Stencils with isotropic discretisation error for differential operators

Michael Patra Mikko Karttunen
Biophysics and Statistical Mechanics Group
Laboratory for Computational Engineering
Helsinki University of Technology
P. O. Box 9203, 02015 HUT, Finland

We derive stencils for differential operators for which the discretisation error becomes isotropic. We treat the Laplacian, the Bilaplacian (=biharmonic operator) and the gradient of a Laplacian both in two and three dimensions. For three dimensions $\mathcal{O}(h^2)$ results are given while for two dimensions both $\mathcal{O}(h^2)$ and $\mathcal{O}(h^4)$ results are presented. The results are also available in electronic form as a Mathematica file. It is shown that the extra computational cost of an isotropic stencil usually is less than 20 %.

I. INTRODUCTION

The first step in discretising a partial differential equation on a regular grid is approximating the differential operator $\mathcal{D}$ by a stencil $\mathcal{S}$. The latter can be written as a matrix S such that

$$\mathcal{D}f(x,y) \approx \sum_{ij}^{-r\ldots r} S_{ij} f(x+ih, y+jh) \; , \tag{1.1}$$

where h is the grid spacing of the system, and $2r+1$ is the extension of the stencil. A stencil with $r = 1$ is called a compact stencil. (For the ease of notation, we restrict ourselves to the two-dimensional case for the moment.) The "quality" of the stencil S is measured by how quickly the discretisation error disappears as $h \to 0$, and the stencil is said to be of p-th order if

$$\mathcal{D}f(x,y) = \sum_{ij} S_{ij} f(x+ih, y+jh) + \mathcal{O}(h^p) \; . \tag{1.2}$$

Differential equations appear mainly in two different contexts, namely

$$\mathcal{D}f(x,y) = \rho(x,y) \; , \tag{1.3a}$$

$$\frac{d}{dt} f(x,y) = \mathcal{D}f(x,y) \; . \tag{1.3b}$$

In the first equation, the task is to find $f(x,y)$ given the source term (also called the force term) $\rho(x,y)$. Technically speaking, it is a boundary value problem for the unknown function $f(x,y)$. Most of the published work on stencils is geared toward this application. For this, it is beneficial if the stencil is diagonally dominant — which is only possible if it is also compact. Stencils for this kind of problem can be improved significantly when not only $\rho(x,y)$ is known but also its derivatives; the functional form of $\rho(x,y)$ may be known or its derivatives can be computed by application of a suitable stencil. Such stencils are referred to as higher-order compact stencils. We will return to these points in Sec. VI..

In this paper we focus on the second problem defined by Eq. (1.3b). Here the task is to first compute the value of the differential operator given only $f(x,y)$, then the function $f(x,y)$ is integrated in time. The problem thus is an initial value problem (in time) for the unknown function $f(x,y,t)$. The selection process of a suitable stencil is two-fold. First, it is immediately obvious that computational efficiency is essential as Eq. (1.3b) has to be iterated a large number of times. Traditionally it is assumed that the computational cost is proportional to the size $(2r+1)^2$ of the stencil or the number of nonzero elements n_1, suggesting that as small a stencil as possible should be chosen. However, as we will demonstrate in Sec. V., the computational cost increases surprisingly slowly with r and n_1.

The second criterion in selecting a stencil does not always seem to have received appropriate attention in the past. Frequently a stencil is discussed only in terms of its order p — but for given order p there are many different stencils with different discretisation errors $\tilde{\mathcal{D}}$. The proper choice of a stencil with a specific discretisation error can be essential.

Let us illustrate the above argument by a simple example. The two dimensional Laplacian $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is frequently discretised as

$$\Delta f(x,y) = \frac{f(x+h,y)+f(x-h,y)+f(x,y+h)+f(x,y-h)-4f(x,y)}{h^2} + \mathcal{O}(h^2)\ , \tag{1.4}$$

i. e., simply by adding the discretisations for the second derivatives in x and y-direction, respectively. Using the test function $f_1(x,y) = x^4$ gives

$$\mathcal{S} f_1(x,y) = \mathcal{D} f_1(x,y) + 2h^2\ , \tag{1.5}$$

i. e., independent of position, the stencil overestimates the Laplacian by $2h^2$. Upon integration of Eq. (1.3b), $f_1(x,y)$ would thus grow a bit faster than it should.

Let us now rotate the coordinate system by 45 degrees, f_1 thus becomes $f_2(x,y) = (x+y)^4/4$. This gives

$$\mathcal{S} f_2(x,y) = \mathcal{D} f_2(x,y) + h^2\ . \tag{1.6}$$

Also, $f_2(x,y)$ grows faster than it should, but even though f_1 and f_2 are identical up to the arbitrary choice of coordinate system, the excess rate of growth is only half as large as for f_1.

Since having a discretisation error is unavoidable, the numerically computed growth rate will always be different from the exact one. This error can be controlled by a proper choice of grid size h and usually poses no significant problem. The differences in the growth rate in different directions are much more of a problem since small differences introduced in $f(x,y)$ are amplified by the iteration process. The system would thus grow preferably along the coordinate axes. This error (see Fig. 1 for an illustration) is much more problematic since it can change the quantitative behaviour of the system.

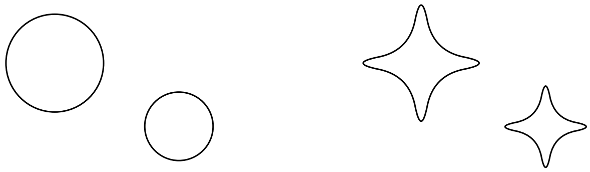


FIG. 1. Starting from homogeneous initial conditions with only two point-like well-separated perturbations, the correct growth process would yield growing spheres (left). If growth is faster along the coordinate axes, the spheres would turn into star-shaped structures (right). This is only an illustration, and in reality the deviation caused by a stencil with anisotropic discretisation error may be be much smaller than shown in this figure.

The only way to circumvent this problem is to use stencils that have an isotropic discretisation error, i. e., the result must not depend on the choice of the coordinate system. Such stencils are presented in this paper. Growth processes are frequently modelled in many areas of physics, chemistry and other natural sciences. The dynamics of these processes is typically described by (coupled) equations of type Eq. (1.3b). Specific examples are reaction-diffusion models (see, e. g., Ref. [1]) and the so-called phase field models [2], in which one identifies the relevant quantities, e. g., conserved, nonconserved and hydrodynamic variables, and uses general symmetry principles to write down the dynamic equations.

Typically, Laplacians, Bilaplacians and the gradients of a Laplacians appear in the resulting equations of motion. For this reason, we concentrate on these three operators in this paper. In two dimensions, we derive $\mathcal{O}(h^2)$ and $\mathcal{O}(h^4)$ stencils, while for three dimensions we restrict ourselves to $\mathcal{O}(h^2)$ stencils. We first derive the general stencils and then identify those ones for which the discretisation error is isotropic.

II. DEFINITIONS

In this paper we consider the Laplacian, the Bilaplacian (also known as biharmonic operator) and the gradient of the Laplacian. For the latter it is sufficient to treat only the x-component, as Eq. (1.3b) has to be integrated in components anyhow. The three operators and our notation for them is as follows,

$$\begin{aligned}
\Delta &= \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} , \\
\Delta^2 &= \left[\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right]^2 , \\
\frac{\partial}{\partial x}\Delta &= \frac{\partial^3}{\partial x^3} + \frac{\partial^3}{\partial x \partial y^2} + \frac{\partial^3}{\partial x \partial z^2} .
\end{aligned}$$

The two-dimensional versions of the operators follow by omitting all derivatives with respect to z.

We present the stencils $\mathcal{S}$ by giving the coefficients of the matrix S. S_{ijk} gives the prefactor of the stencil for the term $f(x+ih, y+jh, z+kh)$. The coefficients are displayed in the order

$$\begin{bmatrix} S_{1,1,\text{-}1} & S_{0,1,\text{-}1} & S_{\text{-}1,1,\text{-}1} \\ S_{1,0,\text{-}1} & S_{0,0,\text{-}1} & S_{\text{-}1,0,\text{-}1} \\ S_{1,\text{-}1,\text{-}1} & S_{0,\text{-}1,\text{-}1} & S_{\text{-}1,\text{-}1,\text{-}1} \end{bmatrix} \begin{bmatrix} S_{1,1,\text{-}0} & S_{0,1,\text{-}0} & S_{\text{-}1,1,\text{-}0} \\ S_{1,0,\text{-}0} & S_{0,0,\text{-}0} & S_{\text{-}1,0,\text{-}0} \\ S_{1,\text{-}1,\text{-}0} & S_{0,\text{-}1,\text{-}0} & S_{\text{-}1,\text{-}1,\text{-}0} \end{bmatrix} \begin{bmatrix} S_{1,1,-1} & S_{0,1,-1} & S_{\text{-}1,1,-1} \\ S_{1,0,-1} & S_{0,0,-1} & S_{\text{-}1,0,-1} \\ S_{1,\text{-}1,-1} & S_{0,\text{-}1,-1} & S_{\text{-}1,\text{-}1,-1} \end{bmatrix} . \tag{2.2}$$

In other words, each two-dimensional matrix corresponds to one fixed value for z, and x is larger on the left than on the right. (These definitions are only relevant when discussing the gradient of the Laplacian since all other stencils are completely symmetric.)

Each "general" stencil $\mathcal{S}$ is defined by a set of conditions for its coefficients which will turn out to be a set of linear equations. "Specific" stencils can be retrieved by setting certain coefficients to zero. One aim can be to minimise the number of nonzero elements n_1. Elements in the interior of the stencil might be accessed by the cpu even if they have the value zero. For this reason, we also introduce the number n_2, denoting the number of elements that are either nonzero or are surrounded by nonzero elements.

Apart from the linear equations for the general stencils, in the following we will also give the coefficients for specific stencils with minimal n_1 or n_2. To aid the reader, a small graphic is displayed showing that stencil. Nonzero elements are indicated by a full square ■. If a zero is lying between two nonzero elements (and hence is included in the count n_2) it is denoted by □. Other zeros are marked by a small dot ·.

III. METHOD

A stencil $\mathcal{S}$ is an $\mathcal{O}(h^n)$ approximation for a differential operator $\mathcal{D}$ of order p if and only if it gives the correct result for any polynomial $P_q(x,y)$ of order $q = p + n - 1$,

$$\sum_{ij} S_{ij} P_q(x+ih, y+jh) = \mathcal{D} P_q(x,y) \quad \text{with} \quad P_q(x,y) = \sum_{i,j=0\ldots q}^{i+j\le q} a_{ij} x^i y^j \ . \tag{3.1}$$

(For convenience the equations are written down only for the two-dimensional case. The three-dimensional case is equivalent.) Since $\mathcal{D}P_q(x,y)$ is easily computed analytically, Eq. (3.1) is sufficient to define all valid stencils S. The amount of work is reduced significantly if S is decomposed into m independent elements connected by symmetry. For example, the Laplacian is invariant under the transformations $x \leftrightarrow -x$, $y \leftrightarrow -y$ and $x \leftrightarrow y$, such that S contains only $m = 3$ independent elements,

$$S = \begin{bmatrix} c_1 & c_2 & c_1 \\ c_2 & c_3 & c_2 \\ c_1 & c_2 & c_1 \end{bmatrix} . \tag{3.2}$$

The stencils derived in this paper contain between 3 and 12 independent elements. Inserting Eq. (3.2) into Eq. (3.1) and demanding the equality of left-hand and right-hand sides for all possible values of $\{a_{ij}\}$, x, y and h gives all sufficient and necessary conditions among the coefficients $c_1, \ldots, c_m$.

Unfortunately, current symbolic algebra software does not seem to be able to handle a direct automatic solution of Eq. (3.1). This problem appears to be related to the task of finding solutions for $\{c_i\}$ that are valid for arbitrary coefficients $\{a_{ij}\}$, whereas symbolic algebra packages are optimised to find solutions for $\{c_i\}$ as a function of $\{a_{ij}\}$.

To circumvent this problem, we break the polynomial $\mathcal{P}$ into its basis functions $x^i y^j$ and replace the single equation Eq. (3.1) by a set of equations — one equation for each basis function. For small stencils, we succeeded in solving this set of equations automatically but for larger stencils the time and memory requirements were prohibitive.

The advisable way of solving this set of equations is as follows. Some of the equations are identically true, others depend on x and/or y while the remaining equations depend only on h and $\{c_i\}$. One of the latter equations is chosen and is solved for c_1. (It might be necessary to solve first for some other coefficient than c_1, but for the ease of notation we assume otherwise.) The solution gives $c_1 = c_1(c_2, \ldots, c_m)$. This expression for c_1 is inserted into all equations and this process is repeated to yield $c_2 = c_2(c_3, \ldots, c_m)$, $c_3 = c_3(c_4, \ldots, c_m)$ and so forth, until after finding and inserting the solution for c_k, $k \le m$, all equations become identically true. Backsubstitution then gives $c_i = (c_{k+1}, \ldots, c_m)$, $i = 1, \ldots, k$. If $k = m$, the stencil is determined uniquely. Otherwise, the freedom to pick $m - k$ coefficients arbitrarily still exists.

The procedure to pick a suitable equation can in principle be automatised, but since for the stencils presented in this paper no more than 10 iterations are needed, we refrained from doing so, and picked the equations by hand. The solution, insertion and simplification were done using the symbolic algebra package Mathematica version 3. (We also tried version 4 but for this particular purpose it seemed to be slower.)

The procedure presented above allows a systematic derivation of the conditions among the coefficients $c_1, \ldots, c_m$ describing a general stencil. Usually $k < m$ such that there is freedom to set some of the coefficients to specific values (e. g., to 0) to yield specific stencil with desirable properties. We are not aware of a similar systematic approach to find stencils with, e. g., minimal n_2.

Equation (3.1) defines only the order of the discretisation error but does not impose any conditions on the precise value of the error. If the truncation error in order n is required to be proportional to the p'-th order differential operator $\tilde{\mathcal{D}}$, then Eq. (3.1) is easily extended to

$$\sum_{ij} S_{ij} P_{q+n'}(x + ih, y + jh) = \mathcal{D} P_{q+n'}(x, y) - c_{m+1} h^n \tilde{\mathcal{D}} P_{q+n'}(x, y) + \mathcal{O}(h^{n+p'}) \ , \quad (3.3)$$

and the procedure described above is applied again. For the Laplacian and the Bilaplacian, it is obvious that the isotropic discretisation error should be proportional to an appropriate power of the Laplacian. Equation (3.3) is thus sufficient for treating the Laplacian and the Bilaplacian.

For the gradient of a Laplacian, the precise form of the "isotropic discretisation error" is not obvious since the gradient itself is an anisotropic operator. Now, the important criterion is rather that the result becomes independent of the choice of the coordinate system. Equation (3.1) is easily adapted also for this case. The polynomial $P_q(x, y)$ is rotated by some angle ϕ, resulting in terms with $\sin\phi$ and $\cos\phi$ in the generated set of equations. The prefactors of those terms have to vanish, giving additional conditions for the coefficients $c_1, \ldots, c_m$ of the stencil. In principle, this approach also works for analysing the Laplacian and the Bilaplacian, but since it is a bit more cumbersome we only used it where needed.

Symbolic mathematics software seems to have been used only very recently to derive stencils for differential operators. To the best of our knowledge, the idea was first used by Gupta and co-workers [3] but their approach is somewhat different from ours. They start with two sets of equations. The first set equates, in our notation, the values of the function $f(x+ih, y+jh)$ at all points of the stencil to the values of the polynomial $P_q(x+ih, y+jh)$. The second set relates the coefficients a_{ij} of the polynomial to the differential operator and certain of its higher derivatives. Solving the complete set of equations gives the coefficients of the stencil, and an expression for the discretisation error in terms of a_{ij} and the higher derivatives of the differential operator. In this approach, one needs to decide a priori which coefficients of the stencil should be nonzero, and their precise choice determines the discretisation error of the stencil. Unfortunately, there is no easy relation between the choice of nonzero coefficients and the properties of the stencil making this method feasible for small (=compact) stencils only, but less so for more complicated problems.

IV. RESULTS

Next, we define the form stencils. We then give the necessary and sufficient conditions among the coefficients that all stencils have to fulfil. Such a stencil has the desired order of the discretisation error but no further conditions on its discretisation error are imposed. This is followed by the conditions for stencils with isotropic discretisation error. Finally, we present a few specific stencils that have "useful" properties. Whenever we managed to find a previous publication of such a stencil, we have provided the references. All stencils are available as Mathematica file from our website http://www.softsimu.org/downloads.shtml. This file contains not only the stencils but also routines that will automatically check that the stencils indeed fulfil Eqs. (3.1) and (3.3), respectively.

A. $\mathcal{O}(h^2)$ stencils for the two-dimensional Laplacian

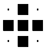
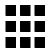
$$\Delta f(x,y) = \frac{1}{h^2} \begin{bmatrix} c_1 & c_2 & c_1 \\ c_2 & c_3 & c_2 \\ c_1 & c_2 & c_1 \end{bmatrix} + \mathcal{O}(h^2) \; . \tag{4.1}$$

Arbitrary stencils are given by

$$c_2 = 1 - 2\,c_1 \; , \qquad c_3 = 4\,c_1 - 4 \; . \tag{4.2}$$

There is only a single isotropic stencil. It has truncation error $(h^2/12)\,\Delta^2 f(x,y) + \mathcal{O}(h^4)$.

c_1	c_2	c_3	n_1	n_2		
0	1	-4	5	5	anisotropic	see any text book
$\frac{1}{6}$	$\frac{2}{3}$	$-\frac{10}{3}$	9	9	isotropic	known under the name "Mehrstellen"

Shapes of these stencils from top to bottom:  

B. $\mathcal{O}(h^4)$ stencils for the two-dimensional Laplacian

$$\Delta f(x,y) = \frac{1}{h^2} \begin{bmatrix} c_1 & c_2 & c_3 & c_2 & c_1 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_3 & c_5 & c_6 & c_5 & c_3 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_1 & c_2 & c_3 & c_2 & c_1 \end{bmatrix} + \mathcal{O}(h^4) \; . \tag{4.3}$$

Arbitrary stencils are given by

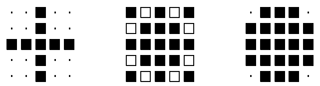
$$c_3 = -\frac{1}{12} - 2\,c_1 - 2\,c_2\ , \qquad c_4 = -16\,c_1 - 8\,c_2\ ,$$
$$c_5 = \frac{4}{3} + 32\,c_1 + 14\,c_2\ , \qquad c_6 = -5 - 60\,c_1 - 24\,c_2\ . \tag{4.4}$$

Isotropic stencils are given by

$$c_2 = -\frac{1}{30} - 4\,c_1\ , \qquad c_3 = 6\,c_1 - \frac{1}{60}\ , \qquad c_4 = \frac{4}{15} + 16\,c_1\ ,$$
$$c_5 = \frac{13}{15} - 24\,c_1\ , \qquad c_6 = 36\,c_1 - \frac{21}{5}\ . \tag{4.5}$$

For all isotropic stencils, the truncation error is $(h^2/90)\,\Delta^3 f(x,y) + \mathcal{O}(h^6)$.

c_1	c_2	c_3	c_4	c_5	c_6	n_1	n_2		
0	0	$-\frac{1}{12}$	0	$\frac{4}{3}$	-5	9	9	anisotropic	Ref. [4]
$-\frac{1}{120}$	0	$-\frac{1}{15}$	$\frac{2}{15}$	$\frac{16}{15}$	$-\frac{9}{2}$	17	25	isotropic	
0	$-\frac{1}{30}$	$-\frac{1}{60}$	$\frac{4}{15}$	$\frac{13}{15}$	$-\frac{21}{5}$	21	21	isotropic	

Shapes of these stencils from top to bottom: 

C. $\mathcal{O}(h^2)$ stencils for the three-dimensional Laplacian

$$\Delta f(x,y,z) = \frac{1}{h^2}\begin{bmatrix} c_1 & c_2 & c_1 \\ c_2 & c_3 & c_2 \\ c_1 & c_2 & c_1 \end{bmatrix}\begin{bmatrix} c_2 & c_3 & c_2 \\ c_3 & c_4 & c_3 \\ c_2 & c_3 & c_2 \end{bmatrix}\begin{bmatrix} c_1 & c_2 & c_1 \\ c_2 & c_3 & c_2 \\ c_1 & c_2 & c_1 \end{bmatrix} + \mathcal{O}(h^2)\ . \tag{4.6}$$

Arbitrary stencils are given by

$$c_3 = 1 - 4\,c_1 - 4\,c_2\ , \qquad c_4 = 16\,c_1 + 12\,c_2 - 6\ . \tag{4.7}$$

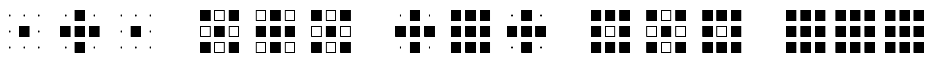
Isotropic stencils are given by

$$c_2 = \frac{1}{6} - 2\,c_1\ , \qquad c_3 = \frac{1}{3} + 4\,c_1\ , \qquad c_4 = -4 - 8\,c_1\ . \tag{4.8}$$

For all isotropic stencils, the truncation error is $(h^2/12)\,\Delta^2 f(x,y,z) + \mathcal{O}(h^4)$.

c_1	c_2	c_3	c_4	n_1	n_2		
0	0	1	-6	7	7	anisotropic	see any text book
$\frac{1}{12}$	0	$\frac{2}{3}$	$-\frac{14}{3}$	15	27	isotropic	Ref. [3]
0	$\frac{1}{6}$	$\frac{1}{3}$	-4	19	19	isotropic	Ref. [5]
$-\frac{1}{12}$	$\frac{1}{3}$	0	$-\frac{10}{3}$	21	27	isotropic	Ref. [3]
$\frac{1}{30}$	$\frac{1}{10}$	$\frac{7}{15}$	$-\frac{64}{15}$	27	27	isotropic	Ref. [6]

Shapes of these stencils from top to bottom:



D. $\mathcal{O}(h^2)$ stencils for the two-dimensional Bilaplacian

$$\Delta^2 f(x,y) = \frac{1}{h^4}\begin{bmatrix} c_1 & c_2 & c_3 & c_2 & c_1 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_3 & c_5 & c_6 & c_5 & c_3 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_1 & c_2 & c_3 & c_2 & c_1 \end{bmatrix} + \mathcal{O}(h^2)\ . \tag{4.9}$$

Arbitrary stencils are given by

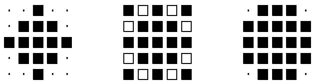
$$\begin{gathered} c_3 = 1 - 2\,c_1 - 2\,c_2\ , \qquad c_4 = 2 - 16\,c_1 - 8\,c_2\ , \\ c_5 = 32\,c_1 + 14\,c_2 - 8\ , \qquad c_6 = 20 - 60\,c_1 - 24\,c_2\ . \end{gathered} \tag{4.10}$$

Isotropic stencils are given by

$$\begin{gathered} c_2 = \frac{1}{3} - 4\,c_1\ , \qquad c_3 = \frac{1}{3} + 6\,c_1\ , \qquad c_4 = 6\,c_1 - \frac{2}{3}\ , \\ c_5 = -\frac{10}{3} - 24\,c_1\ , \qquad c_6 = 12 + 36\,c_1\ . \end{gathered} \tag{4.11}$$

For all isotropic stencils, the truncation error is $(h^2/6)\,\Delta^3 f(x,y) + \mathcal{O}(h^4)$.

c_1	c_2	c_3	c_4	c_5	c_6	n_1	n_2	
0	0	1	2	-8	20	13	13	anisotropic
$\frac{1}{12}$	0	$\frac{5}{6}$	$\frac{2}{3}$	$-\frac{16}{3}$	15	17	25	isotropic
0	$\frac{1}{3}$	$\frac{1}{3}$	$-\frac{2}{3}$	$-\frac{10}{3}$	12	21	21	isotropic

Shapes of these stencils from top to bottom: 

E. $\mathcal{O}(h^4)$ stencils for the two-dimensional Bilaplacian

$$\Delta^2 f(x,y) = \frac{1}{h^4}\begin{bmatrix} c_1 & c_2 & c_3 & c_4 & c_3 & c_2 & c_1 \\ c_2 & c_5 & c_6 & c_7 & c_6 & c_5 & c_2 \\ c_3 & c_6 & c_8 & c_9 & c_8 & c_6 & c_3 \\ c_4 & c_7 & c_9 & c_{10} & c_9 & c_7 & c_4 \\ c_3 & c_6 & c_8 & c_9 & c_8 & c_6 & c_3 \\ c_2 & c_5 & c_6 & c_7 & c_6 & c_5 & c_2 \\ c_1 & c_2 & c_3 & c_4 & c_3 & c_2 & c_1 \end{bmatrix} + \mathcal{O}(h^4)\ . \tag{4.12}$$

Arbitrary stencils are given by

$$\begin{aligned} & c_4 = -\frac{1}{6} - 2\,c_1 - 2\,c_2 - 2\,c_3\ , \qquad c_6 = -\frac{1}{6} - 54\,c_1 - 33\,c_2 - 6\,c_3 - 4\,c_5\ , \\ & c_7 = \frac{7}{3} + 108\,c_1 + 64\,c_2 + 12\,c_3 + 6\,c_5\ , \qquad c_8 = \frac{10}{3} + 351\,c_1 + 192\,c_2 + 30\,c_3 + 16\,c_5\ , \\ & c_9 = -\frac{77}{6} - 594\,c_1 - 318\,c_2 - 50\,c_3 - 24\,c_5\ , \\ & c_{10} = \frac{92}{3} + 976\,c_1 + 512\,c_2 + 80\,c_3 + 36\,c_5\ . \end{aligned} \tag{4.13}$$

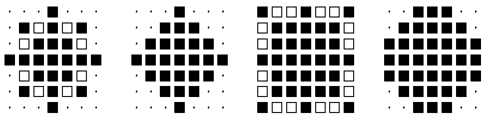
Isotropic stencils are given by

$$c_3 = -\frac{17}{180} - 9\,c_1 - 4\,c_2\ , \qquad c_4 = \frac{1}{45} + 16\,c_1 + 6\,c_2\ , \qquad c_5 = -\frac{29}{180} - 36\,c_1 - 12\,c_2\ ,$$
$$c_6 = \frac{47}{45} + 144\,c_1 + 39\,c_2\ , \quad c_7 = \frac{7}{30} - 216\,c_1 - 56\,c_2\ , \quad c_8 = -\frac{187}{90} - 495\,c_1 - 120\,c_2\ ,$$
$$c_9 = -\frac{191}{45} + 720\,c_1 + 170\,c_2\ , \qquad c_{10} = \frac{779}{45} - 1040\,c_1 - 240\,c_2\ . \tag{4.14}$$

For all isotropic stencils, the truncation error is $(7\,h^4/240)\,\Delta^4 f(x,y) + \mathcal{O}(h^6)$.

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	n_1	n_2
0	0	0	$-\frac{1}{6}$	$-\frac{1}{24}$	0	$\frac{25}{12}$	$\frac{8}{3}$	$-\frac{71}{6}$	$\frac{175}{6}$	21	29
										anisotropic	
0	0	0	$-\frac{1}{6}$	0	$-\frac{1}{6}$	$\frac{7}{3}$	$\frac{10}{3}$	$-\frac{77}{6}$	$\frac{92}{3}$	25	25
										anisotropic	
$-\frac{17}{1620}$	0	0	$-\frac{59}{405}$	$\frac{13}{60}$	$-\frac{7}{15}$	$\frac{5}{2}$	$\frac{187}{60}$	$-\frac{59}{5}$	$\frac{11431}{405}$	33	49
										isotropic	
0	0	$-\frac{17}{180}$	$\frac{1}{45}$	$-\frac{29}{180}$	$\frac{47}{45}$	$\frac{7}{30}$	$-\frac{187}{90}$	$-\frac{191}{45}$	$\frac{779}{45}$	37	37
										isotropic	

Shapes of these stencils from top to bottom:



F. $\mathcal{O}(h^2)$ stencils for the three-dimensional Bilaplacian

$$\Delta^2 f(x,y,z) = \frac{1}{h^4}\begin{bmatrix} c_1 & c_2 & c_3 & c_2 & c_1 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_3 & c_5 & c_8 & c_5 & c_3 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_1 & c_2 & c_3 & c_2 & c_1 \end{bmatrix} \begin{bmatrix} c_2 & c_4 & c_5 & c_4 & c_2 \\ c_4 & c_6 & c_7 & c_6 & c_4 \\ c_5 & c_7 & c_9 & c_7 & c_5 \\ c_4 & c_6 & c_7 & c_6 & c_4 \\ c_2 & c_4 & c_5 & c_4 & c_2 \end{bmatrix} \begin{bmatrix} c_3 & c_5 & c_8 & c_5 & c_3 \\ c_5 & c_7 & c_9 & c_7 & c_5 \\ c_8 & c_9 & c_{10} & c_9 & c_8 \\ c_5 & c_7 & c_9 & c_7 & c_5 \\ c_3 & c_5 & c_8 & c_5 & c_3 \end{bmatrix}$$
$$\begin{bmatrix} c_2 & c_4 & c_5 & c_4 & c_2 \\ c_4 & c_6 & c_7 & c_6 & c_4 \\ c_5 & c_7 & c_9 & c_7 & c_5 \\ c_4 & c_6 & c_7 & c_6 & c_4 \\ c_2 & c_4 & c_5 & c_4 & c_2 \end{bmatrix} \begin{bmatrix} c_1 & c_2 & c_3 & c_2 & c_1 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_3 & c_5 & c_8 & c_5 & c_3 \\ c_2 & c_4 & c_5 & c_4 & c_2 \\ c_1 & c_2 & c_3 & c_2 & c_1 \end{bmatrix} + \mathcal{O}(h^2)\ .(4.15)$$

Arbitrary stencils are given by

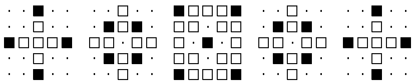
$$c_7 = 2 - 32\,c_1 - 48\,c_2 - 16\,c_3 - 18\,c_4 - 8\,c_5 - 2\,c_6\ ,$$
$$c_8 = 1 - 4\,c_1 - 8\,c_2 - 4\,c_3 - 4\,c_4 - 4\,c_5\ ,$$
$$c_9 = 128\,c_1 + 188\,c_2 + 64\,c_3 + 64\,c_4 + 28\,c_5 + 4\,c_6 - 12\ ,$$
$$c_{10} = 42 - 368\,c_1 - 528\,c_2 - 180\,c_3 - 168\,c_4 - 72\,c_5 - 8\,c_6\ . \tag{4.16}$$

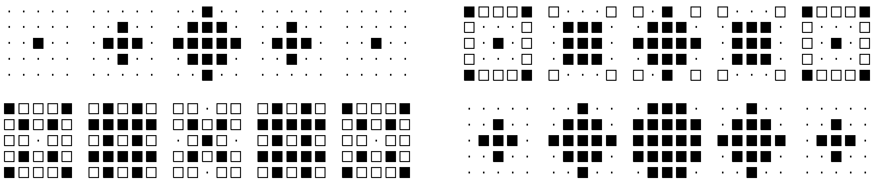
Isotropic stencils are given by

$$\begin{aligned}
&c_1 = \frac{1}{24}(1 - 30\,c_2 - 12\,c_3 - 6\,c_4 - 3\,c_5)\ , \qquad c_6 = 32\,c_2 + 32\,c_3 + 4\,c_4 + 8\,c_5 - \frac{5}{3}\ ,\\
&c_7 = 4 - 72\,c_2 - 64\,c_3 - 18\,c_4 - 20\,c_5\ , \qquad c_8 = \frac{5}{6} - 3\,c_2 - 2\,c_3 - 3\,c_4 - \frac{7}{2}\,c_5\ ,\\
&c_9 = -\frac{40}{3} + 156\,c_2 + 128\,c_3 + 48\,c_4 + 44\,c_5\ ,\\
&c_{10} = 40 - 324\,c_2 - 252\,c_3 - 108\,c_4 - 90\,c_5\ .
\end{aligned} \tag{4.17}$$

For all isotropic stencils, the truncation error is $(h^2/6)\,\Delta^3 f(x,y,z) + \mathcal{O}(h^4)$.

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	n_1	n_2	
0	0	$\frac{1}{4}$	0	0	-1	0	0	0	5	21	67	anisotropic
0	0	0	0	0	0	2	1	-12	42	25	25	anisotropic Ref. [7]
$\frac{1}{24}$	0	0	0	0	$-\frac{5}{3}$	4	$\frac{5}{6}$	$-\frac{40}{3}$	40	41	125	isotropic
$-\frac{1}{36}$	0	0	$\frac{5}{18}$	0	$-\frac{5}{9}$	-1	0	0	10	52	125	isotropic
0	0	0	0	$\frac{1}{3}$	1	$-\frac{8}{3}$	$-\frac{1}{3}$	$\frac{4}{3}$	10	57	57	isotropic

Shapes of these stencils from top to bottom: 



G. $\mathcal{O}(h^2)$ stencils for the gradient of a two-dimensional Laplacian

$$\frac{d}{dx}\Delta f(x,y) = \frac{1}{h^3}\begin{bmatrix} c_1 & c_4 & 0 & -c_4 & -c_1 \\ c_2 & c_5 & 0 & -c_5 & -c_2 \\ c_3 & c_6 & 0 & -c_6 & -c_3 \\ c_2 & c_5 & 0 & -c_5 & -c_2 \\ c_1 & c_4 & 0 & -c_4 & -c_1 \end{bmatrix} + \mathcal{O}(h^2)\ . \tag{4.18}$$

Arbitrary stencils are given by

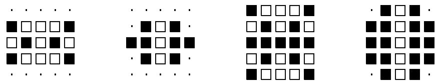
$$c_3 = \frac{1}{2} - 2\,c_1 - 2\,c_2\ , \qquad c_5 = \frac{1}{2} - 8\,c_1 - 2\,c_2 - 4\,c_4\ , \qquad c_6 = -2 + 16\,c_1 + 4\,c_2 + 6\,c_4\ . \tag{4.19}$$

Isotropic stencils are given by

$$\begin{aligned}
c_2 = \frac{1}{6} - 4\,c_1\ , \qquad c_3 = \frac{1}{6} + 6\,c_1\ , \qquad c_4 = \frac{1}{12} - 2\,c_1\ ,\\
c_5 = -\frac{1}{6} + 8\,c_1\ , \qquad c_6 = -\frac{5}{6} - 12\,c_1\ .
\end{aligned} \tag{4.20}$$

For all isotropic stencils, the truncation error is $(h^2/4)\,\frac{d}{dx}\Delta^2 f(x,y)$.

c_1	c_2	c_3	c_4	c_5	c_6	n_1	n_2	
0	$\frac{1}{4}$	0	0	0	-1	6	15	anisotropic
0	0	$\frac{1}{2}$	0	$\frac{1}{2}$	-2	8	12	anisotropic
$\frac{1}{24}$	0	$\frac{5}{12}$	0	$\frac{1}{6}$	$-\frac{4}{3}$	12	25	isotropic
0	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{12}$	$-\frac{1}{6}$	$-\frac{5}{6}$	16	21	isotropic

Shapes of these stencils from top to bottom: 

H. $\mathcal{O}(h^4)$ stencils for the gradient of a two-dimensional Laplacian

$$\frac{d}{dx}\Delta f(x,y) = \frac{1}{h^3}\begin{bmatrix} c_1 & c_5 & c_9 & 0 & -c_9 & -c_5 & -c_1 \\ c_2 & c_6 & c_{10} & 0 & -c_{10} & -c_6 & -c_2 \\ c_3 & c_7 & c_{11} & 0 & -c_{11} & -c_7 & -c_3 \\ c_4 & c_8 & c_{12} & 0 & -c_{12} & -c_8 & -c_4 \\ c_3 & c_7 & c_{11} & 0 & -c_{11} & -c_7 & -c_3 \\ c_2 & c_6 & c_{10} & 0 & -c_{10} & -c_6 & -c_2 \\ c_1 & c_5 & c_9 & 0 & -c_9 & -c_5 & -c_1 \end{bmatrix} + \mathcal{O}(h^4)\ . \tag{4.21}$$

Arbitrary stencils are given by

$$\begin{aligned} c_4 &= -\frac{1}{8} - 2\,c_1 - 2\,c_2 - 2\,c_3\ , \qquad c_7 = -\frac{1}{12} - 36\,c_1 - 16\,c_2 - 4\,c_3 - 9\,c_5 - 4\,c_6\ , \\ c_8 &= \frac{7}{6} + 72\,c1 + 32\,c2 + 8\,c3 + 16\,c5 + 6\,c_6\ , \\ c_{10} &= -\frac{1}{24} - 18\,c_1 - 3\,c_2 - 12\,c_5 - 2\,c_6 - 6\,c_9\ , \\ c_{11} &= \frac{5}{6} + 117\,c_1 + 32\,c_2 + 5\,c_3 + 48\,c_5 + 8\,c_6 + 15\,c_9\ , \\ c_{12} &= -\frac{77}{24} - 198\,c_1 - 58\,c_2 - 10\,c_3 - 72\,c_5 - 12\,c_6 - 20\,c_9\ . \end{aligned} \tag{4.22}$$

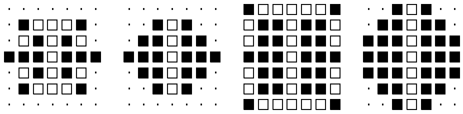
Isotropic stencils are given by

$$\begin{aligned} c_3 &= -\frac{17}{240} - 9\,c_1 - 4\,c_2\ , \qquad c_4 = \frac{1}{60} + 16\,c_1 + 6\,c_2\ , \\ c_6 &= -\frac{29}{360} - 24\,c_1 - 4\,c_2 - 6\,c_5\ , \qquad c_7 = \frac{47}{90} + 96\,c_1 + 16\,c_2 + 15\,c_5\ , \\ c_8 &= \frac{7}{60} - 144\,c_1 - 24\,c_2 - 20\,c_5\ , \qquad c_9 = -\frac{17}{720} - 3\,c_1 - 2\,c_5\ , \\ c_{10} &= \frac{47}{180} + 48\,c_1 + 5\,c_2 + 12\,c_5\ , \qquad c_{11} = -\frac{187}{360} - 165\,c_1 - 20\,c_2 - 30\,c_5\ , \\ c_{12} &= -\frac{191}{180} + 240\,c_1 + 30\,c_2 + 40\,c_5\ . \end{aligned} \tag{4.23}$$

For all isotropic stencils, the truncation error is $(7h^4/120)\frac{d}{dx}\Delta^2 f(x,y) + \mathcal{O}(h^6)$.

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	c_{11}	c_{12}
0	0	0	$-1/8$	0	$-\frac{1}{48}$	0	$\frac{25}{24}$	0	0	$\frac{2}{3}$	$-\frac{71}{24}$
							$n_1 = 14$		$n_2 = 27$	anisotropic	
0	0	0	$-\frac{1}{8}$	0	0	$-\frac{1}{12}$	$\frac{7}{6}$	0	$-\frac{1}{24}$	$\frac{5}{6}$	$-\frac{77}{24}$
							$n_1 = 18$		$n_2 = 23$	anisotropic	
$-\frac{17}{2160}$	0	0	$-\frac{59}{540}$	0	$\frac{13}{120}$	$-\frac{7}{30}$	$\frac{5}{4}$	0	$-\frac{7}{60}$	$\frac{187}{240}$	$-\frac{59}{20}$
							$n_1 = 26$		$n_2 = 49$	isotropic	
0	0	$-\frac{17}{240}$	$\frac{1}{60}$	0	$-\frac{29}{360}$	$\frac{47}{90}$	$\frac{7}{60}$	$-\frac{17}{720}$	$\frac{47}{180}$	$-\frac{187}{360}$	$-\frac{191}{180}$
							$n_1 = 30$		$n_2 = 37$	isotropic	

Shapes of these stencils from top to bottom:



I. $\mathcal{O}(h^2)$ stencils for the gradient of a three-dimensional Laplacian

$$\frac{\partial}{\partial x}\Delta f(x,y,z) = \frac{1}{h^3}\begin{bmatrix} c_1 & c_7 & 0 & -c_7 & -c_1 \\ c_2 & c_8 & 0 & -c_8 & -c_2 \\ c_4 & c_9 & 0 & -c_9 & -c_4 \\ c_2 & c_8 & 0 & -c_8 & -c_2 \\ c_1 & c_7 & 0 & -c_7 & -c_1 \end{bmatrix} \begin{bmatrix} c_2 & c_8 & 0 & -c_8 & -c_2 \\ c_3 & c_{10} & 0 & -c_{10} & -c_3 \\ c_5 & c_{11} & 0 & -c_{11} & -c_5 \\ c_3 & c_{10} & 0 & -c_{10} & -c_3 \\ c_2 & c_8 & 0 & -c_8 & -c_2 \end{bmatrix}$$

$$\begin{bmatrix} c_4 & c_9 & 0 & -c_9 & -c_4 \\ c_5 & c_{11} & 0 & -c_{11} & -c_5 \\ c_6 & c_{12} & 0 & -c_{12} & -c_6 \\ c_5 & c_{11} & 0 & -c_{11} & -c_5 \\ c_4 & c_9 & 0 & -c_9 & -c_4 \end{bmatrix} \begin{bmatrix} c_2 & c_8 & 0 & -c_8 & -c_2 \\ c_3 & c_{10} & 0 & -c_{10} & -c_3 \\ c_5 & c_{11} & 0 & -c_{11} & -c_5 \\ c_3 & c_{10} & 0 & -c_{10} & -c_3 \\ c_2 & c_8 & 0 & -c_8 & -c_2 \end{bmatrix} \begin{bmatrix} c_1 & c_7 & 0 & -c_7 & -c_1 \\ c_2 & c_8 & 0 & -c_8 & -c_2 \\ c_4 & c_9 & 0 & -c_9 & -c_4 \\ c_2 & c_8 & 0 & -c_8 & -c_2 \\ c_1 & c_7 & 0 & -c_7 & -c_1 \end{bmatrix} + \mathcal{O}(h^2) \quad (4.24)$$

Arbitrary stencils are given by

$$\begin{aligned} c_6 &= \frac{1}{2} - 4\,c_1 - 8\,c_2 - 4\,c_3 - 4\,c_4 - 4\,c_5 \ , \\ c_{11} &= \frac{1}{2} - 16\,c_1 - 20\,c_2 - 4\,c_3 - 8\,c_4 - 2\,c_5 - 8\,c_7 - 10\,c_8 - 4\,c_9 - 2\,c_{10} \ , \\ c_{12} &= -3 + 64\,c_1 + 80\,c_2 + 16\,c_3 + 32\,c_4 + 8\,c_5 + 28\,c_7 + 32\,c_8 + 12\,c_9 + 4\,c_{10} \ . \end{aligned} \quad (4.25)$$

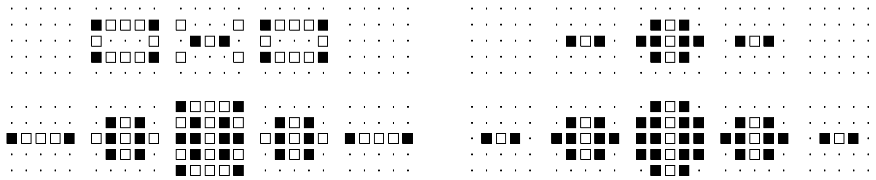
Isotropic stencils are given by

$$\begin{aligned} c_5 &= \frac{1}{6} - 8\,c_1 - 10\,c_2 - 2\,c_3 - 4\,c_4 \ , \qquad c_6 = -\frac{1}{6} + 28\,c_1 + 32\,c_2 + 4\,c_3 + 12\,c_4 \ , \\ c_8 &= \frac{1}{32} - 4\,c_1 - 2\,c_2 - \frac{1}{4}\,c_3 - 2\,c_7 - \frac{1}{8}\,c_{10} \ , \\ c_9 &= \frac{1}{48} + 4\,c_1 + \frac{1}{2}\,c_3 - 2\,c_4 + 2\,c_7 + \frac{1}{4}\,c_{10} \ , \\ c_{11} &= -\frac{11}{48} + 24\,c_1 + 20\,c_2 + \frac{1}{2}\,c_3 + 8\,c_4 + 4\,c_7 - \frac{7}{4}\,c_{10} \ , \\ c_{12} &= -\frac{5}{12} - 80\,c_1 - 64\,c_2 - 2\,c_3 - 24\,c_4 - 12\,c_7 + 3\,c_{10} \ . \end{aligned} \quad (4.26)$$

For all isotropic stencils, the truncation error is $(h^2/4)\frac{d}{dx}\Delta^2 f(x,y) + \mathcal{O}(h^6)$.

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	c_{11}	c_{12}	n_1	n_2	
0	0	$\frac{1}{8}$	0	0	0	0	0	0	0	0	-1	10	23	anisotropic
0	0	0	0	0	$\frac{1}{2}$	0	0	0	0	$\frac{1}{2}$	-3	12	17	anisotropic
0	0	0	$\frac{1}{24}$	0	$\frac{1}{3}$	0	0	0	$\frac{1}{4}$	$-\frac{1}{3}$	$-\frac{2}{3}$	28	57	isotropic
0	0	0	0	$\frac{1}{6}$	$-\frac{1}{6}$	0	0	$\frac{1}{12}$	$\frac{1}{4}$	$-\frac{2}{3}$	$\frac{1}{3}$	36	49	isotropic

Shapes of these stencils from top to bottom:



V. COMPUTATIONAL CONSIDERATIONS

A naive picture of a computer suggests that it has a certain amount of homogeneous memory. In reality, only a small part of the memory, known as cache, is able to supply data to the cpu at a speed that is comparable to the speed at which the cpu can perform simple operations such as additions or multiplications of floating point numbers. (Modern computers frequently have different levels of caches but we will ignore this complication here.) Moving data from the normal memory to the cache can be a rate-limiting process.

Let us assume that the cache is able to hold M floating point numbers. We want to apply a differential operator to a system with linear extension Nh and dimension d ($d = 2, 3$) stored in some matrix A. If $M \geq 2N^d$, then the application of the stencil to the entire system can be done entirely inside the cache. (The factor 2 appears since the result also has to be stored somewhere.) In this case, the time needed is basically proportional to the number of floating point operations, and thus approximately proportional to the number of nonzero elements n_1.

If A no longer entirely fits into the cache, parts of A are loaded into the cache, are then being processed, and then discarded from the cache again. If A is traversed in some sensible order, most of the cache contents are used several times before they are replaced with new data. (This is the reason why arrays in Fortran should have their left-most index vary fastest whereas in C it should be the right-most index. Professional linear algebra packages such as LAPACK [8] use algorithms known as blocked algorithms specially designed to take account of finite cache size.) To process the entire matrix A, each element does not need to be loaded into the cache several times but once only if the condition $M \geq (2r+1)N^{d-1}$ is fulfilled. Assuming that loading data into the cache from normal memory takes much longer than typical arithmetic cpu operations, the computing time becomes independent of the stencil.

In addition to the throughput considerations described above, the question of latencies is also important. The maximal throughput is determined by the hardware and cannot be increased by software means. Latencies can be reduced by issuing prefetch commands for certain data some time before they are actually needed. Modern compilers are able to perform prefetching (unfortunately with varying benefit).

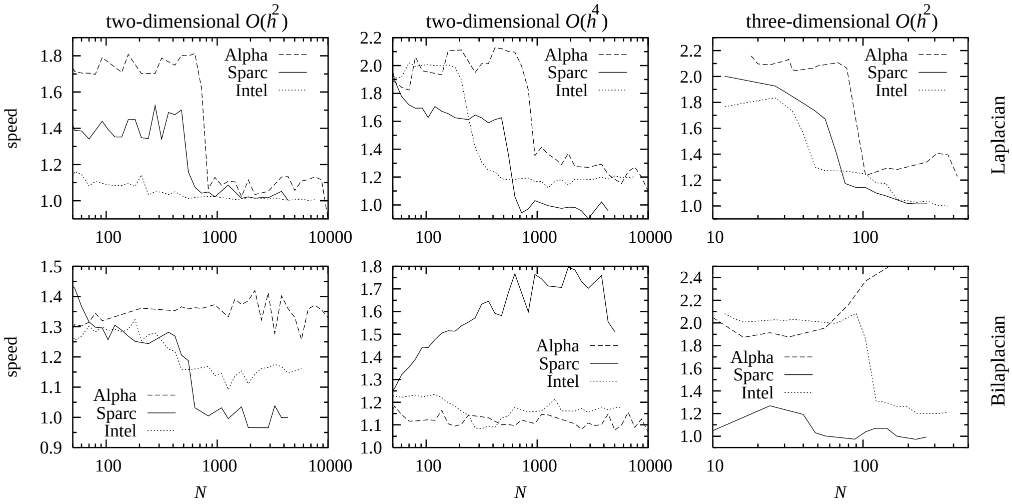


FIG. 2. Ratio of the time needed for the fastest isotropic stencil compared to the time divided by the fastest anisotropic stencil with the same order. A value of "1.6" means that the isotropic stencil needs 60 % longer to compute than the anisotropic stencil of same order. Data for the Laplacian (top) and the Bilaplacian (bottom) is shown for (from left to right) two-dimensional $\mathcal{O}(h^2)$, two-dimensional $\mathcal{O}(h^4)$ and three-dimensional $\mathcal{O}(h^2)$ stencils; see also the labels on the top and on the right.

To check the speed of the stencils derived in this paper, we have implemented the Laplacian and the Bilaplacian (since they are the most commonly used) on three different systems, namely a SparcStation Ultra 60 (using UltraSparc processors at 450 MHz), an AlphaServer ES45 (using EV6.8CB processors at 1 GHz) and an Intel Pentium IV computer (running at 3 GHz). For the first two systems, the program was written in Fortran 90 and compiled with highest optimisation using the native compilers supplied by the vendors. For the Intel system, we used SSE2 intrinsics to increase the speed of the program, and compilation was done using the Intel compilers at standard optimisation. For each system, the speed of the fastest anisotropic and of the fastest isotropic stencil stencil was measured as a function of matrix size N. Periodic boundary conditions were applied, and the application of the stencil was iterated sufficiently often to obtain statistically significant timing data. The ratio of those two speeds quantifies the extra cost of using an isotropic stencil.

The graphs for the Laplacian in Fig. 2 show an effect of finite cache size that agrees very well with the theoretical throughput argument presented above. While the matrix still fits into the cache, the speed of the algorithm depends on the complexity of the stencil. After that, the speed becomes almost independent of the stencil, as re-loading the cache becomes the limiting step. The different cache sizes of the different processors are seen very clearly.

For the Bilaplacian, the situation is a bit different. Some of the curves in Fig. 2 behave as expected, while for others the compiler apparently behaves strangely when generating code for certain stencils. It should be noted that the "noise" in all figures is not statistical sampling noise. Instead the precise locations of the peaks and valleys are reproducible. The raw timing data indicates that there are some combinations of N and stencil $\mathcal{S}$ for which some compilers generate inefficient code while the usual code quality (and thus

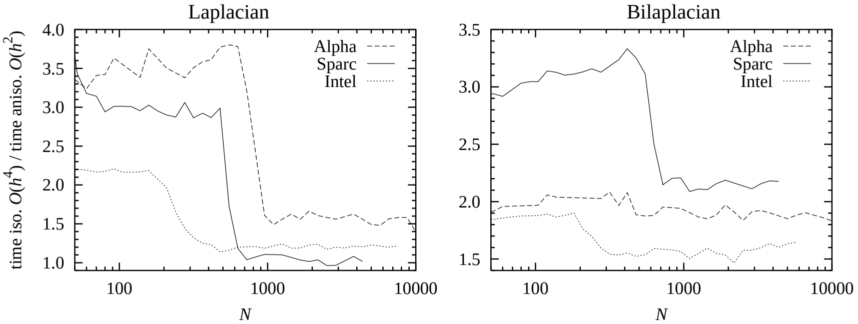


FIG. 3. Comparison of the fastest isotropic $\mathcal{O}(h^4)$ stencil with the fastest anisotropic $\mathcal{O}(h^2)$ stencil for the two-dimensional Laplacian (left) and Bilaplacian (right). For the Laplacian, the extra cost of using the better stencil is only of order 20 % whereas for the Bilaplacian it can be of order 100 %.

speed) is retained when a slightly different value for N is used. We do not believe that a deeper discussion of this would be fruitful. We rather encourage the reader to check the behaviour with their own hardware using their own compilers at different optimisation levels since they might behave very differently from the computers that we have studied.

The main conclusion from Fig. 2 is that for large enough matrices the overhead to be paid for using isotropic stencils for Laplacian and Bilaplacian usually is no more than 20 % — even though the stencils might contain more than twice as many elements. All current research studies systems that no longer fit into the cache, so that those simulations are in the large-matrix regime. We may thus conclude that there is little reason to use an anisotropic stencil in favour of an isotropic stencil of the same order.

For the two-dimensional case, there still is the question of the overhead of using a higher-order rather than a lower-order stencil. In Fig. 3 it can be seen that the extra cost when going from a simple anisotropic $\mathcal{O}(h^2)$ stencil to an isotropic $\mathcal{O}(h^4)$ stencil can be significant, depending on the computer system, and on whether the Bilaplacian is also needed. Then it has to be decided whether the effect of using a coarser grid, as sometimes being possible when going from $\mathcal{O}(h^2)$ to $\mathcal{O}(h^4)$, can compensate for some of the additional computational cost.

VI. DISCUSSION

As could be seen from the references given in Sec. IV., only a few of the stencils have been published before. This simply reflects the fact that most of the work geared towards stencils has focused on solving partial-differential equations, not on integrating them. (For the ease of notation, we use the word “solving” when referring to solving a partial-differential equation with boundary conditions, and “integrating” when computing the solution of a partial differential equation in an initial-value problem in time.)

To focus on the stencils for the Laplacian first: The isotropic $\mathcal{O}(h^2)$ stencils presented in this paper have already been published before, but none of the isotropic $\mathcal{O}(h^4)$ stencils have. The reason for this is that partial differential equations can be solved at a higher order than the order of the applied stencil. Let us demonstrate this by solving the equation

$$\Delta f(x,y) = \rho(x,y) \;, \tag{6.1}$$

by applying an isotropic $\mathcal{O}(h^2)$ stencil $\mathcal{S}$,

$$\Delta f(x,y) = \mathcal{S} f(x,y) + d\, h^2 \Delta^2 f(x,y) + \mathcal{O}(h^4) \;. \tag{6.2}$$

One notes that $\Delta^2 f(x,y) \equiv \Delta\rho(x,y)$ can be evaluated with the same stencil, giving

$$\Delta f(x,y) = \mathcal{S} f(x,y) + d\, h^2 \mathcal{S} \rho(x,y) + \mathcal{O}(h^4) \;. \tag{6.3}$$

The expression $\Delta f(x,y)$ can thus be computed with an error of $\mathcal{O}(h^4)$, even though the stencil is only of order $\mathcal{O}(h^2)$ — since the force term $\rho(x,y)$ is known and does not depend on $f(x,y)$. (This is the important difference from the case where f is integrated forward in time.) Such a stencil is frequently called a higher-order compact stencil, and for the above reason the need to develop explicit $\mathcal{O}(h^4)$ stencils has not been felt as urgently as it is for integration of partial differential equations.

In a more general context, the single application of a large stencil can be replaced by multiple applications of a smaller stencil. (Higher-order compact stencils are an example of this.) The efficiency of this replacement depends among other things on the boundary conditions. If conditions on the gradient of the solution are imposed on the boundary, a modified stencil has to be used in the neighbourhood of the boundary. The larger the stencil, the larger the number of different stencils which have to be designed. This imposes a strong penalty on using large stencils. For many initial value problems, periodic boundary conditions in space are imposed. Typically, the ultimate goal is to simulate a system that is infinitely large in space, and periodic boundary conditions are generally accepted to introduce the smallest artefacts when a finite system size has to be chosen. With periodic boundary conditions, the same stencil can be used everywhere and explicit $\mathcal{O}(h^4)$ stencils as presented in this paper become feasible.

An additional issue should be raised here: The discretisation of Eq. (1.3a) on a grid leads to a matrix equation with banded matrices where the width of the band depends on the extension of the stencil and on the system size. The resulting equations are easier to solve if the stencils are diagonally dominant. All $\mathcal{O}(h^2)$ stencils for the Laplacian presented in this paper are diagonally dominant (except for the 21-point stencil in three dimensions). Non-compact stencils cannot be diagonally dominant. For the Bilaplacian no compact explicit stencils exist, and numerical difficulties consequently arise already in two-dimensions [9, 10]. The precise conditions for "good" or "bad" stencils in this context depend on the precise method used for solving the matrix equations and are thus outside the scope of this paper.

Finally, we would like to comment briefly on previously reported stencils for computing the Bilaplacian. There are no compact stencils that would allow a direct computation of the Bilaplacian. Most of the previous work has focused on introducing auxiliary variables, such that the stencils applied on those auxiliary variables remain compact. The first possibility is to split $\Delta^2 f = \rho$ into the two equations $\Delta f = u$, $\Delta u = \rho$ (see e. g., Refs. [11, 12, 13] and references therein). The second option is to introduce the gradients of f in all spatial directions as additional variables (see Ref. [13] for 2D and Ref. [14] for 3D). For a boundary value problem involving the Bilaplacian, additional boundary conditions have to be specified, since the partial-differential equation is of fourth order. Frequent choices are Dirichlet boundary conditions of the first or second kind, where both the value of the function and the first or the second derivate, respectively, in normal direction are specified, and the optimal auxiliary variables depend on the kind of boundary conditions.

The additional computational cost of introducing auxiliary variables is not always as high as it might seem at first glance since the grid size can be increased if only the Bilaplacian appears in the differential equation. If other operators appear also, the grid size is determined by them and no increase of grid size is possible. The latter is the normal case for most differential operators studied in the context of growth processes such that explicit stencils for the Bilaplacian should be used even if they are non-compact. Consequently, some $\mathcal{O}(h^2)$ stencils for the Bilaplacian have been published. These are, however, anisotropic stencils that were derived by simply applying the one-dimensional stencils several times. Still, the pen-and-paper work can be significant, and the 25-point formula for the three-dimensional Bilaplacian was, to the best of our knowledge, published only in 1967 [7] (reprinted e. g. in Ref. [14]).

VII. CONCLUSIONS

In this paper, we have presented different stencils for the Laplacian, the Bilaplacian and the gradient of the Laplacian. The stencils are also available as a Mathematica file that includes routines to verify that the coefficients indeed correspond to correct stencils. Using the coefficients from this file avoids the risk of typos.

The focus of our paper is on isotropic stencils, i. e., on stencils where the discretisation error does not depend on the choice of coordinate system. These stencils contain significantly more elements than the traditionally used anisotropic stencils. We have shown, however, that in most situations the limited speed of data transfer between memory and cache leads to a decrease in speed of no more than 20 % when isotropic stencils are used instead of anisotropic ones.

This small price to pay means that there are only few situations where anisotropic stencils should still be used. We hope that our work will encourage the use of isotropic stencils in wider areas of science than it is done at the moment.

We would like to acknowledge valuable discussions with Timo Eirola. This work has been supported by the European Union through the Marie Curie fellowship HPMF-CT-2002-01794 (M. P.) and the Academy of Finland (M. K.).

REFERENCES

1. James D. Murray, Mathematical Biology, Springer, Berlin, 1993.
2. Pierre C. Hohenberg and Bertrand I. Halperin, Theory of dynamical critical phenomena, Rev Mod Phys 49 (1977), 435–479.
3. Murli M. Gupta and Jules Kouatchou, Symbolic derivation of finite difference approximations for the three-dimensional Poisson equation, Numer Meth Part Diff Eq 14 (1998), 593–606.
4. Milton Abramowitz and Irene A. Stegun, Handbook of Mathematical Functions, Dover, New York, 1970.

5. U. Ananthakrishnaiah, Ram P. Manohar, and John W. Stephenson, Fourth-order finite difference methods for three-dimensional general linear elliptic problems with variable coefficients, Num Meth Part Diff Eq 3 (1987), 229–240.
6. William F. Spotz and Graham F. Carey, A high-order compact formulation for the 3D Poisson equation, Numer Meth Part Diff Eq 12 (1996), 235–243.
7. J. Ribeiro Dos Santos, Équations aux différences finies pour l'équation biharmonique dans l'espace à trois dimensions, C R Acad Sci Paris Sér A–B 264 (1967), A291–A293.
8. Ed Anderson, Zhao-Jun Bai, Chris Bischof, L. Susan Blackford, Jim Demmel, Jack Dongarra, Jeremy du Cro, Anne Greenbaum, Sven Hammarling, Alan McKenney, and Danny Sorensen, LAPACK Users' Guide, SIAM, Philadelphia, third edn., 1999.
9. Samuel D. Conte and Ralph T. Dames, On an alternating direction method for solving the plate problem with mixed boundary conditions, J Assoc Comput Mach 7 (1960), 264–273.
10. Murli M. Gupta and Ram P. Manohar, Direct solution of the biharmonic equation using noncoupled approach, J Comput Phys 33 (1979), 236–248.
11. Murli M. Gupta, Discretization error estimates for certain splitting procedures for solving first biharmonic boundary value problems, SIAM J Numer Anal 12 (1975), 364–377.
12. Roland Glowinski and Olivier Pironneau, Numerical methods for the first biharmonic equations and for the two-dimensional Stokes problem, SIAM Rev 21 (1979), 167–212.
13. Irfan Altas, Jonathan Dym, Murli M. Gupta, and Ram P. Manohar, Multigrid solution of automatically generated high-order discretizations for the biharmonic equation, SIAM J Sci Comput 19 (1998), 1575–1585.
14. Irfan Altas, Jocelyne Erhel, and Murli M. Gupta, High accuracy solution of the three-dimensional biharmonic equations, Num Algorithms 29 (2002), 1–19.