MATHEMATICS OF COMPUTATION
Volume 68, Number 227, Pages 1121–1141
S 0025-5718(99)01033-9
Article electronically published on February 10, 1999

FOR NUMERICAL DIFFERENTIATION, DIMENSIONALITY CAN BE A BLESSING!

ROBERT S. ANDERSSEN AND MARKUS HEGLAND

ABSTRACT. Finite difference methods, such as the mid-point rule, have been applied successfully to the numerical solution of ordinary and partial differential equations. If such formulas are applied to observational data, in order to determine derivatives, the results can be disastrous. The reason for this is that measurement errors, and even rounding errors in computer approximations, are strongly amplified in the differentiation process, especially if small step-sizes are chosen and higher derivatives are required.

A number of authors have examined the use of various forms of averaging which allows the stable computation of low order derivatives from observational data. The size of the averaging set acts like a regularization parameter and has to be chosen as a function of the grid size h.

In this paper, it is initially shown how first (and higher) order single-variate numerical differentiation of higher dimensional observational data can be stabilized with a reduced loss of accuracy than occurs for the corresponding differentiation of one-dimensional data. The result is then extended to the multivariate differentiation of higher dimensional data. The nature of the trade-off between convergence and stability is explicitly characterized, and the complexity of various implementations is examined.

1. INTRODUCTION

It is assumed that given observational data can be characterised by a model of the form

$$y_{\mathbf{j}} = f(\mathbf{j}h) + \epsilon_{\mathbf{j}}, \quad h = 1/n, \quad \mathbf{j} = (j_1, \ldots, j_d) \in \{0, 1, \ldots, n\}^d, \tag{1}$$

where the function $f \in C^p(\mathbb{R}^d)$, for some appropriate choice of p, denotes the unknown trend, and the $\epsilon_{\mathbf{j}}$ the measurement errors, which are assumed to be independent and identically distributed normal variables with expectation 0 and variance σ^2.

The numerical differentiation problem consists in determining an approximation to the derivative Df of f, where D denotes a differentiation operator; e.g.

$$D = \sum_{|\mathbf{p}| \le q} c_{\mathbf{p}}(\mathbf{x}) \frac{\partial^{|\mathbf{p}|}}{\partial^{\mathbf{p}}\mathbf{x}}, \tag{2}$$

where the $c_{\mathbf{p}}(\mathbf{x})$ denote arbitrary coefficients, $\mathbf{p} = (p_1, \ldots, p_d)$ with $p_i \ge 0$, $\partial^{\mathbf{p}}\mathbf{x} = \partial^{p_1} x_1 \cdots \partial^{p_d} x_d$, and $|\mathbf{p}| = p_1 + \cdots + p_d$. Examples include the Laplacian, divergence, gradient and curl operators as well as other partial derivatives.

Received by the editor April 29, 1997 and, in revised form, October 9, 1997.
1991 *Mathematics Subject Classification.* Primary 65D25.
Key words and phrases. Numerical differentiation.

Numerical differentiation is ill-posed, and standard finite difference formulas, which have proved very useful for the solution of partial differential equations, often amplify the measurement error when applied to observational data. In particular, for small step sizes h, the good approximation properties of a finite difference formula are completely masked by the amplified measurement errors.

In [2], a scheme is proposed which allows one to construct stable estimates for the differentiation of one-dimensional observational data. This scheme uses averages of finite difference approximations over different step sizes to approximate the derivatives. For second derivatives, one such scheme is given by [1]

$$y_i^{[2]} := (2r+1)^{-3}h^{-2}\sum_{j=-r}^{r} y_{i+j+2r+1} - 2y_{i+j} + y_{i+j-2r-1}. \tag{3}$$

Even if the data y_i are contaminated by observational errors of the above type, this approximation converges, as $h \to 0$, to the second derivative $f^{(2)}(ih)$ as long as r behaves like h^{s-1} with $0 < s < 0.2$, and f has a continuous second derivative.

In the sequel, similar formulas for partial derivatives are developed. For example, it will be shown that the second partial derivative $\partial^2 f/\partial x^2$ can be approximated in a stable manner by

$$y_{i_1,i_2}^{[2,0]} := (2r+1)^{-4}h^{-2}\sum_{j_1=-r}^{r}\sum_{j_2=-r}^{r} y_{i_1+j_1+2r+1,i_2+j_2} - 2y_{i_1+j_1,i_2+j_2} + y_{i_1+j_2-2r-1,i_2+j_2}. \tag{4}$$

The condition which guarantees convergence is $r = h^{s-1}$ with $0 < s < 0.66$. An independent analysis of a different class of methods for higher-order numerical differentiation of multi-dimensional observational data can be found in Müller [10, p.77ff].

Compared with numerical differentiation, which is local and sensitive to the presence of observational errors in the given data, numerical integration, which is non-local, is normally well-posed, in the sense that any numerical quadrature estimate of the value of the integral is not sensitive to random perturbations in the evaluation of the integrand. However, in higher dimensions, because of the non-local character of integration, an exponentially large number of function evaluations must be performed, as the dimension increases, in order to achieve a given accuracy for the numerical estimate of the integral. On the other hand, the stability of the resulting estimate, even in the presence of observational errors, is automatically guaranteed because integration is well-posed. This need for an excessive number of function evaluations is often referred to as the *"Curse of Dimensionality"* (cf. [11, pp. 1–2]). For example, consider the integral

$$\mathcal{I} = \int_{[0,1]^d} f(\mathbf{x})d\mu \quad .$$

over the d-dimensional unit cube, where $d\mu = dx_1 \cdots dx_d$. Classical multi-dimensional integration rules for the integral $\mathcal{I}$ use Cartesian products of one-dimensional rules and take, for uniform grids, the form

$$\mathcal{I}_h = h^d \sum_{\mathbf{j}\in\{0,\ldots,n\}^d} w_{\mathbf{j}} y_{\mathbf{j}}.$$

The most popular product rule is the one based on the trapezoidal rule. The approximation error of the trapezoidal rule is $O(h^2)$, if all second partial derivatives $\partial^2 f/\partial x_i^2$ are continuous. Because the total number of function evaluations required for this approximation is $N = (n+1)^d$, the approximation error for the product trapezoidal rule is $O(N^{2/d})$. Consequently, for a given scheme, in order to achieve a 10 times decrease in the current approximation error, one requires a $10^{d/2}$ increase in the number of function evaluations. This yields an explicit characterization of the aforementioned "Curse of Dimensionality".

Note. In other contexts, such as data smoothing (cf. Hastie and Tibshirani [8]), the increase in function evaluations associated with the increase in dimension is also viewed as a "curse".

In numerical differentiation, there is no such curse. This is a direct consequence of the local nature of differentiation. In fact, in order to compute the derivative of a function at a certain point, only the behaviour of that function in the neighbourhood of that point needs to be known. For integration, however, the behaviour of the function throughout the domain of integration has an influence on the value of the integral. The interesting, even in some ways amazing, fact is that some of the "Curse of Dimensionality", associated with numerical integration, can help to "Cure" the "Curse of Ill-Posedness", associated with the evaluation of partial derivatives of multi-dimensional observational data. The ramifications of this observation are the focus of this paper.

2. First order differentiation of two-dimensional data

2.1. **One-dimensional averaging.** By analogy with the model for observational data introduced in equation (1), it is assumed that the given two-dimensional observational data have the form

$$y_{i,j} = f_{i,j} + \epsilon_{i,j}, \quad h = 1/n, \quad f_{i,j} = f(ih, jh), \quad i,j \in \{0,1,\ldots,n\}, \tag{5}$$

where the errors $\epsilon_{i,j}$ are taken to be independent normal random variables with mean 0 and variance σ^2. In addition, it is assumed throughout this section that $f \in C^{(3)}(\mathbb{R}^2)$. Even though the errors depend on two spatial parameters i and j, they are only assumed to be sampled from a one-dimensional distribution, since each data point (observation, measurement) $y_{i,j}$ of $f_{i,j}$ is performed in the same manner, independently of the spatial position defined by (ih, jh).

A standard approximation for the partial derivative $\partial f(x,y)/\partial x$ at the grid-points uses the midpoint rule

$$\dot{f}_{i,j}^{(mid)} = \frac{f_{i+1,j} - f_{i-1,j}}{2h}. \tag{6}$$

These formulas are defined for all i, j, except in the vicinity of the boundary of the region on which the differentiation is being performed, where different rules must be applied depending on the regularity of f in the neighbourhood of the boundary. If one applies this mid-point rule to the data of equation (5) instead of the function

values $f_{i,j}$, then one obtains

$$\hat{f}_{i,j}^{(mid)} = \frac{y_{i+1,j} - y_{i-1,j}}{2h} \tag{7}$$

$$= \frac{f(ih+h, jh) - f(ih-h, jh)}{2h} + \frac{\epsilon_{i+1,j} - \epsilon_{i-1,j}}{2h} \tag{8}$$

$$= \frac{\partial f}{\partial x}(ih, jh) + h^2 \frac{\partial^3 f}{\partial x^3}(ih + s_i h, jh) + \zeta_{i,j}, \quad s_i \in (-1, 1). \tag{9}$$

Though the errors $\zeta_{i,j} = (\epsilon_{i+1,j} - \epsilon_{i-1,j})/(2h)$ will be normally distributed random variables with zero mean and variance $h^{-2}\sigma^2/2$, they will not necessarily be independent (e.g. $\zeta_{i+2,j}$ and $\zeta_{i-2,j}$ are not independent, because they involve the common term $\epsilon_{i,j}$). However, the lack of independence is a minor matter compared with the value of the variance, which becomes arbitrarily large as the step size h tends to zero, since σ^2 is fixed (even if quite small). Consequently, when evaluated on the observational data $y_{i,j}$, the mid-point formula (6) fails to yield a convergent approximation to the partial derivative $\partial f(ih, jh)/\partial x$ as the step size h tends to zero. This illustrates why standard finite difference formulas for partial differentiation will tend to be unstable when evaluated on observational data.

The goal of this paper is the construction of stable alternatives to the standard formulas. The essence of the strategy to be adopted is encapsulated in the expression (8). The approximation error, determined by the second term on the right hand side of (9), is $O(h^2)$, and, consequently, is very small when h is small. Thus, the aim is to identify strategies which allow the error associated with $\zeta_{i,j}$ to be decreased at the expense of an increase in the approximation error. This kind of trade-off actually lies at the heart of any regularization method used to solve improperly posed problems. It is also the basis for the variance-bias interpretation of data smoothing.

For example, in standard numerical analysis texts, where it is tacitly assumed that one has control over the choice of the step length h, such a trade-off is achieved by defining the optimal choice $\bar{h}$ of h to be the value which guarantees that

$$h^2 \frac{\partial^3 f}{\partial x^3}(ih + s_i h, jh) = \zeta_{i,j}.$$

However, this strategy is limited to situations where the variance of $\zeta_{i,j}$ is relatively small, such as occurs in exact numerical situations where the only error is computer rounding error.

Consequently, in a deeper technical sense, it avoids the real issue as to how such a trade-off can be achieved as $h \to 0$. It is this aspect which is the focus of this paper. In fact, the key question being examined is:

> *"For given data where the size of the step length h has already been determined, how does one perform the numerical differentiation in order to fully utilize all the available data and achieve the type of trade-off mentioned above."*

A similar interpretation holds for the kernel smoothing methods applied to the numerical differentiation of observational data by statisticians (cf. Wand and Jones [12]).

If one keeps j fixed, numerical partial differentiation reduces to the classical one-dimensional situation, as is clear from equation (8). Stable finite difference formulas for first order differentiation, based on averaging, have been proposed by

Anderssen and de Hoog [2]. Such formulas achieve stability by averaging the values obtained from the application of the mid-point rule on a variety of different sized grids. In particular, consider the one-parameter family of mid-point rules

$$\dot{f}_{i,j}^{(mid)}[s] = \frac{f_{i+s,j} - f_{i-s,j}}{2sh}, \qquad s = 1,\ 2,\ \ldots .$$

If one applies these mid-point rules to the data of equation (5) instead of the function values $f_{i,j}$, then they generate a one-parameter family of identically distributed normal random variables

$$\tilde{\dot{f}}_{i,j}^{(mid)}[s] = \frac{y_{i+s,j} - y_{i-s,j}}{2sh} = \frac{f_{i+s,j} - f_{i-s,j}}{2sh} + \frac{\epsilon_{i+s,j} - \epsilon_{i-s,j}}{2sh}, \qquad s = 1,\ 2,\ \ldots ,$$

with mean $(\partial f(ih, jh)/\partial x + O(s^2h^2))$ and variance $2\sigma^2/(2sh)^2$. These random variables are independent because, for $s = 1,\ 2,\ \ldots$, each is evaluated on different data $y_{i,j}$ from equation (5). Consequently, for each i and j, one can average the mid-point estimates $\tilde{\dot{f}}_{i,j}^{(mid)}$ to obtain the following random variable:

$$\bar{\dot{f}}_{i,j}^{(mid)} = \frac{1}{r}\sum_{s=1}^{r} \tilde{\dot{f}}_{i,j}^{(mid)}[s] \tag{10}$$

$$= \frac{1}{r}\sum_{s=1}^{r} \frac{(f_{i+s,j} - f_{i-s),j})}{2sh} + \sum_{s=1}^{r} \frac{(\epsilon_{i+s,j} - \epsilon_{i-s,j})}{2rsh}. \tag{11}$$

Its mean and variance are given, respectively, by

$$\frac{1}{r}\sum_{s=1}^{r}(\partial f(ih, jh)/\partial x + O(s^2h^2)), \qquad 2\sum_{s=1}^{r}\sigma^2/(2rsh)^2.$$

This is the approach pursued by Anderssen and de Hoog [2], though in greater generality than indicated here. Among other things, they showed that, if r behaves like $O(h^\beta)$ with $\frac{2}{3} < \beta < 1$, then, asymptotically, this average has mean $\dot{f}_{i,j}$ and bounded variance.

Subsequently, Anderssen *et al.* [1] examined, again for one-dimensional differentiation (but extended to orders higher than the first), *"spatial neighbourhood averaging"*. Here, one first chooses a particular member of the one-parameter family by specifying the value of s. One then evaluates this mid-point formula on spatially adjacent grids as

$$\tilde{\dot{f}}_{i+k,j}^{(mid)}[s] = \frac{y_{i+k+s,j} - y_{i+k-s,j}}{2sh}, \qquad k = -r,\ \cdots ,\ r,$$

and averages them to generate the following random variable:

$$\bar{\dot{f}}_{i,j}^{(mid)} = \frac{1}{2r+1}\sum_{k=-r}^{r} \tilde{\dot{f}}_{i+k,j}^{(mid)}[s] \tag{12}$$

$$= \frac{1}{2r+1}\sum_{k=-r}^{r} \frac{(f_{i+k+s,j} - f_{i+k-s,j})}{2sh} + \sum_{k=-r}^{r} \frac{(\epsilon_{i+k+s,j} - \epsilon_{i+k-s,j})}{2(2r+1)sh}. \tag{13}$$

As long as r is less than the value of s, the individual random variables, corresponding to each mid-point formula, will be independent for $k = -r, \ldots, r$, since each of the data values $y_{i,j}$ entering the formulas $\tilde{\dot{f}}_{i+k,j}^{(mid)}[s]$, $k = -r, \cdots, r$, only appears once. Consequently, because the mean and variance of these individual

spatial mid-point formulas are, respectively, $(\partial f((i+k)h, jh)/\partial x + O(s^2h^2))$ and $2\sigma^2/(2sh)^2$, it follows that their average is a normally distributed random variable with mean and variance given by

$$\frac{1}{2r+1}\sum_{k=-r}^{r}\left(\frac{\partial f((i+k)h, jh)}{\partial x} + O(s^2h^2)\right), \qquad \frac{2}{2r+1}\sum_{k=-r}^{r}\frac{\sigma^2}{(2(2r+1)sh)^2}.$$

This step clearly indicates the additional technical considerations introduced by such spatial-neighbourhood averaging. In [1], as indicated above, one averages over different estimates of the same derivative, whereas, in this generalization, one is averaging over the same estimate of spatially-adjacent derivatives. The variance of the resulting random variable is easily computed (cf. Finney [7, Section 5.8]) to be

$$\frac{2\sigma^2}{(2r+1)(2sh)^2}.$$

The evaluation of the approximation error can be more complex, as one must average over the approximation errors arising from each spatial contribution, as well as over the discretization errors. In particular, because of the standard Taylor series result that

$$\begin{aligned}&\frac{\partial f((i+k)h, jh)}{\partial x} + \frac{\partial f((i-k)h, jh)}{\partial x}\\ &\quad = 2\frac{\partial f(ih, jh)}{\partial x} + \frac{\partial^3 f((i+\theta k)h, jh)}{\partial x^3}O((kh)^2), \qquad -1 < \theta < 1,\end{aligned}$$

the mean of $\overset{=(mid)}{f}_{i,j}$ can be rewritten as

$$\frac{1}{2r+1}\left(\sum_{k=-r}^{r}\frac{\partial f(ih, jh)}{\partial x} + O(k^2h^2)\right) + \frac{1}{2r+1}\left(\sum_{k=-r}^{r}O(s^2h^2))\right).$$

Consequently, the averaging of the individual approximation errors generates two terms, which will be referred to as the *"averaging error"* and the *"averaged discretization error"*. In [1], as indicated above, the averaging is arranged so that one only has to analyse an averaged discretization error, whereas, in the spatial neighbourhood averaging being examined here, one will have both. Because $0 \leq k \leq r$ and $\sum_{k=1}^{r} k^2 \sim r^3$, the averaged discretization error will dominate and the mean of $\overset{=(mid)}{f}_{i,j}$ becomes, with $r < s$,

$$\frac{\partial f(ih, jh)}{\partial x} + O(s^2h^2),$$

if it is assumed, as indicated earlier, that $\partial^3 f((i+\theta k)h, jh)/\partial x^3$ is bounded. Thus, the average and the mean of the random variable $\overset{=(mid)}{f}_{i,j}$ take, respectively, the form

$$\frac{\partial f(ih, jh)}{\partial x} + O(s^2h^2), \qquad \frac{2\sigma^2}{(2r+1)(2sh)^2}. \tag{14}$$

From the point of view of subsequent deliberations, it is important to note, at this stage, that the averaging has a linear effect on the approximation error associated with the accuracy of the mean, and a quadratic effect on the value of the variance. In fact, it is the exploitation of this difference which allows the type of results given below to be derived.

The variance of this averaged midpoint estimate is approximately $0.125r^{-3}$ times the variance of the standard midpoint estimate of equation (7). If r is taken to be of order $h^{-2/3}$, this averaged midpoint value is stable, as there is no growth of the variance as $h \to 0$. The price paid to achieve this stability is an increase in the approximation error, resulting from the use of mid-point formulas with a step size sh as well as from the averaging. Together, these two processes induce an approximation error of order $O(h^{14/19})$ [1].

2.2. **Two-dimensional averaging.** For the two-dimensional data $y_{i,j}$ (where j varies as well as i), the key observation is that, in estimating the partial derivative $\partial f/\partial x$, one is not limited to doing the spatial-neighbourhood averaging only in the x-direction, as examined and motivated above. One is now free to also do the spatial neighbourhood averaging in the second dimension; i.e. with respect to some specified choice of s, average the mid-point estimates

$$\tilde{\dot{f}}^{(mid)}_{i,j+k}[s] = \frac{y_{i+s,j+k} - y_{i-s,j+k}}{2sh}, \quad k = -r, \cdots, r.$$

The clear advantage of this step is that it removes the problem of error correlation, because there is no duplication of the data values entering the above family of mid-point estimates. In this way, there is not longer a need to demand that $r < s$. For example, one could construct the estimate of $\dot{f}_{i,j}$ as the average

$$\bar{\dot{f}}^{(mid)}_{i,j} = \frac{1}{2r+1} \sum_{k=-r}^{r} \tilde{\dot{f}}^{(mid)}_{i,j+k}[1]. \tag{15}$$

For each i, j, this defines a normally distributed random variable with mean and variance given, respectively, by

$$\frac{1}{2r+1} \sum_{k=-r}^{r} \frac{\partial f}{\partial x}(ih, (j+k)h) + O(h^2), \qquad \frac{\sigma^2}{2(2r+1)h^2}.$$

Here, however, one finds, on appealing to standard Taylor series result of the type given above, that the averaging error dominates, and the value of the mean becomes

$$\frac{\partial f(ih, jh)}{\partial x} + O(r^2h^2).$$

The effect of this type of averaging is to reduce the variance associated with the standard mid-point formula (7) by a factor of $O(r^{1/2})$. The cost for this reduction has been achieved at the expense of the size of the approximation error, which has been increased by a factor of order $O(r^2h^2)$. But, because stability can only be guaranteed if r is of order h^{-2}, the size of the corresponding approximation error increases, to become of order $O(h^{-2})$. Consequently, this scheme is only convergent when f is only a function of the second variable. It therefore follows that the larger step size in the scheme which generates $\bar{\dot{f}}^{(mid)}_{i,j}$ plays a crucial role. It is needed not only to guarantee the independence of the random variables which are averaged, but also to guarantee its simultaneous convergence and stability.

Introducing a larger step size into the chosen mid-point formula which is spatially neighbourhood averaged in the second dimension, one obtains the following random

variable estimate for $\dot{f}_{i,j}$:

$$\hat{\dot{f}}_{i,j}^{(mid)} = \frac{1}{2r+1} \sum_{k=-r}^{r} \frac{y_{i+s,j+k} - y_{i-s,j+k}}{2sh}. \tag{16}$$

For each i, j, this random variable has mean and variance given, respectively, by

$$\frac{1}{2r+1} \sum_{k=-r}^{r} \left(\frac{\partial f(ih,(j+k)h)}{\partial x} + O(s^2h^2) \right), \qquad \frac{2}{2r+1} \sum_{k=-r}^{r} \frac{\sigma^2}{(2(2r+1)sh)^2},$$

and, hence,

$$\frac{\partial f(ih,jh)}{\partial x} + O(r^2h^2) + O(s^2h^2), \qquad \frac{\sigma^2}{2(2r+1)s^2h^2}.$$

Stability is guaranteed if rs^2 is of the order of h^{-2}. As well as the averaged discretization error $O(s^2h^2)$, associated with the chosen mid-point formula, the spatial neighbouring averaging generates an averaging error of order $O((rh)^2)$.

For small h, one aims to balance the averaging and the averaged discretization error, because, if one dominated, the other could be increased without much loss of accuracy, but an improvement in the stability. Consequently, r and s are chosen to be of order $O(h^{-2/3})$. Contrary to what might have been anticipated, this type of averaging does not improve on the spatial neighbourhood averaging in the x-direction. One still has a method with an accuracy of $O(h^{2/3})$.

The next possibility combines the two previous approaches. The resulting spatial neighbourhood averaged midpoint rule takes the form

$$\hat{\dot{f}}_{i,j} = \frac{1}{(2r_1+1)(2r_2+1)} \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} \frac{y_{i+m+s,\,j+k} - y_{i+m-s,\,j+k}}{2sh}, \qquad r_1 < s. \tag{17}$$

The constraint $r_1 < s$ ensures that no correlated errors are generated. The mean and variance of this normally distributed random variable estimate of $\dot{f}_{i,j}$ are given, respectively, by

$$\frac{1}{(2r_1+1)(2r_2+1)} \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} \left(\frac{\partial f((i+m)h,(j+k)h)}{\partial x} + O(s^2h^2) \right)$$
$$= \frac{\partial f(ih,jh)}{\partial x} + O((r_1^2 + r_2^2 + s^2)h^2),$$

and

$$\frac{2\sigma^2}{(2r_1+1)(2r_2+1)(2sh)^2}.$$

Consequently, stability is guaranteed if $r_1 r_2 s^2$ is of order h^{-2}. For the reasons outlined above, r_1, r_2 and s are chosen to have the same order, which, in the current situation, implies $r_1 \sim r_2 \sim s \sim O(h^{-1/2})$. Thus, as a direct result of the additional spatial neighbourhood averaging in the y-direction, one obtains a stable scheme which has an accuracy of $O(h)$. In this way, dimensionality has become a blessing for numerical differentiation, because it has allowed one to construct a stable scheme with an improved accuracy over that obtainable from the standard schemes.

2.3. **Implementation.** The estimate $\hat{\hat{f}}_{i,j}$ can be evaluated in two separate ways. On the one hand, one first computes the differences

$$u_{i+m,j+k} := \frac{y_{i+m+s,\,j+k} - y_{i+m-s,\,j+k}}{2sh}, \quad k = -r_1, \cdots, r_1, \quad m = -r_2, \cdots, r_2,$$

and then averages them to obtain

$$\hat{\hat{f}}_{i,j} = \frac{1}{(2r_1+1)(2r_2+1)} \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} u_{i+m,\,j+k}.$$

On the other hand, one first constructs the averages

$$w_{i+t,j} := \frac{1}{(2r_1+1)(2r_2+1)} \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} y_{i+t+m,\,j+k}, \qquad t = \pm s,$$

and then computes the difference

$$\hat{\hat{f}}_{i,j} = \frac{w_{i+s,\,j} - w_{i-s,\,j}}{2sh}.$$

For the evaluation of a single derivative, the first implementation involves $2(2r_1+1)(2r_2+1) - 1$ additions (subtractions) and $(2r_1+1)(2r_2+1)$ divisions, while the second involves the same number of additions, but only 5 divisions. Consequently, if divisions should be minimized, then the second implementation is always the preferred option. On the other hand, because one is dividing by terms which are not dependent on the specific locations of the data points, both can be rewritten so that the division by $2(2r_1+1)(2r_2+1)sh$ is performed as the last step. In this way, the goal reduces to one of minimizing the number of additions, and, hence, the minimization of the number of times that the expensive averaging-step must be performed. If only a single partial derivative is required, then both schemes are equivalent, since they both require the averaging to be performed twice. However, the situation changes when more complex partial derivatives are required over an array of locations, which is a commonly occurring situation in various applications including, for example, the evaluation of a two-dimensional velocity field.

In fact, the results are slightly counter-intuitive. Consider an $N \times M$ (square lattice) array of locations, over which some specified partial differential operator is required. For the gradient $\nabla f = (\partial f/\partial x_1, \partial f/\partial x_2)$, the first procedure reduces to initially computing the differences

$$u_{i,j} := y_{i+s_1,\,j} - y_{i-s_1,\,j}$$

and

$$v_{i,j} := y_{i,\,j+s_2} - y_{i,\,j-s_2},$$

on the $N \times M$ array, and then evaluating the gradient using the two averaging steps

$$(f_{x_1})_{i,j} := \frac{1}{2(2r_1+1)(2r_2+1)s_1h} \sum_{k=-r_1}^{r_1} \cdot \sum_{m=-r_2}^{r_2} u_{i+m,\,j+k}$$

and

$$(f_{x_2})_{i,j} := \frac{1}{2(2r_1+1)(2r_2+1)s_2h} \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} v_{i+m,\,j+k}.$$

This procedure therefore involves $2NM$ applications of the averaging step.

On the other hand, the second procedure reduces to initially performing the following summations for the averaging:

$$w_{i,j} := \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} y_{i+m,j+k},$$

on the $N \times M$ array, and then evaluating the gradient as

$$(f_{x_1})_{i,j} := \frac{w_{i+s_1,\,j} - w_{i-s_1,\,j}}{2(2r_1+1)(2r_2+1)s_1 h}$$

and

$$(f_{x_2})_{i,j} := \frac{w_{i,\,j+s_2} - w_{i,\,j-s_2}}{2(2r_1+1)(2r_2+1)s_2 h}.$$

Typically, since it only involves NM applications of the averaging step, the second procedure is (approximately) twice as fast for the numerical evaluation of the gradient computation as the first.

For the computation of the divergence $F = \nabla \cdot (f,\ g) = (\partial f/\partial x_1 + \partial g/\partial x_2)$, it is the first procedure which has the better complexity. One starts with the two functions f and g, and then computes derivatives which are added to give the scalar result. For the first procedure, one combines the computation of the differences and their addition in order to determine the values

$$\hat{F}_{i,j} := f_{i+s,\,j} - f_{i-s,\,j} + g_{i,\,j+s} - g_{i,\,j-s},$$

on the $N \times M$ array, and then evaluates the divergence as

$$\bar{F}_{i,j} := \frac{1}{2(2r_1+1)(2r_2+1)sh} \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} \hat{F}_{i+m,\,j+k},$$

which only involves NM applications of the averaging step. In order to reduce the number of divisions to one, which is performed as the last step, one must choose $s_1 = s_2 = s$. For the second procedure, twice the number of applications of the averaging is required. One initially computes the summations

$$w_{i,j} := \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} f_{i+m,\,j+k}$$

and

$$z_{i,j} := \sum_{k=-r_1}^{r_1} \sum_{m=-r_2}^{r_2} g_{i+m,\,j+k},$$

on the $N \times M$ array, and then evaluates the divergence as

$$\bar{F}_{i,j} := \frac{w_{i+s,\,j} - w_{i-s,\,j} + z_{i,\,j+s} - z_{i,\,j-s}}{2(2r_1+1)(2r_2+1)sh}.$$

The evaluation of the curl (i.e. $\partial f/\partial x_2 - \partial g/\partial x_1$) proceeds in exactly the same way as for the divergence.

Remark. This ability to move the division to the last step, and, thereby, reduce complexity considerations to the number of averaging steps performed, ceases, if the grid does not remain a square lattice.

3. Higher order derivatives and higher dimensions: Stability and convergence

3.1. **Basic theory.** The basic method of the previous section is now generalised to the numerical evaluation of the q-th order (homogeneous) differential operator

$$Df(\mathbf{x}) = \sum_{|\mathbf{p}|=q} c_{\mathbf{p}} \frac{\partial^q f(\mathbf{x})}{\partial^{\mathbf{p}} \mathbf{x}},$$

where the function $f : \Omega \subset \mathbb{R}^d \to \mathbb{R}$ is assumed to have sufficient smoothness and the $c_{\mathbf{p}}$ are constants. The approximation is computed from the following set of observed function values:

$$y_{\mathbf{j}} = f(\mathbf{j}h) + \epsilon_{\mathbf{j}}, \quad h = 1/n, \quad \mathbf{j} = (j_1, \dots, j_d) \in I = \{0, 1, \dots, n\}^d,$$

where I denotes the index set of the locations of the data, $hI \subset \Omega$, and the observational errors $\epsilon_{\mathbf{j}}$ are normally distributed independent random variables with expectation 0 and variance σ^2. The multidimensional data array with components $y_{\mathbf{j}}$ will be denoted by $\mathbf{y}$, while the linear space of all such multidimensional data arrays will be denoted by $\mathbb{R}^I$. Furthermore, let

$$f_{\mathbf{j}} = f(\mathbf{j}h), \quad h = 1/n, \quad \mathbf{j} = (j_1, \dots, j_d),$$

and let $\mathbf{f} \in \mathbb{R}^I$ denote the multidimensional array containing the exact function values. The Laplacian

$$Df = \triangle f = \sum_{i=1}^{d} \frac{\partial^2 f}{\partial x_i^2}$$

is an example of such a differential operator.

The generalization consists of two steps; namely, an averaging step followed by a differentiation step. In order to simplify the algebra, it is assumed that $\mathbf{y}$ has been extended periodically to $\mathbb{Z}^d$. Let $V = \{-r_1, \dots, r_1\} \times \cdots \times \{-r_d, \dots, r_d\}$, where $r_i \geq 0$ are given integers, define the index set of the template on which the averaging is performed, and M_V denote the standard averaging operator on this template, where each data point averaged has equal weight. The index set $V \subset I$ will be called the support of M_V. A smooth estimate $\widehat{f}_{\mathbf{j}}$ of $f_{\mathbf{j}}$ is generated by applying the *averaging operator* M_V to the data $\mathbf{y}$,

$$\widehat{f}_{\mathbf{j}} = (M_V \mathbf{y})_{\mathbf{j}} = \frac{1}{|V|} \sum_{\mathbf{k} \in V} y_{\mathbf{j}-\mathbf{k}}, \quad \mathbf{j} \in I, \tag{18}$$

where $|V|$ denotes the size (number of members) of the index set V. Because of the way in which V has been defined, M_V takes the form of a convolution.

Since, in general, f is not periodic, the estimates $\widehat{f}_{\mathbf{j}}$ will only be good approximations of $f_{\mathbf{j}}$ when $\mathbf{j} \in I'$, where I' is the index set of the array of averaged values, i.e., the maximal set such that

$$I' + V \subset I.$$

For the error analysis presented below, the second derivative of f will be required. The Hessian of f will be denoted by

$$H_f(\mathbf{x}) = \left[\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} \right]_{i,j=1,\dots,d},$$

and its matrix 2-norm by $\|H_f(\mathbf{x})\|$. The second moment of the characteristic function of V is defined by

$$\mu_2(V) = \frac{1}{|V|}\sum_{\mathbf{k}\in V}\|\mathbf{k}\|^2 = \frac{1}{3}\sum_{i=1}^{d} r_i(r_i+1)\left(\leq \frac{2}{3}\|\mathbf{r}\|^2\right), \tag{19}$$

where $\|\mathbf{k}\|^2 = k_1^2 + \cdots + k_d^2$ and $\mathbf{r} = (r_1, \ldots, r_d)$.

Proposition 1. *If $f \in C^2(\Omega)$, $f_{\mathbf{j}} = f(\mathbf{j}h)$, and $\bar{\mathbf{f}} = M_V\mathbf{f}$, then*

$$\left|\bar{f}_{\mathbf{j}} - f_{\mathbf{j}}\right| \leq \frac{h^2}{2}\|H_f(\mathbf{z})\|\,\mu_2(V), \quad \mathbf{j} \in I',$$

where $|z_i - j_i h| \leq hr_i, \quad i = 1, \ldots, d.$

Proof. From Taylor's theorem, one obtains

$$f_{\mathbf{j}-\mathbf{k}} = f((\mathbf{j}-\mathbf{k})h) = f(\mathbf{j}h) - h\nabla f(\mathbf{j}h)^T\mathbf{k} + \frac{1}{2}h^2\mathbf{k}^T H_f(\xi')\mathbf{k}$$

for some $\xi' \in \mathbf{j}h + hV$. Since V is symmetric, averaging over V eliminates any linear expression in $\mathbf{k}$. In particular, $\sum_{\mathbf{k}\in V}\mathbf{k} = 0$. The proposition is proved by applying M_V to the the standard bound

$$|\mathbf{k}^T H_f(\mathbf{z})\mathbf{k}| \leq \|\mathbf{k}\|^2\|H_f(\mathbf{z}')\|$$

and then invoking the convexity of the averaging performed by M_V, the continuity of the L_2-norm (equivalent to the largest singular value), and the Bolzano intermediate value theorem for $\|H_f(\mathbf{z}')\|$. □

This approximation error can be viewed as an undesirable side-effect of the averaging. On the other hand, its advantage is the potential reduction it can generate in the variance of the random errors in the data. The nature of this reduction is encapsulated in the following proposition.

Proposition 2. *If the errors $\epsilon_{\mathbf{j}}$ are i.i.d. $\mathcal{N}(0,\sigma^2)$ random variables, then the $(M_V\epsilon)_{\mathbf{j}}$ are $\mathcal{N}(0,\sigma^2/|V|)$ random variables. Furthermore, if $\mathbf{j}-\mathbf{j}' \notin 2V$, then the random variables $(M_V\epsilon)_{\mathbf{j}}$ and $(M_V\epsilon)_{\mathbf{j}'}$ are independent.*

Proof. The variance reduction is standard. Furthermore, $(M_V\epsilon)_{\mathbf{j}}$ depends only on $\epsilon_{\mathbf{i}}$ for $\mathbf{i} \in \mathbf{j} + V$. Consequently, if $\mathbf{j}-\mathbf{j}' \notin 2V$, then $(M_V\epsilon)_{\mathbf{j}}$ and $(M_V\epsilon)_{\mathbf{j}'}$ depend on disjoint sets of components of ϵ, and, therefore, are independent. □

The condition $\mathbf{j}-\mathbf{j}' \notin 2V$ is equivalent to the condition $|j_i - j_i'| \geq 2r_i + 1$ for $i = 1, \ldots, d$.

The *numerical evaluation* of the derivative Df on the data $\mathbf{y}$ at the point $\mathbf{j}$ can be formulated as the application of a linear operator $A[\mathbf{s}]$ on $\mathbb{R}^I$ defined by

$$(A[\mathbf{s}]\,\mathbf{y})_{\mathbf{j}} = \sum_{\mathbf{i}\in I}\alpha_{\mathbf{j}-\mathbf{i}}\,y_{\mathbf{i}},$$

where

1. the coefficients $\alpha_{\mathbf{i}}$ are assumed to be periodic with period I,
2. with respect to some given index set $\mathbf{s} = (s_1, \ldots, s_d) \in \mathbb{N}^d$, which characterises the support (spacing) of the points at which the operator $A[\mathbf{s}]$ acts, the components $\alpha_{\mathbf{i}}$ are only nonzero, if, for $k_i \in \mathbb{Z}$, $i = 1, \ldots d$,

$$\mathbf{i} = (k_1 s_1, \ldots, k_d s_d),$$

3. the difference approximation $A[\mathbf{s}]$ approximates the differential operator D in the sense that, if $f'_{\mathbf{j}} = (Df)(\mathbf{j}h)$ and $\dot{\mathbf{f}} = A[\mathbf{s}]\,\mathbf{f}$, then

$$|\dot{f}_{\mathbf{j}} - f'_{\mathbf{j}}| \leq h^2 \|\mathbf{s}\|^2 C(f), \tag{20}$$

where $C(f)$ depends on the values of certain $q+2$nd partial derivatives of f.

As the $\alpha_{\mathbf{j}}$ are periodic, the linear operators $A[\mathbf{s}]$ correspond to convolutions. One is now in a position to combine the *numerical differentiation* operator $A[\mathbf{s}]$ with the averaging operator M_V, in order to construct stable numerical differentiation rules. In fact, because M_V is also a convolution, it commutes with $A[\mathbf{s}]$, i.e.,

$$M_V A[\mathbf{s}] = A[\mathbf{s}] M_V.$$

Examples of such numerical differentiation rules include the well-known midpoint rules discussed above in Section 2.

The following proposition yields a bound on the approximation error of the averaged finite difference.

Proposition 3. *Let M_V be the averaging operator with $V = \{-r_1, \ldots, r_1\} \times \cdots \times \{-r_d, \ldots, r_d\}$, let $A[\mathbf{s}]$ be a difference operator, and let $\bar{\dot{\mathbf{f}}} = A[\mathbf{s}] M_V \mathbf{f}$. Furthermore, let $f'_{\mathbf{j}} = (Df)(\mathbf{j}h)$. If the function f is such that $Df \in C^2$, then*

$$|\bar{\dot{f}}_{\mathbf{j}} - f'_{\mathbf{j}}| \leq h^2 \|\mathbf{s}\|^2 C(f) + \frac{h^2}{2} \|H_{Df}\|\, \mu_2(V),$$

where $I' \subset I$ is the index set of the array of averaged values.

Proof. From the triangle inequality, one obtains

$$|\bar{\dot{f}}_{\mathbf{j}} - f'_{\mathbf{j}}| \leq |\bar{\dot{f}}_{\mathbf{j}} - w_{\mathbf{j}}| + |w_{\mathbf{j}} - f'_{\mathbf{j}}|,$$

where $\mathbf{w} = M_v \mathbf{f}'$. The first term on the right side can be written as $(M_V \mathbf{u})_{\mathbf{j}}$, where $\mathbf{u} = A[\mathbf{s}]\mathbf{f} - \mathbf{f}'$. The values $|u_{\mathbf{j}}|$ of the components of the multidimensional array $\mathbf{u}$ are bounded by $h^2 \|\mathbf{s}\|^2 C(f)$. Furthermore, the averaging operator is uniformly bounded, as can be proved by applying Schwarz' inequality. In particular, if $|u_{\mathbf{j}}| \leq c$ for some constant c, then $|(M_V \mathbf{u})_{\mathbf{j}}| \leq c$, since

$$(M_V \mathbf{u})_{\mathbf{j}}^2 \leq \frac{1}{|V|^2} |V| \sum_{\mathbf{k} \in V} u_{\mathbf{j}+\mathbf{k}}^2.$$

Thus, the effect of the averaging $M_V \mathbf{u}$ can be bounded by

$$|\hat{\dot{f}}_{\mathbf{j}} - w_{\mathbf{j}}| = |(M_V \mathbf{u})_{\mathbf{j}}| \leq h^2 \|\mathbf{s}\|^2 C(f).$$

For the estimation of the second term in the triangle inequality, one uses the smoothness of f (i.e., $Df \in C^2$) and Proposition 1 to obtain

$$|w_{\mathbf{j}} - f'_{\mathbf{j}}| \leq \frac{h^2}{2} \|H_{Df}(\mathbf{v})\|\, \mu_2(V), \quad \mathbf{j} \in I',$$

where the choice of $\mathbf{v}$ is characterised in Proposition 1. The required bound of this proposition is now an immediate consequence of these two results. □

One now turns to the estimation of bounds for the amplification of the measurement errors. A first lemma shows that, if the "spacing" $\mathbf{s}$ of the finite difference formula is sufficiently large, then the error amplification is essentially the 2-norm of the coefficients of the difference stencil (template).

Lemma 1. *Let M_V be the averaging operator with $V = \{-r_1, \dots, r_1\} \times \cdots \times \{-r_d, \dots, r_d\}$, and let $A[\mathbf{s}]$ be a difference operator. If $\epsilon_{\mathbf{j}}$ are independent normally distributed random variables with expectation 0 and variance σ^2 and if*

$$s_i \geq 2r_i + 1,$$

then the $\zeta_{\mathbf{j}} := (A[\mathbf{s}]M_V\epsilon)_{\mathbf{j}}$ are normally distributed random variables with expectation 0 and variance

$$\operatorname{var}(\zeta_{\mathbf{j}}) = \frac{\sigma^2}{|V|}\sum_{\mathbf{i}\in I}\alpha_{\mathbf{i}}^2 = \frac{\sigma^2}{|V|}\|A[\mathbf{s}]\mathbf{e}\|^2,$$

where $\mathbf{e}$ is the array with all its components zero except for the origin $\mathbf{j} = 0$, where $e_{\mathbf{j}} = 1$.

Proof. Let $\eta = M_V\epsilon$. Because M_V and $A[\mathbf{s}]$ commute, it follows that

$$\zeta = M_V A[\mathbf{s}]\epsilon = A[\mathbf{s}]M_V\epsilon = A[\mathbf{s}]\eta.$$

The linearity of the operators involved implies that $E(\zeta) = 0$, where $E(\cdot)$ denotes the expectation operator. Thus, one obtains that

$$\operatorname{var}(\zeta_{\mathbf{j}}) = E(\zeta_{\mathbf{j}}^2) = \sum_{\mathbf{i},\mathbf{k}\in I}\alpha_{\mathbf{j}-\mathbf{i}}\alpha_{\mathbf{j}-\mathbf{k}}E(\eta_{\mathbf{i}}\eta_{\mathbf{k}}).$$

The only terms which contribute to the sum are the ones for which $\mathbf{j} - \mathbf{i} = N_1\mathbf{s}$ and $\mathbf{j} - \mathbf{k} = N_2\mathbf{s}$, and, for these terms, $\mathbf{k} - \mathbf{i} = N_3\mathbf{s}$, where the $N_t = \operatorname{diag}(n_1, \dots, n_d)$, $n_i \in \mathbb{N}$, $t = 1, 2, 3$, denote integer matrices. Furthermore, for $\mathbf{k} - \mathbf{i} \notin 2V$, $E(\overline{\eta}_{\mathbf{i}}\overline{\eta}_{\mathbf{k}}) = 0$. Since, for $s_i \geq r_i + 1$, the only point $N\mathbf{s}$ which is an element of $2V$ is 0, the result of the lemma follows. □

The next lemma yields a bound for the error amplification. The condition on $A[\mathbf{s}]$ guarantees the validity of the intermediate mean-value theorem for differentiation, and holds for all standard finite difference formulas.

Lemma 2. *Let $A[\mathbf{s}]$ be a difference operator which satisfies the consistency condition that, for all periodic f with continuous Df, there exists, for every $\mathbf{k} \in I$, a vector $\mathbf{z} \in \Omega$ such that $[A[\mathbf{s}]\mathbf{f}]_{\mathbf{k}} = Df(\mathbf{z})$. Then*

$$\|A[\mathbf{s}]\mathbf{e}\| \leq (2\pi)^q h^{-(q+d/2)}\sum_{|\mathbf{p}|=q}|c_{\mathbf{p}}|\prod_{i=1}^{d} s_i^{-(p_i+1/2)}.$$

Proof. Let $f(\mathbf{x}) = \prod_{i=1}^{d} f_i(x_i)$ with

$$f_i(x_i) = \frac{s_i}{n}\sum_{k=0}^{n/s_i-1}\exp(-2\pi\sqrt{-1}\,k\,x_i).$$

It follows that f is periodic and C^∞. Furthermore, $f_i(s_i h\,k)$ is one, if $k = 0$, and zero otherwise; i.e., $f(\mathbf{x})$ is a Lagrange function for the grid points given by $N\mathbf{s}$. This proves that the function f interpolates $\mathbf{e}$. Consequently, if $f_{\mathbf{j}} = f(\mathbf{j}h)$ then $A[\mathbf{s}]\mathbf{f} = A[\mathbf{s}]\mathbf{e}$ and, for every $\mathbf{k}$, there is a $\mathbf{z}$ such that (mean value theorem)

$$[A[\mathbf{s}]\mathbf{e}]_{\mathbf{k}} = Df(\mathbf{z}).$$

This proves that $\mathbf{f}$ interpolated $\mathbf{e}$. Since the derivatives of the component functions f_i are bounded by

$$\left| \frac{d^{p_i} f_i(x_i)}{dx_i^{p_i}} \right| \leq \left(\frac{2\pi n}{s_i} \right)^{p_i},$$

it follows that, since $n = 1/h$,

$$|Df(\mathbf{z})| \leq \sum_{|\mathbf{p}|=q} |c_{\mathbf{p}}| \prod_{i=1}^{d} \left(\frac{2\pi}{h s_i} \right)^{p_i}.$$

Because these bounds are independent of $\mathbf{z}$, and because at most $\prod_{i=1}^{d} \frac{n}{s_i}$ of the components of $A[\mathbf{s}]\mathbf{e}$ are different from zero, one obtains the bound

$$\|A[\mathbf{s}]\mathbf{e}\| \leq \left(\prod_{i=1}^{d} \frac{n}{s_i} \right)^{1/2} \sum_{|\mathbf{p}|=q} |c_{\mathbf{p}}| \prod_{i=1}^{d} \left(\frac{2\pi}{h s_i} \right)^{p_i}.$$

Rearranging terms yields the given bound. □

For many practical stencils this bound is rather pessimistic, as it only assumes consistency. However, often one can explicitly compute $\|A[\mathbf{s}]\mathbf{e}\|$ and apply Lemma 1 directly.

Proposition 4. *Let M_V be the averaging operator with $V = \{-r_1, \ldots, r_1\} \times \cdots \times \{-r_d, \ldots, r_d\}$, and let $A[\mathbf{s}]$ be a difference operator. In addition, let $\epsilon_{\mathbf{j}}$ be i.i.d. normal random variables with expectation 0 and variance σ^2, and $\zeta = A[\mathbf{s}]M_V\epsilon$. Furthermore, assume that the consistency condition of the previous lemma holds for $A[\mathbf{s}]$. If*

$$s_i \geq 2r_i + 1,$$

then $\zeta_{\mathbf{j}}$ are normally distributed random variables with expectation 0 and variance bounded by

$$\begin{aligned} \operatorname{var}(\zeta_{\mathbf{j}}) &\leq (2\pi)^{2q} \sigma^2 h^{-(2q+d)} \prod_{i=1}^{d} (2r_i + 1)^{-1} \left(\sum_{|\mathbf{p}|=q} |c_{\mathbf{p}}| \prod_{i=1}^{d} s_i^{-(p_i+1/2)} \right)^2 \\ &\leq (2\pi)^{2q} \sigma^2 h^{-(2q+d)} \left(\sum_{|\mathbf{p}|=q} |c_{\mathbf{p}}| \prod_{i=1}^{d} (2r_i + 1)^{-(p_i+1)} \right)^2. \end{aligned}$$

Proof. The proposition follows by combining the results of the two previous lemmas. □

A numerical differentiation rule is said to be *stable*, if the error amplification is bounded. As a consequence of the previous proposition, one obtains

Corollary 1. *If, for all $c_{\mathbf{p}} \neq 0$ with $\mathbf{p} = (p_1, \ldots, p_d)$, there is a constant K such that the inequality*

$$\prod_{i=1}^{d} (2r_i + 1)^{p_i+1} \geq K h^{-(q+d/2)}$$

holds, then the averaged differentiation rules, defined above, are stable. They are also convergent, if, for all $i = 1, \ldots, d$, *there exist constants* K_1 *and* K_2 *and* $\beta_1, \beta_2 < 1$ *such that*

$$s_i \leq K_1 h^{-\beta_1} \quad \textit{and} \quad r_i \leq K_2 h^{-\beta_2}.$$

Any realistic template (stencil) V must not be too elongated in any one of the coordinate directions. A natural constraint, which achieves this, is to require that

$$r_1 \sim r_2 \sim \cdots \sim r = \frac{|V|^{1/d}}{2}.$$

On the other hand, from the earlier assumptions about the index set I it follows that

$$n + 1 = |I|^{1/d},$$

and hence

$$h \sim |I|^{-1/d}.$$

In this way, the study of the stability and convergence of multi-dimensional numerical (partial) differentiation reduces, for a given partial derivative and multi-dimensional space with dimension d, to specifying how the size of $|V|$ must change relative to the size of $|I|$. For example, if $p_i = p$, $i = 1, 2, \cdots, d$, then $q = pd$. It therefore follows from Corollary 1 that stability is guaranteed if

$$(2r)^{q+d} \geq K h^{-(q+d/2)} \geq K n^{(q+d/2)},$$

and, hence, that

$$|V|^{(p+1)} \geq K 2^{-(q+d/2)} |I|^{p+1/2}.$$

This represents a heuristic proof that stability is guaranteed if, for a suitable chosen ν,

$$|V| \sim |I|^{\nu}, \ 0 < \nu < 1.$$

The admissible range of values of ν which guarantee both convergence and stability now follows on combining this result with the estimate of Proposition 3.

3.2. **Implementation.** As for the gradient in two dimensions (cf. Section 2.3), the way the computations are performed can affect the number of floating point operations required to evaluate multi-dimensional numerical derivatives. In order to quantify this, it is necessary to examine the complexity of the various implementations.

For example, the evaluation of any partial derivative over the whole domain involves the formation of the double matrix vector product $A[\mathbf{s}]M_V y$. This is done in two stages which essentially consist of averaging and differencing. The division associated with the averaging stage is moved into the differencing stage. Consequently, the first stage consists in forming the sum

$$u_{\mathbf{j}} = \sum_{\mathbf{k} \in V} y_{\mathbf{j}-\mathbf{k}}, \quad \mathbf{j} \in I'.$$

These sums are only computed for $\mathbf{j} \in I'$, as the function is not assumed periodic. (The evaluation of derivatives near the boundary will involve different approximations, which are not discussed in this paper.) The actual evaluation of the double

matrix vector product will involve $|I'|(|V|-1)$ additions, but no multiplications nor divisions. The difference step reduces to the evaluation of

$$|V|^{-1}(A[\mathbf{s}]u)_{\mathbf{j}} = \sum_{\mathbf{k}\in I}(\alpha_{\mathbf{k}}/|V|)u_{\mathbf{j}-\mathbf{k}}, \quad \mathbf{j}\in I''.$$

It is assumed that the coefficients $\alpha_{\mathbf{k}}/|V|$ have been precomputed. The differences are only evaluated on the set I'', which guarantees that only components of $u_{\mathbf{k}}$ with $\mathbf{k}\in I'$ are used. An essential observation is that the $\alpha_{\mathbf{k}}$ are only zero for a small subset of indices which shall be denoted by S. Thus, the difference step requires $|I''|(|S|-1)$ additions and $|I''|\cdot|S|$ multiplications. Except for the trivial precomputation step, mentioned above, no divisions are required.

The total amount of operations for the formation of $A[\mathbf{s}]M_V y$ is the sum of these two expressions. For very large data sets, the number of data in the vicinity of the boundary will be small when compared with the number in the interior of the domain I. Consequently, I' and I'' are asymptotically of the same size as I, and the total number of operations required to form $A[\mathbf{s}]M_V$ becomes $|I|\cdot(|V|+|S|-2)$ additions and $|I|\cdot|S|$ multiplications. (In the sequel, only asymptotic estimates will be derived.) Furthermore, as discussed in the previous section, the size of V is related to that of I by a relationship of the form $|V|=c|I|^{\nu}$, where $0<\nu<1$. Finally, on modern computers, floating point additions and multiplications require about the same number of operations, and so the total floating point operation count is asymptotically $|I|(c|I|^{\nu}+2|S|-2)=O(|I|^{\nu+1})$.

An alternative way to proceed is to compute the action of $A[\mathbf{s}]M_V$ on some given data $\mathbf{y}$. Here, one turns to the application of the fast Fourier transform (FFT), where the complexity will be $O(|I|\log_2(I))$. Thus, there is a crossover point defined by an equation of the form

$$|I|^{\nu} = C\log_2(|I|)$$

where C is a constant depending on the choice of V and on how the FFTs are performed. Consequently, for larger I, the FFT approach will be more efficient. For this "indirect" method, a d-dimensional FFT will be required. An efficient implementation for a general d can be found in [9].

For the evaluation of a gradient, or Hessian, there will be several, say t, independent derivatives to be computed. If these derivatives are again required over the whole domain, this reduces to the evaluation of a matrix vector product of the form

$$(A_1[\mathbf{s}],\ldots,A_t[\mathbf{s}])^T M_V\mathbf{y}.$$

If the averaging is performed first on the scalar data, this will require (asymptotically) $|I|\cdot(|V|-1+t(|S|-1))$ additions and $t|I|\cdot|S|$ multiplications. On the other hand, if the difference operators were applied first and the results were all averaged independently, then the number of operations would be t times what is needed for the computation of one such partial derivative. Consequently, differencing first will be much more expensive than averaging first.

On the other hand, if a reduction operation takes place (such as occurs in the evaluation of a divergence), the differencing should be performed first; i.e., evaluate

the formula as

$$M_V(A_1[\mathbf{s}], \ldots, A_t[\mathbf{s}]) \begin{bmatrix} y_1 \\ \vdots \\ y_t \end{bmatrix}.$$

Compared with gradient-like operations, an additional summation must be performed which involves $(t-1)|I|$ additions. Consequently, one requires a total of $|I| \cdot (|V| + t|S| - 2)$ additions and $t|I| \cdot |S|$ multiplications.

When either $|S|$ or $|V|$ is large, FFT's, based on the discrete Fourier transform, can be applied to the stencil of the averaging. The stencil of the averaging, or multi-point differencing method, can be represented by

$$M_V^T A[\mathbf{s}]^T e_1.$$

This stencil can be precomputed with little work. Because the supports for the differentiation and the averaging do not overlap, the number of nonzero elements of the stencil is $|S| \cdot |V|$. Consequently, the application of the stencil requires $|S| \cdot |V| - 1$ additions and $|S| \cdot |V|$ multiplications. Having computed the stencil, we can apply it to determine the derivative at a variety of points, and in particular, all points of I''. Note, however, that for this the number of additions and multiplications required would be (asymptotically) $|I|(|S| \cdot |V| - 1)$ and $|I| \cdot |S| \cdot |V|$. This is much more than the separate application of $A[\mathbf{s}]$ and M_V uses. However, if the values of several derivatives at only one point are required, then the number of additions and multiplications reduces to $t|S| \cdot |V| - 1$ and $t|S| \cdot |V|$. A similar estimate holds if a set of derivatives are only required at a limited number of points

As a conclusion, when implementing algorithms for the computation of derivatives, one should carefully consider the characteristics of the problem in order to choose an efficient method. In summary, if derivatives on the full range of points in the domain are to be computed, one fares best when the averaging and the differencing are done separately. For the computation of a set of different derivatives of the same scalar function, one should consider averaging before differentiating. If the result required is a linear combination of a set of derivatives of multiple functions, one should do the differentiation first and the averaging second. For large and complicated derivatives and large averages, multidimensional FFTs should be used for best performance.

4. Application and exemplification

4.1. **Applications.** The above analysis has been performed under the assumption that the errors are generated by iid Gaussian random variables, as this yields a framework in which explicit results about the trade-off between convergence and stability can be constructed. Nevertheless, on various grounds, it can be argued that the results apply more generally to a wide variety of error situations. For example, if the errors are positively correlated, then the correlation process will tend to reduce the amount of scatter in the errors locally, with the potential stability associated with the differentiation of the corresponding data improved. An illustration of this point can be found in terms of the behaviour of correlation functions in spatial statistics [5, Ch. 2].

Any process involving a spatial aspect leads naturally to the need to evaluate partial derivatives. An essential feature of such applications, of which fluid dynamics is a good example, is that various combinations of partial derivatives, which define

appropriate physical concepts, are required rather than single partial derivative. In fluid dynamics, a classical example is the (two-dimensional) vorticity

$$\omega(x,y,t) = \left(\frac{\partial v_2}{\partial x} - \frac{\partial v_1}{\partial y}\right)(x,y,t),$$

where the (two-dimensional) velocity vector $\mathbf{v}$ has the components $(v_1,\ v_2)$.

The velocity data $d_{l,(i,j)} = v_l(x_i, y_j) + \epsilon_{i,j},\ l = 1,\ 2$ is often the end product of a preprocessing, using conditional averaging [6], of the time series records of the measurements of the velocity components of the fluid flow at a grid of points (x_i, y_j) by laser doppler anemometers. In the study of forced convection and vortex shedding [4], estimates of the vorticity are required for input into the Howe theory of aeroacoustics.

The estimation of the distribution and size of thermal and magnetic sources, as well as fluid input sources into confined aquifiers [3], reduces to the evaluation of multi-dimensional Sturm-Liouville (Laplacian) operators on observational data.

In many practical applications, the components of some underlying stress tensor must be estimated. Because the continuity and momentum conditions only involve three equations, whereas a symmetric stress tensor has six components, it is more natural to first compute the displacements or velocities and then evaluate the components of the required stress tensor, than to compute the components of the stress tensor directly, since that would involve the formulation of additional consistency conditions in order to obtain six equations for the six unknowns.

4.2. **Computational example.** The proposed method is now demonstrated on simulated data. All the computations were done with MATLAB. The chosen function was

$$f(x,y,z) = \exp(-x^2 - y^2 - z^2)$$

and the simulated data had the form

$$d_{i,j,k} = f(ih, jh, kh) + \epsilon_{i,j,k}, \quad i,j,k = -n, \ldots, n,$$

where $h = 2/(n-1)$ and $\epsilon_{i,j,k}$ are independent normally distributed random variables with expectation 0 and standard deviation $\sigma = 5/1000$.

From the simulated data, the Laplacian Δf has been computed using averaging and the midpoint rule as proposed in the previous sections. For averaging, the set $V = \{-r, \ldots, r\}^3$ was used and the standard 9-point stencil was chosen for the discrete Laplacian $A[\mathbf{s}]$ where $\mathbf{s} = (s,s,s)$ with $s = 2r+1$. Now one gets from equation (19)

$$\mu_2(V) = r(r+1) = \frac{1}{4}(m^2 - 1).$$

The Laplacian and the Hessian of the Laplacian of f can be computed, and one obtains (with some assistance from the MAPLE package) the bound

$$\|H_{\Delta f}\| \leq 20.$$

The constant $C(f)$ is obtained from a Taylor expansion of the truncation error of the stencil, and one finds that

$$C(f) = \frac{1}{36}\left\|\frac{\partial^4 f}{\partial x^4} + \frac{\partial^4 f}{\partial y^4} + \frac{\partial^4 f}{\partial z^4}\right\|_\infty \leq 1.$$

TABLE 1. Root mean squared error for the 3D Laplacian.

n	9	17	33	65	129	257
s	2	2	4	6	9	15
error	0.032	0.020	0.013	0.0097	0.0083	0.0059

The approximation error bound from Proposition 3 thereby becomes

$$|\bar{\dot{f}}_{\mathbf{j}} - f'_{\mathbf{j}}| \leq 3h^2 s^2 + 2.5h^2(s^2 - 1).$$

The variance bound from Proposition 4 is

$$\operatorname{var}(\zeta_{\mathbf{j}}) \leq (2\pi)^4 \sigma^2 h^{-7} s^{-10}.$$

This bound was derived for a very general case, but here one can actually compute the variance explicitly. One can see that $\|A[\mathbf{s}]\mathbf{e}\|^2 = 42s^{-4}h^{-4}$, and so one obtains from Lemma 1:

$$\operatorname{var} \zeta_{\mathbf{j}} = 42\sigma^2 s^{-7} h^{-4}.$$

Though the choice of the parameter s is crucial, its estimation is outside the scope of this work. For example, in cross validation s is estimated as the minimizer of an estimate of the mean squared error [12]. If such an estimate is unbiased, its expected value is bounded by

$$30.2s^4h^4 + 42\sigma^2 s^{-7} h^{-4},$$

as can be seen from the previous approximation estimate and the variance formula. For demonstration purposes, the s chosen here is the minimizer of this function of s. This gives

$$s = \lceil 1.1 h^{-8/11} \sigma^{2/11} \rceil,$$

where $\lceil \cdot \rceil$ denotes the next larger integer.

In Table 1, the root mean square errors (scaled with $\Delta(f)(0)$) are displayed. One can see clear convergence, and closer inspection reveals that the convergence rate is $O(h^{6/11})$, as predicted by the theory. In particular, Table 1 shows that convergence occurs as h tends to zero. If a similar method is applied to one-dimensional data to compute second derivatives, it can be shown that the convergence rate is $O(h^{2/9})$. Thus the convergence rate in h for the three-dimensional data is 2.5 times higher than for the one-dimensional data. This demonstrates the effectiveness of the volume averaging compared with lower dimensional averaging in increasing the convergence, and, thus, the beneficial effects of dimensionality. Note that there is a slight trade-off between convergence in σ and h.

REFERENCES

[1] R.S. Anderssen, F. de Hoog, and M. Hegland, *A stable finite difference ansatz for higher order differentiation of non-exact data*, Mathematics Research Report MRR96-023, ANU, School of Mathematical Sciences, 1996, `ftp://nimbus.anu.edu.au:/pub/Hegland/andhh96.ps.gz`.

[2] R.S. Anderssen and F.R. de Hoog, *Finite difference methods for the numerical differentiation of non-exact data*, Computing **33** (1984), 259–267. MR **86e:**65032

[3] S. S. Choi and R. S. Anderssen, *Determination of the transmissitivity zonata using a linear function strategy*, Inverse Problems **7** (1991), 831–851.

[4] R. R. Clements, *An inviscid model of two-dimensional vortex shedding*, J. Fluid Mech. **75** (1976), 209–231.

[5] N. Cressie, *Statistics for spatial data*, J. Wiley and Sons, 1991. MR **92k:**62166

[6] M. E. Davies, *A comparison of the wake structure of a stationary and oscillating bluff body, using a conditional averaging technique*, J. Fluid Mech. **75** (1976), 209–231.
[7] D.J. Finney, *Statistics for mathematicians: an introduction*, Oliver and Boyd, Edinburgh, 1968.
[8] T.J. Hastie and R.J. Tibshirani, *Generalized additive models*, Monographs on statistics and applied probability, vol. 43, Chapman and Hall, 1990. MR **92e:**62177
[9] Markus Hegland, *An implementation of multiple and multi-variate Fourier transforms on vector processors*, SIAM J. Sci. Comp. **16** (1995), no. 2, 271–288. MR **96b:**65129
[10] H.G. Müller, *Nonparametric regression analysis of longitudinal data*, Lecture Notes in Statistics, vol. 46, Springer, 1987. MR **89i:**62003
[11] H. Niederreiter, *Random number generation and quasi-Monte Carlo methods*, SIAM, 1992. MR **93h:**65008
[12] M.P. Wand and M.C. Jones, *Kernel smoothing*, Monographs on statistics and applied probability, vol. 60, Chapman and Hall, 1995. MR **96k:**62119

CSIRO Mathematical and Information Sciences, GPO Box 1965, Canberra ACT 2601, Australia
E-mail address: Bob.Anderssen@cmis.csiro.au

Computer Sciences Laboratory, Australian National University, Canberra ACT 0200, Australia
E-mail address: Markus.Hegland@anu.edu.au