unconstrained case.

In order to achieve this increased model accuracy, constrained regression does require additional computation effort. Up to five points are added to each discretized Phase II problem. In most cases, if the trial model violates a bound, all five points are returned. However, some problems resulted in fewer than five violated points. The constrained regression statistics (feasible set size, total constrained regression iterations, and regression re-solves required) for this problem set are provided in Table 12. Constraining the regression problem over the expanded domain requires more regression re-solves and iterations than constraining the original problem domain.

For the problems shown, constraints enforcing bounds on the response variables not only generate more physically realizable

Table 9
Training and validation errors for surrogate models in Example 3.

Method	Training error	Test error		
		Original domain	Lower ext. domain	Upper ext. domain
Small training set				
Unconstrained	0.0021	0.6816	0.4062	2.7257
Constrained	0.1318	0.1656	0.1763	0.0393
Large training set				
Unconstrained	0.0862	0.1834	0.0983	0.3447
Constrained	0.1382	0.1593	0.1622	0.0599

Table 10
Description of problem types in the computational test set.

Problem type	Description	Data set sizes	Random parameters
Batch reactor	Batch reactor concentration of B, $A\xrightarrow{k_a}B\xrightarrow{k_b}C$, $[A]_0=1$, $[B]_0=0$, and $[C]_0=0$.	3, 10, 10, 10	$k_a\in[0.1, 2]$, $k_b\in[0.1, 2]$
Normal distribution	Normal distribution with mean μ and standard deviation σ	3, 10, 10, 10	$\mu\in[1, 10]$, $\sigma\in[1, 3]$
Logisitic curve	Logistic function with multiplier a and offset b, $1/(1+\exp(-a(x-b)))$	3, 10, 10, 10	$a\in[0, 5]$, $b\in[3, 7]$

Table 11
Constrained regression specifications.

Problem type	Bounds enforced	Original problem domain	Extended problem domain
Batch reactor	$0\le[B]\le[A]_0$	$t\in[0.6, 10]$	$t\in[0.5, 11]$
Normal distribution	$0\le z$	$x\in[1, 10]$	$x\in[0.5, 11]$
Logisitic curve	$0\le z\le 1$	$x\in[1, 10]$	$x\in[0.5, 11]$

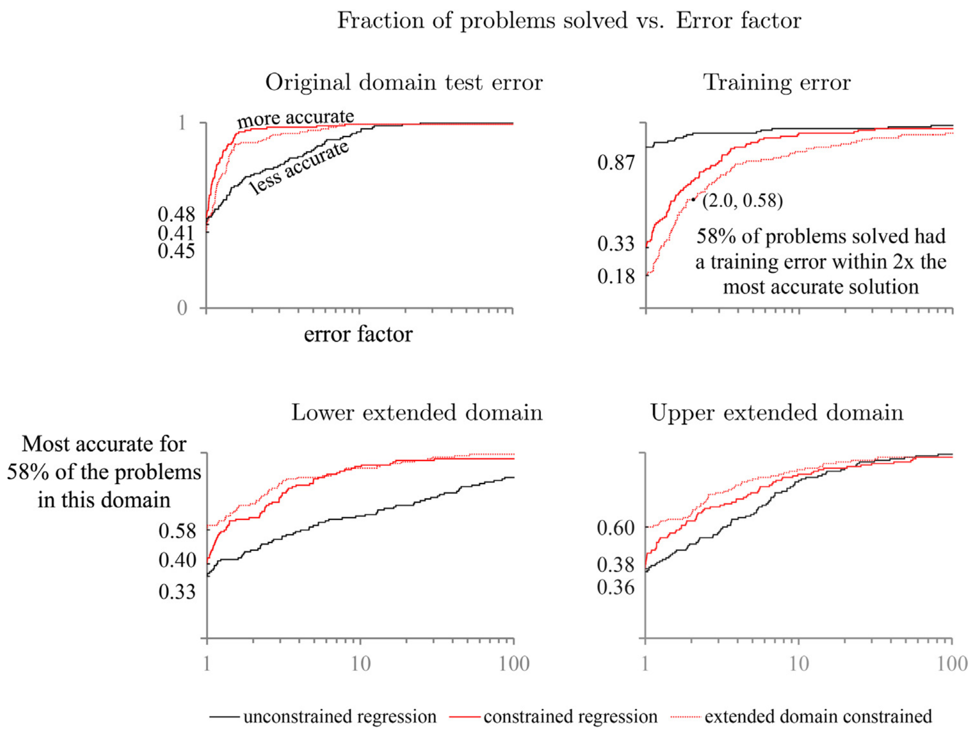

Fig. 7. Computational results.

Table 12
Constrained regression solution statistics.

	Points in the final feasible set		Constrained regression iter		Additional regression problems solved	
	Mean	Range	Mean	Range	Mean	Range
Original domain	7.52	[0, 25]	1.93	[1, 3]	1.58	[0, 7]
Extended domain	10.7	[0, 35]	2.28	[1, 6]	2.50	[0, 18]

models, but use this information to build more accurate models and to enhance extrapolation accuracy.

7. Conclusions

The combination of data-driven modeling and theory-driven *a priori* knowledge results in higher quality surrogate models. The improvements are measured by both physical relevance and model accuracy. We introduced a novel approach to constrained regression: restricting the original problem in the space of predictor and response variables. Constraints of this *more intuitive* form can be used to reveal hidden relationships between regression coefficients that are otherwise unknown to the modeler. In particular, we described several classes of problems that lead to intuitive constraints including bounds of response variables, thermodynamic limitations, safe extrapolation, boundary conditions, and enforcing favorable numerical properties on first and second derivatives. Through extensive testing, we demonstrated the capability of these restrictions to improve model accuracy in addition to ensuring adherence to physical limits.

We conclude that *a priori* information from first principles, intuition, and other system knowledge can be used to enhance data-driven modeling with many model selection and regression methods, from ordinary least squares regression to subset selection and regularized regression.

Acknowledgments

As part of the National Energy Technology Laboratory's Regional University Alliance (NETL-RUA), a collaborative initiative of the NETL, this technical effort was performed under the RES contract DE-FE0004000, as part of the Carbon Capture Simulation Initiative. This report was prepared as an account of work sponsored by an Agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

Bard Y. Nonlinear parameter estimation. Academic Press; 1974.

Chang Y, Sahinidis NV. Steady-state process optimization with guaranteed robust stability under parametric uncertainty. AIChE J 2011;57:3395–407.

Cozad A, Sahinidis NV, Miller DC. Learning surrogate models for simulation-based optimization. AIChE J 2014;60:2211–27.

Gibbons DI, McDonald GC. Constrained regression estimates of technology effects on fuel economy. J Qual Technol 1999;31:235–45.

Goberna MA, López MA. Linear semi-infinite programming theory: an updated survey. Eur J Oper Res 2002;143:390–405.

Hettich R, Kortanek KO. Semi-infinite programming: theory, methods, and applications. SIAM Rev 1993;35:380–429.

Hurvich CM, Tsai CL. A corrected Akaike information criterion for vector autoregressive model selection. J Time Ser Anal 1993;14:271–9.

John F. Extremum problems with inequalities as subsidiary condition. In: Studies and essays, courant anniversary volume, interscience; 1948. p. 187–204.

Judge GG, Takayama T. Inequality restrictions in regression analysis. J Am Stat Assoc 1966;61:166–81.

Knopov PS, Korkhin AS. Regression analysis under a priori parameter restrictions. Springer; 2011.

Korkhin AS. Certain properties of the estimates of the regression parameters under a priori constraint-inequalities. Cybernetics 1985;21:858–70.

Korkhin AS. Parameter estimation accuracy for nonlinear regression with nonlinear constraints. Cybern Syst Anal 1998;34:663–72.

Korkhin AS. Solution of problems of the nonlinear least-squares method with nonlinear constraints. J Autom Inf Sci 1999;6:110–20.

Korkhin AS. Estimation accuracy of linear regression parameters with regard to inequality constraints based on a truncated matrix of mean square errors of parameter estimates. Cybern Syst Anal 2002;38:900–3.

Korkhin AS. Determining sample characteristics and their asymptotic linear regression properties estimated using inequality constraints. Cybern Syst Anal 2005;41:445–56.

Korkhin AS. Using a priori information in regression analysis. Cybern Syst Anal 2013;91:41–54.

Liew CK. Inequality constrained least-squares estimation. J Am Stat Assoc 1976;71:746–51.

McKay MD, Beckman RJ, Conover WJ. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 1979;21:239–45.

Montgomery DC, Peck EA, Vining GG. Introduction to linear regression analysis. Wiley; 2012.

Nelles O. Nonlinear system identification: from classical approaches to neural networks and fuzzy models. Springer; 2001.

Pearson RK, Pottmann M. Gray-box identification of block-oriented nonlinear models. J Process Control 2000;10:301–15.

Rao CR. Linear statistical inference and its applications. Wiley; 1965.

Reemtsen R, Görner S. Numerical methods for semi-infinite programming: a survey. In: Reemtsen R, Rückmann" JJ, editors. Semi-infinite programming. Boston, MA: Kluwer Academic Publishers; 1998. p. 195–275.

Reemtsen R, Rückmann JJ. Semi-infinite programming. Boston, MA: Kluwer Academic Publishers; 1998.

Rezk G. Inequality restrictions in regression analysis. J Dev Econ 1976;71: 746–51.

Rios LM, Sahinidis NV. Derivative-free optimization: a review of algorithms and comparison of software implementations. J Global Optim 2013;56: 1247–93.

Sahinidis NV. BARON, user's manual; 2014, Available at http://www.gams.com/dd/docs/solvers/baron.pdf

Simpson TW, Peplinski J, Koch PN, Allen JK. Metamodels for computer-based engineering design: survey and recommendations. Eng Comput 2001;17: 129–50.

Tawarmalani M, Sahinidis NV. A polyhedral branch-and-cut approach to global optimization. Math Program 2005;103:225–49.

Thompson M. Some results on the statistical properties of an inequality constraints least squares estimator in a linear model with two regressors. J Econom 1982;19:215–31.

˜Zaković S, Rustem B. Semi-infinite programming and applications to minimax problems. Ann Oper Res 2002;124:81–110.