From Discrete Measurements to Bounded Gradient Estimates: A Look at Some Regularizing Structures

Gene A. Bunin, Grégory François, and Dominique Bonvin*

Laboratoire d'Automatique, Ecole Polytechnique Fédérale de Lausanne (EPFL)

CH-1015 Lausanne, Switzerland

E-mail: dominique.bonvin@epfl.ch

Phone: +41 21 6933843. Fax: +41 21 6932574

Abstract

Obtaining a reliable gradient estimate for an unknown function when given only its discrete measurements is a common problem in many engineering disciplines. While there are many approaches to obtaining an estimate of a gradient, obtaining lower and upper bounds on this estimate is an issue that is often overlooked, as rigorous bounds that are not overly conservative usually require additional assumptions on the function that may either be too restrictive or impossible to verify. In this work, we try to make some progress in this direction by considering four general structural assumptions as a means of bounding the function gradient in a rigorous likelihood sense. After proposing an algorithm for calculating these bounds, we compare their accuracy and precision across different scenarios in an extensive numerical study.

Introduction

In most experimental settings, measurements act as the fundamental tool for learning about the nature of the measured quantity (modeling), confirming the statistical validity of a statement (hy-

*To whom correspondence should be addressed

pothesis testing), or applying feedback to obtain some desired behavior (control/optimization). However, while the knowledge obtained from measurements is often priceless, it generally comes with two innate limitations – not only is it cost-prohibitive to obtain an exhaustive set of measurements to *completely* identify an experimental quantity and its relation to the associated variables, but the measurements that are collected are themselves, often due to sensor inaccuracies/noise, subject to error and uncertainty. Mathematically, we may phrase this by defining the experimental quantity of interest as some function, $f : \mathbb{R}^{n_u} \to \mathbb{R}$, for which a discrete, sampled set of N input-output pairings is available:

$$y_i = f(\mathbf{u}_i) + w_i, \quad i = 1, ..., N, \tag{1}$$

with $\mathbf{u} \in \mathbb{R}^{n_u}$ denoting the inputs (the manipulated variables), y the output (measurement), and w the measurement noise.

The particular problem that we consider in this work is the following:

> Given the set of inputs $\mathbf{u}_1, ..., \mathbf{u}_N$, the corresponding measured output values $y_1, ..., y_N$, and the assumption that $f(\mathbf{u})$ is differentiable, calculate the bounds, $\nabla \underline{f}(\overline{\mathbf{u}})$ and $\nabla \overline{f}(\overline{\mathbf{u}})$, on the function gradient, $\nabla f(\overline{\mathbf{u}})$, such that $\nabla \underline{f}(\overline{\mathbf{u}}) \preceq \nabla f(\overline{\mathbf{u}}) \preceq \nabla \overline{f}(\overline{\mathbf{u}})$ for some specified $\overline{\mathbf{u}} \in \{\mathbf{u}_1, ..., \mathbf{u}_N\}$.

We start by presenting a simpler version of this problem, which involves only obtaining an estimate of the function gradient (without bounding its uncertainty) and arises in a number of different, often unrelated, engineering fields such as geology,[1] image rendering,[2] and particle filtering.[3] The authors' own interest in the problem comes from its applicability to real-time optimization,[4,5,6,7] where the objective is to steer a process towards economically optimal conditions while working only with discrete measurements – using, for example, a gradient-descent method. We also note that this problem may play an important role, for almost identical reasons, in numerical black-box optimization when function evaluations are time consuming.[8]

A key trait of estimating the gradient, or the derivatives, of a function from its discrete mea-

surements is that the problem is ill-posed and inverse,[9] since applying the standard definition of a derivative and taking the limit of a finite difference with a step size that tends to 0 will inevitably yield an estimate of $\pm\infty$ in the presence of noise. The standard way to solve such problems is through the use of regularization, which, roughly speaking, calls for an assumption on the structure or nature of $f(\mathbf{u})$. Perhaps the simplest assumption is the one of Lipschitz continuity with respect to the different input directions:

$$\underline{\kappa}_j \leq \frac{\partial f}{\partial u_j}\Big|_{\overline{\mathbf{u}}} \leq \overline{\kappa}_j, \quad \forall \overline{\mathbf{u}} \in \{\mathbf{u}_1, ..., \mathbf{u}_N\}, \quad j = 1, ..., n_u, \tag{2}$$

i.e. that the derivatives $\frac{\partial f}{\partial u_j}\Big|_{\overline{\mathbf{u}}}$ are bounded by some finite constants $\underline{\kappa}_j$ and $\overline{\kappa}_j$ over the relevant input set. This hypothesis is useful in that it is generally true (for most functions, one can always choose the Lipschitz bounds to be sufficiently low and high) and in that it succeeds in regularizing the problem by constraining the derivatives to lie in a finite-size box. Alternatively, one may also regularize by imposing stricter structural assumptions, such as assuming $f(\mathbf{u})$ to be linear or quadratic and obtaining a regularized estimate by regressing the available input-output data with these structures. Finally, we may also assume that $f(\mathbf{u})$ is smooth to some extent, which leads to general-purpose regularization methods like those of Tikhonov,[10] where one enforces that the finite-difference approximations of the function's *curvature* be sufficiently small via the penalization of their norm in a multi-criterion optimization problem.[9,11]

With the existence of all these methods, it should be clear that obtaining a regularized gradient estimate is relatively straightforward. Unfortunately, the quality of any such estimate remains suspect unless something can be said about its uncertainty. As a very simple example of why this might matter, consider the problem of estimating the absorptivity of a liquid according to the Beer-Lambert law $A = \varepsilon C$, where A is the measured absorbance, ε the (unknown) absorptivity, and C the concentration. A typical procedure would be to prepare solutions with different values of C, to measure A for each solution, and to then regress the data so as to obtain $\hat{\varepsilon}$ as the slope (see Figure 1 for a qualitative illustration). While one could make certain statistical claims about $\hat{\varepsilon}$ (usually in the least-squares sense), a stronger result would be to also give lower and upper bounds ($\underline{\varepsilon}$ and $\overline{\varepsilon}$) on

the absorptivity, which may in turn let the experimenter know if more measurements are needed or if the current bounds are sufficiently tight. As a second, less trivial, example, we note that recent work in real-time optimization[12] has formalized a robust procedure that can achieve optimality provided that bounds on the gradients, rather than the gradients themselves, are available. More abstractly, one may envision that being able to provide uncertainty bounds for gradient estimates would be welcomed in many applications where a gradient estimate is desired.

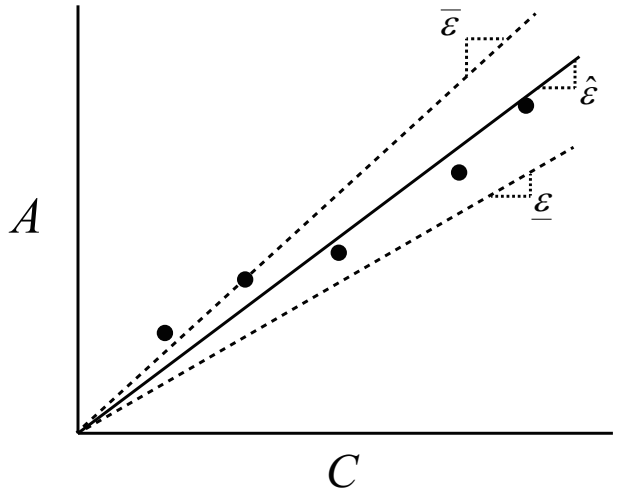


Figure 1: Linear regression of concentration-absorbance data (five discrete points) to obtain $\hat{\varepsilon}$. The lines defined by $\underline{\varepsilon}$ and $\overline{\varepsilon}$ may be seen as the regressions corresponding to the possible lower and upper fits, which define the smallest and largest possible absorptivities with corresponding fits that are still statistically valid.

Not surprisingly, obtaining bounds that are relatively tight, and therefore useful, requires a bit more work than just obtaining an estimate. To the best of the authors' knowledge, the most relevant existing methods for doing this are those for quantifying the estimation error in finite-difference estimates, where the standard approach consists in writing out the Taylor series expansion of $f(\mathbf{u})$ and upper bounding its nonlinear components (to bound the truncation error) and then bounding the measurement noise w (to bound the error due to noise).[7,13] Such approaches are innately conservative as they consider both the worst-case deterministic and stochastic elements, and are generally not proposed as methods to obtain uncertainty bounds on the gradient estimate – rather, they are used to choose the input points $\mathbf{u}$ so as to either minimize the worst-case error from the forthcoming finite-difference estimate or to keep its norm below some specified value. Analogous

concepts are also used to bound the error of the gradient of a regression model in derivative-free optimization schemes,[8] but are also more conceptual than practically useful, and serve largely to check that the input data set is well-poised for regression.

So as to circumvent these difficulties, the method proposed in this work follows a significantly different philosophy. Namely, we propose to obtain gradient uncertainty bounds by exploiting a *local* structural assumption on $f(\mathbf{u})$ together with the probability density function (pdf) of the noise, which allows us to regress the observed data with the assumed structure subject to a likelihood criterion. We then proceed to calculate the minimal and maximal derivatives that the regressed structure may allow without losing statistical validity. The benefits of our approach lie in that:

(a) if the structural assumption is correct, the calculated bounds are *guaranteed* to be accurate and to contain the true gradient with a *chosen* probability P,

(b) the method is general and admits many different structural assumptions, with the overall trend that more general structures yield more accurate, but less precise, bounds (analogous to the bias-variance tradeoff in statistical learning[14]).

The theoretical foundations for our method, as well as its algorithmic implementation, are discussed in Sections 2 and 3 and comprise the first half of the paper.

In practice, it is often the case that the structure of $f(\mathbf{u})$ is not known, even locally, and so assuming an incorrect structure may lead to inaccuracy, where the calculated uncertainty interval does *not* contain the true gradient. For this reason, we dedicate the second half of the paper (Section 4) to an exhaustive set of numerical experiments, where different regularizing structures for various functions $f(\mathbf{u})$ are investigated in an attempt to evaluate the accuracy and precision of each choice of regularizing structure on average. The studies also touch upon a number of other factors that may influence the quality of the bounds and, although it is impossible to evaluate the bounds' absolute quality outside of any specific context or application, are complemented by efforts to give insight and recommendations wherever it is possible to do so.

Local Likelihood Regularization with Structural Constraints

In this section, we discuss the theoretical aspects of our approach, which is centered around the following maximum-likelihood (ML) regularization of a local subset of some $n \leq N$ measured data points subject to structural constraints:

$$\begin{aligned} \underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{d}}}{\text{maximize}} \quad & \prod_{i=1}^{n} p(\hat{w}_i) \\ \text{subject to} \quad & \hat{w}_i = y_i - \hat{y}_i, \quad i = 1,...,n \\ & \hat{\mathbf{y}},\hat{\mathbf{F}},\hat{\mathbf{d}} \in S \end{aligned} \quad , \tag{3}$$

where $p : \mathbb{R} \to \mathbb{R}$ is the pdf of the noise, $\hat{\mathbf{y}} \in \mathbb{R}^n$ the estimated noise-free output values, $\hat{\mathbf{w}} \in \mathbb{R}^n$ the estimated noise values, $\hat{\mathbf{F}} \in \mathbb{R}^{n_u \times n}$ the matrix of gradient estimates $[\nabla\hat{f}(\mathbf{u}_1)...\nabla\hat{f}(\mathbf{u}_n)]$, and $\hat{\mathbf{d}}$ a vector of estimated auxiliary variables that depends on the specific definition of the structure S, the latter usually corresponding either to parameters in a parametric structure or to discrete values in a non-parametric one. In simpler terms, Problem (3) constructs a function that satisfies the structure-induced constraints S (by specifying $\hat{\mathbf{y}},\hat{\mathbf{F}},\hat{\mathbf{d}}$) in a way that maximizes the joint probability of the resulting noise realization ($\hat{\mathbf{w}}$). It is assumed that the noise values are independent and that the corresponding pdf is available. One may also extend (3) to accommodate different pdfs for different measurements, but the discussion here is limited to a sole pdf for simplicity.

It is crucial to note that without the structural constraints $\hat{\mathbf{y}},\hat{\mathbf{F}},\hat{\mathbf{d}} \in S$ the above regularization is essentially useless, as it would simply set every $\hat{w}$ value to that which corresponds to the maximum of the pdf (i.e. the mode). As such, S is *the* regularizing element in this problem, and may be seen abstractly as any additional assumption on $f(\mathbf{u})$. More specifically, the previously given examples of $f(\mathbf{u})$ being Lipschitz-continuous, linear or quadratic, or having curvature that is subject to certain smoothness conditions may then all contribute to define S via precisely formulated mathematical constraints.

We now proceed to discuss the concept of structure validity, propose four types of structures that may be used in (3), and finish by showing how one may calculate the gradient bounds in this

framework.

Valid and Invalid Structures

We start by making the somewhat evident statement that regularizing with valid structures (i.e. assumptions that the true function $f(\mathbf{u})$ satisfies) leads to more accurate results than regularizing with invalid ones. However, as confirming the validity of a structure is usually not possible without sampling the entire input space, we are forced to work from the other end and to focus our efforts on rejecting the structures that can be guaranteed, with a certain probability, to be invalid.

This may be statistically quantified by defining a likelihood value that serves as a lower bound, with probability P, on the likelihood for the true measurement noise of n samples. Denoting this value by L_P, we define it implicitly as follows:

$$\text{prob}(\prod_{i=1}^{n} p(w_i) \geq L_P) = P. \tag{4}$$

Analytically, one would calculate this quantity by integrating the pdf of $\prod_{i=1}^{n} p(w_i)$, but this is not possible in the general case since such an integration may be intractable. We may, however, obtain a very good approximation of L_P by Monte Carlo sampling. This is essentially done by drawing n samples a large number of times and building the approximate pdf of the likelihood.

We are now in the position to define the concept of a probabilistically invalid structure.

Definition 1 (P-invalidity)**.** *The structure set S is said to be P-invalid, or invalid with probability P, if the set of all feasible solutions to Problem (3) with a likelihood greater than or equal to L_P, denoted by $\mathscr{E}$, is empty:*

$$\mathscr{E} = \{\hat{\mathbf{y}}, \hat{\mathbf{w}}, \hat{\mathbf{F}}, \hat{\mathbf{d}} : \prod_{i=1}^{n} p(\hat{w}_i) \geq L_P;\ \hat{\mathbf{w}} = \mathbf{y} - \hat{\mathbf{y}};\ \hat{\mathbf{y}}, \hat{\mathbf{F}}, \hat{\mathbf{d}} \in S\} = \varnothing. \tag{5}$$

We may then think of L_P as a threshold below which a certain invalid structure is deemed as statistically unlikely, since enforcing such a structure would force the resulting noise estimates to

have a low likelihood. A simple way to check whether a structure can be declared as P-invalid is formulated next.

Lemma 1 (Guarantee of P-invalidity)**.** *Let* $\{\hat{\mathbf{y}}^*, \hat{\mathbf{w}}^*, \hat{\mathbf{F}}^*, \hat{\mathbf{d}}^*\}$ *be the global maximum of (3). If* $\prod_{i=1}^{n} p(\hat{w}_i^*) < L_P$*, then S is P-invalid.*

Proof. By the global optimality of $\hat{\mathbf{w}}^*$, it follows that $\prod_{i=1}^{n} p(\hat{w}_i) \leq \prod_{i=1}^{n} p(\hat{w}_i^*) < L_P$ for all $\hat{\mathbf{w}}$ in the feasible set of (3), which implies $\mathscr{E} = \varnothing$ and thus the P-invalidity of S. □

As a simple example of P-invalidity, consider the case where $f(\mathbf{u})$ is quadratic and the regularizing structural assumption is that of linearity. Problem (3) can then be written as:

$$\begin{aligned} \underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{a}},\hat{b}}{\text{maximize}} \quad & \prod_{i=1}^{n} p(\hat{w}_i) \\ \text{subject to} \quad & \hat{w}_i = y_i - \hat{y}_i, \quad i = 1,...,n \\ & \hat{y}_i = \hat{\mathbf{a}}^T \mathbf{u}_i + \hat{b}, \quad i = 1,...,n \\ & \nabla \hat{f}(\mathbf{u}_i) = \hat{\mathbf{a}}, \quad i = 1,...,n \end{aligned} \quad , \tag{6}$$

where we make the link with the general formulation in (3) by noting that $\hat{\mathbf{d}} = \{\hat{\mathbf{a}}, \hat{b}\}$, and that $S = \{\hat{\mathbf{y}}, \hat{\mathbf{F}}, \hat{\mathbf{a}}, \hat{b} : \hat{y}_i = \hat{\mathbf{a}}^T \mathbf{u}_i + \hat{b},\ i = 1,...,n;\ \nabla \hat{f}(\mathbf{u}_i) = \hat{\mathbf{a}},\ i = 1,...,n\}$.

Clearly, enforcing this structure for a data set that extends beyond a local (linear) neighborhood would lead to a poor fit with significant residuals and a low value of $\prod_{i=1}^{n} p(\hat{w}_i^*)$. This is illustrated for a one-dimensional case in Figure 2.

Having treated the case of rejecting the structural hypothesis when it is false, we now consider the alternative of rejecting the structural hypothesis when it is, in fact, true.

Lemma 2 (False P-invalidity)**.** *Consider the case where* $f(\mathbf{u})$ *meets the structural assumptions of S. The probability of S being declared P-invalid as by Lemma 1 is at most* $1 - P$*.*

Proof. By our definition of P-invalidity in (5), we need simply to prove that the probability of $\mathscr{E} = \varnothing$ is at most $1 - P$ or, equivalently, that the probability of $\mathscr{E} \neq \varnothing$ is at least P. We prove the latter by considering the following choice of variables: $\hat{y}_i = f(\mathbf{u}_i), \hat{w}_i = w_i, \nabla \hat{f}(\mathbf{u}_i) = \nabla f(\mathbf{u}_i), i = 1,...,n,$

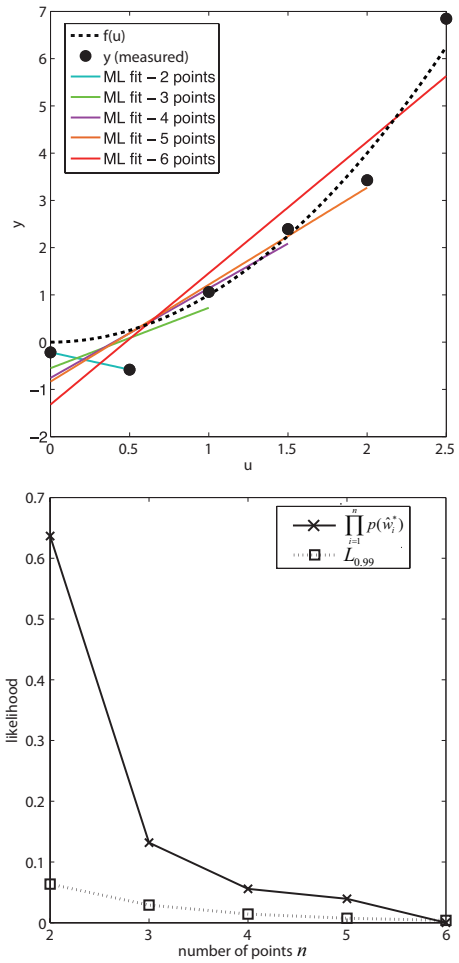


Figure 2: P-invalidity of a linear structure for data generated from the quadratic function $f(u) := u^2$, with an additive measurement noise $w \in \mathcal{N}(0, 0.25)$. Here, we illustrate the case of iteratively fitting a linear function to a data set that is augmented by one point at a time (starting from a set of two points, $u = 0$ and $u = 0.5$, and appending from left to right as shown in the figure). For the cases of 5 points or less, the ML fit of Problem (6) has a likelihood greater than $L_{0.99}$ and so the linear structure cannot be rejected. For 6 points, however, the fit is sufficiently unlikely so as to allow rejection of the linear structure with a probability of 0.99.

together with $\hat{\mathbf{d}} = \mathbf{d}$, where $\mathbf{d}$ denote the auxiliary variables that define $f(\mathbf{u})$. We show that this choice of variables belongs to $\mathscr{E}$ with a probability of P by considering the three sets of constraints that define $\mathscr{E}$. First, it follows that this choice satisfies $\hat{\mathbf{y}}, \hat{\mathbf{F}}, \hat{\mathbf{d}} \in S$, or $f(\mathbf{u})$ would violate the structural constraints of S, which it cannot by assumption. The condition that $\hat{\mathbf{w}} = \mathbf{y} - \hat{\mathbf{y}}$ is clearly satisfied as well, since it simply becomes the measurement equality (1). As such, both of these conditions are satisfied with a probability of 1. The remaining condition, $\prod_{i=1}^{n} p(\hat{w}_i) = \prod_{i=1}^{n} p(w_i) \geq L_P$, is satisfied with a probability of P by the definition of L_P in (4), and so the joint probability of all three conditions being satisfied is P. Having thereby provided *one* choice of variables for which $\mathscr{E} \neq \varnothing$ with a probability of P, we now note that other choices of $\hat{\mathbf{y}}, \hat{\mathbf{w}}, \hat{\mathbf{F}}, \hat{\mathbf{d}}$ belonging to $\mathscr{E}$ may also exist, thereby allowing us to state that the probability of $\mathscr{E} \neq \varnothing$ must be at least P. □

Lemma 2 may thus be seen as an upper bound on the probability of a false negative – the probability of rejecting a structure as P-invalid when it is valid. The practical consequence is that, for a sufficiently high P, a valid structure is almost always guaranteed to be retained.

Choice of Structure

While the structural constraints of S can take a number of different forms, it should be clear that tighter, more constrained sets should be much preferred, provided that they are not invalid, as doing so places more restrictions in the regularization and increases its usefulness. Of particular interest are the limitations that different choices of S place on $\hat{\mathbf{F}}$, and thereby on the gradient estimate, $\nabla \hat{f}(\overline{\mathbf{u}})$. We now proceed to define four structures that succeed in constraining $\hat{\mathbf{F}}$, each to a different extent, and show how they may be formulated in the framework of (3). The Lipschitz assumption of (2) is also incorporated into S, where we note that although obtaining the Lipschitz constants $\underline{\kappa}$ and $\overline{\kappa}$ may range from trivial to very difficult in practice,[12] very conservative estimates will suffice here, since the essential role of these constants is to define a compact domain for the gradients and to thereby ensure boundedness of the estimates.

Linear Functions

Linearity is perhaps the most frequently used and the most restricting assumption, being only (approximately) valid in a small local neighborhood for most nonlinear functions. With respect to gradient estimation, the characteristic trait of the linear structure is that the gradient is constant:

$$\begin{aligned} \underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{a}},\hat{b}}{\text{maximize}} \quad & \prod_{i=1}^{n} p(\hat{w}_i) \\ \text{subject to} \quad & \hat{w}_i = y_i - \hat{y}_i, \quad i = 1,...,n \\ & \hat{y}_i = \hat{\mathbf{a}}^T \mathbf{u}_i + \hat{b}, \quad i = 1,...,n \\ & \nabla \hat{f}(\mathbf{u}_i) = \hat{\mathbf{a}}, \quad i = 1,...,n \\ & \underline{\kappa} \preceq \nabla \hat{f}(\mathbf{u}_i) \preceq \overline{\kappa}, \quad i = 1,...,n \end{aligned} \quad . \tag{7}$$

Quadratic Functions

The assumption of quadratic structure is also standard (it is, for example, particularly common in response surface methods[15]). Here, we consider the version without interaction terms between the variables, i.e. where the matrix of quadratic coefficients $\hat{\mathbf{Q}} \in \mathbb{R}^{n_u \times n_u}$ is diagonal:

$$\begin{aligned} \underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{Q}},\hat{\mathbf{a}},\hat{b}}{\text{maximize}} \quad & \prod_{i=1}^{n} p(\hat{w}_i) \\ \text{subject to} \quad & \hat{w}_i = y_i - \hat{y}_i, \quad i = 1,...,n \\ & \hat{y}_i = \mathbf{u}_i^T \hat{\mathbf{Q}} \mathbf{u}_i + \hat{\mathbf{a}}^T \mathbf{u}_i + \hat{b}, \quad i = 1,...,n \\ & \nabla \hat{f}(\mathbf{u}_i) = 2\hat{\mathbf{Q}} \mathbf{u}_i + \hat{\mathbf{a}}, \quad i = 1,...,n \\ & \underline{\kappa} \preceq \nabla \hat{f}(\mathbf{u}_i) \preceq \overline{\kappa}, \quad i = 1,...,n \end{aligned} \quad . \tag{8}$$

The choice of a *diagonal* $\hat{\mathbf{Q}}$ is motivated by the fact that including interaction terms, while entirely valid, significantly weakens the regularization power of the structure (the degrees of freedom scaling quadratically in n_u rather than linearly). As with the linearity assumption above, we note that the parametric nature of $\hat{\mathbf{d}}$ (which is, in this case, $\{\hat{\mathbf{Q}}, \hat{\mathbf{a}}, \hat{b}\}$) leads to equality constraints on the gradients.

Convex/Concave Functions

This type of regularization, although not new conceptually,[11,16] appears to be much less common and uses the first-order convexity/concavity conditions to constrain the gradient. Unlike in the linear and quadratic cases, this structure is not parametric and is described by *inequalities*:

$$\begin{aligned} \underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}}}{\text{maximize}} \quad & \prod_{i=1}^{n} p(\hat{w}_i) \\ \text{subject to} \quad & \hat{w}_i = y_i - \hat{y}_i, \quad i = 1,...,n \\ & \hat{y}_i + \nabla \hat{f}(\mathbf{u}_i)^T (\mathbf{u}_j - \mathbf{u}_i) \leq \hat{y}_j, \quad i,j = 1,...,n \ (i \neq j) \\ & \underline{\kappa} \preceq \nabla \hat{f}(\mathbf{u}_i) \preceq \overline{\kappa}, \quad i = 1,...,n \end{aligned} \quad , \tag{9}$$

$$\begin{aligned} \underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}}}{\text{maximize}} \quad & \prod_{i=1}^{n} p(\hat{w}_i) \\ \text{subject to} \quad & \hat{w}_i = y_i - \hat{y}_i, \quad i = 1,...,n \\ & \hat{y}_i + \nabla \hat{f}(\mathbf{u}_i)^T (\mathbf{u}_j - \mathbf{u}_i) \geq \hat{y}_j, \quad i,j = 1,...,n \ (i \neq j) \\ & \underline{\kappa} \preceq \nabla \hat{f}(\mathbf{u}_i) \preceq \overline{\kappa}, \quad i = 1,...,n \end{aligned} \quad , \tag{10}$$

with (9) and (10) corresponding to the convex and concave cases, respectively.

Difference of Bounded Convex Functions

The difference of convex (DC) functions – or any function that can be expressed as the sum of a convex and a concave function – are well-trodden ground in the domain of global optimization,[17] but have not, to the best of the authors' knowledge, been proposed for use in regularization. This is not surprising, given the generality of DC functions (any twice continuously differentiable function is proven to be DC), as such a regularization would ultimately fail due to any measured data always being described perfectly by some DC structure. To make the structure restricting, and thereby apt for regularization, we propose to use "difference of bounded convex" (DBC) functions with additional box constraints on both the convex and concave components forcing them to lie inside the infinity norm of the data (with additional slack for noise), together with analogous constraints on the derivatives:

$$
\begin{aligned}
\underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{y}}_{cvx},\hat{\mathbf{y}}_{ccv},\hat{\mathbf{F}}_{cvx},\hat{\mathbf{F}}_{ccv}}{\text{maximize}} \quad & \prod_{i=1}^{n} p(\hat{w}_i) \\
\text{subject to} \quad & \hat{w}_i = y_i - \hat{y}_i, \quad i = 1,...,n \\
& \hat{\mathbf{y}} = \hat{\mathbf{y}}_{cvx} + \hat{\mathbf{y}}_{ccv} \\
& \hat{\mathbf{F}} = \hat{\mathbf{F}}_{cvx} + \hat{\mathbf{F}}_{ccv} \\
& \hat{y}_{cvx,i} + \nabla \hat{f}_{cvx}(\mathbf{u}_i)^T (\mathbf{u}_j - \mathbf{u}_i) \leq \hat{y}_{cvx,j}, \quad i,j = 1,...,n \; (i \neq j) \\
& \hat{y}_{ccv,i} + \nabla \hat{f}_{ccv}(\mathbf{u}_i)^T (\mathbf{u}_j - \mathbf{u}_i) \geq \hat{y}_{ccv,j}, \quad i,j = 1,...,n \; (i \neq j) \\
& -\|\mathbf{y}\|_\infty - \max\left(|\underline{w}|, |\overline{w}|\right) \leq \hat{y}_{cvx,i} \leq \|\mathbf{y}\|_\infty + \max\left(|\underline{w}|, |\overline{w}|\right), \\
& i = 1,...,n \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad , \qquad (11) \\
& -\|\mathbf{y}\|_\infty - \max\left(|\underline{w}|, |\overline{w}|\right) \leq \hat{y}_{ccv,i} \leq \|\mathbf{y}\|_\infty + \max\left(|\underline{w}|, |\overline{w}|\right), \\
& i = 1,...,n \\
& \underline{\boldsymbol{\kappa}} \preceq \nabla \hat{f}(\mathbf{u}_i) \preceq \overline{\boldsymbol{\kappa}}, \quad i = 1,...,n \\
& -\max\left(|\underline{\boldsymbol{\kappa}}_j|, |\overline{\boldsymbol{\kappa}}_j|\right) \leq \left.\frac{\partial \hat{f}_{cvx}}{\partial u_j}\right|_{\mathbf{u}_i} \leq \max\left(|\underline{\boldsymbol{\kappa}}_j|, |\overline{\boldsymbol{\kappa}}_j|\right) \\
& i = 1,...,n, \; j = 1,...,n_u \\
& -\max\left(|\underline{\boldsymbol{\kappa}}_j|, |\overline{\boldsymbol{\kappa}}_j|\right) \leq \left.\frac{\partial \hat{f}_{ccv}}{\partial u_j}\right|_{\mathbf{u}_i} \leq \max\left(|\underline{\boldsymbol{\kappa}}_j|, |\overline{\boldsymbol{\kappa}}_j|\right) \\
& i = 1,...,n, \; j = 1,...,n_u
\end{aligned}
$$

with $\underline{w}$ and $\overline{w}$ defined as $\{\underline{w}, \overline{w} : \text{prob}(w_i \leq \overline{w}) \geq 0.99, \text{prob}(w_i \geq \underline{w}) \geq 0.99\}$, i.e. as the 99%-probability worst-case lower and upper deviations due to the noise. Problem (11) is an extension of (9) and (10) in that it constructs bounded convex and concave parts and then uses their sum (the DC function) to regularize the data. Note that the choice to bound the convex/concave components and their derivatives as done here is just one option, other choices being possible. However, the particular definition of (11) allows us to obtain either (9) or (10) by simply setting either $\hat{\mathbf{y}}_{ccv}, \hat{\mathbf{F}}_{ccv}$ or $\hat{\mathbf{y}}_{cvx}, \hat{\mathbf{F}}_{cvx}$ to $\mathbf{0}$, respectively, thereby allowing the DBC structure to be seen as a generalization of the convex/concave ones.

Table 1 summarizes the computational aspects of Problems (7)-(11), while Table 2 establishes

the concrete links with the general formulation given in (3) by defining S and $\hat{\mathbf{d}}$ for each case.

Table 1: The computational aspects – number of variables, constraints (inequality, equality, and box), and degrees of freedom – for the different regularization problems. We note that the convex/concave and DBC regularizations are more prone to scale-up issues with an increasing number of data points, as the number of inequality constraints grows quadratically in n.

Structure	Variables	Inequalities	Equalities	Box	DoF
Linear	$n(n_u+2)+n_u+1$	$-$	$n(n_u+2)$	$2nn_u$	n_u+1
Quadratic	$n(n_u+2)+2n_u+1$	$-$	$n(n_u+2)$	$2nn_u$	$2n_u+1$
Convex/Concave	$n(n_u+2)$	$n(n-1)$	n	$2nn_u$	$n(n_u+1)$
DBC	$n(3n_u+4)$	$2n(n-1)$	$n(n_u+2)$	$2n(3n_u+2)$	$2n(n_u+1)$

The choice of these four structures is strategic in that one sees a progression in generality, as shown below:

$$\text{linear} \subseteq \begin{array}{c} \text{quadratic (semidefinite)} \subseteq \text{convex/concave} \subseteq \text{DBC} \\ \text{quadratic (indefinite)} \end{array}, \tag{12}$$

with the class of DBC functions subsuming everything but the indefinite quadratic, and the class of linear functions being the least general and subsumed by all of the others. The natural consequence of this order is that we would expect DBC functions to be valid for many more cases than convex/concave or quadratic functions, which should in turn be valid more often than linear functions (it should be noted that, although we are unable to prove that DBC functions as we have defined them will always be more general than *all* quadratics, this often appears to be the case and will be confirmed in the numerical trials later). Generality comes, however, at the cost of regularization power – regularization using linear functions is the tightest (note that the gradient is constrained by equality and that the number of degrees of freedom in the regularization problem is the lowest by far), while DBC regularization is very loose (seen, again, purely by the number of degrees of freedom given to the regularization). Quadratic and convex/concave structures represent, in some sense, a middle ground between these two.

Another, more practical, justification for these choices is that they all yield S that are convex polyhedra and thereby avoid major tractability issues. As an example of an extension with such

Table 2: Definitions of S and $\hat{\mathbf{d}}$ for the regularizations given in (7)-(11).

Structure	S	$\hat{\mathbf{d}}$
Linear	$\{\hat{\mathbf{y}},\hat{\mathbf{F}},\hat{\mathbf{a}},\hat{b}:\hat{y}_i=\hat{\mathbf{a}}^T\mathbf{u}_i+\hat{b},\ i=1,...,n;$ $\nabla\hat{f}(\mathbf{u}_i)=\hat{\mathbf{a}},\ i=1,...,n;$ $\underline{\kappa}\preceq\nabla\hat{f}(\mathbf{u}_i)\preceq\overline{\kappa},\ i=1,...,n\}$	$\{\hat{\mathbf{a}},\hat{b}\}$
Quadratic	$\{\hat{\mathbf{y}},\hat{\mathbf{F}},\hat{\mathbf{Q}},\hat{\mathbf{a}},\hat{b}:\hat{y}_i=\mathbf{u}_i^T\,\hat{\mathbf{Q}}\,\mathbf{u}_i+\hat{\mathbf{a}}^T\mathbf{u}_i+\hat{b},\ i=1,...,n;$ $\nabla\hat{f}(\mathbf{u}_i)=2\,\hat{\mathbf{Q}}\,\mathbf{u}_i+\hat{\mathbf{a}},\ i=1,...,n;$ $\underline{\kappa}\preceq\nabla\hat{f}(\mathbf{u}_i)\preceq\overline{\kappa},\ i=1,...,n\}$	$\{\hat{\mathbf{Q}},\hat{\mathbf{a}},\hat{b}\}$
Convex	$\{\hat{\mathbf{y}},\hat{\mathbf{F}}:\hat{y}_i+\nabla\hat{f}(\mathbf{u}_i)^T(\mathbf{u}_j-\mathbf{u}_i)\leq\hat{y}_j,\ i,j=1,...,n\ (i\neq j);$ $\underline{\kappa}\preceq\nabla\hat{f}(\mathbf{u}_i)\preceq\overline{\kappa},\ i=1,...,n\}$	$\varnothing$
Concave	$\{\hat{\mathbf{y}},\hat{\mathbf{F}}:\hat{y}_i+\nabla\hat{f}(\mathbf{u}_i)^T(\mathbf{u}_j-\mathbf{u}_i)\geq\hat{y}_j,\ i,j=1,...,n\ (i\neq j);$ $\underline{\kappa}\preceq\nabla\hat{f}(\mathbf{u}_i)\preceq\overline{\kappa},\ i=1,...,n\}$	$\varnothing$
DBC	$\{\hat{\mathbf{y}},\hat{\mathbf{F}},\hat{\mathbf{y}}_{cvx},\hat{\mathbf{y}}_{ccv},\hat{\mathbf{F}}_{cvx},\hat{\mathbf{F}}_{ccv}:\hat{\mathbf{y}}=\hat{\mathbf{y}}_{cvx}+\hat{\mathbf{y}}_{ccv};\hat{\mathbf{F}}=\hat{\mathbf{F}}_{cvx}+\hat{\mathbf{F}}_{ccv};$ $\hat{y}_{cvx,i}+\nabla\hat{f}_{cvx}(\mathbf{u}_i)^T(\mathbf{u}_j-\mathbf{u}_i)\leq\hat{y}_{cvx,j},\ i,j=1,...,n\ (i\neq j);$ $\hat{y}_{ccv,i}+\nabla\hat{f}_{ccv}(\mathbf{u}_i)^T(\mathbf{u}_j-\mathbf{u}_i)\geq\hat{y}_{ccv,j},\ i,j=1,...,n\ (i\neq j);$ $-\|\mathbf{y}\|_\infty-\max(\lvert\underline{w}\rvert,\lvert\overline{w}\rvert)\leq\hat{y}_{cvx,i}\leq\|\mathbf{y}\|_\infty+\max(\lvert\underline{w}\rvert,\lvert\overline{w}\rvert),\ i=1,...,n;$ $-\|\mathbf{y}\|_\infty-\max(\lvert\underline{w}\rvert,\lvert\overline{w}\rvert)\leq\hat{y}_{ccv,i}\leq\|\mathbf{y}\|_\infty+\max(\lvert\underline{w}\rvert,\lvert\overline{w}\rvert),\ i=1,...,n;$ $\underline{\kappa}\preceq\nabla\hat{f}(\mathbf{u}_i)\preceq\overline{\kappa},\ i=1,...,n;$ $-\max\left(\lvert\underline{\kappa}_j\rvert,\lvert\overline{\kappa}_j\rvert\right)\leq\left.\dfrac{\partial\hat{f}_{cvx}}{\partial u_j}\right\rvert_{\mathbf{u}_i}\leq\max\left(\lvert\underline{\kappa}_j\rvert,\lvert\overline{\kappa}_j\rvert\right),$ $i=1,...,n,\ j=1,...,n_u;$ $-\max\left(\lvert\underline{\kappa}_j\rvert,\lvert\overline{\kappa}_j\rvert\right)\leq\left.\dfrac{\partial\hat{f}_{ccv}}{\partial u_j}\right\rvert_{\mathbf{u}_i}\leq\max\left(\lvert\underline{\kappa}_j\rvert,\lvert\overline{\kappa}_j\rvert\right),$ $i=1,...,n,\ j=1,...,n_u\}$	$\{\hat{\mathbf{y}}_{cvx},\hat{\mathbf{y}}_{ccv},$ $\hat{\mathbf{F}}_{cvx},\hat{\mathbf{F}}_{ccv}\}$

issues, we note that past work by the authors has shown *quasiconvex* structures to be useful for bounding the gradient as well.[18] Unfortunately, regularizing with a quasiconvex or quasiconcave function generally leads to an intractable S unless certain relaxations are made (a combinatorial algorithm to perform such a regularization is currently under development, however).

Calculating the Bounds on the True Gradient

We note that assuming a certain structure to be valid for a set of n data points (i.e. not dismissing it as P-invalid) is already sufficient to put implicit bounds on the gradient estimates $\hat{\mathbf{F}}$, as they are restricted to those values that belong to $\mathscr{E}$. To make these bounds explicit and more suited for use in applications, we simply propose to calculate the minimal and maximal values of each derivative over $\mathscr{E}$ for some specified $\mathbf{u}_i$:

$$\frac{\partial \underline{f}}{\partial u_j}\Big|_{\mathbf{u}_i} := \min_{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{d}}\in\mathscr{E}} \quad \frac{\partial \hat{f}}{\partial u_j}\Big|_{\mathbf{u}_i}, \quad j=1,...,n_u \quad , \tag{13}$$

$$\frac{\partial \overline{f}}{\partial u_j}\Big|_{\mathbf{u}_i} := \max_{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{d}}\in\mathscr{E}} \quad \frac{\partial \hat{f}}{\partial u_j}\Big|_{\mathbf{u}_i}, \quad j=1,...,n_u \quad , \tag{14}$$

which represents a conservative but robust outer approximation, as we essentially construct a box around all of the feasible values that each gradient may take.

We now prove the following result regarding the accuracy of these bounds with respect to the true gradient $\nabla f(\mathbf{u}_i)$.

Theorem 1 (Accuracy of Calculated Bounds)**.** *Let $f(\mathbf{u})$ have the structure S and let $\nabla\underline{f}(\mathbf{u}_i)$ and $\nabla\overline{f}(\mathbf{u}_i)$ be calculated according to (13) and (14). Then, $\nabla\underline{f}(\mathbf{u}_i) \preceq \nabla f(\mathbf{u}_i) \preceq \nabla\overline{f}(\mathbf{u}_i)$ is satisfied with a probability of at least P.*

Proof. Using the same argument as in Lemma 2, we consider the choice $\hat{y}_i = f(\mathbf{u}_i), \hat{w}_i = w_i, \nabla\hat{f}(\mathbf{u}_i) = \nabla f(\mathbf{u}_i), i = 1,...,n$, $\hat{\mathbf{d}} = \mathbf{d}$, which we know belongs to $\mathscr{E}$ with a probability of P.

Supposing that this choice belongs to $\mathscr{E}$, we now prove that $\nabla\underline{f}(\mathbf{u}_i) \preceq \nabla f(\mathbf{u}_i) \preceq \nabla\overline{f}(\mathbf{u}_i)$ with a probability of 1. Proceeding by contradiction, we suppose that $\nabla\underline{f}(\mathbf{u}_i) \preceq \nabla f(\mathbf{u}_i) \preceq \nabla\overline{f}(\mathbf{u}_i)$ does

not hold, which implies that at least one of the inequalities $\frac{\partial \underline{f}}{\partial u_j}\Big|_{\mathbf{u}_i} \leq \frac{\partial f}{\partial u_j}\Big|_{\mathbf{u}_i} \leq \frac{\partial \overline{f}}{\partial u_j}\Big|_{\mathbf{u}_i}$ is not met. We choose to suppose that the lower bound is violated, with $\frac{\partial \underline{f}}{\partial u_j}\Big|_{\mathbf{u}_i} > \frac{\partial f}{\partial u_j}\Big|_{\mathbf{u}_i}$ (a symmetrical argument follows for the upper bound). If this is true, then, as there is no element in $\mathscr{E}$ with a value strictly inferior to $\frac{\partial \underline{f}}{\partial u_j}\Big|_{\mathbf{u}_i}$, it follows that there is no element in $\mathscr{E}$ with a value equal or inferior to $\frac{\partial f}{\partial u_j}\Big|_{\mathbf{u}_i}$. However, this is clearly false since the chosen point belongs to $\mathscr{E}$ and has a value of $\frac{\partial \hat{f}}{\partial u_j}\Big|_{\mathbf{u}_i} = \frac{\partial f}{\partial u_j}\Big|_{\mathbf{u}_i}$.

We have thereby provided one choice of variables for which the joint probability of their belonging to $\mathscr{E}$ and of $\nabla \underline{f}(\mathbf{u}_i) \preceq \nabla f(\mathbf{u}_i) \preceq \nabla \overline{f}(\mathbf{u}_i)$ is P. As other choices may also exist, it follows that the probability of $\nabla \underline{f}(\mathbf{u}_i) \preceq \nabla f(\mathbf{u}_i) \preceq \nabla \overline{f}(\mathbf{u}_i)$ is at least P. □

The practical significance of Theorem 1 is to guarantee that, when the chosen structure is correct, (13) and (14) yield *accurate* bounds that contain the true gradient with a probability of at least P.

An Algorithm for Computing Gradient Bounds

As only very general structures will be valid for all of the N available data points when the data cover a significant portion of the input space, we propose an algorithm to *locally* search and discover the neighborhood where the proposed structure is not P-invalid. Once such a neighborhood is found, we compute the gradient bounds over the set $\mathscr{E}$ for that neighborhood by employing the methods described in the previous section.

The algorithm to compute $\nabla \underline{f}(\overline{\mathbf{u}})$ and $\nabla \overline{f}(\overline{\mathbf{u}})$ for the point $\overline{\mathbf{u}}$ given the input set $\mathbf{u}_1, ..., \mathbf{u}_N$ and the output measurements $y_1, ..., y_N$ is as follows:

1. Choose a value of P and a structure S (e.g. one of the four structures given in the preceding section). Compute U as the set of all input points ordered increasingly in their Euclidean distance from $\overline{\mathbf{u}}$ (with $\overline{\mathbf{u}}$ being the first element). Likewise, apply the same ordering to the output measurements $\mathbf{y}$ to compute Y. Let $\tilde{(\cdot)}$ denote the change of indices due to the ordering (e.g., with $\tilde{\mathbf{u}}_1 = \overline{\mathbf{u}}$ and $\tilde{y}_1 = \overline{y}$, where $\overline{y}$ is the measurement corresponding to $\overline{\mathbf{u}}$). Since this also affects the definition of S, denote by $\tilde{S}$ the structural constraints with the ordering applied

(note that the same definitions as in Table 2 may be used for $\tilde{S}$ if the substitutions $\mathbf{u} \to \tilde{\mathbf{u}}$ are made).

2. Let n_L and n_U denote the lower and upper bounds, respectively, on the number of points closest to $\overline{\mathbf{u}}$ for which the chosen structure is not P-invalid. Set $n_L := 1$ and $n_U := N$.

3. Define the test number $n_T := \lceil 0.5n_L + 0.5n_U \rceil$ if $n_U > 2$ and $n_T := 1$ if $n_U = 2$, and define U_T and Y_T as the subsets of U and Y with the first n_T elements retained. Likewise, let $\tilde{S}_T$ denote the structural constraints corresponding to only n_T data points.

4. Use Monte Carlo sampling to approximate $L_{P,T}$ – the value of the L_P bound for n_T data points.

5. Check the P-invalidity of the structure on U_T by solving the following log-likelihood version of Problem (3) to global optimality:

$$\begin{aligned} \hat{\mathbf{y}}^*, \hat{\mathbf{w}}^*, \hat{\mathbf{F}}^*, \hat{\mathbf{d}}^* = \arg\underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{d}}}{\text{maximize}} \quad & \sum_{i=1}^{n_T} \log p(\hat{w}_i) \\ \text{subject to} \quad & \hat{w}_i = \tilde{y}_i - \hat{y}_i, \quad i = 1, ..., n_T \\ & \hat{\mathbf{y}}, \hat{\mathbf{F}}, \hat{\mathbf{d}} \in \tilde{S}_T \end{aligned} \quad . \tag{15}$$

If $\sum_{i=1}^{n_T} \log p(\hat{w}_i^*) < \log L_{P,T}$, then the structure is P-invalid on U_T, in which case set $n_U := n_T$. Otherwise, set $n_L := n_T$. If $n_U - n_L < 2$, proceed to Step 6. Otherwise, return to Step 3.

6. Compute the gradient bounds $\nabla \underline{f}(\overline{\mathbf{u}})$ and $\nabla \overline{f}(\overline{\mathbf{u}})$ by solving the following optimization problems to global optimality for each of the input variables ($j = 1, ..., n_u$):

$$\begin{aligned} \frac{\partial \underline{f}}{\partial u_j}\Big|_{\overline{\mathbf{u}}} = \arg\underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{d}}}{\text{minimize}} \quad & \frac{\partial \hat{f}}{\partial u_j}\Big|_{\tilde{\mathbf{u}}_1} \\ \text{subject to} \quad & \hat{w}_i = \tilde{y}_i - \hat{y}_i, \quad i = 1, ..., n_L \\ & \hat{\mathbf{y}}, \hat{\mathbf{F}}, \hat{\mathbf{d}} \in \tilde{S}_L \\ & \sum_{i=1}^{n_L} \log p(\hat{w}_i) \geq \log L_{P,L} \end{aligned} \quad , \tag{16}$$

$$\left.\frac{\partial \overline{f}}{\partial u_j}\right|_{\overline{\mathbf{u}}} = \underset{\hat{\mathbf{y}},\hat{\mathbf{w}},\hat{\mathbf{F}},\hat{\mathbf{d}}}{\arg\,\text{maximize}} \quad \left.\frac{\partial \hat{f}}{\partial u_j}\right|_{\tilde{\mathbf{u}}_1}$$
$$\text{subject to} \quad \hat{w}_i = \tilde{y}_i - \hat{y}_i, \quad i = 1,...,n_L$$
$$\hat{\mathbf{y}},\hat{\mathbf{F}},\hat{\mathbf{d}} \in \tilde{S}_L$$
$$\sum_{i=1}^{n_L} \log p(\hat{w}_i) \geq \log L_{P,L} \qquad (17)$$

We now make some remarks regarding the algorithm's implementation.

- The gradient bounds computed in Step 6 are guaranteed to be accurate with a probability of at least P when the assumed structure over U_L matches the structure of the true function (Theorem 1).

- The overall computational burden may be summarized as follows:

  - Both re-ordering the points with respect to $\overline{\mathbf{u}}$, as well as approximating $L_{P,T}$, are negligible with respect to the other computations (in our experience, 1000 sets of n_T samples are more than sufficient for a good estimate of $L_{P,T}$ in the Monte Carlo sampling).

  - Steps 3-5 essentially describe a bisection algorithm, and thus (15) needs to be solved, approximately, $\log_2(N-1)$ times.

  - As Problems (15), (16), and (17) require global solutions, the practical implication that follows is that the pdf $p(w)$ be log-concave (e.g. Gaussian, Laplacian, or uniform) if these problems are to be solved efficiently via convex optimization methods. [11] When this is so, the main computational burden of our algorithm is approximately $2n_u + \log_2(N-1)$ convex optimization problems. When the pdf is not log-concave, true global optimization methods are required and the computational cost may grow drastically. A similar difficulty is encountered if structures resulting in a nonconvex S are chosen.

- The algorithm supposes knowledge of the noise pdf, whose availability in practice depends on the ease with which the sample noise can be modeled or obtained from experiments.

- The algorithm does not provide an estimate of the gradient, although one is readily obtained for the linear or quadratic structures if (7) or (8) are solved. For the non-parametric structures, it is up to the user to decide what the estimate should be, with the bounds obtained by the algorithm acting to limit the decision space.

- Any choice of the structures presented here is guaranteed to not be declared as P-invalid for $n_T := 1$, as a regularization with a single point is essentially free of structural constraints (the point can take any value while satisfying these constraints). This serves to guarantee that the bisection algorithm converges, as there exists some $n_T \geq 1$ for which the structure is not P-invalid. Practically, larger values of n_T are desired as they result in stronger regularizations with more constraints.

- The initial choice of n_T, which is by definition $\lceil 0.5 + 0.5N \rceil$ here (assuming $N > 2$), could be modified if the user expects S to be valid for either a small or large number of points, in which case n_T could be initialized as 1 or as N, respectively. This is unlikely to be crucial, but could speed up the bisection algorithm.

Numerical Studies

Using the four structures proposed, the algorithm is tested on noisy data generated from the following six functions (Figure 3):

$$\begin{aligned} f_1(\mathbf{u}) &= u_1 + u_2 \\ f_2(\mathbf{u}) &= u_1^2 - u_2^2 \\ f_3(\mathbf{u}) &= u_1^4 + u_2^4 \\ f_4(\mathbf{u}) &= u_1^2 u_2^2 + 5\sin(3u_1)\sin(3u_2) \\ f_5(\mathbf{u}) &= u_2 \log(u_1 + 3) \\ f_6(\mathbf{u}) &= u_1 u_2^3 \end{aligned} \quad , \tag{18}$$

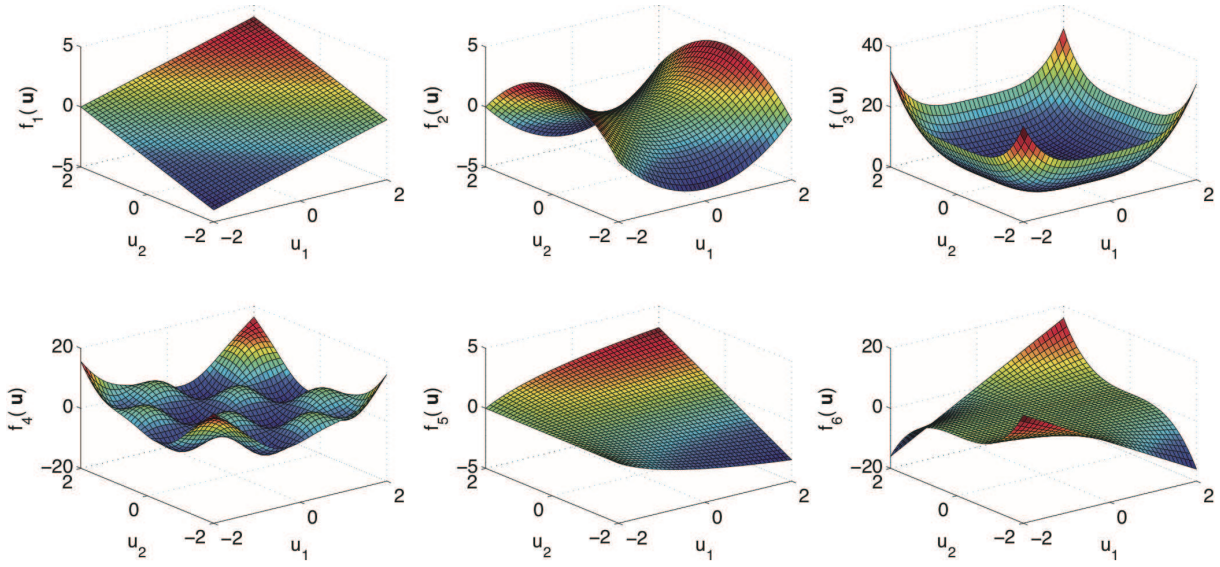


Figure 3: The six functions used to test the algorithm for computing gradient bounds.

where we restrict the relevant input space to $-\mathbf{2} \preceq \mathbf{u} \preceq \mathbf{2}$ and consider a total of 100 points regularly distributed on a 10 by 10 grid defined by the Cartesian product of $[-2, -\frac{14}{9}, ..., \frac{14}{9}, 2] \times [-2, -\frac{14}{9}, ..., \frac{14}{9}, 2]$. The functions $f_1(\mathbf{u}), f_2(\mathbf{u})$, and $f_3(\mathbf{u})$ represent functions where some of the structural assumptions are correct: $f_1(\mathbf{u})$ is linear, quadratic, convex/concave, and DBC, $f_2(\mathbf{u})$ is quadratic and DBC, and $f_3(\mathbf{u})$ is convex and DBC. The functions $f_4(\mathbf{u}), f_5(\mathbf{u})$, and $f_6(\mathbf{u})$ are of more practical interest as they represent functions where none of the structural hypotheses hold – they are neither linear, quadratic, nor convex/concave. From a quick analytical inspection it is unlikely that they are DBC either – none of the three can be written as a sum of a convex and concave function, let alone a convex and concave function with boundedness constraints. The purpose of these three functions is thus to test the accuracy and precision of the bounded estimates when the structural hypotheses are not satisfied (i.e. when Theorem 1 does not apply).

Prior to defining our metrics for accuracy and precision, we give, in Table 3, the Lipschitz constants for the six functions, defined here as twice the minimal and maximal derivatives over the relevant input space. As mentioned before, the Lipschitz constants act as ultimate, though not very precise, global bounds on the gradient, and are used to bound the gradient estimates in the regularization. In this study, they also serve a second purpose and are used for scaling the computed

bounds so as to evaluate their accuracy and precision (α_a and α_p, respectively) as follows:

$$\begin{aligned}\alpha_a &:= 1 - \frac{1}{n_u}\sum_{j=1}^{n_u}\frac{1}{\overline{\kappa}_j - \underline{\kappa}_j}\min_{\frac{\partial g}{\partial u_j}\in\left[\frac{\partial \underline{f}}{\partial u_j}\Big|_{\overline{\mathbf{u}}},\frac{\partial \overline{f}}{\partial u_j}\Big|_{\overline{\mathbf{u}}}\right]}\left\|\frac{\partial g}{\partial u_j} - \frac{\partial f}{\partial u_j}\Big|_{\overline{\mathbf{u}}}\right\|_1 \\ \alpha_p &:= 1 - \frac{1}{n_u}\sum_{j=1}^{n_u}\frac{1}{\overline{\kappa}_j - \underline{\kappa}_j}\left(\frac{\partial \overline{f}}{\partial u_j}\Big|_{\overline{\mathbf{u}}} - \frac{\partial \underline{f}}{\partial u_j}\Big|_{\overline{\mathbf{u}}}\right)\end{aligned}, \tag{19}$$

where the accuracy measure is defined as the average distance by which the true gradient falls outside of the calculated bounds ($\alpha_a = 1$ if it lies within them) scaled by the respective ranges. The precision measure reflects the average reduction in distance between the lower and upper bounds that is achieved with respect to the full range ($\alpha_p = 1$ if the bounds are tight and define a point, and $\alpha_p = 0$ if they are simply the lower and upper Lipschitz constants). Figure 4 illustrates these definitions geometrically.

Table 3: The Lipschitz constants used for the numerical studies.

	$\underline{\kappa}$	$\overline{\kappa}$
$f_1(\mathbf{u})$	$[-2;-2]$	$[2;2]$
$f_2(\mathbf{u})$	$[-8;-8]$	$[8;8]$
$f_3(\mathbf{u})$	$[-64;-64]$	$[64;64]$
$f_4(\mathbf{u})$	$[-62;-62]$	$[62;62]$
$f_5(\mathbf{u})$	$[-4;-2\log 5]$	$[4;2\log 5]$
$f_6(\mathbf{u})$	$[-16;-48]$	$[16;48]$

We corrupt the measurement at each point with additive Gaussian noise, $w_i \in \mathcal{N}(0,\sigma^2)$, $i = 1,...,100$, and vary the standard deviation as $\sigma = 0.1, 0.4, 0.7, 1.0$ to test for the effect of noise on the estimates.

With regard to implementation specifics, the algorithm of Section 3 is always run once for *all* of the available data points (i.e. by setting each point as $\overline{\mathbf{u}}$ once) and the results averaged – unless otherwise stated, we do not focus on the quality of the calculated bounds for any particular $\overline{\mathbf{u}}$. A P value of 0.99 is used unless stated otherwise. For the solution of (15), (16), and (17) we employed the CVX-SeDuMi modeling-solver package[19,20] in the MATLAB interface. In the rare ($< 1\%$) cases where the solver encountered numerical difficulties and failed to yield a solution, we simply left the results out of the averaging.

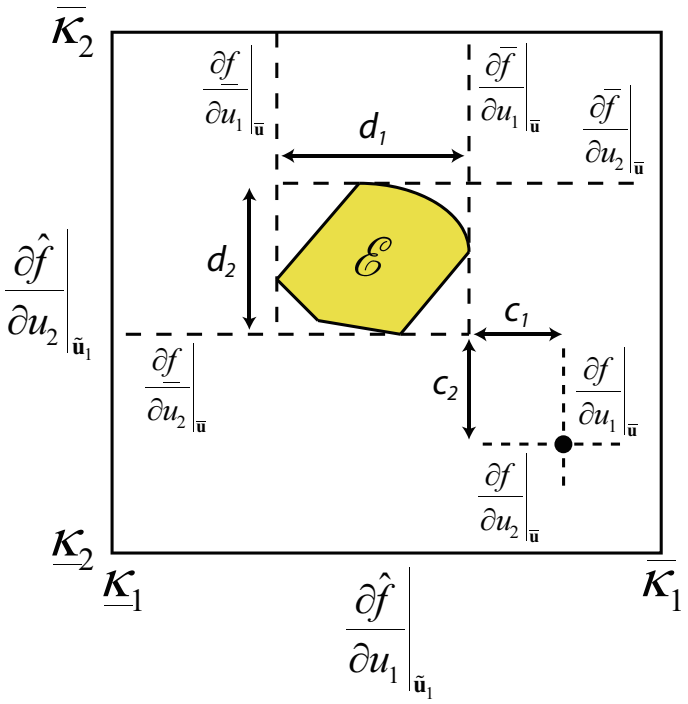


Figure 4: A geometric illustration of how accuracy and precision were calculated in this study for the gradient bounds obtained from (16) and (17). Here, the true gradient (the point in the bottom right) falls outside of the box constructed around the constrained space $\mathscr{E}$. By our metric, the accuracy of the bounds is calculated as $\alpha_a = 1 - 0.5\frac{c_1}{\overline{\kappa}_1 - \underline{\kappa}_1} - 0.5\frac{c_2}{\overline{\kappa}_2 - \underline{\kappa}_2}$. The precision, which reflects the size of the calculated box in comparison to the original one defined by the Lipschitz constants, is calculated as $\alpha_p = 1 - 0.5\frac{d_1}{\overline{\kappa}_1 - \underline{\kappa}_1} - 0.5\frac{d_2}{\overline{\kappa}_2 - \underline{\kappa}_2}$.

Finally, for the case where the structure is assumed to be convex/concave, we perform the regularization with both convex and concave functions in parallel and then simply choose the one that has the larger n_L. If n_L is the same for both, we choose the function that has the larger ML value for the n_L data points. As such, the worst-case computational burden is doubled for this specific choice of S. The justification for running the convex/concave regularization in parallel (i.e. lumping both convex and concave into a single structural hypothesis) comes from the fact that the two are mutually exclusive unless the function is linear.

We proceed to report the results for the different tests.

Study 1: General Trends in Algorithm Performance

To obtain a good idea of the average quality of the estimated bounds, we conduct a three-level study with the levels being:

- the function $f(\mathbf{u})$ used to generate the data (6 choices),
- the regularizing structure S (4 choices),
- the noise level σ (4 choices),

for a total of 96 cases. The results for $f_1(\mathbf{u})$ and $f_4(\mathbf{u})$, considered as the most representative, are given in Tables 4 and 5 (we refer the reader to the Supporting Information for the results for the other functions).

Table 4: Results of Study 1 for $f_1(\mathbf{u})$, expressed in the form of calculated mean $\pm$ calculated standard deviation.

	Linear	Quadratic	Convex/Concave	DBC
$\sigma = 0.1$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.9722 \pm 0.0014$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.9332 \pm 0.0197$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.7367 \pm 0.1664$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0065 \pm 0.0177$
$\sigma = 0.4$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.9003 \pm 0.0040$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.7551 \pm 0.0732$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.4729 \pm 0.1517$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0000 \pm 0.0000$
$\sigma = 0.7$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.7964 \pm 0.0053$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.5976 \pm 0.0966$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.3302 \pm 0.1577$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0000 \pm 0.0000$
$\sigma = 1.0$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.7745 \pm 0.0082$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.5679 \pm 0.1001$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.2728 \pm 0.1640$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0000 \pm 0.0000$

Table 5: Results of Study 1 for $f_4(\mathbf{u})$.

	Linear	Quadratic	Convex/Concave	DBC
$\sigma = 0.1$	$\alpha_a = 0.9769 \pm 0.0174$ $\alpha_p = 0.9303 \pm 0.1484$	$\alpha_a = 0.9948 \pm 0.0049$ $\alpha_p = 0.8259 \pm 0.2323$	$\alpha_a = 0.9978 \pm 0.0055$ $\alpha_p = 0.8437 \pm 0.1141$	$\alpha_a = 0.9998 \pm 0.0012$ $\alpha_p = 0.4386 \pm 0.3205$
$\sigma = 0.4$	$\alpha_a = 0.9914 \pm 0.0098$ $\alpha_p = 0.9364 \pm 0.0874$	$\alpha_a = 0.9959 \pm 0.0100$ $\alpha_p = 0.8276 \pm 0.2198$	$\alpha_a = 0.9981 \pm 0.0058$ $\alpha_p = 0.8104 \pm 0.1166$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.1993 \pm 0.1929$
$\sigma = 0.7$	$\alpha_a = 0.9930 \pm 0.0090$ $\alpha_p = 0.9421 \pm 0.0342$	$\alpha_a = 0.9923 \pm 0.0124$ $\alpha_p = 0.8786 \pm 0.1739$	$\alpha_a = 0.9965 \pm 0.0074$ $\alpha_p = 0.7984 \pm 0.1158$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.1063 \pm 0.1342$
$\sigma = 1.0$	$\alpha_a = 0.9866 \pm 0.0158$ $\alpha_p = 0.9454 \pm 0.0317$	$\alpha_a = 0.9876 \pm 0.0146$ $\alpha_p = 0.9214 \pm 0.1028$	$\alpha_a = 0.9933 \pm 0.0127$ $\alpha_p = 0.8117 \pm 0.1231$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0651 \pm 0.0895$

We start by cautioning the reader that these numbers may not be interpreted in any sort of absolute sense, as they are defined with respect to the Lipschitz constants and could be, in principle, made arbitrarily close to 1 by increasing the size of the box defined by $\overline{\kappa} - \underline{\kappa}$. Furthermore, we have no way of stating which accuracy or precision values are "good" or "satisfactory", as this is completely application-dependent and must be decided by the user – one should not, therefore, be misled by the high (>0.9500) values of α_a and α_p as being a testament to the high quality of the bounds. Having said this, we go on to highlight the general trends witnessed in the numerical trials:

- Theorem 1 is verified in all trials where the structural assumption is correct. This is witnessed by $\alpha_a = 1$ for all the relevant cases for the functions $f_1(\mathbf{u})$, $f_2(\mathbf{u})$, and $f_3(\mathbf{u})$ (even though there remains, at most, a 0.01 probability of inaccuracy). This property is not influenced by the noise level.

- When the assumed structure is correct, increasing the noise clearly reduces the precision of the gradient bounds (seen in all columns of Table 4, for example). The exact trend does not appear to be linear, however, and may vary with the structure and nature of the function. We also note that this trend does not appear to hold when the assumed structure is incorrect.

- The expected accuracy vs. precision tradeoff is observed, perhaps most clearly for $f_1(\mathbf{u})$ (see Table 4), where the linear structure gives the most precise bounds, followed by the quadratic, the convex/concave, and finally the DBC. On the other hand, we often see that the linear structure gives the least accurate estimates for functions where none of the structural

hypotheses are correct (see Table 5), with convex/concave often attaining a better accuracy than the quadratic in these cases. DBC functions justify their expected generality by giving perfect accuracy scores in almost all cases (the case with $\sigma = 0.1$ for $f_4(\mathbf{u})$ being the only exception). However, one also sees that, for the majority of the cases, DBC functions are no more useful than the Lipschitz constants, as they generally fail with respect to precision ($f_4(\mathbf{u})$ being the only case where a relatively large reduction appears to be achieved).

We attempt to give some general guidance based on these observations:

- If the structure of the true sampled function is known, then it is probably best to use this structure to regularize. If the estimated bounds are not tight enough to serve their desired purpose, use a more restricting, but less accurate, structure. If multiple valid structures are available, it is unequivocally best to use the more restricting one.

- It is unlikely that DBC functions would be useful for regularization in most cases, since good precision may often be needed. However, if the goal of the estimated bounds is to refine overly conservative Lipschitz constants (or to obtain an estimate of the Lipschitz constants from the available data), then the DBC approach may actually be quite useful as it appears to be generally accurate and as such would yield valid bounds.

- As a single, general-purpose, robust choice when the structure of the true function is unknown, the convex/concave regularization appears to achieve a respectable tradeoff between accuracy and precision in many cases. As such, it could be used as a first-choice regularizing structure for such cases. However, it is not difficult to envision a multiple-structure algorithm that is tailored to the application – i.e. one that starts with the least restricting functions (DBC) and proceeds down towards the most restricting ones (linear) until sufficiently tight bounds are obtained.

We now proceed to study the sensitivity of α_a and α_p with respect to other factors – namely, the value of P, the geometric location of the point being tested, the number of data points available,

and the dispersion of the points. So as not to overwhelm the reader with too many results, we only choose the specific case of $f_6(\mathbf{u})$ with $\sigma = 0.7$ as the basic reference for the studies that follow (this choice being somewhat arbitrary).

Study 2: Effect of P

As P plays a crucial role in checking the validity of a structure with regard to local data, we expect that lowering its value would lead to the following:

- Fewer data points used in the regularization (smaller n_L), since the odds of a structure being declared P-invalid increase as P is lowered (the constraint on the likelihood is tightened – see Definition 1).
- A higher probability of the bounds being incorrect when the structure is valid (by Theorem 1). A similar result may be expected when the structure is approximately correct (e.g. a linear structure over a small neighborhood).

Considering only the case of $f_6(\mathbf{u})$ and $\sigma = 0.7$, we set P to values of 0.10, 0.70, 0.80, and 0.90, with the results reported in Table 6. No major changes are observed when P is lowered from 0.99 to 0.70, and while some degradation in performance does seem to occur for $P = 0.10$ (lowered accuracy for the linear structure, lowered precision for both the linear and quadratic structures, more variation in precision for all structures), it does not appear to be drastic given the drastic change in P. As such, while we would rather be conservative and take $P = 0.99$, the effects of not doing so and using lower values do not appear to be too critical for the data set used here (suggesting, perhaps, that the importance of P is largely theoretical).

Study 3: Effect of Point Location

It is important to note that the (non-parametric) convex/concave and DBC structures only constrain the gradient by inequalities. While this is a reflection of their generality over the linear and

Table 6: Results of Study 2, with lower P values being used.

	Linear	Quadratic	Convex/Concave	DBC
$P = 0.99$	$\alpha_a = 0.9933 \pm 0.0149$	$\alpha_a = 0.9924 \pm 0.0144$	$\alpha_a = 0.9981 \pm 0.0076$	$\alpha_a = 1.0000 \pm 0.0000$
	$\alpha_p = 0.9441 \pm 0.0485$	$\alpha_p = 0.9274 \pm 0.0776$	$\alpha_p = 0.8326 \pm 0.1103$	$\alpha_p = 0.0300 \pm 0.0428$
$P = 0.90$	$\alpha_a = 0.9924 \pm 0.0135$	$\alpha_a = 0.9926 \pm 0.0149$	$\alpha_a = 0.9978 \pm 0.0057$	$\alpha_a = 1.0000 \pm 0.0000$
	$\alpha_p = 0.9446 \pm 0.0452$	$\alpha_p = 0.9274 \pm 0.0854$	$\alpha_p = 0.8275 \pm 0.1152$	$\alpha_p = 0.0332 \pm 0.0451$
$P = 0.80$	$\alpha_a = 0.9938 \pm 0.0126$	$\alpha_a = 0.9942 \pm 0.0133$	$\alpha_a = 0.9983 \pm 0.0055$	$\alpha_a = 1.0000 \pm 0.0000$
	$\alpha_p = 0.9417 \pm 0.0441$	$\alpha_p = 0.9084 \pm 0.1236$	$\alpha_p = 0.8248 \pm 0.1100$	$\alpha_p = 0.0326 \pm 0.0456$
$P = 0.70$	$\alpha_a = 0.9943 \pm 0.0109$	$\alpha_a = 0.9936 \pm 0.0151$	$\alpha_a = 0.9988 \pm 0.0037$	$\alpha_a = 1.0000 \pm 0.0000$
	$\alpha_p = 0.9420 \pm 0.0445$	$\alpha_p = 0.9012 \pm 0.1405$	$\alpha_p = 0.8258 \pm 0.1081$	$\alpha_p = 0.0328 \pm 0.0459$
$P = 0.10$	$\alpha_a = 0.9879 \pm 0.0147$	$\alpha_a = 0.9951 \pm 0.0181$	$\alpha_a = 0.9976 \pm 0.0128$	$\alpha_a = 1.0000 \pm 0.0000$
	$\alpha_p = 0.9295 \pm 0.1015$	$\alpha_p = 0.8544 \pm 0.1865$	$\alpha_p = 0.8218 \pm 0.1318$	$\alpha_p = 0.0368 \pm 0.0492$

quadratic assumptions, there is an added weakness in regularization with these structures that occurs whenever the point of interest is located on the edge of the input space. We illustrate this idea in Figure 5, where we have taken a one-dimensional case and generated data (with negligible noise) that clearly may be described by a convex function. As shown by the dashed lines, the assumption of convex structure constrains the gradient at the center point to lie in the sector described by the angle θ_1, as leaving this sector would imply that the gradient no longer supports the other points of the function (thereby violating the first-order condition of convexity). Furthermore, it is evident that the two neighboring points play a crucial role in constraining the gradient in such a manner. For the rightmost point, however, no such constraint exists to bound the maximal value of the gradient, which could then, in the worst case, stretch up to $+\infty$ (in our trials, this would simply be the upper Lipschitz constant). As a result, the precision of the gradient bounds achieved with these structures is likely to suffer significantly when $\overline{\mathbf{u}}$ is an edge or corner point.

We study this phenomenon by considering progressively smaller inner subsets of the generated data, where we retain all of the data but compute gradient bounds only for the inner sets as shown in Figure 6. Table 7 provides the results, which confirm our expectations, as significant improvements in precision are noted for both the convex/concave and DBC structures as we go deeper and deeper into the interior of the data set. Some improvement is noted for the linear and quadratic structures as well, but this is expected to be due to the fact that the inner layers of function $f_6(\mathbf{u})$ are progressively more linear (see Figure 3), thereby leading to better estimation with these structures.

It is difficult to make any practical recommendations with regard to this behavior when the

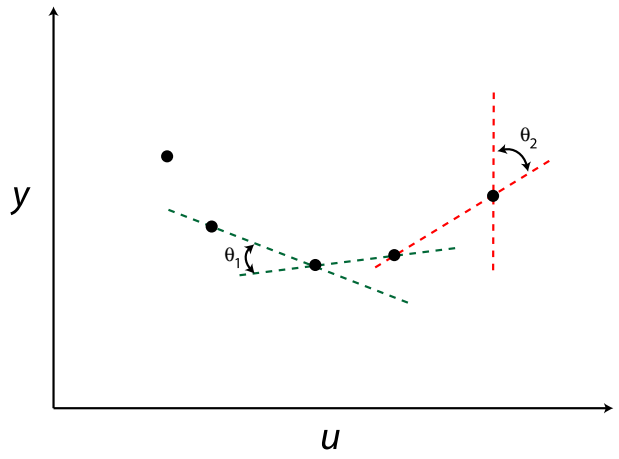


Figure 5: The necessity of having well-distributed neighboring points for obtaining precise gradient bounds in a convex regularization.

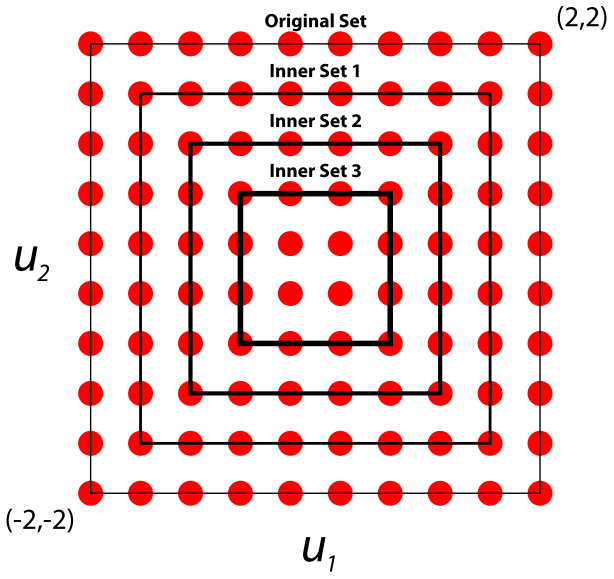


Figure 6: Sets that are deeper in the interior of the full data set are expected to yield more precise gradient bounds when regularized with convex/concave or DBC structures.

Table 7: Results of Study 3, which demonstrates the increased precision of bounds obtained for convex/concave and DBC structures when points deeper in the interior of the data set are considered.

	Linear	Quadratic	Convex/Concave	DBC
Original Set	$\alpha_a = 0.9933 \pm 0.0149$ $\alpha_p = 0.9441 \pm 0.0485$	$\alpha_a = 0.9924 \pm 0.0144$ $\alpha_p = 0.9274 \pm 0.0776$	$\alpha_a = 0.9981 \pm 0.0076$ $\alpha_p = 0.8326 \pm 0.1103$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0300 \pm 0.0428$
Inner Set 1	$\alpha_a = 0.9979 \pm 0.0046$ $\alpha_p = 0.9562 \pm 0.0343$	$\alpha_a = 0.9960 \pm 0.0072$ $\alpha_p = 0.9647 \pm 0.0226$	$\alpha_a = 0.9996 \pm 0.0011$ $\alpha_p = 0.8917 \pm 0.0662$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0420 \pm 0.0476$
Inner Set 2	$\alpha_a = 0.9973 \pm 0.0045$ $\alpha_p = 0.9714 \pm 0.0153$	$\alpha_a = 0.9954 \pm 0.0056$ $\alpha_p = 0.9741 \pm 0.0176$	$\alpha_a = 0.9992 \pm 0.0027$ $\alpha_p = 0.9369 \pm 0.0328$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0512 \pm 0.0466$
Inner Set 3	$\alpha_a = 0.9992 \pm 0.0011$ $\alpha_p = 0.9778 \pm 0.0093$	$\alpha_a = 0.9991 \pm 0.0009$ $\alpha_p = 0.9775 \pm 0.0111$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.9478 \pm 0.0195$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0771 \pm 0.0352$

input data set is fixed, apart from the encouragement to keep in mind that the precision of the estimates obtained by convex/concave or DBC regularization may vary significantly depending on how "well-surrounded" the data point in question is. When the input set is not fixed and can be changed (as is the case in optimization[7,8]), this behavior should motivate the user to choose input points that are somehow favorable to this regularization.

Study 4: Effect of Data Size

It is not difficult to surmise that having more (less) available data is likely to lead to better (worse) and tighter (looser) estimates, since every additional data point translates into a potential new constraint for the regularization problem. To confirm this formally, we carry out tests for coarser grids with 64, 36, and 16 points, noting that these are still regularly distributed across the space defined by $-\mathbf{2} \preceq \mathbf{u} \preceq \mathbf{2}$.

Table 8 provides the results, which show that the precision of the bounded estimates undergoes a consistent degradation as fewer and fewer data points are used, although this is not so evident with the linear structure, which appears to give fairly precise (though not necessarily accurate) estimates regardless of the data size. The accuracy of the bounded estimates is not affected significantly by using more or fewer data in this case.

Table 8: Results of Study 4, which show the effect of using less data on the quality of the bounds.

	Linear	Quadratic	Convex/Concave	DBC
100 points	$\alpha_a = 0.9933 \pm 0.0149$ $\alpha_p = 0.9441 \pm 0.0485$	$\alpha_a = 0.9924 \pm 0.0144$ $\alpha_p = 0.9274 \pm 0.0776$	$\alpha_a = 0.9981 \pm 0.0076$ $\alpha_p = 0.8326 \pm 0.1103$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0300 \pm 0.0428$
64 points	$\alpha_a = 0.9878 \pm 0.0216$ $\alpha_p = 0.9408 \pm 0.0496$	$\alpha_a = 0.9912 \pm 0.0141$ $\alpha_p = 0.8918 \pm 0.1322$	$\alpha_a = 0.9979 \pm 0.0057$ $\alpha_p = 0.8072 \pm 0.1200$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0201 \pm 0.0367$
36 points	$\alpha_a = 0.9905 \pm 0.0142$ $\alpha_p = 0.9201 \pm 0.0512$	$\alpha_a = 0.9932 \pm 0.0248$ $\alpha_p = 0.7440 \pm 0.2354$	$\alpha_a = 0.9983 \pm 0.0051$ $\alpha_p = 0.7872 \pm 0.1155$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0123 \pm 0.0232$
16 points	$\alpha_a = 0.9751 \pm 0.0191$ $\alpha_p = 0.9238 \pm 0.0372$	$\alpha_a = 0.9975 \pm 0.0045$ $\alpha_p = 0.5399 \pm 0.2547$	$\alpha_a = 0.9959 \pm 0.0111$ $\alpha_p = 0.7152 \pm 0.1266$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0031 \pm 0.0122$

The natural and expected recommendation that comes out of this study is that more experiments may be carried out if more precise bounds are desired.

Study 5: Effect of the Distribution of the Data

Finally, we consider the scenario where the points are irregularly distributed. In particular, we consider the effect of poorly-poised data for which the regularized maximum-likelihood fit is not uniquely defined (or where the likelihood value is very insensitive to perturbations near the maximum)[8] due to a lack of proper "excitation" in the different input directions. To study this, we generate the 100 data points ($i = 1, ..., 100$) in the following random manner:

$$\begin{aligned} u_{i,1} &\in \mathscr{U}(-2,2) \\ u_{i,2} &= u_{i,1} + \mathscr{N}(0,s^2) \\ u_{i,2} < -2 &\rightarrow u_{i,2} := -2 \\ u_{i,2} > 2 &\rightarrow u_{i,2} := 2 \end{aligned}, \tag{20}$$

with s acting as a scatter parameter. For $s = 0$ or very small, the data are not well-poised as they are one-dimensional (along $u_1 = u_2$), but become well-poised as s is increased, with a projection onto the box $-\mathbf{2} \preceq \mathbf{u} \preceq \mathbf{2}$ enforced if the generated point fails to fall inside. The cases tested, from $s = 0$ to $s = 0.4$, are illustrated in Figure 7 and the corresponding results are presented in Table 9.

Table 9: Results of Study 5, with irregularly dispersed data points.

	Linear	Quadratic	Convex/Concave	DBC
Standard	$\alpha_a = 0.9933 \pm 0.0149$ $\alpha_p = 0.9441 \pm 0.0485$	$\alpha_a = 0.9924 \pm 0.0144$ $\alpha_p = 0.9274 \pm 0.0776$	$\alpha_a = 0.9981 \pm 0.0076$ $\alpha_p = 0.8326 \pm 0.1103$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0300 \pm 0.0428$
$s = 0$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.3263 \pm 0.0231$	$\alpha_a = 0.9992 \pm 0.0067$ $\alpha_p = 0.3386 \pm 0.0797$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.2545 \pm 0.0413$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0315 \pm 0.0350$
$s = 0.1$	$\alpha_a = 0.9891 \pm 0.0214$ $\alpha_p = 0.8792 \pm 0.0590$	$\alpha_a = 0.9952 \pm 0.0116$ $\alpha_p = 0.8953 \pm 0.0545$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.2915 \pm 0.0569$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0121 \pm 0.0182$
$s = 0.2$	$\alpha_a = 0.9940 \pm 0.0113$ $\alpha_p = 0.9174 \pm 0.0459$	$\alpha_a = 0.9920 \pm 0.0182$ $\alpha_p = 0.9421 \pm 0.0301$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.4434 \pm 0.1146$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0173 \pm 0.0245$
$s = 0.3$	$\alpha_a = 0.9948 \pm 0.0132$ $\alpha_p = 0.9323 \pm 0.0440$	$\alpha_a = 0.9893 \pm 0.0171$ $\alpha_p = 0.9626 \pm 0.0227$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.5526 \pm 0.1428$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0125 \pm 0.0202$
$s = 0.4$	$\alpha_a = 0.9933 \pm 0.0126$ $\alpha_p = 0.9479 \pm 0.0344$	$\alpha_a = 0.9884 \pm 0.0156$ $\alpha_p = 0.9652 \pm 0.0246$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.6130 \pm 0.1584$	$\alpha_a = 1.0000 \pm 0.0000$ $\alpha_p = 0.0125 \pm 0.0214$

The observed effect of poisedness (in this case, the size of s) on the precision of the bounds is somewhat expected – better poised data lead to more precise bounds for all cases except those corresponding to the DBC structure (where the precision appears to be so low so as not to suffer

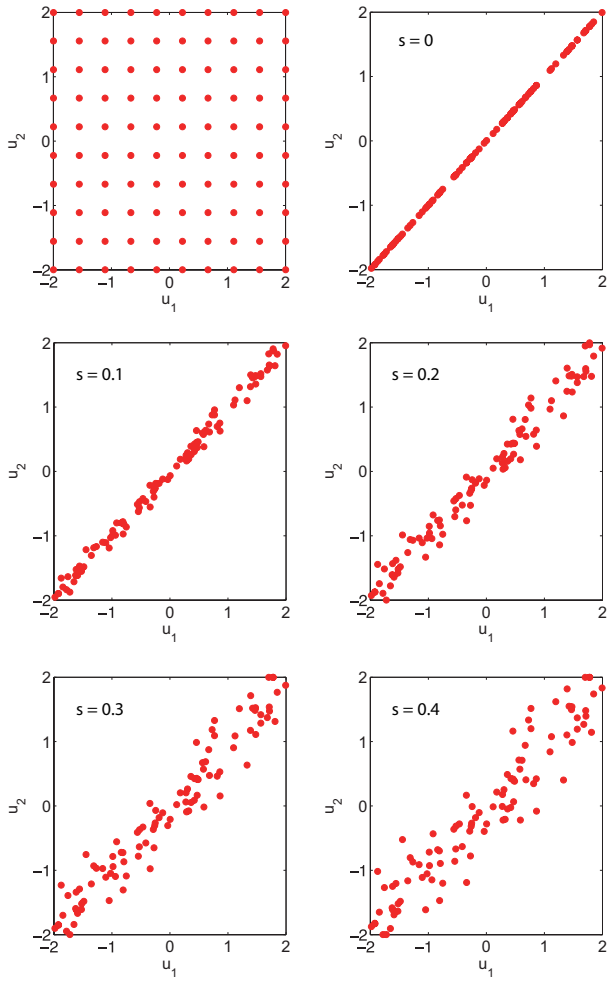


Figure 7: Cases with irregularly distributed data points for different values of the scatter parameter s (the top-left case being the standard).

much, if at all, from the lack of poisedness). The natural recommendation is therefore to make sure to work with well-poised data whenever possible, as failing to do so will lead to imprecision in the bounds.

We finish with a numerical warning concerning data points that are very close together in irregularly dispersed sets, as such a case may cause numerical issues in the regularization problem due to scaling. This is mostly relevant for the convex/concave and DBC structures, where constraints such as $\hat{y}_i + \nabla\hat{f}(\mathbf{u}_i)^T(\mathbf{u}_j - \mathbf{u}_i) \leq \hat{y}_j$ become extremely ill-conditioned as $\|\mathbf{u}_j - \mathbf{u}_i\| \to 0$. Either removing some of the data or binning it may be necessary in such cases.

Conclusions

This paper has discussed the problem of obtaining uncertainty bounds for the gradient estimate of a discretely sampled function at a given input point, and we have proposed the use of maximum-likelihood regularization subject to structural constraints as a means of obtaining useful gradient bounds. In particular, the problem appears to be largely solved for the cases where both the structure of the function and the pdf of the measurement noise are known, as here it is possible to guarantee that the obtained bounds be accurate with a chosen probability. This, together with the trend that more restricting structures lead to more precise bounds (smaller uncertainty intervals), has been confirmed in the numerical trials, with a number of other relevant effects being explored as well.

Many questions remain unanswered for the general case where the noise pdf is only approximated and where the structural hypotheses are false, however. Our numerical study has addressed the latter issue, showing that the DBC structural assumption is almost always accurate (but imprecise), and that the convex/concave structure is often a good compromise when more precision is needed. There are many cases where the linear and quadratic regularizations also perform comparably well with respect to accuracy, but it remains difficult to judge the results in general terms, as the quality of the uncertainty bounds, with regard to both accuracy and precision, will inevitably

vary from application to application. In this sense, we hope to have provided a number of options (the regularizing structures) for the potential user to try so as to find the one that is the best for them. Clearly, other general structures could also be proposed (e.g. quasiconvexity or log-convexity/concavity).

Finally, it is difficult to judge if the six functions tested here truly represent the full variety of cases that may be encountered in practice (a very likely answer is "no"). As such, the ideas presented could benefit from exposure to real applications, and some are already being carried out by the authors in the real-time optimization context. Another relevant aspect that has not been sufficiently treated here is dimensionality (only $n_u = 2$ was considered in the tests). Although neither the theory nor the algorithm appear to suffer from scale-up issues, this should nevertheless be confirmed.

Supporting Information Available

The results of Numerical Study 1 that have been omitted here are available in the Supporting Information. This information is available free of charge via the Internet at `http://pubs.acs.org/`.

References

1. Meyer, T.; Eriksson, M.; Maggio, R. Gradient estimation from irregularly spaced data sets. *Math. Geol.* **2001**, *33*, 693–717.

2. Correa, C.; Hero, R.; Ma, K. A comparison of gradient estimation techniques for volume rendering of unstructured meshes. *IEEE Trans. Vis. Comput. Graph.* **2011**, *17*, 305–319.

3. Daum, F.; Huang, J.; Krichman, M.; Kohen, T. Seventeen dubious methods to approximate the gradient for nonlinear filters with particle flow. *Signal and Data Processing of Small Targets (San Diego)* **2009**, *7445*.

4. Mansour, M.; Ellis, J. Comparison of methods for estimating real process derivatives in on-line optimization. *Appl. Math. Model.* **2003**, *27*, 275–291.

5. Brdys, M.; Tatjewski, P. *Iterative Algorithms for Multilayer Optimizing Control*; Imperial College Press, 2005.

6. Gao, W.; Engell, S. Iterative set-point optimization of batch chromatography. *Comput. Chem. Eng.* **2005**, *29*, 1401–1409.

7. Marchetti, A.; Chachuat, B.; Bonvin, D. A dual modifier-adaptation approach for real-time optimization. *J. Process Control* **2010**, *20*, 1027–1037.

8. Conn, A.; Scheinberg, K.; Vicente, L. *Introduction to Derivative-Free Optimization*; Cambridge University Press, 2009.

9. Engl, H.; Hanke, M.; Neubauer, A. *Regularization of Inverse Problems*; Kluwer Academic Publishers, 2000.

10. Yeow, Y.; Isac, J.; Khalid, F.; Leong, Y.; Lubansky, A. A method for computing the partial derivatives of experimental data. *AIChE J.* **2010**, *56*, 3212–3224.

11. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press, 2008.

12. Bunin, G.; François, G.; Bonvin, D. Performance of real-time optimization schemes - II. Implementation issues. *Comput. Chem. Eng. (submitted)* **2013**,

13. Brekelmans, R.; Driessen, L.; Hamers, H.; den Hertog, D. Gradient estimation schemes for noisy functions. *J. Optim. Theory Appl.* **2005**, *126*, 529–551.

14. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed.; Springer, 2009.

15. Myers, R.; Montgomery, D.; Anderson-Cook, C. *Response Surface Methodology*; John Wiley & Sons, 2009.

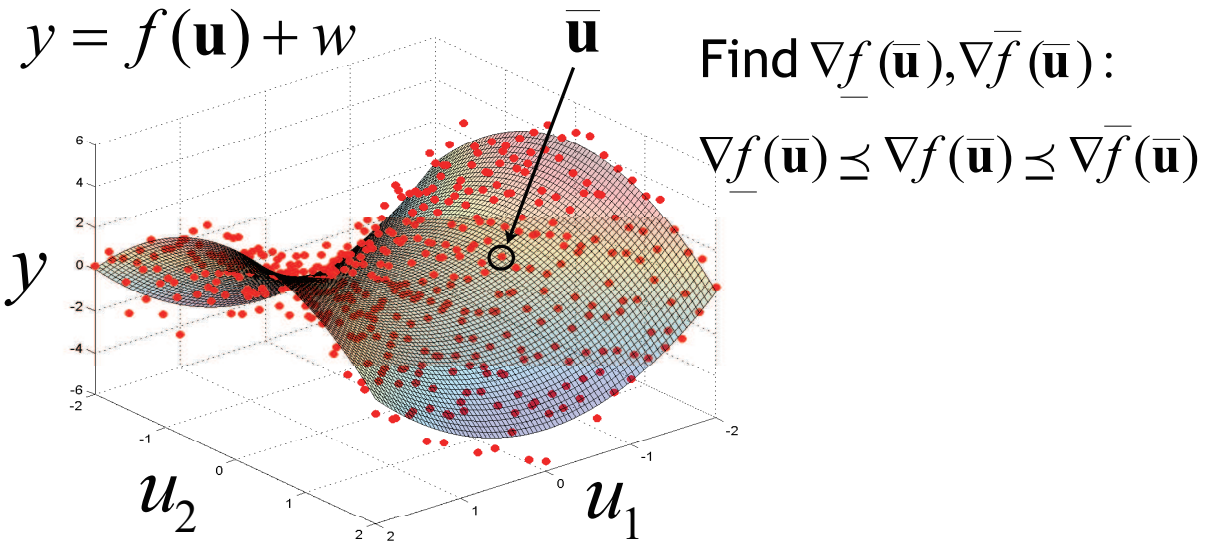
16. Ubhaya, V. Regression by special functions: algorithms and complexity. *Encyclopedia of Optimization* **2009**, 3268–3273.

17. Tuy, H. *Convex Analysis and Global Optimization*; Kluwer Academic Publishers, 1998.

18. Bunin, G.; François, G.; Bonvin, D. Exploiting local quasiconvexity for gradient estimation in modifier-adaptation schemes. *Proceedings of the 2012 American Control Conference (Montréal)* **2012**, 2806–2811.

19. Sturm, J. F. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optim. Methods Softw.* **1999**, *11*, 625–653.

20. CVX Research, I. CVX: MATLAB software for disciplined convex programming, version 2.0 beta. 2012.



For Table of Contents Only.