International Journal for Uncertainty Quantification, 5 (5): 453–468 (2015)

A GRADIENT-ENHANCED SPARSE GRID ALGORITHM FOR UNCERTAINTY QUANTIFICATION

Jouke H. S. de Baar & Brendan Harding*

Mathematical Sciences Institute, Australian National University, John Dedman Building 27, Union Lane, Canberra, ACT, 2601, Australia

*Address all correspondence to Jouke H. S. de Baar E-mail: j.h.s.debaar@gmail.com

Original Manuscript Submitted: 07/16/2015; Final Draft Received: 09/30/2015

Adjoint-based gradient information has been successfully incorporated to create surrogate models of the output of expensive computer codes. Exploitation of these surrogates offers the possibility of uncertainty quantification, optimization and parameter estimation at reduced computational cost. Presently, when we look for a surrogate method to include gradient information, the most common choice is gradient-enhanced Kriging (GEK). As a competitor, we develop a novel method: gradient-enhanced sparse grid interpolation. Results for two test functions, the Rosenbrock function and a test function based on the drag of a transonic airfoil with random shape deformations, show that the gradient-enhanced sparse grid interpolation is a reliable surrogate that can incorporate the gradient information efficiently for high-dimensional problems.

KEY WORDS: *high-dimensional surrogates, uncertainty quantification, sparse grids*

1. INTRODUCTION

A surrogate model provides an approximation of the output of an input-output system, such as an expensive computational fluid dynamics (CFD) solver. The surrogate approximation is based on a set of training data—a small set of evaluations of the solver—sampled at certain points in the input parameter space. Since their introduction by Sacks et al. [1], surrogate models have been shown to be effective in reducing the computational cost of uncertainty quantification, optimization, and parameter estimation. The common objective is to arrive at an accurate surrogate using only a small number of training data.

The main problem with surrogate models is that when we increase the number of input parameters, the number of required training data increases exponentially. This effect is known as the "curse of dimensionality" [2]. A promising approach to mitigate this curse of dimensionality is to use gradient information [3–5]. When gradient information is obtained from an adjoint solver, the full gradient is available at the cost of only one additional solve. Adjoint-based gradients are now becoming available in a range of CFD solvers, such as Stanford University's SU^2 [6], NASA's FUN3D [7], the German Aerospace Center (DLR)'s TAU [8], OpenFOAM [9], Rolls-Royce's HYDRA [10], and ANSYS' Fluent [11]. One practical problem with adjoint-based gradients is that they can contain numerical noise [12–14], an issue which we will address in Section 3.

Most research on gradient-enhanced surrogate models has focused on gradient-enhanced Kriging (GEK). Kriging was developed independently by Matheron [15] and Gandin [16], in the fields of geology and meteorology, respectively. A complete discussion of Kriging can be found in [17, 18]. GEK is a form of co-Kriging, where the partial derivatives are added as co-variables. GEK has been used in a number of applications, for example drag minimization for a supersonic business jet [19], prediction of the drag of a transonic airfoil [20], uncertainty quantification for a transonic airfoil [8], prediction of the performance of nuclear reactors [21], uncertainty quantification for a pitching airfoil [22], uncertainty quantification for the k-ϵ turbulence model [23], shape optimization of a high-pressure jet

2152–5080/15/$35.00

engine turbine [24], and high-dimensional uncertainty quantification for a transonic airfoil [25]. Because GEK has been widely used in engineering problems, we presently consider it as the reference gradient-enhanced surrogate method. Other gradient-enhanced surrogate methods include weighted least squares [26] and gradient-assisted radial basis functions [27].

In this paper, we develop gradient-enhanced sparse grid interpolation. Sparse grids were developed to reduce the curse of dimensionality [28]. The sparse grid combination technique combines a number of low-frequency grids in order to approximate the interpolation based on a single high-frequency grid, without significant loss of accuracy [29–31]. The combination technique approach enables the construction of a sparse grid without using a hierarchical basis. An important difference between Kriging and sparse grids is the way one constructs the sampling plan: Kriging usually uses a latin-hypercube sampling plan, which is random and can be computationally expensive, while the sparse grid sampling plan is created deterministically and is relatively cheap to construct.

Sparse grids were first used for uncertainty quantification by Ganapathysubramanian and Zabaras [32]. Further use of sparse grids for uncertainty quantification is discussed by Le Maître and Knio [33] and references therein. Instead of using the sparse grid directly in a quadrature rule, we treat the sparse grid interpolation as a surrogate model, which is then exploited for uncertainty quantification. In this framework, extending the sparse grid method to include gradient information provides an alternative to GEK.

2. METHOD

A general, robust, and non-intrusive method for uncertainty quantification is the Monte Carlo method of statistical sampling [34]. For an input-output system, given a multivariate input distribution $p(\vec{x})$, the Monte Carlo method allows us to accurately estimate the statistical moments of the output $u(\vec{x})$. For example, the mean of the output is estimated by

$$\mathrm{E}[u] \approx \frac{1}{M} \sum_{i=1}^{M} u(\vec{x}_i), \tag{1}$$

where $\vec{x}_i$ are M random vectors, which are sampled according to the input distribution.

However, an accurate Monte Carlo estimate requires that M is very large. Therefore, if the evaluation of the input-output system u is expensive, direct evaluation of (1) becomes impracticable. One approach is to approximate u with a much cheaper surrogate model u_m, which is based on a small set of $m \ll M$ training data, obtained from evaluating the input-output system only m times. Instead of (1) we then estimate

$$\mathrm{E}[u] \approx \frac{1}{M} \sum_{i=1}^{M} u_m(\vec{x}_i), \tag{2}$$

at much lower computational cost, by randomly sampling the cheap surrogate instead of the expensive input-output system. The challenge is then to find an efficient surrogate method, that provides an accurate approximation using only a small number m of training data. In the following subsections, we present gradient-enhanced sparse grid interpolation. This method can provide an efficient surrogate in high-dimensional problems.

2.1 Sparse Grids

In d dimensions consider approximating a function $u : [0,1]^d \mapsto \mathbb{R}$ as piecewise multi-linear. For $i \in \mathbb{N}$ we define the equidistant points

$$\Omega_i := \begin{cases} \{j2^{-i} : j = 0, 1, \ldots, 2^i\} & \text{if } i > 0, \\ \left\{\frac{1}{2}\right\} & \text{if } i = 0, \end{cases}$$

and for $\underline{i} \in \mathbb{N}^d$ we define the uniform grid $\Omega_{\underline{i}} = \Omega_{i_1} \times \cdots \times \Omega_{i_d}$. We denote the piecewise multi-linear interpolant of the samples $u(\Omega_{\underline{i}})$ by $u_{\underline{i}}$. Given the piecewise linear basis functions

$$\phi_{i_k,j_k}(x_k) := \begin{cases} \max\{0, 1 - |2^{i_k} x_k - j_k|\} & \text{if } i_k > 0, \\ 1 & \text{if } i_k = 0, \end{cases}$$

then piecewise multi-linear nodal basis functions are defined by

$$\phi_{\underline{i},\underline{j}}(\vec{x}) := \prod_{k=1}^{d} \phi_{i_k,j_k}(x_k)\,.$$

Defining the index set $B_{i_k} := \{0, 1, \ldots, 2^{i_k}\}$ if $i_k > 0$, $B_0 := \{0\}$ and $B_{\underline{i}} := B_{i_1} \times \cdots \times B_{i_d}$, then we may express $u_{\underline{i}}$ as

$$u_{\underline{i}}(\vec{x}) = \sum_{\underline{j} \in B_{\underline{i}}} u_{\underline{i},\underline{j}} \phi_{\underline{i},\underline{j}}(\vec{x})\,,$$

where $u_{\underline{i},\underline{j}} := u(\vec{x}_{\underline{i},\underline{j}})$ with $\vec{x}_{\underline{i},\underline{j}} = (x_{i_1,j_1}, \ldots, x_{i_d,j_d})$ such that each x_{i_k,j_k} is the j_kth member of Ω_{i_k}, i.e.,

$$x_{i_k,j_k} := \begin{cases} j_k 2^{-i_k} & \text{if } i_k > 0, \\ \dfrac{1}{2} & \text{if } i_k = 0. \end{cases}$$

This definition differs from classical piecewise multi-linear approximations in that the basis function is constant when $i_k = 0$ and is determined by the value at $x_k = 1/2$ (as opposed to a linear approximation determined by the value at the endpoints 0 and 1).

The combination technique constructs the sparse grid interpolant of u with depth $n \in \mathbb{N}$ via

$$u_n^c := \sum_{k=0}^{d-1} (-1)^k \binom{d-1}{k} \sum_{|\underline{i}|=n-k} u_{\underline{i}}\,. \tag{3}$$

Error estimates for sparse grid interpolation have been well established, see for example [29–31]. It is well known that the approximation error of sparse grids as described above is $\mathcal{O}(n^{d-1}4^{-n})$.

2.2 Gradient-Enhanced Sparse Grids

Here we are concerned with incorporating gradient information into the sparse grid interpolant to improve our approximation of the function u. While there are several ways in which one might construct an interpolant using gradient information we focus on piecewise cubic Hermite interpolation in this paper. This approach is related to high-order quadrature on sparse grids, which has been studied in [35]. The difference here as that in dimensions $d > 1$ we do not have enough information to construct a full multi-dimensional piecewise cubic Hermite interpolant of u on each of the $u_{\underline{i}}$ (which requires all mixed derivatives $D^{\underline{i}}u$ with $\underline{0} < \underline{i} \leq \underline{1}$ in addition to the function values on $\Omega_{\underline{i}}$). In particular we only have estimates of $\partial u/\partial x_1, \ldots, \partial u/\partial x_d$ at each sample point. To get around this we construct several interpolants which each use piecewise cubic Hermite interpolation on at most one of the spatial dimensions and then use the inclusion-exclusion principle over p space to construct a gradient-enhanced interpolant for each $u_{\underline{i}}$. These are then combined using the classical combination technique (3) to obtain a gradient enhanced sparse grid interpolant.

In one dimension a cubic Hermite spline may be expressed via the sum of two cubic nodal basis functions, one which incorporates the function value and a second which incorporates the gradient. We define the basis functions

$$\psi_{i_k,j_k,\ell}(x_k) := \begin{cases} (2\phi_{i_k,j_k}(x_k) - 1)\phi_{i_k,j_k}(x_k)^2 & \text{if } \ell = k \text{ and } i_k > 0, \\ \phi_{i_k,j_k} & \text{otherwise,} \end{cases}$$

and

$$\zeta_{i_k,j_k,\ell}(x_k) := \begin{cases} (x_k - j_k 2^{-i_k})\phi_{i_k,j_k}(x_k)^2 & \text{if } \ell = k \text{ and } i_k > 0, \\ 0 & \text{if } \ell = k \text{ and } i_k = 0, \\ \phi_{i_k,j_k} & \text{otherwise,} \end{cases}$$

for constructing piecewise cubic Hermite splines (or constant functions if $i_k = 0$). In $d = 1$ dimension with $u(x_1) : [0,1] \to \mathbb{R}$, $i_1 > 0$ and $\ell = 1$ we have the piecewise cubic Hermite interpolant between samples Ω_{i_1} via

$$\tilde{u}_{i_1,\ell} \sum_{j_1 \in B_{i_1}} u(x_{i_1,j_1}) \psi_{i_1,j_1,\ell}(x_1) + \frac{\partial u(x_{i_1,j_1})}{\partial x_1} \zeta_{i_1,j_1,\ell}(x_1) \,.$$

For $d > 1$ dimensions we have d different interpolants which each incorporate the gradient information into one of the d dimensions. Specifically for $\ell = 1, \ldots, d$ we define the basis functions

$$\psi_{\underline{i},\underline{j},\ell}(\vec{x}) = \prod_{k=1}^{d} \psi_{i_k,j_k,\ell}(x_k) \qquad \text{and} \qquad \zeta_{\underline{i},\underline{j},\ell}(\vec{x}) = \prod_{k=1}^{d} \zeta_{i_k,j_k,\ell}(x_k) \,,$$

with which we obtain the interpolants

$$\tilde{u}_{\underline{i},\ell} := \sum_{\underline{j} \in B_{\underline{i}}} u(\vec{x}_{\underline{i},\underline{j}}) \psi_{\underline{i},\underline{j},\ell}(\vec{x}) + \frac{\partial u(\vec{x}_{\underline{i},\underline{j}})}{\partial x_\ell} \zeta_{\underline{i},\underline{j},\ell}(\vec{x}) \,.$$

Each $\tilde{u}_{\underline{i},\ell}$ consists of a piecewise linear approximation along the dimensions $k \in \{1, \ldots, d\}$, with $k \neq \ell$ and piecewise cubic Hermite spline interpolation along the dimension $k = \ell$ (with the exception when $i_k = 0$ corresponds to a constant function). To construct an approximation of u that uses all of the gradient information we compute

$$\hat{u}_{\underline{i}} := -(d-1) u_{\underline{i}} + \sum_{\ell=1}^{d} \tilde{u}_{\underline{i},\ell} \,. \tag{4}$$

With respect to hp-approximation theory, one may interpret (3) as an inclusion-exclusion principle over h space while (4) may be interpreted as an inclusion-exclusion principle over p space. Notice also that coefficients of (4) are analogous to those of the combination technique with a depth of one. Given a function $u(\vec{x})$ which can be decomposed as a sum of functions in each dimension, that is $u(\vec{x}) = v_1(x_1) + \cdots + v_d(x_d)$, then $\hat{u}_{\underline{i}}$ is precisely a sum of piecewise cubic Hermite interpolants over each of the $v_k(x_k)$.

The classical combination technique (3) is now applied over these $\hat{u}_{\underline{i}}$ to obtain the gradient-enhanced sparse grid interpolant

$$\hat{u}_n^c := \sum_{k=0}^{d-1} (-1)^k \binom{d-1}{k} \sum_{|\underline{i}|=n-k} \hat{u}_{\underline{i}} \,.$$

In Fig. 1 we depict the computation of $\hat{u}_2^c$ in $d = 2$ dimensions. Figure 1(a) shows the sparse grid, which is the result of the combination technique illustrated in Figs. 1(b)–1(f). Figure 1(b) depicts the basis functions used to construct $u_{\underline{i}}$ for each combination grid, with the basis functions shown in blue at the top and the left. The second and third rows, i.e., Figs. 1(c)–1(d) and 1(e)–1(f), depict the basis functions used to construct $\hat{u}_{\underline{i},1}$ and $\hat{u}_{\underline{i},2}$, respectively, for each combination grid, with the left figures depicting $\psi_{\underline{i},\underline{j},\ell}$ and the right figures depicting $\zeta_{\underline{i},\underline{j},\ell}$.

This method of interpolation has been implemented in MATLAB and has been tested on two different problems which are discussed in Section 3. The code includes several options, including the option to use the linear approximation between the two end points when $i_k = 0$. We chose to use the constant function for $i_k = 0$ by default as this allows the implementation to be used in very high dimensions as the grid for $u_{\underline{i}}$ with $\underline{i} = \underline{0}$ consists of only one center point as opposed to the 2^d corner points. A second option allows one to use the gradient information to obtain a linear approximation along a dimension k when using the center point for $i_k = 0$ rather than the default approximation as a constant function. We use constant functions in the following test cases; other options were not investigated.

3. RESULTS

In Section 2 we presented a gradient-enhanced sparse grid algorithm. In the present section, we apply this algorithm to two different test functions: the multi-dimensional Rosenbrock function and a transonic airfoil with random shape deformations.

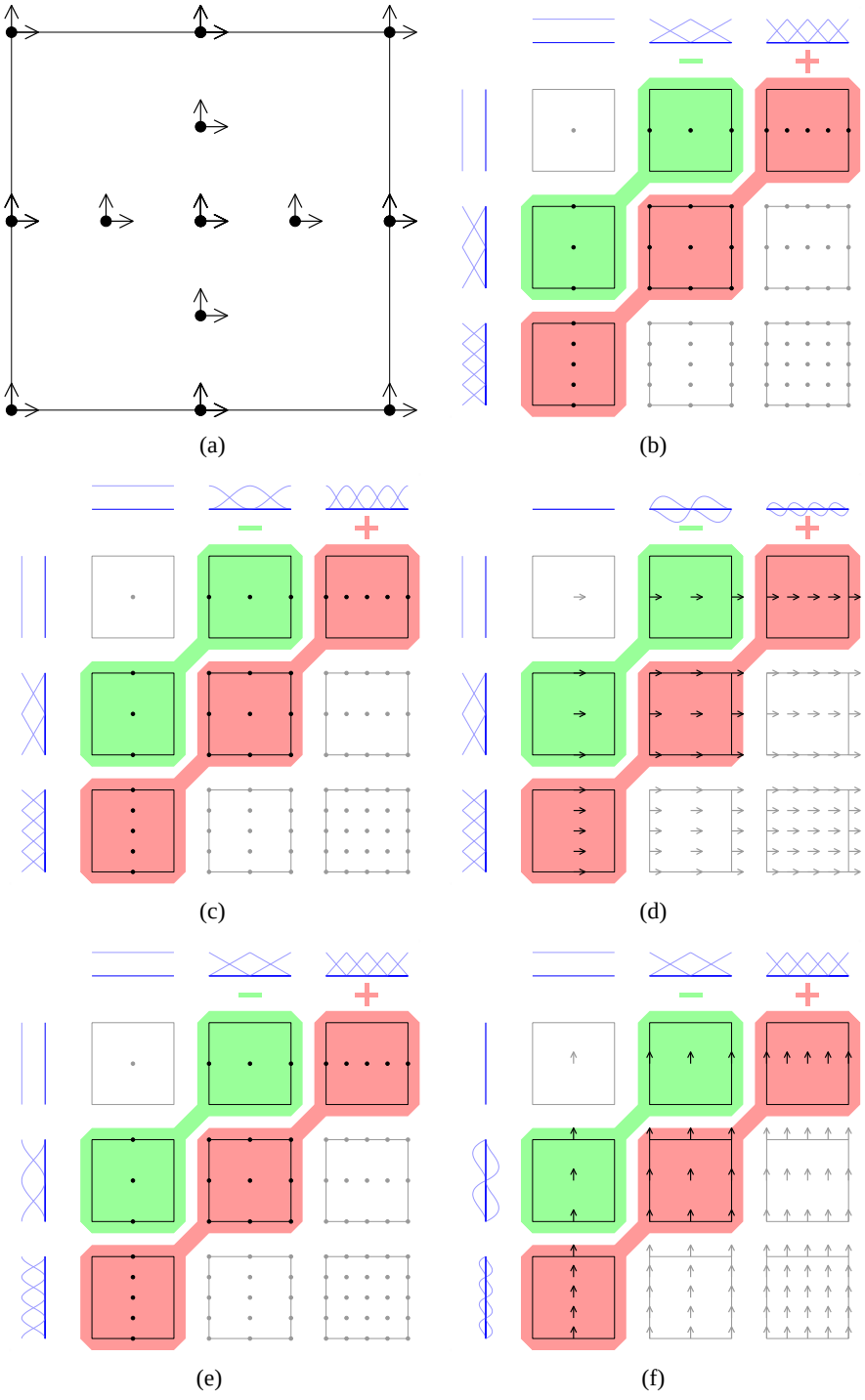


FIG. 1: Illustration of the basis functions for each of the combination grids in $d = 2$ dimensions with depth $n = 2$. The interpolants are combined by adding the second and third rows and subtracting the first row. (a) Sparse grid corresponding to $\hat{u}_n^c$ with $n = d = 2$. Basis functions for the (b) $u_{\underline{i}}$ in the combination $\hat{u}_2^c$, (c) ϕ component of each $\tilde{u}_{\underline{i},1}$ in the combination $\hat{u}_2^c$, (d) ζ component of each $\tilde{u}_{\underline{i},1}$ in the combination $\hat{u}_2^c$, (e) ϕ component of each $\tilde{u}_{\underline{i},2}$ in the combination $\hat{u}_2^c$, and (f) ζ component of each $\tilde{u}_{\underline{i},2}$ in the combination $\hat{u}_2^c$.

3.1 Rosenbrock Function

The first test function is the Rosenbrock function [36]. It is a well-known test function, which is often used to test optimization algorithms. The multi-dimensional Rosenbrock function is given by

$$f(\vec{x}) = \sum_{k=1}^{d-1} 100\ (x_{k+1} - x_k^2)^2 + (1 - x_k)^2 \tag{5}$$

on the domain $[-2, 2]^d$. We consider the rescaled Rosenbrock function

$$g(\vec{x}) = f(4\vec{x} - 2) \tag{6}$$

on the domain $[0, 1]^d$. Figure 2 is an illustration of the two-dimensional case. The minimum, located at $\vec{x} = [3/4, 3/4]$, is indicated by the cross hair. For $d > 3$ there is a global minimum at $\vec{x} = [3/4, 3/4, \ldots, 3/4]$ as well as a local minimum at $\vec{x} = [1/4, 3/4, \ldots, 3/4]$.

In the case of the Rosenbrock function, we obtain the gradient information by means of a complex step finite difference method. In terms of function evaluations, this is an unrealistically expensive approach. In Section 3.1.6 we assume that the gradient information comes at the cost of a single solve, as is the case for adjoint-based derivatives.

3.1.1 Convergence for Increasing Number of Samples

Figure 3 shows the relative prediction error for $d = 2$ dimensions for an increasing number of samples (i.e., an increasing sparse grid depth). Compared with the results for a regular sparse grid, the gradient-enhanced sparse grid is more accurate and shows a higher rate of convergence. For a large number of samples, the prediction error is limited by machine precision.

3.1.2 Scaling with Number of Dimensions

In Fig. 4 we consider the number of samples required to achieve a relative prediction error of 0.05. The number of required samples increases dramatically with the number of dimensions, an effect known as the "curse of dimensionality." However, in Fig. 4 we see that adding gradient information in the sparse grid reduces the number of required samples significantly.

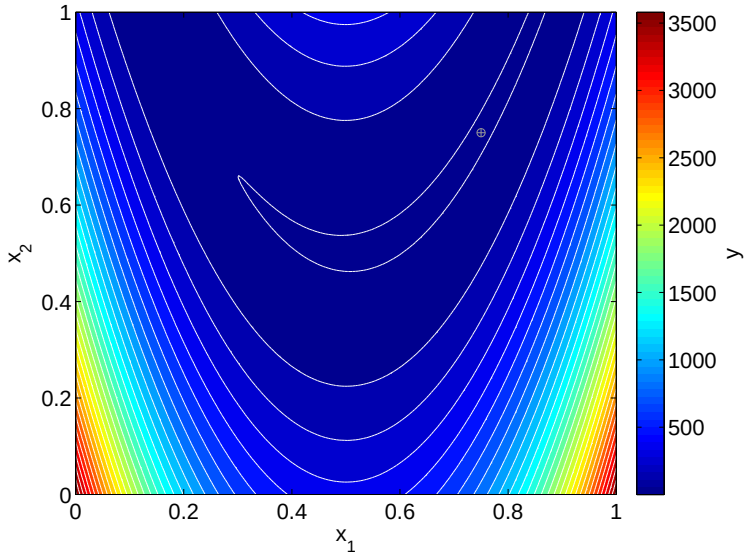


FIG. 2: First test function is the Rosenbrock function. The cross-hair indicates the location of the minimum.

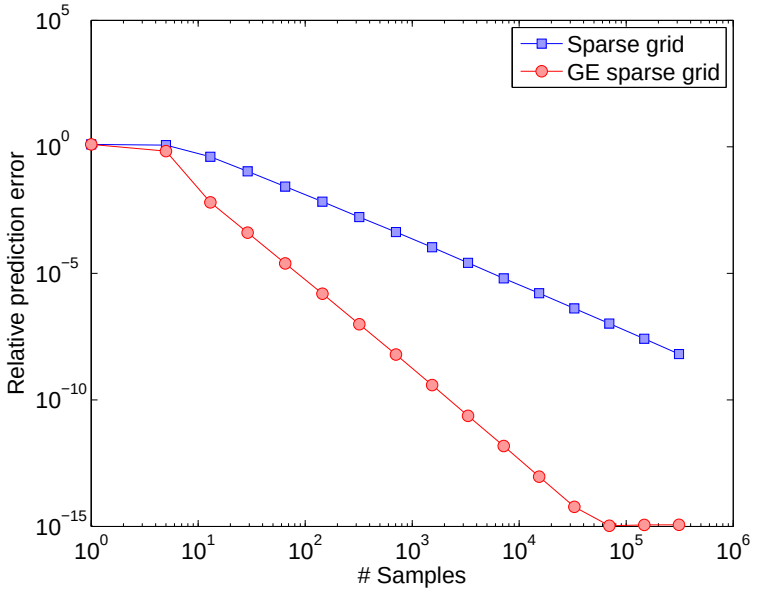


FIG. 3: Convergence of the relative prediction error of the Rosenbrock function for an increasing number of samples. The gradient-enhanced sparse grid prediction is more accurate and has a higher rate of convergence.

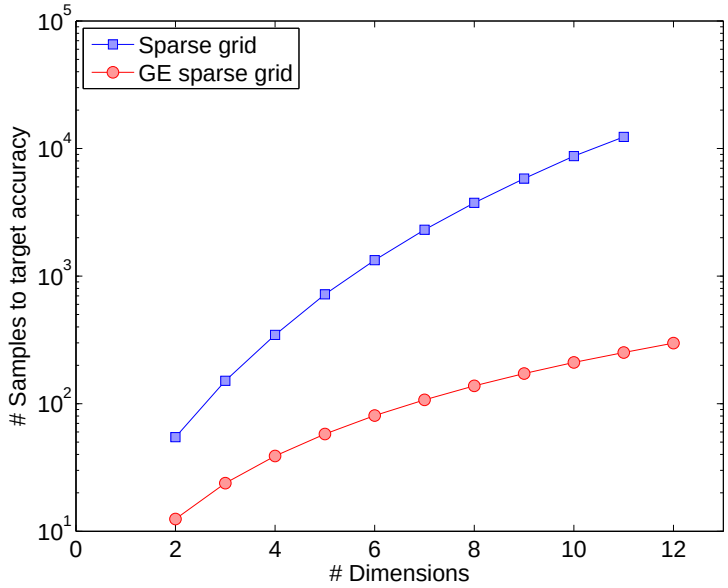


FIG. 4: Curse of dimensionality for the Rosenbrock function: a dramatic increase of the number of samples required to reach a certain target accuracy of the sparse grid prediction. Adding gradient information reduces this effect significantly.

3.1.3 Effect of gradient noise

An important issue with adjoint-based gradients is that they can contain a significant amount of numerical noise [12–14]. Therefore, we consider the effect of adding gradient noise. In Fig. 5 the blue line indicates the sparse grid prediction error for a two-dimensional grid of depth 2 (the gradient noise has no effect as the gradient information

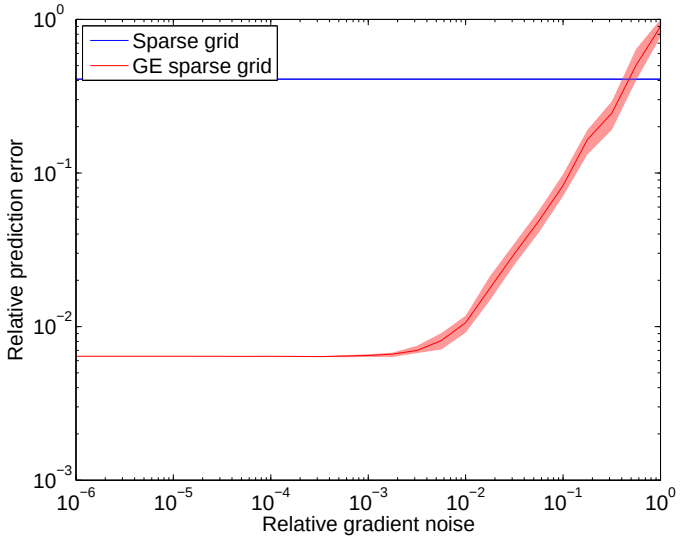


FIG. 5: Effect of gradient noise on the prediction error of the Rosenbrock function. The red line is the median, the shaded area indicates 0.2 and 0.8 quantiles.

is not used in this case). The red line indicates the effect of gradient noise on the gradient-enhanced sparse grid prediction error. Because the noise is random, we have repeated this experiment 20 times; the red line is the median and the shaded area indicates the 0.2 and 0.8 quantiles. For a relative noise magnitude of >0.01, the error starts to increase.

3.1.4 Comparison with Kriging

We compare the sparse grid interpolation with a Kriging surrogate model [15–17]. In a Bayesian framework, it is straightforward to incorporate gradient information in the Kriging prediction, resulting in GEK [23]. Figure 6 shows the prediction error of the sparse grid interpolation and the Kriging surrogate, both cases with and without gradient information. Because we use a random latin-hypercube design of experiment [37] for Kriging and GEK, we repeat the experiment 20 times, and plot the median and 0.2 and 0.8 quantiles. Interestingly, Kriging is more accurate than the sparse grid prediction. However, the gradient-enhanced sparse grid prediction is more accurate than GEK, although it shows a similar rate of convergence.

3.1.5 Computational Cost

In order to effectively use the gradient-enhanced sparse grid algorithm in a computer experiment, the computational cost of running the sparse grid algorithm should be lower than the cost of running the solver. Figure 7 shows CPU timings on a Intel Core 2.8 GHz processor, for a sparse grid of depth 2. For an increase in the number of dimensions, we see an increase of the computational cost required to construct the grid and make either a sparse grid or gradient-enhanced sparse grid prediction. The discontinuity in the cost of constructing the grid, at 36 dimensions, is likely where we have completely filled the CPU cache.

For high dimensions, incorporating the gradient information makes the algorithm roughly 5 times as expensive. The cost at 64 dimensions is considerable, however, we should consider that this corresponds to 8×10^3 samples, which is likely to come at a significant solver cost as well.

3.1.6 Uncertainty Quantification

We conclude this subsection on the Rosenbrock function with an uncertainty quantification analysis. We consider the $d = 2$ dimensional case, with an uncertainty in the input parameters of

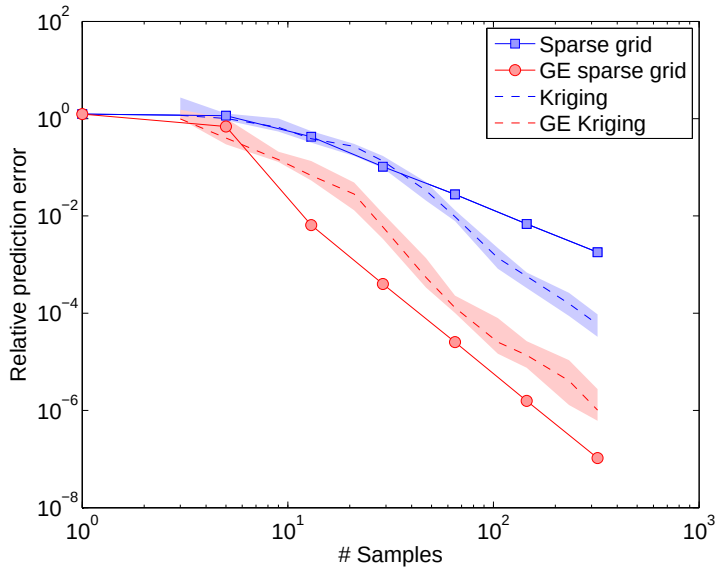


FIG. 6: Sparse grid versus Kriging prediction of the Rosenbrock function for $d = 2$ dimensions. For Kriging, we show the median as well as 0.2 and 0.8 quantiles.

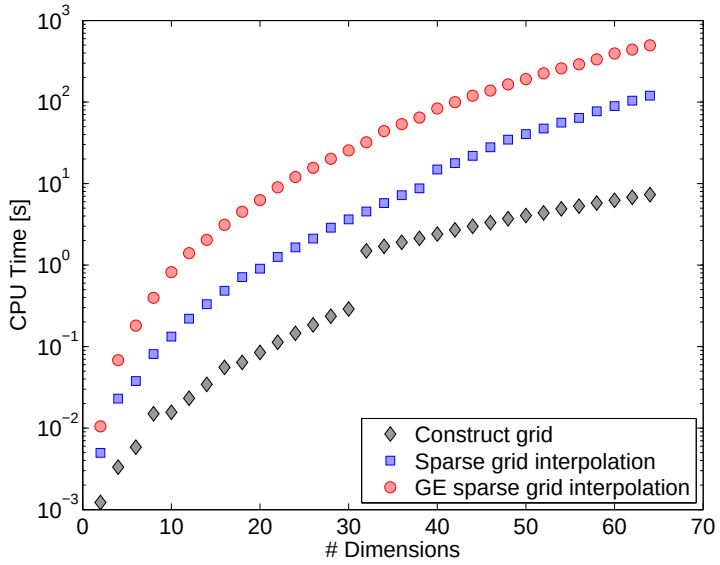
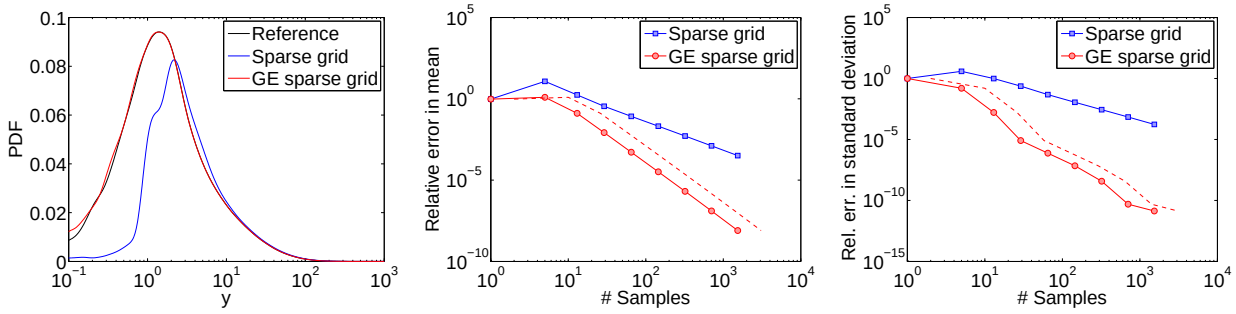


FIG. 7: Computational cost of creating a sparse grid and making either a sparse grid or gradient-enhanced sparse grid prediction of the Rosenbrock function for an increasing number of dimensions. Timings are on a Intel Core 2.8 GHz processor.

$$[x_1, x_2] \sim \mathcal{N}\left([0.5, 0.5], \sigma^2 I\right), \tag{7}$$

with $\sigma = 0.1$. We truncate the distribution to the domain $[0, 1]^d$. We aim to propagate this input uncertainty to obtain a probability density function (PDF) of the output. Figure 8(a) shows a reference PDF in black, obtained from kernel density estimate of a Monte Carlo simulation with $M = 10^5$ random samples [34]. The PDF is highly non-symmetric, because at the center of the input PDF the output is close to the minimal value of the test function. The blue and red lines show the resulting PDF using a sparse grid and a gradient-enhanced sparse grid, respectively, both of a depth



(a) (b) (c)

FIG. 8: Uncertainty quantification for the Rosenbrock function. The gradient-enhanced sparse grid prediction results in a PDF which is closer to the reference. The red dotted lines indicate the shift of the results when plotting against the number of solves, instead of the number of samples. (a) Output PDF based on either a sparse grid or gradient-enhanced sparse grid prediction, (b) relative error in the predicted mean of the output PDF, and (c) relative error in the predicted standard deviation of the output PDF.

of 4. Here, we use the (gradient-enhanced) sparse grid prediction as a surrogate, in order to propagate the uncertainty. Clearly, the PDF based on the gradient-enhanced sparse grid is much closer to the reference. To quantify this, Figs. 8(b) and 8(c) show the relative error in the predicted mean and standard deviation, respectively.

We have been comparing the sparse grid and the gradient-enhanced sparse grid predictions on based on the number of samples. Assuming that all gradients are obtained from one adjoint solve having the same cost as the primal solve, then plotting against the number of solves instead of the number of samples would move the red line in Figs. 8(a) and 8(b) to the right, as indicated by the red dotted line.

3.2 Transonic Airfoil

As a second test function, we use the response of an actual engineering problem. We consider the aerodynamical drag coefficient of the transonic airfoil in Fig. 9, which operates at an angle of attack of $\alpha = 1.25°$ and a Mach number of $M_\infty = 0.8$. The baseline wing geometry is the FFAST airfoil [39, p. 110], however, small deformations of the geometry might have a considerable effect on the shock locations and, therefore, on the drag. These deformations are parameterized using the basis functions [25]

$$\bar{\xi} = \frac{\xi - c_0}{1 - 2c_0}, \quad \xi \in [c_0, 1 - c_0]$$

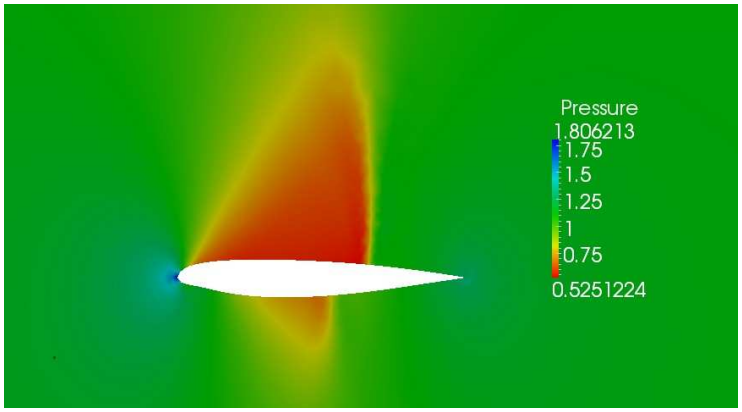


FIG. 9: Pressure field for the baseline transonic airfoil at Mach = 0.8, with visible shocks. From [38].

$$f_i(\xi) = \sin(\pi\bar{\xi})\frac{\sin(i\pi\bar{\xi})}{i}, \tag{8}$$

with ξ the chordwise coordinate, $i \in \{1, \ldots, d/2\}$ and $c_0 = 0.15$ to avoid deformations near the leading and trailing edge. With these basis functions, the deformations on the top and bottom of the airfoil are, respectively,

$$\Delta_{\text{top}}(\xi) = \sum_{i=1}^{d/2} x_{i,\text{top}} f_i(\xi), \tag{9}$$

$$\Delta_{\text{bottom}}(\xi) = -\sum_{i=1}^{d/2} x_{i,\text{bottom}} f_i(\xi), \tag{10}$$

with d the total number of shape parameters and $x_i \in [-0.0125, 0.0125]$ the shape parameters.

For the dimensions $d = [2, 3, 4, 6, 8]$, [25] provides 200 samples on a latin-hypercube design of experiment, including adjoint-based gradient information, obtained by solving the Euler equations with the open source solver SU^2 [6]. We use these data to train a GEK surrogate, which then serves as the test function for this section. As a minor modification, we rescale the shape parameters to $[0, 1]^d$, with the baseline geometry remaining in the center.

The resulting test function for $d = 2$ dimensions is illustrated in Fig. 10. In this figure, the first shape parameter determines the deformation of the bottom of the airfoil, while the second shape parameter determines the deformation of the top of the airfoil. Clearly, the optimal design, with minimal drag, is close to the baseline; this is not a surprise, since the FFAST geometry is based on an optimized transonic wing design.

In the case of the transonic airfoil test function, GEK provides analytical gradient information. In Section 3.2.4 we assume that the gradient information comes at the cost of one additional solve, as is the case for adjoint-based gradients.

3.2.1 Convergence for Increasing Number of Samples

Figure 11 shows the convergence of the relative prediction error for an increasing number of samples. For a small number of samples, the gradient-enhanced prediction is more accurate than the sparse grid prediction. However, for a larger number of samples the accuracy of both methods becomes similar.

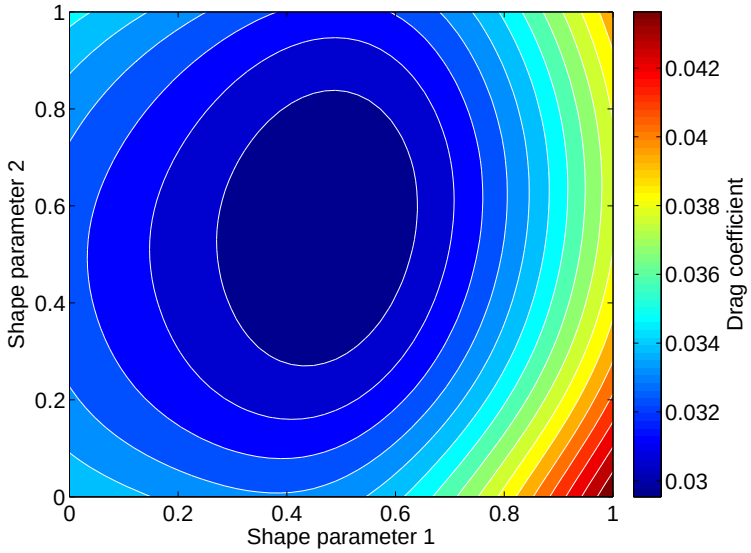


FIG. 10: Drag coefficient of the transonic airfoil as a function of two shape parameters, which define the deformation of the bottom and top of the airfoil. The optimum is close to the baseline design at $[0.5, 0.5]$.

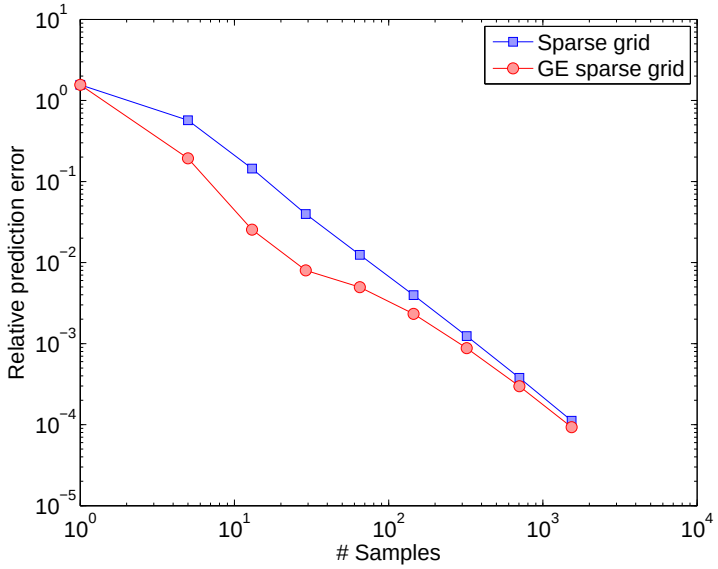


FIG. 11: Convergence of the relative prediction error of the transonic airfoil for an increasing number of samples. The gradient-enhanced sparse grid prediction is more accurate for a small number of samples, but is close to the accuracy of the sparse grid prediction for a larger number of samples.

3.2.2 Scaling with Number of Dimensions

In Fig. 12, for the transonic airfoil the transonic airfoil we see a full curse of dimensionality, with an exponential increase of the number of samples required to achieve a relative target accuracy of 0.05. Adding gradient information reduces the number of samples, but not as radically as in the case of the Rosenbrock function in Fig. 4.

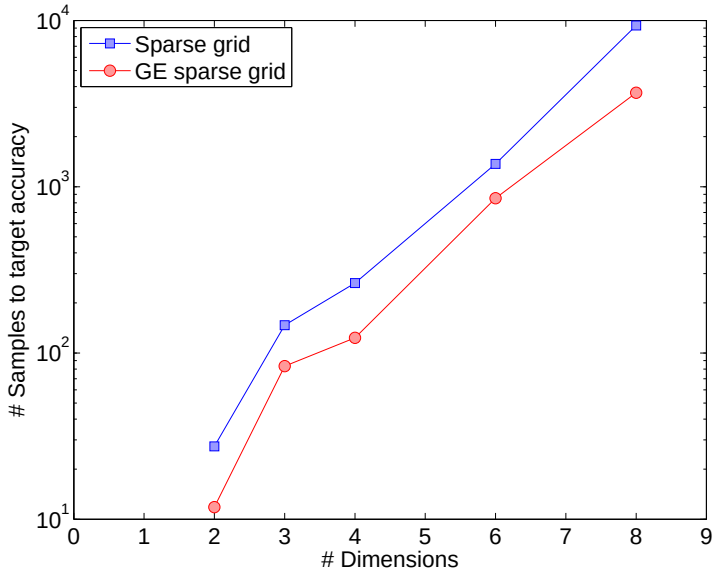


FIG. 12: Number of samples required to achieve a given target accuracy, as a function of the number of shape parameters.

3.2.3 Effect of Gradient Noise

In the case of the transonic airfoil, the standard deviation of the relative gradient noise was found to be 0.09. Because this is a significant amount of noise, Fig. 13 illustrates the effect of gradient noise on the accuracy of the gradient-enhanced sparse grid prediction. As we are simulating the gradient noise with uncorrelated Gaussian noise, we repeat the experiment 20 times; the red line is the median, the shaded area shows the 0.2 and 0.8 quantiles. For a relative noise level of 0.09, indicated by the black dotted line, the accuracy is still close to that of the noise-free prediction. We note that, for a similar case, Lukaczyk et al. [14] have proposed to reduce the effect of gradient noise by filtering the surface sensitivities before projecting them into the parameter space.

3.2.4 Uncertainty Quantification

We aim to propagate the same truncated normal distribution (7) as was used for the Rosenbrock function. Along the same lines, we obtain the black reference PDF in Figure 14(a). In the same figure, we show the PDF obtained from a regular sparse grid and a gradient-enhanced sparse grid, both for a depth of 1. The PDF based on the gradient-enhanced sparse grid is closer to the reference PDF.

Figures 14(b) and 14(c) show the relative error in the mean and standard deviation, respectively. Again, we have used a dotted red line to illustrate the shift that would occur if we would plot against the number of solves instead of the number of samples, again assuming that the gradient would be computed in a single adjoint solve. Both the error in the mean and in the standard deviation is lower when we use gradient information, although it approaches the error obtained using a regular sparse grid for a larger number of samples.

4. CONCLUSIONS

We have developed a gradient-enhanced sparse grid interpolation method. We have evaluated the performance of this method for up to 64 dimensions. We have demonstrated that this method can be used as a surrogate model for uncertainty quantification.

For the Rosenbrock test function, including gradient information significantly reduces the curse of dimensionality. For two dimensions, the accuracy of the sparse grid and gradient-enhanced sparse grid surrogate is similar to that

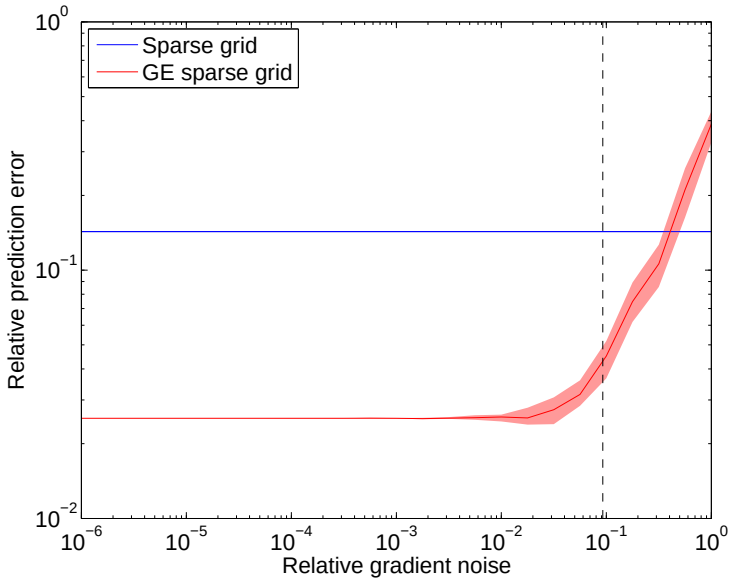
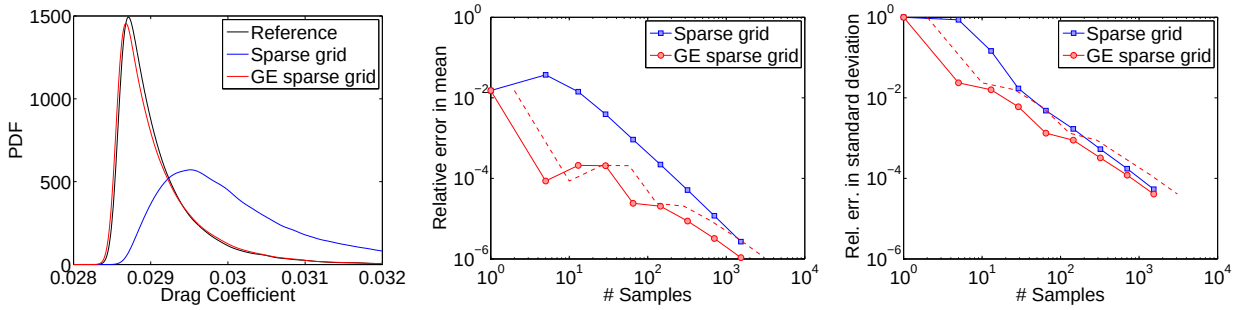


FIG. 13: Effect of gradient noise on the relative prediction error for the transonic airfoil. The dotted black line indicates the observed noise level. The red line is the median error, the shaded area indicates 0.2 and 0.8 quantiles.



(a) (b) (c)

FIG. 14: Uncertainty quantification for the transonic airfoil test function. For a small number of samples, the gradient-enhanced sparse grid prediction results in a PDF which is closer to the reference. The red dotted lines indicate the shift of the results when plotting against the number of solves, instead of the number of samples. (a) Output PDF based on either a sparse grid or gradient-enhanced sparse grid prediction, (b) relative error in the predicted mean of the output PDF, and (c) relative error in the predicted standard deviation of the output PDF.

of Kriging and GEK, respectively. Using gradient information reduces the cost of uncertainty quantification. For the transonic airfoil test function, including gradient information does not reduce the curse of dimensionality significantly. Using gradient information reduces the cost of uncertainty quantification when we consider the analysis for a small number of CFD solves.

Future work will include further investigation of the reduction of the curse of dimensionality for different test functions, as well as possible development of a direct gradient-enhanced sparse grid quadrature method. Based on the present results, gradient-enhanced sparse grid interpolation can become a serious competitor of GEK.

SUPPLEMENTARY MATERIAL

GESPGR, the Matlab code for sparse grid and gradient-enhanced sparse grid interpolation, is available on the Journals website (http://uncertainty-quantification.com/), as well as through: http://www.mathworks.com/matlabcentral/fileexchange/53386-gradient-enhanced-sparsegrid.

REFERENCES

1. Sacks, J., Welch, W., Mitchell, T., and Wynn, H., Design and analysis of computer experiments, *Stat. Sci.*, 4(4):409–435, 1989.
2. Bellman, R. E., *Adaptive Control Processes: A Guided Tour*, Princeton University Press, Princeton, NJ, 1961.
3. Oden, T., Moser, R., and Ghattas, O., Computer predictions with quantified uncertainty (Part 1), *SIAM News*, 43(9), 2010.
4. Oden, T., Moser, R., and Ghattas, O., Computer predictions with quantified uncertainty (Part 2), *SIAM News*, 43(10), 2010.
5. Viana, F. A., Simpson, T. W., Balabanov, V., and Toropov, V., Metamodeling in multidisciplinary design optimization: How far have we really come?, *AIAA J.*, 52(4):670–690, 2014.
6. Palacios, F., Alonso, J., Duraisamy, K., Colonno, M., Hicken, J., Aranake, A., Campos, A., Copeland, S., Economon, T., Lonkar, A., Lukaczyk, T., and Taylor, T., Stanford University Unstructured (SU^2): An open-source integrated computational environment for multi-physics simulation and design, in *AIAA Paper 2013-0287, 51st AIAA Aerospace Sciences Meeting and Exhibit*, Grapevine, Texas, Jan. 7–10, 2013.
7. Nielsen, E. J. and Diskin, B., Discrete adjoint-based design for unsteady turbulent flows on dynamic overset unstructured grids, *AIAA J.*, 51(6):1355–1373, 2013.
8. Dwight, R. P. and Han, Z.-H., Efficient uncertainty quantification using gradient-enhanced Kriging, *11th AIAA Non-Deterministic Approaches Conference*, Palm Springs, California, May 4–7, 2009.

9. Jasak, H., Popovac, M., and Rusche, H., Implementation and numerical stabilisation of adjoint flow and turbulence model in OpenFOAM, in *11th World Congress on Computational Mechanics*, Barcelona, July 20–25, 2014.
10. Ntanakas, G. D. and Meyer, M., Towards unsteady adjoint analysis for turbomachinary applications, in *"ECCOMAS"*, Barcelona, July 20–25, 2014.
11. Han, T., Hill, C., and Jindal, S., Adjoint method for aerodynamic shape improvement in comparison with surface pressure gradient method, *SAE Int. J. Passenger Cars—Mech. Syst.*, 4:100–107, 2011.
12. Giles, M., Duta, M., Muller, J., and Pierce, N., Algorithm developments for discrete adjoint methods, *AIAA J.*, 41(2):198–205, 2003.
13. Dwight, R. and Brezillon, J., Effect of approximations of the discrete adjoint on gradient-based optimization, *AIAA J.*, 44(12):3022–3071, 2006.
14. Lukaczyk, T., Taylor, T., Palacios, F., and Alonso, J., Managing gradient inaccuracies while enhancing optimal shape design methods, in *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Grapevine, Texas, Jan. 7–10, 2013.
15. Matheron, G., Principles of geostatistics, *Econ. Geol.*, 58:1246–1266, 1963.
16. Gandin, L., *Objective Analysis of Meteorological Fields: Gidrometeorologicheskoe Izdatel'stvo (GIMIZ), Leningrad*, Translated by Israel Program for Scientific Translations, Jerusalem, 1965.
17. Cressie, N., *Statistics for Spatial Data*, Wiley, New York, 1993.
18. Webster, R. and Oliver, M. A., *Geostatistics for Environmental Scientists*, 2nd ed., Wiley, New York, 2007.
19. Chung, H.-S. and Alonso, J. J., Using gradients to construct Co-kriging approximation models for high-dimensional design optimization problems, in *AIAA 40th Aerospace Sciences Meeting and Exhibit*, Reno, NV, Jan. 14–17, 2002.
20. Laurenceau, J. and Sagaut, P., Building efficient response surfaces of aerodynamic functions with Kriging and Co-kriging, *AIAA J.*, 46(2):498–507, 2008.
21. Lockwood, B. A. and Anitescu, M., Gradient-enhanced universal Kriging for uncertainty propagation, *Nucl. Sci. Eng.*, 170(2):168–195, 2012.
22. Rumpfkeil, M. P., Yamazaki, W., and Mavriplis, D. J., Uncertainty analysis utilizing gradient and Hessian information, in Kuzmin, A. (Ed.), *Computational Fluid Dynamics*, Springer, Berlin, pp. 261–268, 2011.
23. de Baar, J., Dwight, R., and Bijl, H., Improvements to gradient-enhanced Kriging using a Bayesian interpretation, *Int. Jo. Uncertainty Quantification*, 4:205–223, 2014.
24. Huijbregts, F. P., *Gradient-Enhanced, Response Surface Based, Aerodynamic Shape Optimization of a One-Stage High-Pressure Turbine*, MSc Thesis, TU Delft, The Netherlands, 2014.
25. de Baar, J., Scholcz, T., and Dwight, R., Exploiting adjoint derivatives in high-dimensional metamodels, *AIAA J.*, 53(5):1391–1395, 2015.
26. Keulen, F. and Vervenne, K., Gradient-enhanced response surface building, *Struct. Multidiscip. Optimization*, 27(5):337–351, 2004.
27. Kampolis, I. C., Karangelos, E. I., and Giannakoglou, K. C., Gradient-assisted radial basis function networks: theory and applications, *Appl. Math. Modell.*, 28(2):197–209, 2004.
28. Zenger, C., Sparse grids, In: Hackbush, W. (Ed.), *Parallel Algorithms for Partial Differential Equations, Notes on Numerical Fluid Mechanics, 31*, Vieweg, Braunschweig, 1991.
29. Griebel, M., Schneider, M., and Zenger, C., A combination technique for the solution of sparse grid problems, in Groen, P. and Beauwens, R. (Eds.), *Iterative Methods in Linear Algebra*, IMACS, Elsevier, North Holland, pp. 263–281, 1992. Also as SFB Bericht, 342/19/90 A, Institut für Informatik, TU München, 1990.
30. Bungartz, H.-J. and Griebel, M., Sparse grids, *Acta Num.*, 13:147–269, 2004.
31. Garcke, J., Sparse grids in a nutshell, in Garcke, J. and Griebel, M. (Eds.), *Sparse Grids and Applications*, Vol. 88 of Lecture Notes in Computational Science and Engineering, Springer, Berlin, pp. 57–80, 2013.
32. Ganapathysubramanian, B. and Zabaras, N., Sparse grid collocation schemes for stochastic natural convection problems, *J. Comput. Phys.*, 225:652–685, 2007.

33. Maître, O. L. and Knio, O., *Spectral Methods for Uncertainty Quantification; With Applications to Computational Fluid Dynamics*, Springer, Berlin, 2010.

34. Metropolis, N. and Ulam, S., The Monte Carlo method, *J. Am. Stat. Assoc.*, 44:335–341, 1949.

35. Bungartz, H.-J. and Dirnstorfer, S., Higher order quadrature on sparse grids, in Bubak, M., Albada, G. D., Sloot, P. M. A., and Dongarra, J. (Eds.), *International Conference on Computational Science*, Vol. 3039 of Lecture Notes in Computer Science, Springer, Berlin, pp. 394–401, 2004.

36. Rosenbrock, H. H., An automatic method for finding the greatest or least value of a function, *Computer J.*, 3:175–184, 1960.

37. Forrester, A. I., Sobester, A., and Keane, A. J., *Engineering Design via Surrogate Modelling, a Practical Guide*, Wiley, New York, 2008.

38. de Baar, J., *Stochastic Surrogates for Measurements and Computer Models of Fluids*, PhD Thesis, Delft University of Technology, Delft, 2014.

39. Jones, D. and Gaitonde, A., Future fast methods for loads calculations: The "FFAST" project, in Knörzer, D. and Szodruch, J. (Eds.), *Innovation for Sustainable Aviation in a Global Environment*, IOS Press BV, pp. 110–115, 2012.