Attenuating and enhancing properties of the approximate deconvolution method based on higher-order explicit and compact filters

L. CABAN, A. TYLISZCZAK

Department of Thermal Machinery, Czestochowa University of Technology,
Faculty of Mechanical Engineering and Computer Science,
Armii Krajowej 21, 42-201 Czestochowa, Poland, e-mail: artur.tyliszczak@pcz.pl

WE ANALYSE THE ACCURACY OF A DECONVOLUTION (inverse filtering) method in 1D and 2D periodic domains. The deconvolution is performed by applying the iterative van Cittert method using explicit and compact filters of the 2nd to 8th order. We consider cases in which an approximate inverse filter G_a^{-1} formulated to deconvolve an original function from a filtered one $(\overline{f} = G * f)$ is constructed based on: (a) G the same as used to define $\overline{f} = G * f$; (b) G different than the one used to define $\overline{f} = G * f$. In case (a), the convergence rate of the deconvolution process is much better when compact filters are used. This is attributed to a flatter transfer function of this type of filter and thus a smaller deterioration of the input function $\overline{f}$. Case (b) reflects a real situation in which the precise definition of a basic filter G used in $\overline{f} = G * f$ is unknown. We found that when G_a^{-1} is formulated based on G of a higher order than the one used to define $\overline{f}$ the reconstructed function $f^* = G_a^{-1} * \overline{f}$ is suppressed compared to the original function f. On the other hand, the deconvolution process performed with the use of G_a^{-1} defined based on G of a lower order than the order of the basic filter significantly amplifies the reconstructed function f^*. As a result, the function f^* contains more energy than the function f, especially in the range of small and high-frequency scales. This effect is particularly strong when explicit filters of different orders are used. The impact of the filter type in the practical application of deconvolution is demonstrated based on large eddy simulations (LES) of a 2D decaying homogenous turbulent flow. LES combined with an approximate deconvolution method (ADM) for the computation of sub-filter terms shows better accuracy than in the case when these terms are modelled using the classical Smagorinsky model or when they are neglected (no-model approach). This analysis consists of comparisons of the evolution of total energy, energy spectra, and higher-order moments (variance, skewness, kurtosis) of the velocity components and vorticity. We found that more accurate results are obtained when the deconvolution is performed using the explicit filters even if the deconvolution process based on the compact filters was found to converge faster in 1D and 2D test cases. Most likely this is because in the performed LES the explicit filters correspond better to an unknown filter induced by discretisation.

Key words: explicit and compact filters, van Cittert deconvolution method, signal analysis, sub-filter scales reconstruction, Large Eddy Simulation.

1. Introduction

Filtering is commonly used in data acquisition and digital signal processing. It is applied purposely to extract features from a signal distorted during its processing (measurement noise) and/or is implicitly introduced by a limited range of operation of measurement tools. The filtering can be applied to both continuous and discrete signals in both time and spatial domains. From a theoretical point of view, defining a general variable t representing time or space, the filtering in a 1D domain is defined by the convolution operation

$$(1.1) \qquad \overline{f}(t) = f(t) * G(t) = \int\limits_{-\infty}^{\infty} f(\tau)G(t - \tau)\,\mathrm{d}\tau,$$

where $f(t)$ is an input signal, $G(t)$ is a filter kernel function, shortly referred to as filter, and $\overline{f}(t)$ is an output signal. In the frequency domain ω, this operation is equivalent to the multiplication of Fourier transforms of the signal $\widehat{f}(\omega) = \mathcal{F}[f(t)]$ and the filter function $\widehat{G}(\omega) = \mathcal{F}[G(t)]$. Thus, one can write

$$(1.2) \qquad \mathcal{F}[\overline{f}(t)] = \widehat{f}(\omega)\widehat{G}(\omega), \quad f(t) * G(t) = \mathcal{F}^{-1}[\widehat{f}(\omega)\widehat{G}(\omega)],$$

where $\mathcal{F}[\cdot]$ and $\mathcal{F}^{-1}[\cdot]$ stand for the Fourier transform and its inverse; $\widehat{G}(\omega)$ is called the transfer function [1] and the knowledge of it allows examining the impact of the filter in the frequency domain. The process inverse to filtering is deconvolution, also called inverse filtering. Assuming that the recorded signal $\overline{f}(t)$ is the result of the filtering process, deconvolution allows obtaining a 'clean' original signal $f(t)$. The deconvolution operation is defined as the reciprocal of the convolution operation (1.1) and relies on the determination of $f(t)$ based on $G(t)$ and a given $\overline{f}(t)$. The accuracy of this process depends on the knowledge of $G(t)$ and the correctness of computing its inverse $G^{-1}(t)$.

The deconvolution methods have numerous applications in various fields. In radio astronomy, deconvolution is used to describe solar distributions of brightness temperature, distributions of cosmic noise, or aerial smoothing [2]. In spectroscopy, it has been applied to describe the blur of optical spectra [3–5], X-ray spectra [6–8] and infrared spectra [9, 10]. In scanning electron microscopy (SEM), deconvolution is used to post-process images to remove blurs caused by the existence of a finite electron-beam size. Yano *et al.* [11] applied deconvolution to reconstruct high spatial frequency information which has been lost due to the size of the electron beam. They showed that this method can be combined with all material analysis techniques where the beam size determines the instrumental resolution, e.g., scanning transmission electron microscopy (STEM), auger electron spectroscopy (AES), or secondary ion mass spectroscopy (SIMS). Deconvolution is also an excellent tool for increasing the resolution of an electron probe [12] and has been successfully applied to deconvolve real electron

microprobe data in diffusion tests [13]. In digital image processing, the deconvolution methods are widely used to restore blurred images [14] and can be combined with machine learning methods [15]. Kawata and Ichioka [16, 17] proposed an iterative image-restoration method to solve systems of linear equations occurring in a linear imaging model. Deconvolution finds applications in reconstructing tomography images from radiographic projections in Computer Tomography (CT) [18, 19]. CT image reconstruction methods include a filtered-back-projection (FBP) algorithm, back-project-filter (BPF) and a recovery strategy based on Fourier's theorem. A high spatial resolution and low-noise image-domain BPF reconstruction method can be obtained by applying deep learning techniques [20–22]. The so-called blind deconvolution approach in which neither the filter operator nor the input signal is known [23] finds applications in image processing [24, 25], tomography [26], ultrasound imaging [27–30], magnetic resonance imaging [31], remote sensing imaging [32], acoustics, where the signal is usually distorted by an unknown filter function, acoustic rooms for echos and reverberation cancellation, speech processing system identification [33], or restoring old acoustic recordings [34]. The deconvolution of acoustic emission signals can also be used for damage identification in composites [35], for instance, in reinforced concrete bridge slab [36]. Moreover, blind seismic deconvolution is an important tool in seismography based on the recovery of the reflectivity sequence from the seismic records when the seismic wave is unknown [37–39]. Wu *et al.* [40] used this technique to describe a stochastic behaviour of seismic reflection coefficients, and more recently, Lary *et al.* [41] proposed a new non-stationary blind deconvolution of seismic records. In telecommunications, Castaldi *et al.* [42] used blind estimation and deconvolution of communication channels with unbalanced noise. Recently, Cang *et al.* [43] have applied blind deconvolution for underwater acoustic channels using wide-band integrated dictionaries.

Deconvolution methods are also commonly used in computational fluid dynamics to recover small turbulent flow scales [44]. Their accurate representation in numerical simulations is precluded by a computational mesh on which the minimum length of resolved flow structures is limited by the Nyquist theorem and filtering. Regarding filtering, in general, there are two sources. The first one is an explicit filtering of variables by a known filter during a solution procedure [45–47]. The second is an implicit filtering induced by discretisation of the flow governing equations (Navier–Stokes equations, species transport equations, energy transport equation, etc.) by applying finite difference or finite volume type methods [48, 49]. With filtering, the so-called sub-filter term arises [48], which affects mainly the smallest flow scales resolved on a given computational mesh. To model the sub-filter scales in the large eddy simulation (LES), Stolz and Adams [50] proposed the approximate deconvolution method (ADM) based

on an iterative van Cittert deconvolution that is also the main deconvolution tool in this paper. In this approach, the approximate inverse $f^*(t)$ of the filtered function $\overline{f}(t)$ is calculated using the series of ν terms defined as

$$(1.3) \qquad f^*(t) = \sum_{n=0}^{\nu} (I - G(t))^n \overline{f}(t).$$

ADM quickly gained popularity in LES and has been used in various applications, e.g., in modelling of near-wall turbulent boundary layers [51–54], compressible flows with shock boundary layer interaction [55–57], decaying isotropic turbulence [58, 59, 61], large-scale ocean circulation [62, 63], and atmospheric boundary layer [64–66]. In reactive flows, ADM was used in the reconstruction of small-scale variations of species and enthalpy to model chemical source terms [67–69]. ADM has been also applied in the time domain [70], and to formulate the boundary conditions [71] and a relaxation term model [53, 72] replacing classical LES sub-grid models.

The above review shows that deconvolution methods play a very important role in data analysis and numerical simulations. In this paper, we focus on analysing the deconvolution method with implications for the blind deconvolution approach, which in practice seems to have more applications. In reality, except for the explicitly defined filter functions, the precise form of $G(t)$ is not known, and, moreover, it can be a conglomerate of different filter sources, as shown in Fig. 1. The original signal can be biased by external disturbances, the presence of measurement probes, the noise of experimental apparatus, or errors caused by a limited sampling frequency. These factors can distort the original signal in a broadband or narrow frequency range ω, by either amplifying or damping the signal at a given ω. They can be grouped in the function $G(t)$, not necessarily meaning the filter function in a general sense. We focus on the
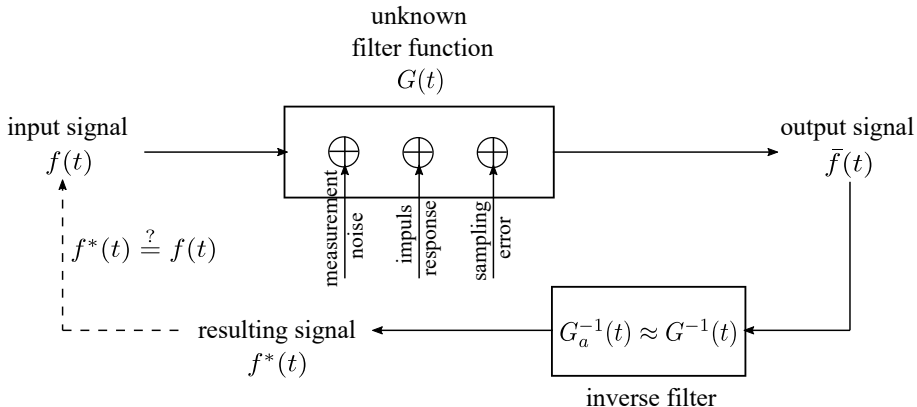

Fig. 1. Schematic representation of the filtering and deconvolution process.

accuracy of the deconvolution operation of $\overline{f}(t)$ by constructing the inverse operators $G_a^{-1}(t)$ to $G(t)$ using the ADM based on the explicit and compact Padé filtering methods. We analyse the convergence of ADM, i.e., $f^*(t) \to f(t)$, in the function of ν terms in (1.3) in two cases. At first, we consider $G_a^{-1}(t)$ constructed on the corresponding $G(t)$ function. DUNCA and LEWANDOWSKI [73] performed a theoretical analysis of ADM formulated for Helmholtz and Gaussian filters with the application to LES in a 3D periodic domain. It was shown that in some cases the number of terms ν needed to significantly reduce the error $\epsilon = f^*(t) - f(t)$ is so large that the van Cittert deconvolution seems not suitable for practical applications. As they concluded, their results were in contradiction with the findings of STOLZ et al. [51], who claimed that only a few terms in (1.3) are sufficient to quickly decrease ϵ and thus properly reconstruct sub-filter scales. Next, we investigate what happens with the deconvolved signal, $f^*(t)$ in Fig. 1, when $G_a^{-1}(t)$ is formulated based on $G(t)$ different than that used to obtain $\overline{f}(t)$. Unlike the theoretical analysis presented in [73], the present investigations are carried out for explicitly defined test functions of various complexity in 1D and 2D periodic domains. Finally, we analyse the accuracy of LES-ADM with various filters for modelling a 2D decaying turbulent field. The simulations are carried out using a high-order numerical code based on the 6th-order compact difference spatial discretisation and the 4th-order Runge–Kutta method for time integration. We compare energy spectra, temporal evolutions of turbulent kinetic energy (TKE), variance, and higher-order moments (skewness and kurtosis) of the velocity and vorticity fields. A sensitivity analysis of LES-ADM to the filter type was previously presented in [59] for 2D and 3D Taylor–Green flow employing a low-order discretisation method, and also for 1D Burgers equation [60]. The comparisons were made mainly for the TKE evolutions and revealed a strong dependence of the simulation accuracy on the choice of the filter type and its parameters. Although a clear superiority of one filter type over the other was not evidenced, it was suggested that the compact Padé filters ensure the best solutions.

2. Basic definitions

The deconvolution operation does not have a strict definition, constitutes an ill-conditioned inverse problem, and in some cases is not unique [74]. Namely, if $f_1(t)$ is a solution to the deconvolution problem, then $f_2(t) = f_1(t) + \eta(t)$ is also the solution, where $\eta(t)$ satisfies the homogeneous equation [75]

$$(2.1) \qquad \int_{-\infty}^{\infty} \eta(\tau) G(t - \tau) \, \mathrm{d}\tau = 0.$$

The ill-conditioning of the deconvolution means that small perturbations of the output signal $\overline{f}(t)$ or errors in defining the filter lead to large perturbations in $f(t)$. Therefore, the noisy data or inaccurate determination of the inverse filter can provide a significantly different solution than the real one. One of the most powerful tools used in solving inverse problems is the regularisation method, which consists in finding an approximate solution that fulfils additional constraints resulting from the physics of the problem [76].

In real applications, the continuous signal is discretised so that the continuous variable $f(t)$ and the filter function $G(t)$ are known only in discrete points i. Thus, Eq. (1.1) can be expressed as

$$(2.2) \qquad \overline{f}_i = \sum_j G_{i-j} f_j,$$

which in the matrix form reads as

$$(2.3) \qquad \overline{\mathbf{f}} = \mathbf{G}\mathbf{f},$$

where $\overline{\mathbf{f}}$ and $\mathbf{f}$ are the column vectors and $\mathbf{G}$ is the square matrix whose singularity or ill-conditioning [77] precludes the accurate calculation of the inverse filter by the simple inversion of the $\mathbf{G}$ matrix. A remedy for this problem is iterative methods [9, 10, 78–80] involving relaxation and optimisation techniques. In this paper, we compute $\mathbf{G}^{-1}$ using an approximate deconvolution method introduced by van Cittert [3] and Burger and van Cittert [4] based on an iterative convolution of the filter $\mathbf{G}$ and $\overline{\mathbf{f}}$. The output signal $\overline{\mathbf{f}}$ is taken as the initial approximation $\mathbf{f}_0^*$ of the input signal $\mathbf{f}$, and subsequent approximations $\mathbf{f}_n^*$ are obtained by adding the correction term being the difference between $\overline{\mathbf{f}}$ and its approximation $\overline{\mathbf{f}}_{n-1} = \mathbf{G}\mathbf{f}_{n-1}^*$ to the previous iteration. Writing these steps as:

$$(2.4) \qquad \begin{aligned} \mathbf{f}_0^* &= \overline{\mathbf{f}}, \\ \mathbf{f}_1^* &= \mathbf{f}_0^* + [\overline{\mathbf{f}} - \mathbf{G}\mathbf{f}_0^*], \\ \mathbf{f}_n^* &= \mathbf{f}_{n-1}^* + [\overline{\mathbf{f}} - \mathbf{G}\mathbf{f}_{n-1}^*], \end{aligned}$$

and continuing for the next ν iterations leads to the general form given as

$$(2.5) \qquad \mathbf{f}_\nu^* = \left\{ \mathbf{I} + [\mathbf{I} - \mathbf{G}] + [\mathbf{I} - \mathbf{G}]^2 + \cdots + [\mathbf{I} - \mathbf{G}]^\nu \right\} \overline{\mathbf{f}} = \sum_{n=0}^{\nu} [\mathbf{I} - \mathbf{G}]^n \overline{\mathbf{f}},$$

where $\mathbf{I}$ is the identity matrix. Thus, the expression

$$(2.6) \qquad \left\{ \mathbf{I} + [\mathbf{I} - \mathbf{G}] + [\mathbf{I} - \mathbf{G}]^2 + \cdots + [\mathbf{I} - \mathbf{G}]^\nu \right\} = \sum_{n=0}^{\nu} [\mathbf{I} - \mathbf{G}]^n = \mathbf{G}_a^{-1}$$

corresponds to an approximate inverse filter, $\mathbf{G}_a^{-1} \approx \mathbf{G}^{-1}$. For all filters analysed in this paper the series (2.6) satisfies the convergence criterion $|\mathbf{I} - \mathbf{G}| < 1$ [2, 81] and for $\nu \to \infty$ it converges such that $\mathbf{G}_a^{-1} \to \mathbf{G}^{-1}$. The van Cittert method is characterised by a noise amplification that increases linearly with the number of iterations. To overcome this problem, PARRUCK and RIAD [82] developed an optimisation procedure that minimises the noise content while maintaining deconvolution accuracy. BENNIA and RIAD [83] managed to optimise the number of iterations required to achieve an acceptable solution at minimum noise influence.

3. Filter definition

We analyse a 1D periodic domain $[0, L]$ consisting of K uniformly distributed nodes with spacings $h = L/K$, i.e., $x_i = h(i-1)$ are $i = 1, \ldots, K$ coordinates of the nodes. The domain $[0, L]$ can be regarded as a time or spatial axis. The formulas derived for the 1D domain can be directly applied to 2D and 3D cases along separate lines in particular directions. First, we consider the explicit finite difference (FD) filtering schemes that are most commonly found in the literature. A general formula for the FD filter can be written as

$$(3.1) \qquad \overline{f_i} = \frac{1}{2} \sum_{j=0}^{N} b_j (f_{i+j} + f_{i-j}),$$

where $\overline{f_i}$ are the filtered values at nodes, x_i and f_i are the known discrete function values and b_j, $j = 1, \ldots, N$, are the filter coefficients. The main feature of explicit filters is that the filtered value is defined only by the unfiltered quantities $f_{i\pm j}$ existing in the neighbouring nodes. The implicit filters are additionally constructed based on the filtered values such as $\overline{f}_{i\pm k}$. Following LELE [84], the implicit compact difference filters (CD) can be represented by the following general formula

$$(3.2) \qquad \overline{f}_i + \sum_{k=1}^{M} a_k (\overline{f}_{i+k} + \overline{f}_{i-k}) = \frac{1}{2} \sum_{j=0}^{N} b_j (f_{i+j} + f_{i-j}).$$

It should be noted that an explicit filter is a special case of a compact filter when $M = 0$. The procedure for determining the filter coefficients is presented in subsection 3.1. Writing (3.2) for all grid points leads to a linear system of equations, which in the matrix notation has the following form

$$(3.3) \qquad \mathbf{A}\overline{\mathbf{f}} = \frac{1}{2}\mathbf{B}\mathbf{f} \quad \to \quad \overline{\mathbf{f}} = \underbrace{\frac{1}{2}\mathbf{A}^{-1}\mathbf{B}}_{\mathbf{G}}\mathbf{f},$$

where $\bar{\mathbf{f}}$ and $\mathbf{f}$ are the vectors of dimensions $K \times 1$ containing the values of the filtered function $\bar{f}_i$ in nodes x_i, and the values of the function f_i in these nodes, respectively, $\mathbf{A}$ and $\mathbf{B}$ are the sparse band matrices of the dimension $K \times K$ composed by the filter coefficients a_k and b_j. For instance, with $M = 2$ and $N = 2$, the matrices corresponding to Eq. (3.2) have the form:

(3.4)

$$
\mathbf{A} = \begin{pmatrix}
1 & a_1 & a_2 & 0 & \dots & \dots & 0 & a_2 & a_1 \\
a_1 & 1 & a_1 & a_2 & 0 & \dots & \dots & 0 & a_2 \\
a_2 & a_1 & 1 & a_1 & a_2 & 0 & \dots & \dots & 0 \\
0 & a_2 & a_1 & 1 & a_1 & a_2 & 0 & \dots & \dots \\
\dots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots \\
\dots & \dots & 0 & a_2 & a_1 & 1 & a_1 & a_2 & 0 \\
0 & \dots & \dots & 0 & a_2 & a_1 & 1 & a_1 & a_2 \\
a_2 & 0 & \dots & \dots & 0 & a_2 & a_1 & 1 & a_1 \\
a_1 & a_2 & 0 & \dots & \dots & 0 & a_2 & a_1 & 1
\end{pmatrix}, \quad
\mathbf{B} = \begin{pmatrix}
2b_0 & b_1 & b_2 & 0 & \dots & \dots & 0 & b_2 & b_1 \\
b_1 & 2b_0 & b_1 & b_2 & 0 & \dots & \dots & 0 & b_2 \\
b_2 & b_1 & 2b_0 & b_1 & b_2 & 0 & \dots & \dots & 0 \\
0 & b_2 & b_1 & 2b_0 & b_1 & b_2 & 0 & \dots & \dots \\
\dots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots \\
\dots & \dots & 0 & b_2 & b_1 & 2b_0 & b_1 & b_2 & 0 \\
0 & \dots & \dots & 0 & b_2 & b_1 & 2b_0 & b_1 & b_2 \\
b_2 & 0 & \dots & \dots & 0 & b_2 & b_1 & 2b_0 & b_1 \\
b_1 & b_2 & 0 & \dots & \dots & 0 & b_2 & b_1 & 2b_0
\end{pmatrix}.
$$

The filter matrix operator $\mathbf{G}$ is then defined as $\mathbf{G} = \frac{1}{2}\mathbf{A}^{-1}\mathbf{B}$. In the case of the explicit filters, the matrix $\mathbf{A} = \mathbf{I}$ and in this case Eq. (3.1) would have the form $\bar{\mathbf{f}} = \frac{1}{2}\mathbf{B}\mathbf{f}$ with $\mathbf{G} = \frac{1}{2}\mathbf{B}$.

3.1. Derivation of the filtering coefficients

FD filters are a special case of the CD filter with $M = 0$ in (3.2). In periodic domains, the filter formulas (3.2) are symmetric with respect to the i-node. Each filter has an associated transfer function $\widehat{G}(\omega)$ representing the filtering in the Fourier space. It is defined as

(3.5)
$$
\widehat{G}(\omega) = \frac{\sum_{j=0}^{N} b_j \cos(j\omega)}{1 + 2\sum_{k=1}^{M} a_k \cos(k\omega)},
$$

where ω is the scaled wavenumber in the range $[0, \pi]$ [1, 84]. Its discrete values are $\omega_n = n2\pi/K$ with $n = 0, 1, \dots, K/2$. The transfer function $\widehat{G}(\omega)$ is real, which means that the filter modifies the amplitude of f without affecting its phase [85]. Expansions of $f_{i\pm j}$ and $\bar{f}_{i\pm k}$ in (3.2) into the Taylor series define non-zero terms of $2n$-th order

$$
\mathcal{M}^{2n}(x_i) = \sum_{j=0}^{N} \frac{(jh)^{2n}}{(2n)!} b_j - 2\sum_{k=1}^{M} \frac{(kh)^{2n}}{(2n)!} a_k.
$$

The filter coefficients are derived based on the conditions put on $\widehat{G}(\omega)$ and $\mathcal{M}^{2n}(x_i)$. First, we require that the filter does not modify a constant value function $f = \text{const}$, nor a constant component of its spectrum $\widehat{f}(\omega = 0)$. Assuming the truncation error of the Taylor series expansion of the order $2N$ $(\mathcal{O}(h)^{2N})$

leads to $N-1$ equations for $\mathcal{M}^{2n}(x_i) = 0$. The shortest wavelength that can be represented on a mesh composed of K nodes is determined by the Nyquist frequency corresponding to the mode $n = K/2$ that translates to $\omega = \pi$. As in [84] we assume that the filter excludes $\widehat{f}(\omega)$ occurring at $\omega = \pi$ that is obtained with $\widehat{G}(\pi) = 0$. Note that with the definition (3.5) the condition $d\widehat{G}/d\omega|_{\omega=\pi} = 0$ also holds automatically. The above conditions lead to a system of linear equations in the form

$$
(3.6) \quad
\begin{cases}
\widehat{G}(0) = 1 & \rightarrow \displaystyle\sum_{j=0}^{N} b_j - 2\sum_{k=1}^{M} a_k = 1, \\
\mathcal{M}^{2n}(x_i) = 0 \rightarrow \displaystyle\sum_{j=0}^{N} j^{2n} b_j - 2\sum_{k=1}^{M} k^{2n} a_k = 0 & \text{for } n = 1, \ldots, N-1, \\
\widehat{G}(\pi) = 0 & \rightarrow \displaystyle\sum_{j=0}^{N} (-1)^j b_j = 0,
\end{cases}
$$

which is a generalisation of the formulas presented in [84]. Note that the condition $\widehat{G}(0) = 1$ is equivalent to $\mathcal{M}^0(x_i) = 1$ in the physical space. The solution of the system (3.6) provides a family of $2N$ order filters with M parameters a_k. As discussed in [84] they can be used to tune the shape of the transfer function at $\omega \to \pi$, e.g., by a constraint $d^2\widehat{G}/d\omega^2 = 0$ or at a specific wave number ω by requiring that $\widehat{G}(\omega)$ equals to an assumed value. Details of various filters can be found in [84] along with detailed expressions of their truncation errors. In this paper, we employ the filters up to the 8th order defined through the coefficients specified in Table 1. The p-values denote the order of the filter $\mathcal{O}(h^p)$. In the case of CD filters, we consider a one-parameter family for $M = 1$ and $M = 2$ with a varying coefficient a_1. Its value changes the shape of the transfer function

TABLE 1. Coefficients of the explicit finite difference (FD) and compact difference (CD) filters defined through (3.2).

Filter	M	N	p	a_2	b_0	b_1	b_2	b_3	b_4
FD2	0	1	2	–	$\frac{1}{2}$	$\frac{1}{2}$	–	–	–
FD4	0	2	4	–	$\frac{5}{8}$	$\frac{1}{2}$	$-\frac{1}{8}$	–	–
FD6	0	3	6	–	$\frac{11}{16}$	$\frac{15}{32}$	$-\frac{3}{16}$	$\frac{1}{32}$	–
FD8	0	4	8	–	$\frac{93}{128}$	$\frac{7}{16}$	$-\frac{7}{32}$	$\frac{1}{16}$	$-\frac{1}{128}$
CD2	1	1	2	–	$\frac{1}{2} + a_1$	$\frac{1}{2} + a_1$	–	–	–
CD4	1	2	4	–	$\frac{5}{8} + *\frac{3}{4}a_1$	$\frac{1}{2} + a_1$	$-\frac{1}{8} + \frac{1}{4}a_1$	–	–
CD6	1	3	6	–	$\frac{11}{16} + \frac{5}{8}a_1$	$\frac{15}{32} + \frac{17}{16}a_1$	$-\frac{3}{16} + \frac{3}{8}a_1$	$\frac{1}{32} - \frac{1}{16}a_1$	–
CD8	2	3	8	$\frac{3}{10} - \frac{1}{5}a_1$	$\frac{1}{2} + \frac{3}{4}a_1$	$\frac{3}{4} + \frac{7}{8}a_1$	$\frac{3}{10} + \frac{1}{20}a_1$	$\frac{1}{20} - \frac{3}{40}a_1$	–

but not the order of the filter. We compare the filtering and deconvolution with $a_1 = 0.25, 0.4, 0.475$. The transfer functions of 2nd, 4th and 6th order explicit filters and a 4th order compact filter with different a_1 are shown in Fig. 2a. The straight horizontal line denotes the reference level ('no-filtering') corresponding to $\overline{\mathbf{f}} = \mathbf{f}$ in (2.3). It can be seen that the increase of the filter order or a_1 causes the rise of $\widehat{G}(\omega)$ at a higher frequency range. Note that $\widehat{G}(\omega)$ of the 4th order filter CD4 ($M = 1$, $N = 2$) with $a_1 = 0.25$ has a similar shape as the 6th order explicit filter FD6 ($M = 0$, $N = 3$). Figure 2b presents the transfer functions of CD filters with $M = 1$ and $M = 2$ with the coefficient $a_1 = 0.4$ and varying N. It is worth noticing that for $M = 1$ the lower the order of the filter the more intense the filtering effect is in the range $\omega > \pi/2$. For $M = 2$ the situation is not so clear and these filters are characterised by $\widehat{G}(\omega)$ which becomes steeper when the order of the filter increases ($N = 2 \rightarrow 4$). For instance, the filter with $M = 2$, $N = 2$ for $\omega < 1.7$ is more aggressive than the filter based on $M = 2$, $N = 4$, but for $\omega > 1.7$ the situation is opposite.
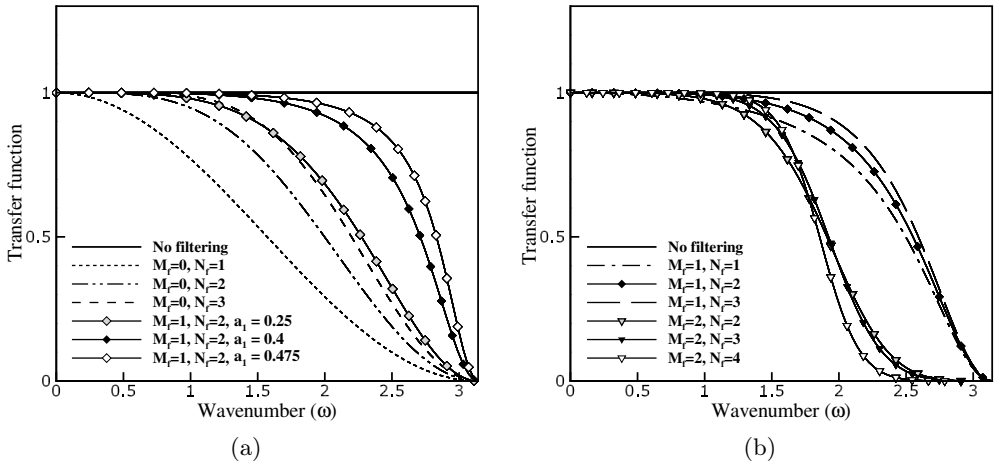

Fig. 2. Transfer functions of the selected explicit and compact filters with varying a_1 (a) and of the different compact filters with $a_1 = 0.4$ (b).

4. Results

In this section, we first analyse the convergence of the van Cittert approximate deconvolution method for the explicit and compact filters of various orders for filtered 1D analytical functions. Next, we focus on the deconvolution in 1D and 2D cases for the cases in which the approximate inverse filter is constructed based on a filter different than that used to prepare the filtered test functions.

4.1. 1D case

The one-dimensional test function has a simple form of K sinusoidal waves

$$(4.1) \qquad f(x_i) = \sum_{k=1}^{K} \sin\left(k \cdot x_i + \varphi\right),$$

where $x_i = (i-1)h$, i is the index of $N = 64$ sampling points (nodes), $h = 2\pi/N$ is the space between them and φ is a random phase. We consider three test functions, f_1 with a moderate range of frequencies ($K = N/2 - 16$), f_2 containing $K = N/2 - 1$ modes, and the function f_3 that extends up to the Nyquist frequency $K = N/2$. All modes of the f function given by the formula (4.1) have the same constant amplitude equal to one. This is a 'difficult' situation created artificially to determine the accuracy of deconvolution. In practice, the amplitude of particular modes decreases with increasing K, because the high-frequency components are related to the small-scale low-amplitude phenomena. The functions f_1, f_2 and f_3 are shown in Fig. 3. It can be seen that the function f_1 is smooth compared to the functions f_2 and f_3 and this has a direct implication on their filtered counterparts $\overline{f} = G * f$. The right vertical axis in Fig. 3 shows the absolute values of the local differences between the original and filtered functions $|f - G * f|$ calculated using the 4th order explicit filter and the 4th order compact filter with $a_1 = 0.4$ defined in Table 1. In the case of the smoother function (f_1), these differences are small, whereas in the cases with the functions f_2 and f_3, they reach values comparable with the function values. In the next section, we analyse how the complexity of the function affects the deconvolution process.

4.1.1. Assessment of the accuracy of the deconvolution in the physical and spectral space. We define the error between the original and deconvolved functions as:

$$(4.2) \qquad \begin{aligned} \mathcal{E} = f - f^* &= f - G_a^{-1} * (G * f) \\ &= \underbrace{(I - G_a^{-1} * G)}_{\mathcal{R}} * f. \end{aligned}$$

From Fig. 2 one may easily infer that $G_a^{-1} * G$ at low ω must be close to identity, hence, the term $\mathcal{R}$ acts as a high-pass filter. In the LES-ADM approach formulated in [51], the term $\mathcal{R}$ constitutes the main part of the relaxation source term in the Navier–Stokes equations. It replaces a classical sub-grid model but plays essentially the same role. It causes energy damping at small scales that stabilises the solution. The term $\mathcal{R}$ and its Fourier transform can be written as:

$$(4.3) \qquad \mathcal{R} = I - G_a^{-1} * G = (I - G)^{\nu+1}; \quad \mathcal{F}[\mathcal{R}] = \widehat{\mathcal{R}}(\omega) = (I - \widehat{G}(\omega))^{\nu+1}.$$
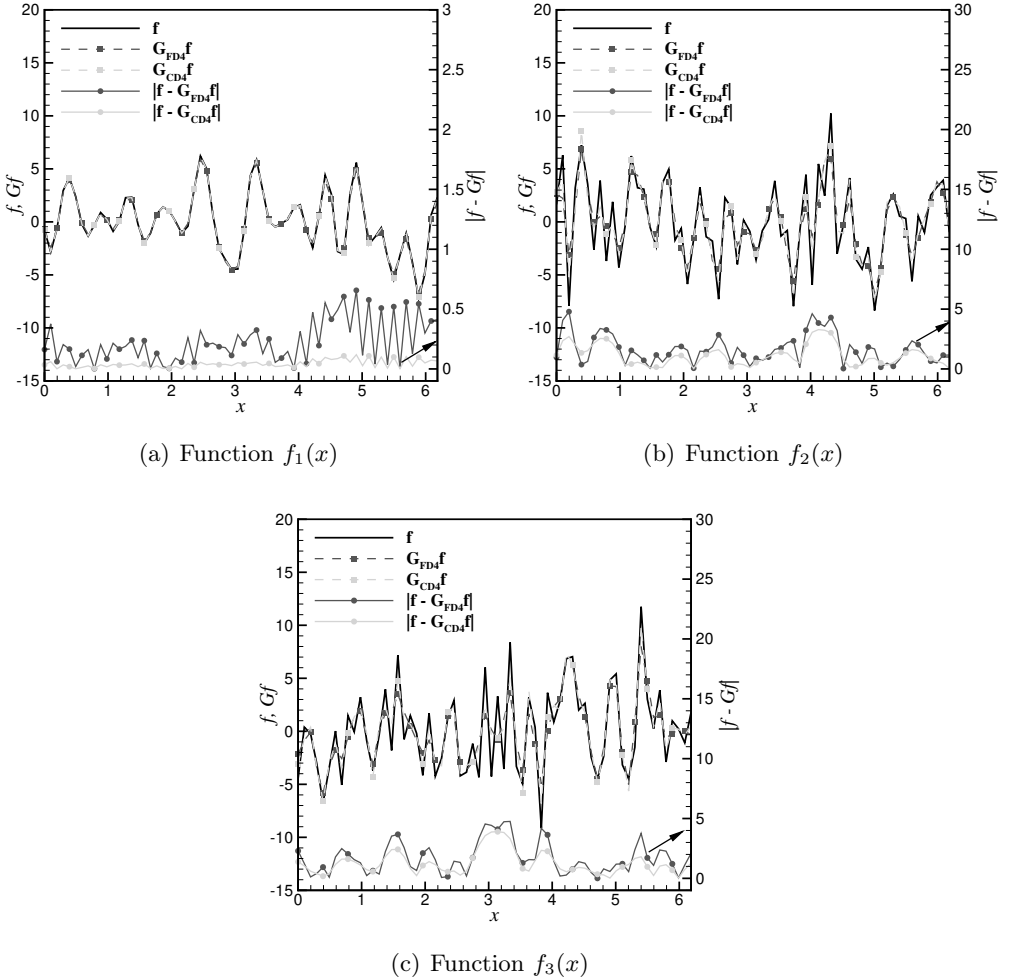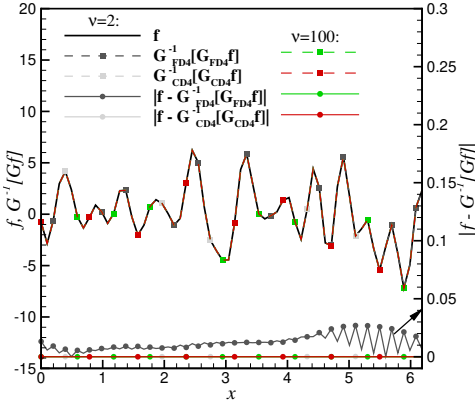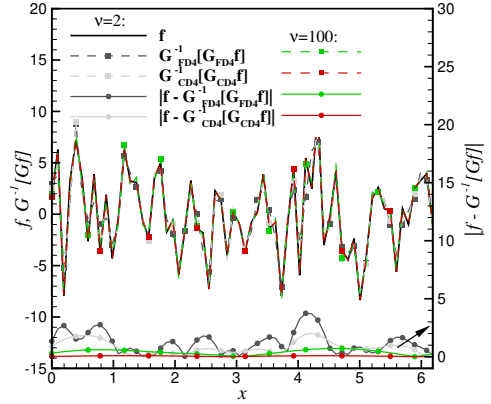
(a) Function $f_1(x)$

(b) Function $f_2(x)$

(c) Function $f_3(x)$

FIG. 3. Functions f_1 (a), f_2 (b) and f_3 (c) with their filtered counterparts $\overline{f} = G * f$ obtained using the 4th order explicit and compact filter with $a_1 = 0.4$. The right vertical axes show the modulus of differences between the original and filtered functions $|f - G * f|$.

The Fourier transform of (4.2) is

$$(4.4) \qquad \widehat{\mathcal{E}}(\omega) = (I - \widehat{G}(\omega))^{\nu+1}\widehat{f}(\omega).$$

This formula shows that both the filter type and the function shape affect the deconvolution error. For low-order filters with $\widehat{G}(\omega)$ acting over a wide range of ω where $\widehat{f}(\omega)$ are non-zero, the deconvolution process requires a large number of iterations. This is why in [73] it was concluded that for functions with $\widehat{f}(\omega)$ large in the range $\omega \to \pi$, a few iterations will not suffice, as often suggested based

on test functions containing only low-frequency modes. In particular, Eq. (4.4) shows that the deconvolution process will never converge for the filters with $\widehat{G}(\pi) = 0$ when $\widehat{f}(\pi) \neq 0$. Below we analyse the convergence of G_a^{-1} to G^{-1} in the function of ν in the physical space through the L_1 norm of the vector of the nodal error values

$$(4.5) \qquad \mathbf{E} = \mathbf{f} - \mathbf{f}^* = \mathbf{f} - \mathbf{G}_a^{-1}(\mathbf{Gf}),$$

where $\mathbf{f}, \mathbf{f}^*$ are the vectors of the test function and the function resulting from the deconvolution. If $L_1(\mathbf{E}) \to 0$ this means that $\mathbf{G}_a^{-1} \to \mathbf{G}^{-1}$ and thus $\mathbf{f}^* \to \mathbf{f}$. For
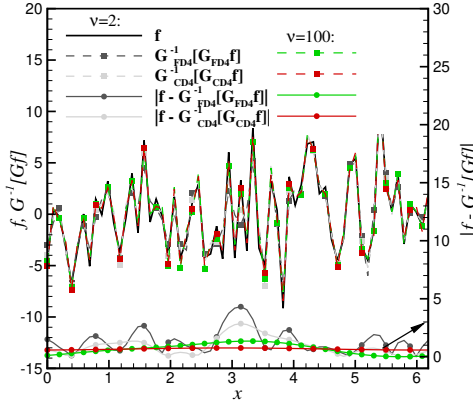

(a) Function $f_1(x)$

(b) Function $f_2(x)$

(c) Function $f_3(x)$

FIG. 4. Functions f_1 (a), f_2 (b) and f_3 (c) with their deconvolved counterparts $f^* = G^{-1}[G * f]$ obtained by the deconvolution based on the 4th order explicit and compact filters with $a_1 = 0.4$, for $\nu = 2$ and $\nu = 100$. The solid lines with the symbols show the absolute values of local errors: $|f - G^{-1}[G * f]|$.

simplicity, the subscript 'a' is omitted in the following notation and $\mathbf{G}^{-1}$ should be always treated as the approximate inverse filter and the subscript denotes the filter type. For instance, $\mathbf{G}_{\mathrm{FD4}}^{-1}$ and $\mathbf{G}_{\mathrm{CD4}}^{-1}$ stand for the approximate inverse filters calculated based on the 4th order explicit and compact filters (FD4 and CD4 defined in Table 1). Figure 4 shows the effect of deconvolution for $\nu = 2$ and $\nu = 100$ applying the 4th order explicit and compact ($a_1 = 0.4$) filters. On the right vertical axes in Fig. 4, the errors in the particular nodes are shown. It can be seen that the function $\mathbf{f}_1^*$ obtained from the deconvolution coincides well with the function $\mathbf{f}_1$. In this case, the applied approximate inverse filter converges quite well already for $\nu = 2$. On the contrary, the values of $\mathbf{f}_2^*$ and $\mathbf{f}_3^*$ for $\nu = 2$ are far from their originals and even for $\nu = 100$, the errors are noticeable.
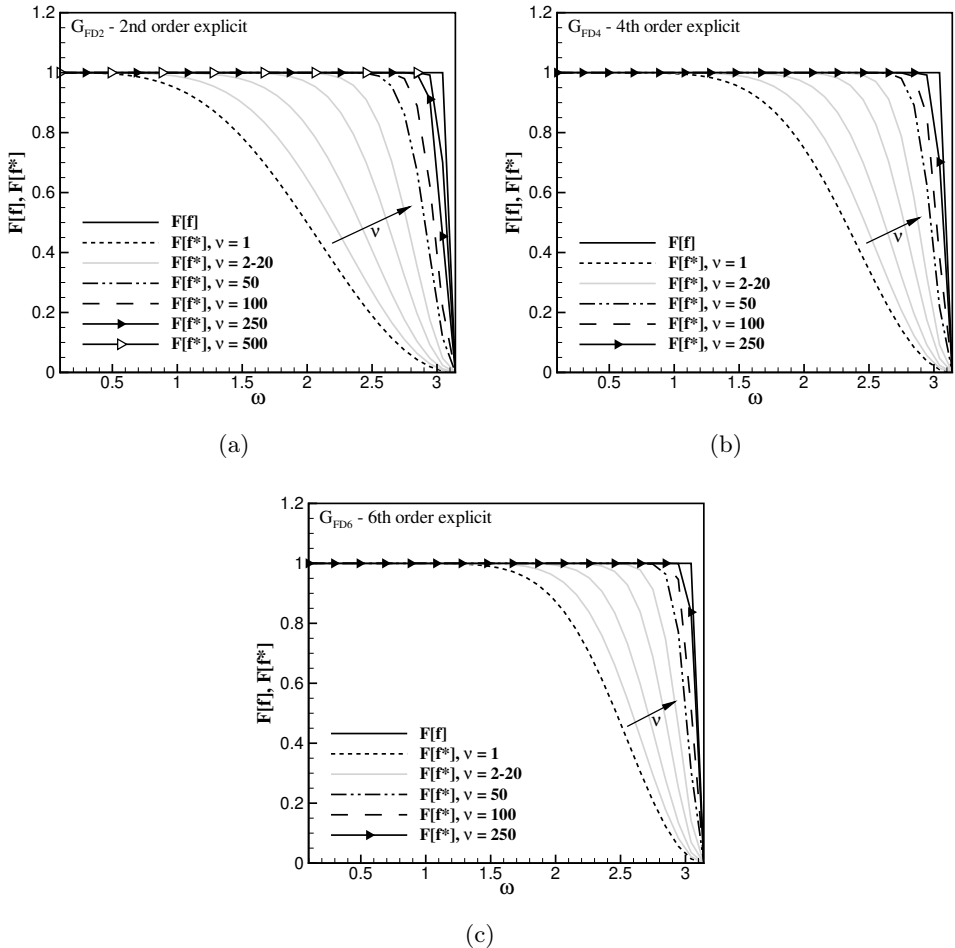

(a)

(b)

(c)

Fig. 5. Convergence of the deconvolution process in spectral space for the function f_2 applying 2nd (a), 4th (b) and 6th (c) order explicit filter.

Analysis of the convergence of the series (2.6) in terms of ν is performed in the spectral space by calculating the Fourier transforms $\mathcal{F}[\mathbf{f}^*] = \widehat{\mathbf{f}}^*$ and comparing their spectra with $\mathcal{F}[\mathbf{f}] = \widehat{\mathbf{f}}$.

Figure 5 shows the convergence process $\widehat{\mathbf{f}}^* \to \widehat{\mathbf{f}}$ for the f_2 function. It can be seen that for small ν, the deconvolution error is large and the amplitudes of particular modes are largely underestimated, especially in the high-frequency range. Note that it would be ideal if they all equalled 1 as defined by the test function. The convergence in terms of ν is asymptotic, which manifests the fact that for an increasing ν the corresponding lines successively tend to the spectrum of the original function. Note that for a smaller value of ν the series (2.6) converges faster for the higher-order filters. For example, for the 2nd order inverse explicit filter ($\mathbf{G}_{\mathrm{FD2}}^{-1}$), an accurate representation of the function $\widehat{\mathbf{f}}_2^*$ is obtained only if ν is large ($\nu = 500$). For the 4th order and 6th order explicit filters ($\mathbf{G}_{\mathrm{FD4}}^{-1}$, $\mathbf{G}_{\mathrm{FD6}}^{-1}$), the error is reduced much faster and $\widehat{\mathbf{f}}^*$ approaches to $\widehat{\mathbf{f}}$ well already for $\nu = 250$. Table 2 shows $L_1(\mathbf{E})$ for selected ν for all three test functions. It can be seen that for the smooth function f_1 the values of $L_1(\mathbf{E})$ decrease very quickly. For instance, for the filter $\mathbf{G}_{\mathrm{FD6}}^{-1}$ the level of machine accuracy ($\mathcal{O}(10^{-12})$) is obtained already for $\nu = 20$. It is worth noticing that a further increase of ν makes the approximation less accurate and this is related to the sensitivity of the van Cittert method to the noise at larger ν values [82, 83]. Figure 6 presents the convergence process $\widehat{\mathbf{f}}^* \to \widehat{\mathbf{f}}$ using the 4th order compact filter (CD4) with the parameter $a_1 = 0.25, 0.4, 0.475$ for the test function f_2. This time a relatively good approximation of the filter $\mathbf{G}^{-1}$ is observed already for $\nu = 50$. Table 3 shows the values of $L_1(\mathbf{E})$ for the selected ν. For the function f_1, we see that $\mathbf{f}_1^* \approx \mathbf{f}_1$ already for $\nu = 10$. When a_1 tends to 0.5 the convergence rate increases. A similar behaviour was

TABLE 2. Detailed values of $L_1(\mathbf{E})$ errors calculated based on the deconvolved function obtained applying the explicit filters.

| ν value	$L_1(\mathbf{E_1})$		$L_1(\mathbf{E_2})$		$L_1(\mathbf{E_3})$	
	$\mathbf{G}_{\mathrm{FD2}}$	$\mathbf{G}_{\mathrm{FD6}}$	$\mathbf{G}_{\mathrm{FD2}}$	$\mathbf{G}_{\mathrm{FD6}}$	$\mathbf{G}_{\mathrm{FD2}}$	$\mathbf{G}_{\mathrm{FD6}}$
$\nu = 1$	0.14×10^2	0.72×10^0	0.10×10^3	0.77×10^2	0.10×10^3	0.85×10^2
$\nu = 2$	0.65×10^1	0.85×10^{-1}	0.94×10^2	0.69×10^2	0.98×10^2	0.78×10^2
$\nu = 4$	0.14×10^1	0.13×10^{-2}	0.81×10^2	0.61×10^2	0.88×10^2	0.69×10^2
$\nu = 10$	0.21×10^{-1}	0.50×10^{-8}	0.66×10^2	0.49×10^2	0.75×10^2	0.56×10^2
$\nu = 20$	0.20×10^{-4}	0.18×10^{-12}	0.56×10^2	0.39×10^2	0.63×10^2	0.50×10^2
$\nu = 50$	0.32×10^{-12}	0.30×10^{-12}	0.42×10^2	0.28×10^2	0.52×10^2	0.44×10^2
$\nu = 100$	0.10×10^{-11}	0.60×10^{-12}	0.34×10^2	0.19×10^2	0.47×10^2	0.42×10^2
$\nu = 150$	0.14×10^{-11}	0.16×10^{-11}	0.29×10^2	0.13×10^2	0.44×10^2	0.42×10^2
$\nu = 200$	0.16×10^{-11}	0.18×10^{-12}	0.25×10^2	0.95×10^1	0.42×10^2	0.42×10^2

(a)

(b)

(c)
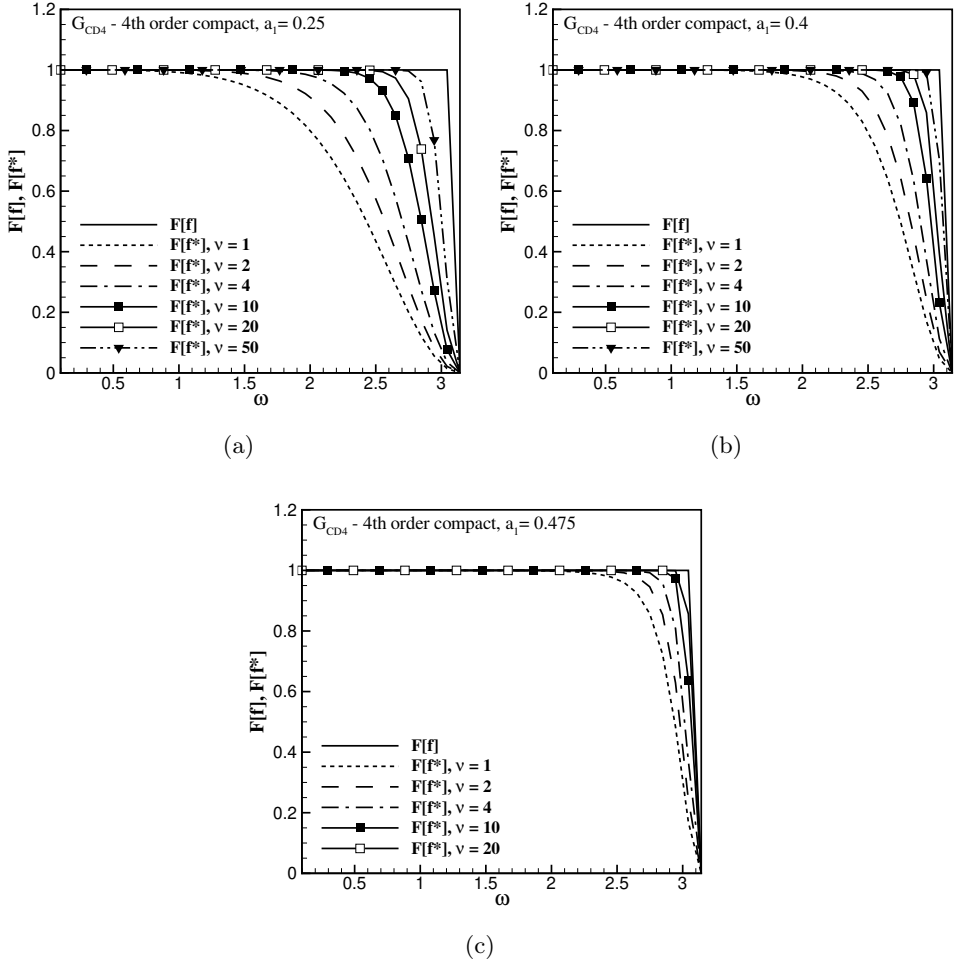
FIG. 6. Convergence of the deconvolution process in spectral space for the function f_2 applying 4th order compact filter with parameters $a_1 = 0.25$ (a), $a_1 = 0.4$ (b), $a_1 = 0.475$ (c).

observed for the increasing order of the explicit filter (see Table 2). However, one should notice that the convergence of the deconvolution process is more dependent on the filter transfer function shape than on the filter order. For instance, comparing the results obtained for the same filter's order, $\mathbf{G}_{FD4}^{-1}$ shown in Fig. 5b and $\mathbf{G}_{CD4}^{-1}$ in Fig. 6b, the error in the latter case is smaller. In fact, it is smaller than in the case when the 6th order explicit filter $\mathbf{G}_{FD6}^{-1}$ is used, see Fig. 5c. On the other hand, applying the filters of different types and orders $\mathbf{G}_{FD6}^{-1}$ and $\mathbf{G}_{CD4}^{-1}$ with $a_1 = 0.25$, but having very similar transfer function shapes (see Fig. 2) the deconvolution errors are comparable. This is confirmed in Fig. 5c and Fig. 6a, and also in Tables 2 and 3. The reason for such be-

TABLE 3. Detailed values of $L_1(\mathbf{E})$ errors calculated based on the deconvolved function obtained applying the 4th order compact filter with different a_1 parameter.

ν value	$L_1(\mathbf{E_1})$		$L_1(\mathbf{E_2})$		$L_1(\mathbf{E_3})$	
	$a_1 = 0.25$	$a_1 = 0.475$	$a_1 = 0.25$	$a_1 = 0.475$	$a_1 = 0.25$	$a_1 = 0.475$
$\nu = 1$	0.75×10^0	0.73×10^{-2}	0.73×10^2	0.38×10^2	0.81×10^2	0.50×10^2
$\nu = 2$	0.87×10^{-1}	0.85×10^{-4}	0.65×10^2	0.33×10^2	0.74×10^2	0.46×10^2
$\nu = 4$	0.13×10^{-2}	0.13×10^{-7}	0.57×10^2	0.26×10^2	0.64×10^2	0.43×10^2
$\nu = 10$	0.50×10^{-8}	0.78×10^{-13}	0.45×10^2	0.14×10^2	0.53×10^2	0.42×10^2
$\nu = 20$	0.19×10^{-12}	0.13×10^{-12}	0.36×10^2	0.58×10^1	0.48×10^2	0.42×10^2
$\nu = 50$	0.38×10^{-12}	0.37×10^{-12}	0.25×10^2	0.36×10^0	0.42×10^2	0.42×10^2
$\nu = 100$	0.91×10^{-12}	0.85×10^{-12}	0.15×10^2	0.36×10^{-2}	0.42×10^2	0.42×10^2
$\nu = 150$	0.15×10^{-11}	0.13×10^{-11}	0.95×10^1	0.35×10^{-4}	0.42×10^2	0.42×10^2
$\nu = 200$	0.20×10^{-11}	0.19×10^{-11}	0.58×10^1	0.35×10^{-6}	0.42×10^2	0.42×10^2

havior is that $\bar{\mathbf{f}}$ obtained by applying the filter with $\widehat{G}(\omega)$ close to one over a wider range of ω is closer to $\mathbf{f}$. Thus, the initial deconvolution errors are smaller.

A convergence rate of $\widehat{\mathbf{f}}^* \to \widehat{\mathbf{f}}$ using particular filters is examined in Fig. 7 presenting a log-lin plot of $L_1(\mathbf{E})$ normalised by L_1 norm of an initial difference between $\mathbf{f}$ and $\bar{\mathbf{f}}$ denoted $L_1(\mathbf{E_0})$. It can be seen that for $\mathbf{f}_1$ and $\mathbf{f}_2$ the errors $L_1(\mathbf{E})/L_1(\mathbf{E_0})$ drop linearly with the slopes dependent on the filter and the test function. From (4.4) we see that Fourier modes of the error normalized by their initial values $\widehat{\mathcal{E}}_0(\omega) = (I - \widehat{G}(\omega))\widehat{f}(\omega)$ are $\widehat{\mathcal{E}}(\omega)/\widehat{\mathcal{E}}_0(\omega) = (I - \widehat{G}(\omega))^\nu$. Taking $\log(\widehat{\mathcal{E}}(\omega)/\widehat{\mathcal{E}}_0(\omega))$ determines the slope of the error drop, which equals to $\log(I - \widehat{G}(\omega))$ in a log-lin plot. Hence, if $(I - \widehat{G}(\omega))$ is small that holds for the filters with a flatter $\widehat{G}(\omega)$ the convergence rate is large. Note that the overall convergence rate for a given test function results from the summation of the errors for all non-zero modes $\widehat{f}(\omega)$. Thus, it is also linear but becomes worse when the test function contains high-frequency modes, here the function f_2. Figure 7c shows the error drop for the function f_3. It can be seen that in this case $L_1(\mathbf{E})/L_1(\mathbf{E_0})$ stagnates at constant levels. The reason for that is the inability of the deconvolution process to recover the component of f_3 at $\omega = \pi$, as explained at the beginning of this section. In terms of ν, the constant error levels are reached faster and are higher for the filters with a flatter $\widehat{G}(\omega)$. The latter effect is because in these cases the initial $L_1(\mathbf{E_0})$ are smaller. The real values of $L_1(\mathbf{E})$ are shown in Tables 2 and 3. It can be seen that regardless of the filter type, they are the same at the large ν. From the definition of the test function (4.1) we see that $L_1(\mathbf{E}) = \sum_{i=1}^N |\sin(\varphi)|$ for $\nu \to \infty$. In the next section, we analyse the deconvolution accuracy in the cases in which the approximate filter $\mathbf{G}^{-1}$ is computed based on $\mathbf{G}$ different than that used to create $\bar{\mathbf{f}}$.
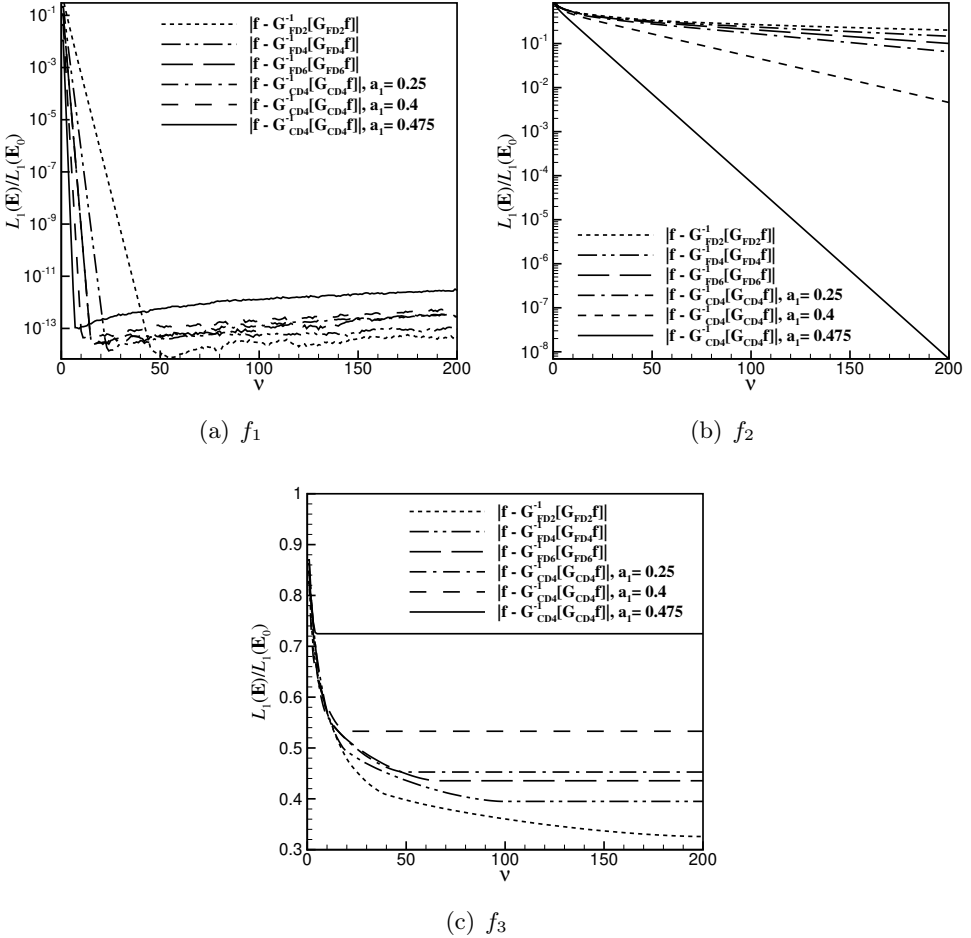
(a) f_1

(b) f_2

(c) f_3

FIG. 7. Log-lin plots of the $L_1(\mathbf{E})$ error in terms of ν parameter for the functions f_1 (a), f_2 (b) using different filters. In subfigure (c) showing $L_1(\mathbf{E})$ for the functions f_3 the axes are linear.

4.1.2. Defiltering with a filter different than the basic one. In practice, the filter (hereinafter referred to as basic) that acts on the input signal is usually unknown. However, the deconvolution process requires its knowledge. In the previous section, it was shown that the van Cittert method quickly converges when the basic filter characterises $\widehat{G}(\omega) \approx 1$ in a possibly the largest range of ω. In this section, we analyse the results of the deconvolution when $\mathbf{G}^{-1}$ is computed using $\mathbf{G}$ different than the basic one and thus having different $\widehat{G}(\omega)$ and order. We compare the cases specified in Table 4 and consider only the function f_2 for which accurate deconvolution requires a substantial number of iterations but is possible.

Table 4. Combinations of the basic and inverse filters.

| Explicit filters | | Compact filters with $a_1 = 0.4$ | |
Filtering	Deconvolution	Filtering	Deconvolution
$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{FD4}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{FD2}}^{-1}[\mathbf{G}_{\text{FD4}}\mathbf{f}_2]$	$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{CD4}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{CD2}}^{-1}[\mathbf{G}_{\text{CD4}}\mathbf{f}_2]$
$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{FD6}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{FD2}}^{-1}[\mathbf{G}_{\text{FD6}}\mathbf{f}_2]$	$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{CD6}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{CD2}}^{-1}[\mathbf{G}_{\text{CD6}}\mathbf{f}_2]$
$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{FD8}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{FD2}}^{-1}[\mathbf{G}_{\text{FD8}}\mathbf{f}_2]$	$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{CD8}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{CD2}}^{-1}[\mathbf{G}_{\text{CD8}}\mathbf{f}_2]$
$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{FD2}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{FD4}}^{-1}[\mathbf{G}_{\text{FD2}}\mathbf{f}_2]$	$\bar{\mathbf{f}}_2 = \mathbf{G}_{\text{CD2}}\mathbf{f}_2$	$\mathbf{f}_2^* = \mathbf{G}_{\text{CD4}}^{-1}[\mathbf{G}_{\text{CD2}}\mathbf{f}_2]$
	$\mathbf{f}_2^* = \mathbf{G}_{\text{FD6}}^{-1}[\mathbf{G}_{\text{FD2}}\mathbf{f}_2]$		$\mathbf{f}_2^* = \mathbf{G}_{\text{CD6}}^{-1}[\mathbf{G}_{\text{CD2}}\mathbf{f}_2]$
	$\mathbf{f}_2^* = \mathbf{G}_{\text{FD8}}^{-1}[\mathbf{G}_{\text{FD2}}\mathbf{f}_2]$		$\mathbf{f}_2^* = \mathbf{G}_{\text{CD8}}^{-1}[\mathbf{G}_{\text{CD2}}\mathbf{f}_2]$
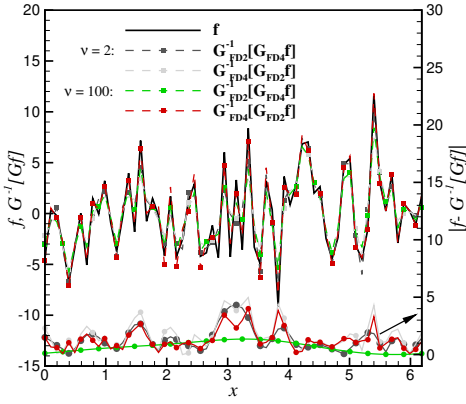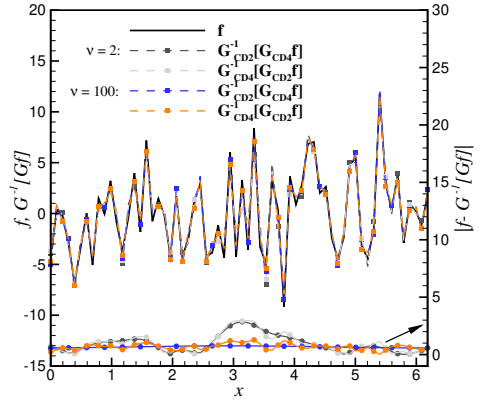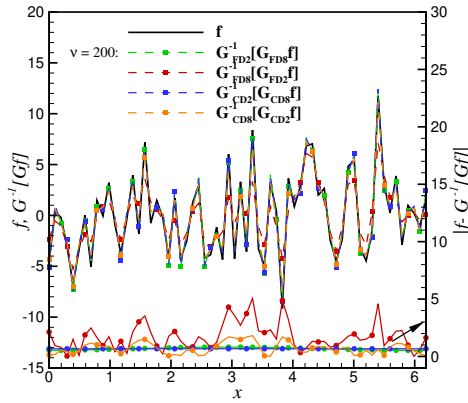
Fig. 8. Comparison of the function f_2 and its deconvolved counterpart f_2^* obtained using the explicit and compact filters different than the basic one for $\nu = 2, 100$ (a, b) and $\nu = 200$ (c).
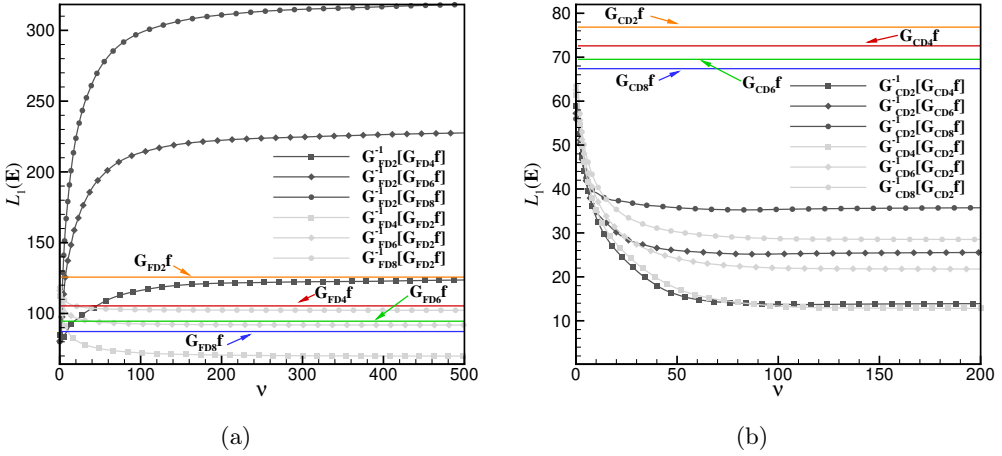
FIG. 9. $L_1(\mathbf{E})$ for the function f_2 computed applying various combinations of FD (a) and CD (b) filters.

Figure 8 shows the results of the deconvolution performed with the filters $\mathbf{G}_{\text{FD2}}$, $\mathbf{G}_{\text{FD4}}$ and $\mathbf{G}_{\text{CD2}}$, $\mathbf{G}_{\text{CD4}}$. It can be clearly seen that the deconvolved function $\mathbf{f}^*$ differs from $\mathbf{f}$, and surprisingly, in some cases, the discrepancies are more pronounced when ν is large. Both the explicit and compact filters show the same tendency, yet, in the latter case, the error of the deconvolution seems smaller. Its detailed comparisons are presented in Fig. 9 along with the colour lines showing $L_1(\mathbf{f} - \bar{\mathbf{f}})$. Note that these lines are independent of ν and represent the reference level. The first very important observation is that the deconvolution does not always lead to $\mathbf{f}^*$ close to $\mathbf{f}$. It turns out that in some cases, the filtered function $\bar{\mathbf{f}}$ is closer to $\mathbf{f}$ than $\mathbf{f}^*$ is. For instance, if explicit filters are used, deconvolution is beneficial only when the basic filter is of a lower order than the one used as the inverse filter. It can be seen in Fig. 9 that the grey lines representing $L_1(\mathbf{f} - \mathbf{f}^*)$ with $\mathbf{f}^*$ deconvolved by $\mathbf{G}_{\text{FD4}}^{-1}$, $\mathbf{G}_{\text{FD6}}^{-1}$, $\mathbf{G}_{\text{FD8}}^{-1}$ are at the lower level than the orange line corresponding to $L_1(\mathbf{f} - \bar{\mathbf{f}})$. In all remaining cases, i.e., when the basic filter is of a higher order than the inverse filter, the deconvolution causes the departure of $\mathbf{f}^*$ from $\mathbf{f}$. The situation is different when the compact filters are used as in this case any combination of $\mathbf{G}^{-1}/\mathbf{G}$ leads to $L_1(\mathbf{f} - \mathbf{f}^*) < L_1(\mathbf{f} - \bar{\mathbf{f}})$. Additional observations concerning the $L_1(\mathbf{E})$ are the following:

1. $L_1(\mathbf{E})$ increases when the deconvolution is performed using $\mathbf{G}^{-1}$ based on $\mathbf{G}$ significantly different than $\mathbf{G}$ applied for the filtering, e.g., $\mathbf{G}_{\text{FD2}}^{-1}[\mathbf{G}_{\text{FD8}}\mathbf{f_2}]$ produces much worse results than $\mathbf{G}_{\text{FD2}}^{-1}[\mathbf{G}_{\text{FD4}}\mathbf{f_2}]$;

2. $L_1(\mathbf{E})$ increases when the deconvolution is performed using $\mathbf{G}^{-1}$ based on $\mathbf{G}$ of the smaller order than $\mathbf{G}$ applied for the filtering, e.g., $\mathbf{G}_{\mathrm{FD2}}^{-1}[\mathbf{G}_{\mathrm{FD8}}\mathbf{f_2}]$ is evidently less accurate than $\mathbf{G}_{\mathrm{FD8}}^{-1}[\mathbf{G}_{\mathrm{FD2}}\mathbf{f}]$ and this can be seen also in Fig. 8c; it is worth noticing that this property holds also for the compact filters, though to a lesser extent.

Besides, taking into account point 2 above, we observe that when the deconvolution is performed using $\mathbf{G}^{-1}$ based on $\mathbf{G}$ of a smaller order than the basic filter $\mathbf{G}$ the error $L_1(\mathbf{E})$ first decreases for small ν and then increases. On the contrary, if $\mathbf{G}^{-1}$ is based on $\mathbf{G}$ of a higher order than the basic filter $\mathbf{G}$ we see that $L_1(\mathbf{E})$ drops with ν. However, in both cases $L_1(\mathbf{E})$ saturates at some value of ν and its further increase has no influence on the results. This is very well seen for the compact filters (see Fig. 9b), where in all cases $L_1(\mathbf{E})$ stagnates from approximately $\nu = 50$. Trying to find the reason for the difference we compare in Fig. 10 and 11 the Fourier transforms of the function $\mathcal{F}[\mathbf{f}]$, its deconvolved counterpart $\mathcal{F}[\mathbf{f}^*] = \mathcal{F}[\mathbf{G}^{-1}[\mathbf{Gf}]]$, and the filtered function $\mathcal{F}[\bar{\mathbf{f}}]$ (colour lines). It can be readily seen that if during the deconvolution the inverse filter is of a higher order than the basic one (Fig. 10a and 11a), the deconvolution error decreases with increasing ν, but only to some level, and then it remains constant. Comparing the amplitudes of $\mathcal{F}[\mathbf{f}]$ with the ones resulting from $\mathcal{F}[\mathbf{f}^*]$ it is seen that the latter are smaller over the entire range of ω. The differences are particularly large at $\omega \to \pi$, especially for the explicit filters. However, if one compares the amplitudes of $\mathcal{F}[\mathbf{f}^*]$ and $\mathcal{F}[\bar{\mathbf{f}}]$, it is evident that the former are closer to $\mathcal{F}[\mathbf{f}]$ that confirms the results presented in Fig. 9. The situation changes
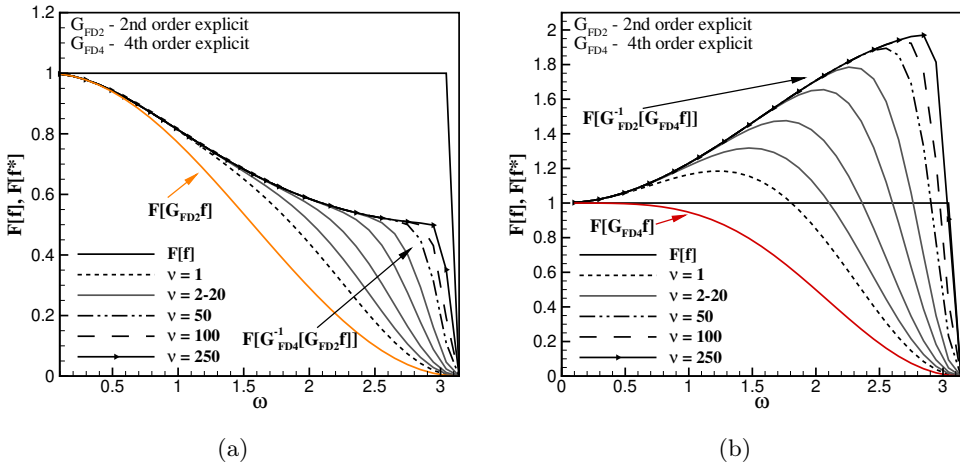

(a)   (b)

FIG. 10. Convergence of the deconvolution process in spectral space for the function f_2 applying the FD filters different than the basic ones. The colour lines denote the spectra of the filtered function.

when the deconvolved filter $\mathbf{G}^{-1}$ is of the lower order, see Figs. 10b and 11b. In this case, when ν is small, the amplitudes of $\mathcal{F}[\mathbf{f}^*]$ are larger than $\mathcal{F}[\mathbf{f}]$ for small ω and smaller for $\omega \to \pi$. However, the convergent solution characterises the amplification of the amplitudes over the entire range of ω. This is why the $L_1(\mathbf{E})$ in Fig. 9b grows for increasing ν values. One should note that when the explicit filters are used the error is particularly large. In fact, the amplitudes of the filtered function $\mathcal{F}[\bar{\mathbf{f}}]$ are closer to $\mathcal{F}[\mathbf{f}]$ than these resulting from $\mathcal{F}[\mathbf{f}^*]$. This means that the deconvolution with large ν makes the function $\mathbf{f}^*$ more different from $\mathbf{f}$ than the filtered function. In conclusion, in the case of explicit filters, when $\mathbf{G}^{-1}$ is constructed based on $\mathbf{G}$ of the lower order than the basic filter, deconvolution is beneficial only when ν is small. However, its precise value cannot be determined a priori without knowing the original function. When the compact filters are employed the amplification of $\mathcal{F}[\mathbf{f}^*]$ is not so strong, as can be seen in Fig. 9b, and deconvolution is beneficial for any combination of $\mathbf{G}_{\mathrm{CD2}}$, $\mathbf{G}_{\mathrm{CD4}}$, $\mathbf{G}_{\mathrm{CD8}}$.
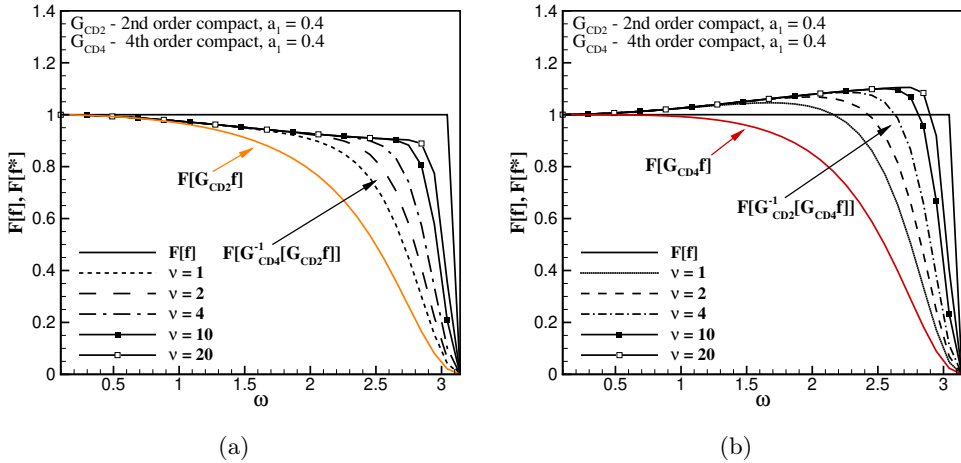

Fig. 11. Convergence of the deconvolution process in spectral space for the function f_2 applying the CD filters different than the basic ones with the parameters $a_1 = 0.4$. The colour lines denote the spectra of the filtered function.

Referring to practical applications, the scenarios when the order of $\mathbf{G}^{-1}$ is obtained based on the lower order than the actual $\mathbf{G}$ seem worse. In acoustics, this kind of deconvolution would lead to the amplification of high-frequency tones usually heard by the human ear as a squeak. In deblurring, it would create the contours of pictures wavier than the original ones. In the modelling of turbulent flows, such a deconvolution would be responsible for adding unphysical energy to the small-scale motion, which in numerical simulations would very likely result in

instability. In the next section, we analyse how the present findings translate to a 2D case and concentrate on the energy content of the original and deconvolved function.

4.2. 2D case

To analyse the deconvolution process in a two-dimensional (2D) domain we define the following test function

$$(4.6) \qquad f(x_i, y_j) = \sin(x_i + y_j)\cos(x_i + 2)$$
$$+ \cos(x_i - y_j)\sin(x_i + 2) + \sin(x_i + y_j)\sin(x_i + y_j)$$
$$+ \sum_{k=1}^{K} [\varphi \sin(k \cdot x_i) + \psi \cos(k \cdot y_j)],$$

where $x_i = (i-1)h$, $y_j = (j-1)h$, $1 \le i, j \le N$, are the discrete nodes in the two-dimensional periodic domain $2\pi \times 2\pi$, $h = 2\pi/N$ is the space between the nodes, φ, ψ are random amplitudes in the range $0 \le \varphi, \psi < 0.05$, $N = 64$. In 2D, the filtering and deconvolution are performed by applying the filter G and accompanying it G^{-1} successively in separate directions. Thus, by analogy with Eq. (1.1) the filtering operation is defined as

$$(4.7) \quad \overline{f}(x,y) = G_y * [G_x * f(x,y)] = \int_{-\infty}^{\infty} G_y(y - \gamma) \int_{-\infty}^{\infty} f(x,y) G_x(x - \tau) \, \mathrm{d}\tau \, \mathrm{d}\gamma.$$

Having the filters defined in the matrix form, one can write

$$(4.8) \qquad \overline{\mathbf{f}} = \mathbf{G}_y [\mathbf{G}_x \mathbf{f}]^T,$$

where $\mathbf{f}$ and $\overline{\mathbf{f}}$ are the matrices consisting of the $f(x_i, y_j)$ and $\overline{f}(x_i, y_j)$ in particular nodes. The nodal values along the x-coordinate are stored in the columns of $\mathbf{f}$ and $\overline{\mathbf{f}}$. The matrices $\mathbf{G}_x$ and $\mathbf{G}_y$ are the filters defined in Section 3.1. They have the same form along the 'x' and 'y' coordinates. The deconvolution process is carried out by changing the order of the applied approximate filters so that the deconvolved function is obtained from

$$(4.9) \qquad \mathbf{f}^* \approx \mathbf{G}_x^{-1} [\mathbf{G}_y^{-1} \overline{\mathbf{f}}^T].$$

4.2.1. Error measure. In 2D, rather than using L_1 error measure, we analyse the accuracy of the deconvolution using the global measure of the error in terms of the function fluctuations with respect to the actually computed mean value $\overline{f} = \frac{1}{N^2} \sum_{i,j} f(x_i, y_j)$. Thus, we define the parameter

$$(4.10) \qquad \sigma(f) = \sum_{i,j} \left(f'(x_i, y_j) \right)^2 = \sum_{i,j} \left(f(x_i, y_j) - \overline{f} \right)^2,$$

which can be interpreted as the energy of the fluctuations of the test function, and calculate the relative error of the deconvolution as

$$E(f^*) = \frac{\sigma(f) - \sigma(f^*)}{\sigma(f)} \cdot 100\%.$$

With such an error definition we are able to show whether the deconvolved 2D function is 'smoother' ($E(f^*) > 0$) or 'wavier' ($E(f^*) < 0$) with respect to the original function. For instance, in deblurring applications, if $E(f^*) > 0$ the obtained pictures would be more blurred than the original one.

4.2.2. Results of the deconvolution in 2D with the filter different than the basic one.

Figure 12 shows the contours of the function $f(x, y)$ and its filtered counterpart $\overline{f}(x, y)$ obtained with the CD filter $\mathbf{G}_{CD2}$ with the parameter $a_1 = 0.4$. The filtered function is smoother than the original one, yet, the main large-scale function shapes are very similar. This is because the 2D filtering, similar to that in the 1D case, acts mainly in the range of high frequencies. Figure 13 shows the parameter $E(f^*)$ for various combinations of $\mathbf{G}^{-1}/\mathbf{G}$ for the explicit and compact filters. It can be seen that when the latter is used, the deconvolution process converges faster and is more accurate. By applying the explicit filters, the deconvolution makes sense only if the inverse filter originates from a filter of a higher order than the basic one. If this is not the case, the filtered function $\overline{f}$ better reflects the original f than the deconvolved function. On the other hand, when compact filters are used, deconvolution always brings the function f^* closer to the true one. However, the important aspect is which filter is used as the
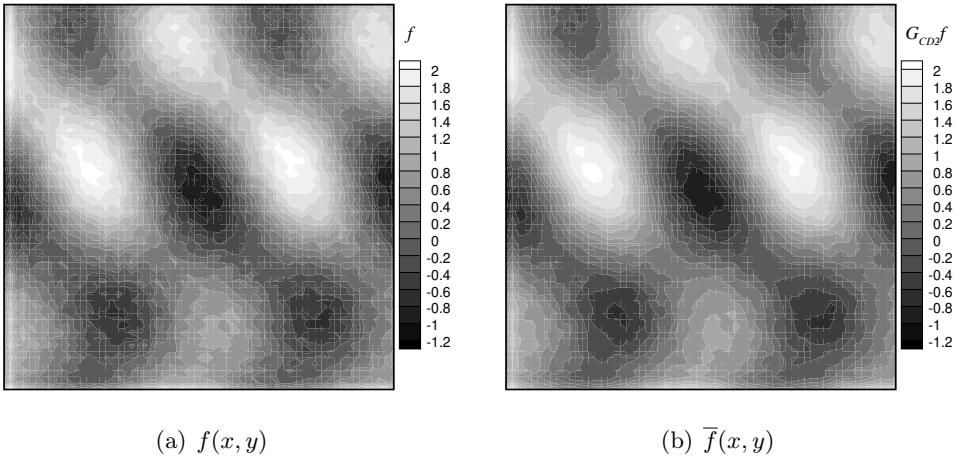

(a) $f(x, y)$ (b) $\overline{f}(x, y)$

Fig. 12. Function $f(x, y)$ (a) and $\overline{f}(x, y)$ (b) obtained using $\mathbf{G}_{CD2}$ with $a_1 = 0.4$.
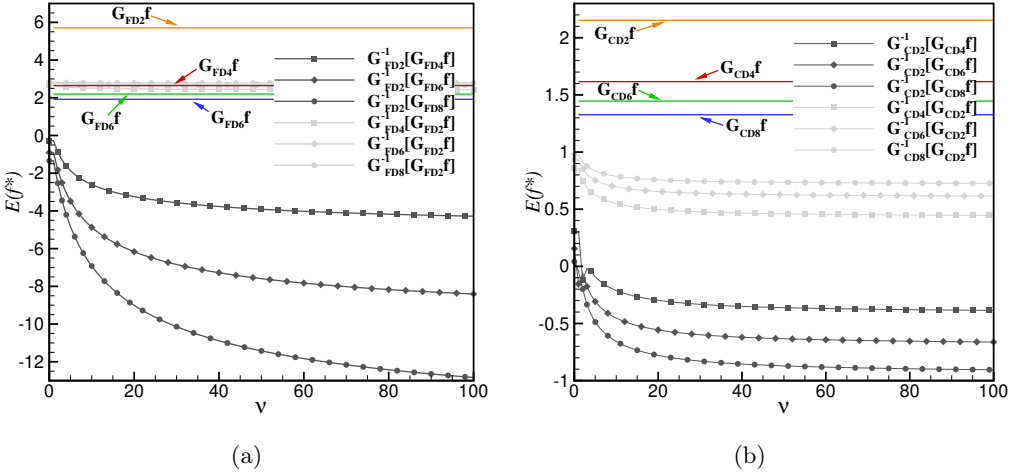
Fig. 13. Parameter $E(f^*)$ for various combinations of $\mathbf{G}^{-1}$ and $\mathbf{G}$ for the explicit (a) and compact (b) filters.

inverse one. The negative values of $E(f^*)$ mean that the deconvolution operation artificially increases the energy of the fluctuations $f'(x_i, y_j)$. It is seen in Fig. 13 that this is the case when the inverse filter is of a lower order than the basic one. Table 5 shows $E(f^*)$ for different combinations of the compact filters. For large ν for which the deconvolution process is convergent, the $E(f^*)$ values in the cases CD4/CD2, CD6/CD2 and CD8/CD2 are positive, whereas for CD2/CD4, CD2/CD6 and CD2/CD8 they are negative. Although the differences between $|E(f^*)|$ are not large one should emphasise that the sign of $E(f^*)$ reveals the opposite character of deconvolution. If $E(f^*) < 0$, the small-scale phenomena are amplified, whereas for $E(f^*) > 0$ they are dampened.

Table 5. Deconvolution error $E(f^*)$ at the selected ν values for different combinations of the compact filters with $a_1 = 0.4$.

ν value	CD4/CD2	CD6/CD2	CD8/CD2	CD2/CD4	CD2/CD6	CD2/CD8
$\nu = 1$	8.57×10^{-1}	9.30×10^{-1}	9.76×10^{-1}	3.05×10^{-1}	1.54×10^{-1}	4.12×10^{-2}
$\nu = 2$	7.50×10^{-1}	8.49×10^{-1}	9.11×10^{-1}	1.20×10^{-1}	-5.86×10^{-2}	-1.99×10^{-1}
$\nu = 4$	6.51×10^{-1}	7.73×10^{-1}	8.51×10^{-1}	-4.43×10^{-2}	-2.52×10^{-1}	-4.22×10^{-1}
$\nu = 10$	5.49×10^{-1}	6.95×10^{-1}	7.88×10^{-1}	-2.11×10^{-1}	-4.52×10^{-1}	-6.56×10^{-1}
$\nu = 20$	4.97×10^{-1}	6.54×10^{-1}	7.56×10^{-1}	-2.97×10^{-1}	-5.56×10^{-1}	-7.79×10^{-1}
$\nu = 50$	4.59×10^{-1}	6.26×10^{-1}	7.34×10^{-1}	-3.61×10^{-1}	-6.33×10^{-1}	-8.70×10^{-1}
$\nu = 100$	4.45×10^{-1}	6.15×10^{-1}	7.26×10^{-1}	-3.85×10^{-1}	-6.62×10^{-1}	-9.05×10^{-1}

4.3. 2D decaying homogenous isotropic turbulence

In the last test case, we focus on a problem typical of fluid mechanics, i.e., modelling of decaying homogeneous isotropic turbulence (HIT). We apply the large eddy simulation (LES) method in which we use the ADM to directly retrieve a part of the small sub-fliter scales close to the filter cutoff. In LES, turbulent flow scales are divided into the large scales that are directly solved on a numerical mesh, and the small scales, called a sub-grid, the effect of which on the resolved scales is modelled. The separation of scales is achieved by a low-pass spatial filtering procedure defined as [86, 87]

$$(4.12) \qquad \overline{f}(\mathbf{x}, t) = \int_{\Omega} f(\mathbf{x}', t) G(\mathbf{x} - \mathbf{x}') \, d\mathbf{x}',$$

where Ω is the flow domain, f denotes an arbitrary variable, and $G(\mathbf{x})$ is the filter function satisfying the condition $\int_{\Omega} G(\mathbf{x} - \mathbf{x}') \, d\mathbf{x}' = 1$. As in [88], we consider a 2D configuration which can only be regarded as a simplified model of a 'real' 3D turbulent velocity field. However, this is sufficient for verifying different ADM variants. In 2D cases, the Navier–Stokes equations in the vorticity–stream function formulation (ω, ψ) in the non-dimensional form are defined as:

$$(4.13) \qquad \frac{\partial \omega}{\partial t} + u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} = \frac{1}{\mathrm{Re}} \left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right),$$

$$(4.14) \qquad \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = -\omega,$$

where $u = \frac{\partial \psi}{\partial y}$, $v = -\frac{\partial \psi}{\partial x}$ are the velocity components, and Re is the Reynolds number. Applying the filter (4.12) to (4.13) and (4.14) gives:

$$(4.15) \qquad \frac{\partial \overline{\omega}}{\partial t} + \overline{u} \frac{\partial \overline{\omega}}{\partial x} + \overline{v} \frac{\partial \overline{\omega}}{\partial y} = \frac{1}{\mathrm{Re}} \left(\frac{\partial^2 \overline{\omega}}{\partial x^2} + \frac{\partial^2 \overline{\omega}}{\partial y^2} \right) - \tau_{\mathrm{SGS}}^{\omega},$$

$$(4.16) \qquad \frac{\partial^2 \overline{\psi}}{\partial x^2} + \frac{\partial^2 \overline{\psi}}{\partial y^2} = -\overline{\omega}.$$

The term $\tau_{\mathrm{SGS}}^{\omega}$, represents the contribution of the unresolved small scale [86, 87]. It originates from the filtering of the non-linear convective terms in (4.13). It has the following form:

$$(4.17) \qquad \tau_{\mathrm{SGS}}^{\omega} = \overline{u \frac{\partial \omega}{\partial x}} + \overline{v \frac{\partial \omega}{\partial y}} - \overline{u} \frac{\partial \overline{\omega}}{\partial x} + \overline{v} \frac{\partial \overline{\omega}}{\partial y}.$$

Neither the unfiltered ω nor the velocity vector $U = [u, v]$ are known, and hence, $\tau_{\mathrm{SGS}}^{\omega}$ has to be modelled. A typical approach relies on modelling $\tau_{\mathrm{SGS}}^{\omega}$

using the so-called sub-grid viscosity models. There is a number of them, e.g., the Smagorinsky, dynamic or Vreman models [86, 87]. Here, we use the ADM approach to compute $\tau_{\mathrm{SGS}}^{\omega}$ directly using the deconvolved variables and compare the results with those obtained using the Smagorinsky model and also with the solutions where $\tau_{\mathrm{SGS}}^{\omega}$ was omitted (no-model approach). Additionally, we perform direct numerical simulations (DNS) on a very dense mesh, which allows us to univocally assess the solution accuracy of the particular approach. As defined in (2.5), the deconvolved variables (ω^* and velocity vector U^*) in ADM are calculated from:

$$(4.18) \qquad \mathbf{f}_\nu^* = \sum_{n=0}^{\nu} [\mathbf{I} - \mathbf{G}]^n \overline{\mathbf{f}}, \quad \mathbf{f} \approx \mathbf{f}_\nu^*.$$

Knowing ω^* and U^* we substitute them into (4.17) and filter the first two terms. Consequently, the last two terms in (4.17) and the convective terms in (4.13) cancel out. Unlike Stolz et al. [51], we do not add the relaxation term. With $\nu = 0$, the present approach would correspond to the model formulated by Layton and Lewandowski [90], which characterises excellent stability. With $\nu > 0$, the stability of LES-ADM has been proven in [91, 92] on the theoretical ground and also in computations [59, 61].

The solution procedure of (4.13)–(4.16) can be found in [89] and is not repeated here. In short, we apply the 4th order Runge–Kutta method for the time integration and two spatial discretisation methods, the Fourier pseudo-spectral (PS) [93] and the 6th order compact difference (CD) [84, 89]. The former is used to perform direct numerical simulations (DNS) on a dense computational mesh to have exemplary data for LES, and also to assess the impact of the discretisation method on the solutions. In LES, the exact form of the filter function $G(\mathbf{x})$ is unknown. The resolved flow scales (RFS) are determined by the mesh, which implicitly filters out the scales shorter than the mesh size. The RSF are sometimes subjected to an explicit filtering ($\tilde{f} = G_e * \overline{f}$, see [45–47]) but we do not apply this here. The RSF are filtered by a filter induced by the numerical discretisation of the convective and diffusive terms ($G_{I,c}(\mathbf{x})$, $G_{I,d}(\mathbf{x})$) [48, 49]. To our best knowledge, the formulation of a single induced filter $G_I(\mathbf{x})$ that would reflect the combination of $G_{I,c}(\mathbf{x})$, $G_{I,d}(\mathbf{x})$ has not been proposed so far. Also, in this work, we do not attempt to precisely define $G_I(\mathbf{x})$. Instead, we calculate the inverse filter based on the explicit and compact filters defined in the previous sections. We demonstrate how the choice of the filter affects the simulation results. We compare the solutions obtained by applying the filters $\mathbf{G}_{\mathrm{FD2}}$, $\mathbf{G}_{\mathrm{CD2}}$ ($a_1 = 0.4$) and $\mathbf{G}_{\mathrm{FD4}}$, $\mathbf{G}_{\mathrm{CD4}}$ ($a_1 = 0.4$).

The computational domain is a 2D box ($2\pi \times 2\pi$) with periodic boundary conditions on all sides. Following [88] we initialise the flow field based on an energy spectrum given as

$$(4.19) \qquad E(k) = \frac{1}{2} \frac{(2s+1)^{s+1}}{2^s s!} \frac{1}{k_p} \left(\frac{k}{k_p} \right)^{2s+1} \exp\left[-\left(s + \frac{1}{2} \right) \left(\frac{k}{k_p} \right)^2 \right],$$

where $k = |\mathbf{k}| = \sqrt{k_x^2 + k_y^2}$ is a wave number module, k_p is the wave number at which the spectrum has the maximum value, and s is a shape parameter. Here, we assume $k_p = 12$ and $s = 3$. The initial vorticity in the spectral space is defined as $\hat{\omega}(\mathbf{k}) = \sqrt{k/\pi E(k)} \exp^{i(\xi(\mathbf{k})+\eta(\mathbf{k}))}$, where $\xi(k)$, $\eta(k)$ are independent random variables, which introduce a phase shift between the particular modes. Taking a 2D inverse FFT gives the vorticity in the physical space, i.e., $\omega(\mathbf{x}) = \mathcal{F}_x^{-1}[\mathcal{F}_y^{-1}[\hat{\omega}(\mathbf{k})]]$, which is then used to compute ψ from (4.16) and the velocity components.
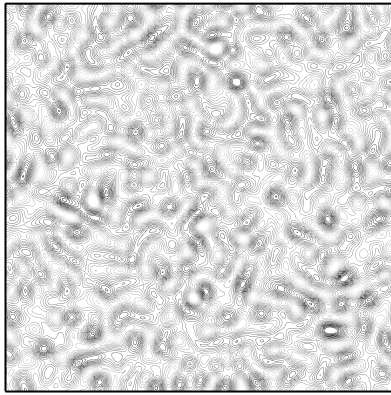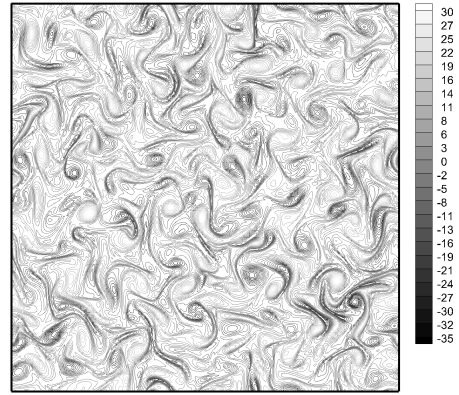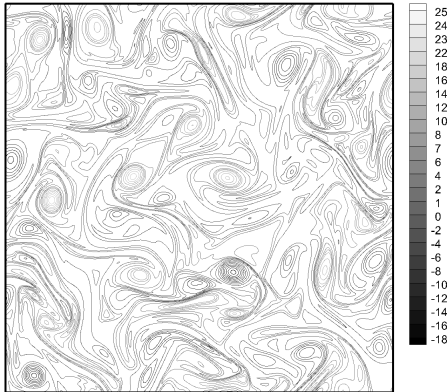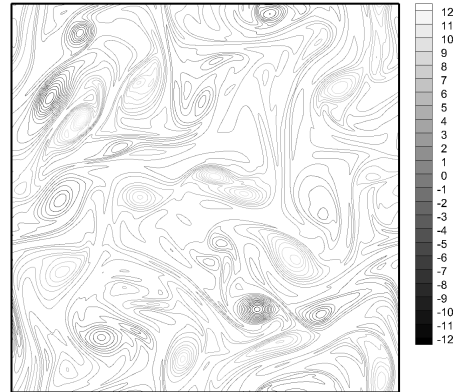

(a) $t = 0$ (b) $t = 1$

(c) $t = 5$ (d) $t = 6$

FIG. 14. Contours of the vorticity field obtained on the mesh with 1025^2 nodes using DNS.

The computations were performed for two Reynolds numbers (Re = 1000 and Re = 3000) on two meshes consisting of $N^2 = 1025^2$ nodes for DNS and $N^2 = 257^2$ nodes for LES. Although in [88] it was shown that the mesh with $N^2 = 1025^2$ is sufficiently dense to obtain accurate DNS solutions for Re = 1000 and Re = 3000 we verify this by performing test computations on the mesh composed of $N^2 = 513^2$ nodes. In all the cases, the initial velocity field was generated on the densest mesh and then projected on the coarser ones. To avoid the impact of time integration errors, the simulations were performed with a small time-step $\Delta t = 1 \times 10^{-6}$, which was two orders of magnitude lower than the stability limit on the densest mesh. The preliminary tests with $\Delta t = 1 \times 10^{-5}$ showed that the time-integration errors have practically no effect on the results. The simulation lasted until $t = 10$. Figure 14 shows the contours of the vorticity field in selected time instances obtained in DNS. Note that the minimum and maximum ranges in the legends change. This is because the vorticity field is quickly dampened over time, and their contours would be almost invisible if they were plotted in the same range of values. In addition to the damping of the vorticity magnitude, one can readily notice that the sizes of the vortices also change. Initial small structures are simultaneously dissipated and merged into larger ones.

The impact of the filter used in ADM on the solution accuracy is assessed based on the following quantities:

1. Instantaneous mean energy

$$(4.20) \qquad E(t) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{1}{2} \left| \overline{U}(x_i, y_j, t) \right|^2 .$$

2. Energy spectrum

$$(4.21) \qquad \widehat{E}(k,t) = 2\pi k \left\langle \frac{1}{2} \left| \widehat{\overline{U}}(\mathbf{k},t) \right|^2 \right\rangle,$$

where the operator $\langle \cdot \rangle$ denotes averaging in a thin shell of wave numbers ($k = k \pm 0.5$).

3. Spatial variance (μ_2) and two higher moments, skewness (μ_3) and kurtosis (μ_4) defined as:

$$(4.22) \qquad \mu_n(\overline{\phi}, t) = \frac{\frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left[\overline{\phi}(x_i, y_j, t) - \mu_1(\overline{\phi}, t) \right]^n}{\mu_2(t)^{\frac{n}{2}}},$$

$$(4.23) \qquad \mu_2(\overline{\phi}, t) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left[\overline{\phi}(x_i, y_j, t) - \mu_1(\overline{\phi}, t) \right]^2 ,$$

$$(4.24) \qquad \mu_1(\overline{\phi}, t) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \overline{\phi}(x_i, y_j, t),$$

where $\overline{\phi}(x_i, y_j, t)$ represents a general variable at the time moment t in the node x_i, y_j.

Additionally, we define the time dependent solution error as

$$(4.25) \qquad \text{Error} = \frac{1}{N_t} \sqrt{\sum_{i=1}^{N_t} \left(F(t)_i^{DNS} - F(t)_i^{LES} \right)^2},$$

where N_t is the number of time steps from $t = 0$ and $F(t)$ represents the above-defined quantities $(E(t), \mu_n)$. Figure 15 shows the comparisons of spectra $\widehat{E}(k, t)$ obtained on the meshes with $N^2 = 1025^2$ and $N^2 = 513^2$ applying the PS and CD methods. The red line represents the initial spectrum (4.19) at $t = 0$ and the rest of the lines and symbols refer to the spectra at $t = 2$ and $t = 10$. The results obtained for Re = 1000 show an excellent agreement both regarding the comparison of discretisation methods as well as the solutions on different meshes. In the inertial range, the spectra are not far from k^{-3} slope to which they should tend when Re $\rightarrow \infty$, according to the Kraichnan–Batchelor–Leith theory. They exhibit better agreement with k^{-3} for $t = 2$ than $t = 10$ and the same behaviour was observed in [88]. This is because of the energy dissipation of small scales that causes bending of the spectrum profiles when the time progresses. Small discrepancies between particular solutions are seen only when the CD method is
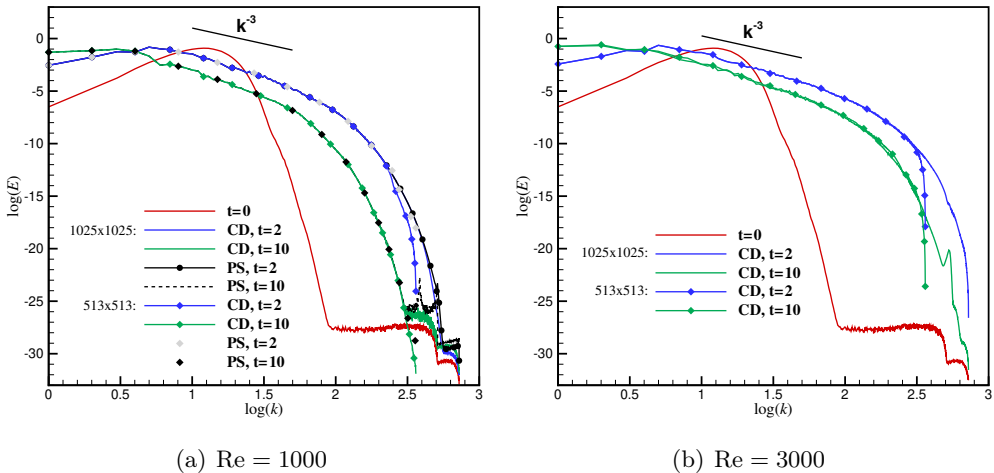

(a) Re = 1000   (b) Re = 3000

Fig. 15. Energy spectra at the time moments $t = 0$, 2 and $t = 10$ obtained in the computations with the use PS and CD discretisation methods.

used, and only close to $k = \sqrt{2}N/2$ (the maximum wave number for which the energy spectrum is computed). Note, that the differences at the level $< 10^{-20}$ are caused more by the round-off error than the mesh density and the discretisation method. For Re = 3000, the PS method diverges because of the aliasing errors to which the PS method is particularly prone [93]. Most likely, applying the de-aliasing techniques would stabilise the solution but we did not attempt to do that. A very good agreement of the solutions obtained with the CD method on two meshes confirms that the results on the denser mesh can be regarded as DNS. In the following comparisons, for both Reynolds numbers, we refer to DNS obtained on the mesh with $N^2 = 1025^2$ nodes using the CD discretisation method.

The energy spectra at $t = 10$ obtained using LES and no-model approach are compared with the DNS data in Fig. 16. The analysis of this figure allows noticing significant differences between particular solutions only in the range of larger wave numbers $(1.5 < \log(k) < 2.25 \rightarrow 32 < k < 181)$. The grid cut-off wave number $k_c = N/2$ for LES is represented by a vertical dashed line. Surprisingly, here the solution obtained with the no-model approach shows the best agreement with DNS. However, as the energy at large wave numbers is small $(\widehat{E}(k) < 10^{-10})$ the observed differences impact only very little on the total kinetic energy (TKE) of the flow. Figure 17 shows a temporal TKE decay. It can be seen that TKE dissipates the fastest when the Smagorinsky model is used. This is, however, not surprising as this model is known to be over dissipative [86, 87]. The use of ADM ensures definitively better results, yet, not much better than the no-model approach. Among the filters applied in ADM, the filter $\mathbf{G}_{\text{FD2}}$ leads to the
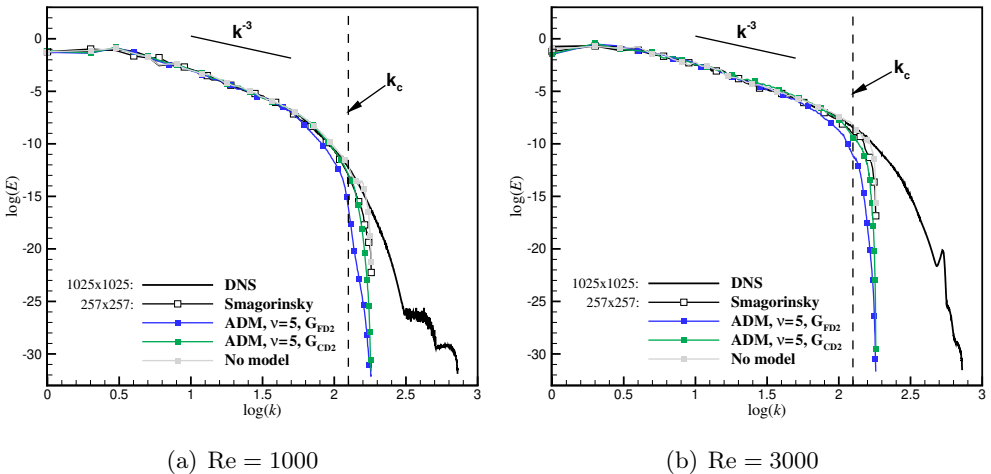

(a) Re = 1000   (b) Re = 3000

Fig. 16. Energy spectra at $t = 10$ obtained for DNS, no-model simulations and LES.

(a) Re = 1000, $\mathbf{G}_{FD2}$, $\mathbf{G}_{CD2}$

(b) Re = 1000, $\mathbf{G}_{FD4}$, $\mathbf{G}_{CD4}$

(c) Re = 3000, $\mathbf{G}_{FD2}$, $\mathbf{G}_{CD2}$

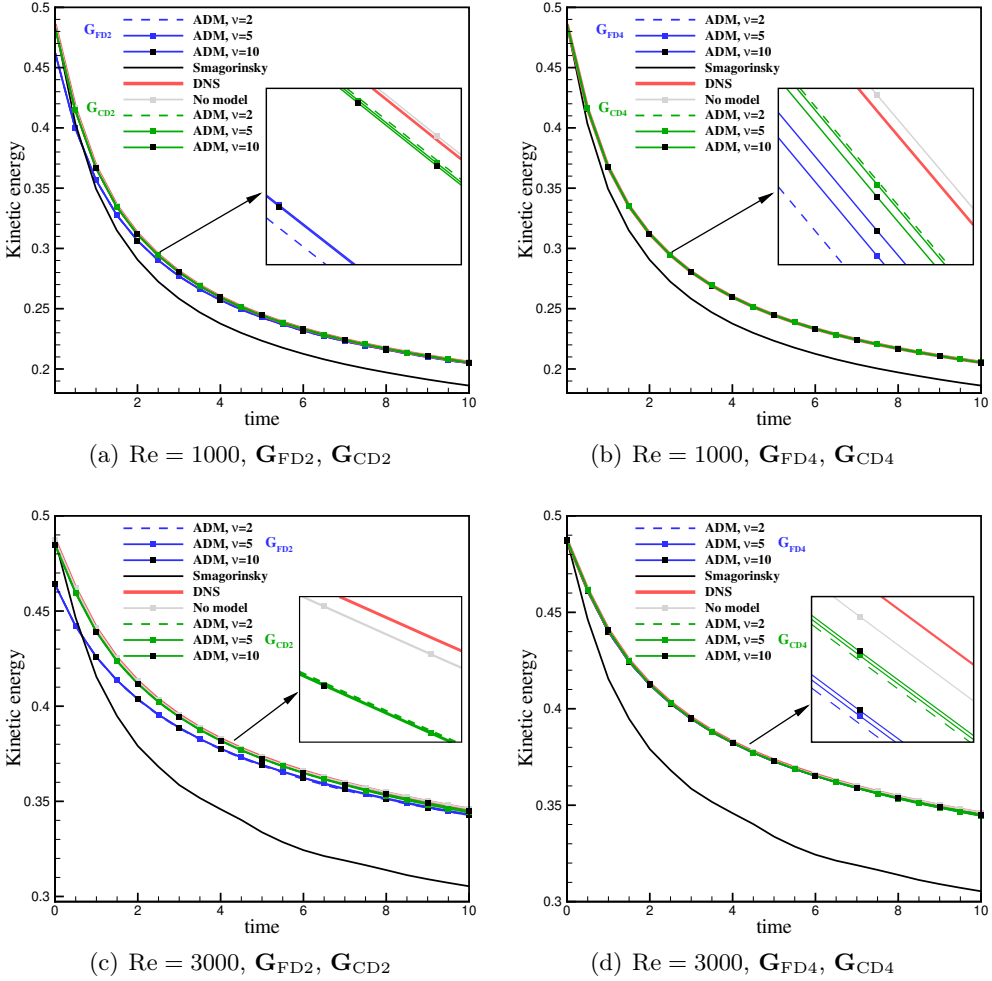(d) Re = 3000, $\mathbf{G}_{FD4}$, $\mathbf{G}_{CD4}$

Fig. 17. Temporal TKE decay obtained in the computations on the meshes with $N^2 = 1025^2$ (DNS) and $N^2 = 257^2$ nodes using various LES sub-grid models and no-model approach.

largest divergence from DNS, particularly in the initial phase of the simulation. Figures 18 and 19 show the temporal evolutions of the higher statistical moments (variance, skewness, and kurtosis) of the u velocity component obtained using DNS, no-model approach, LES with the Smagorinsky model, and LES with ADM ($\nu = 2, 5, 10$). For both Reynolds numbers, LES and no-model results follow the DNS solutions only for some time and then diverge significantly. This is due to the non-linear character of the interactions between the flow scales. A relatively small difference at an early simulation time leads to large differences at a later time.
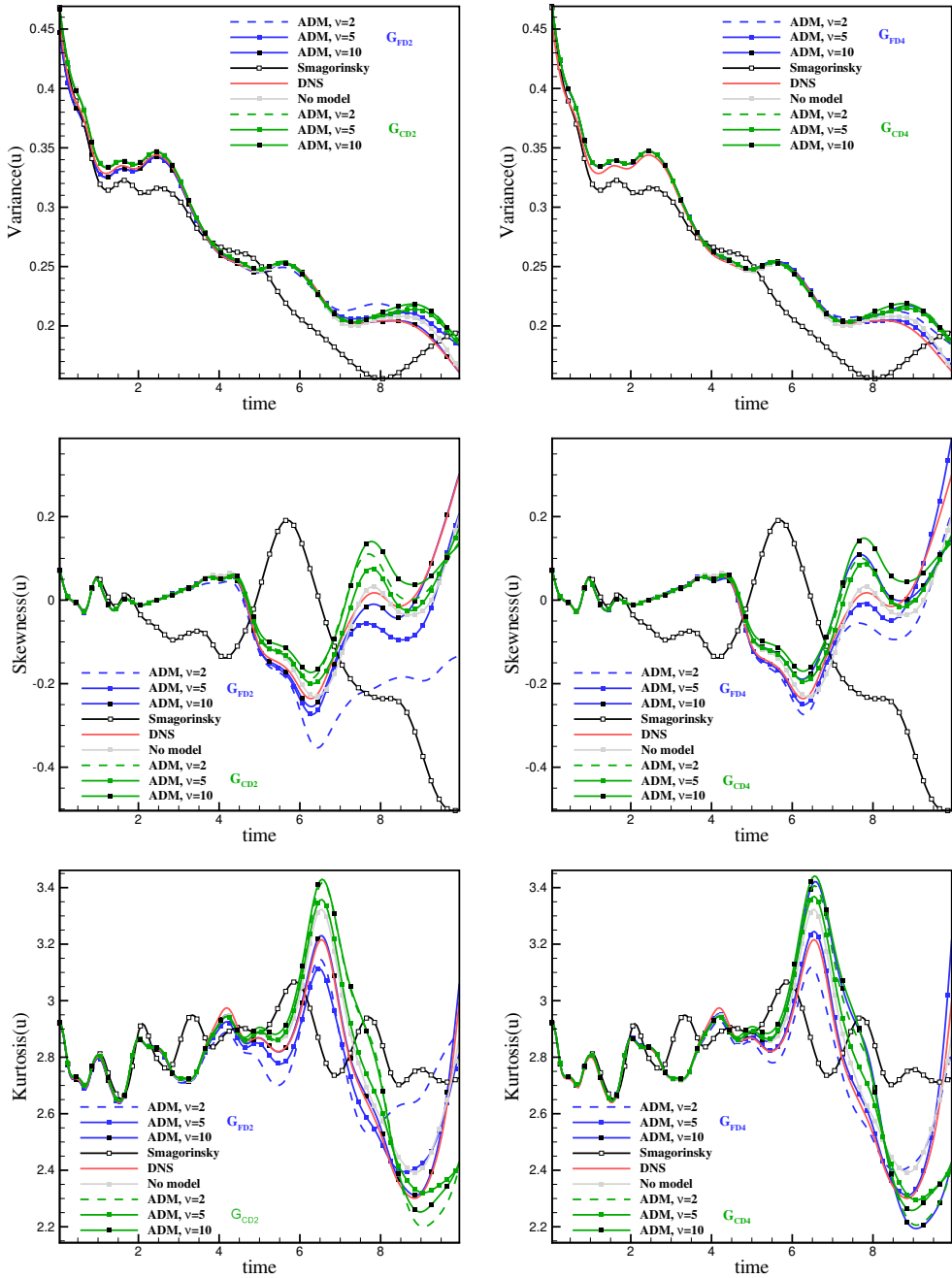
FIG. 18. Evolution of the variance, skewness and kurtosis of the u velocity component for $Re = 1000$. DNS results, no-model and LES results with the Smagorinsky model and ADM with various parameter ν and the filters $\mathbf{G}_{FD2}$, $\mathbf{G}_{FD4}$, $\mathbf{G}_{CD2}$ and $\mathbf{G}_{CD4}$.
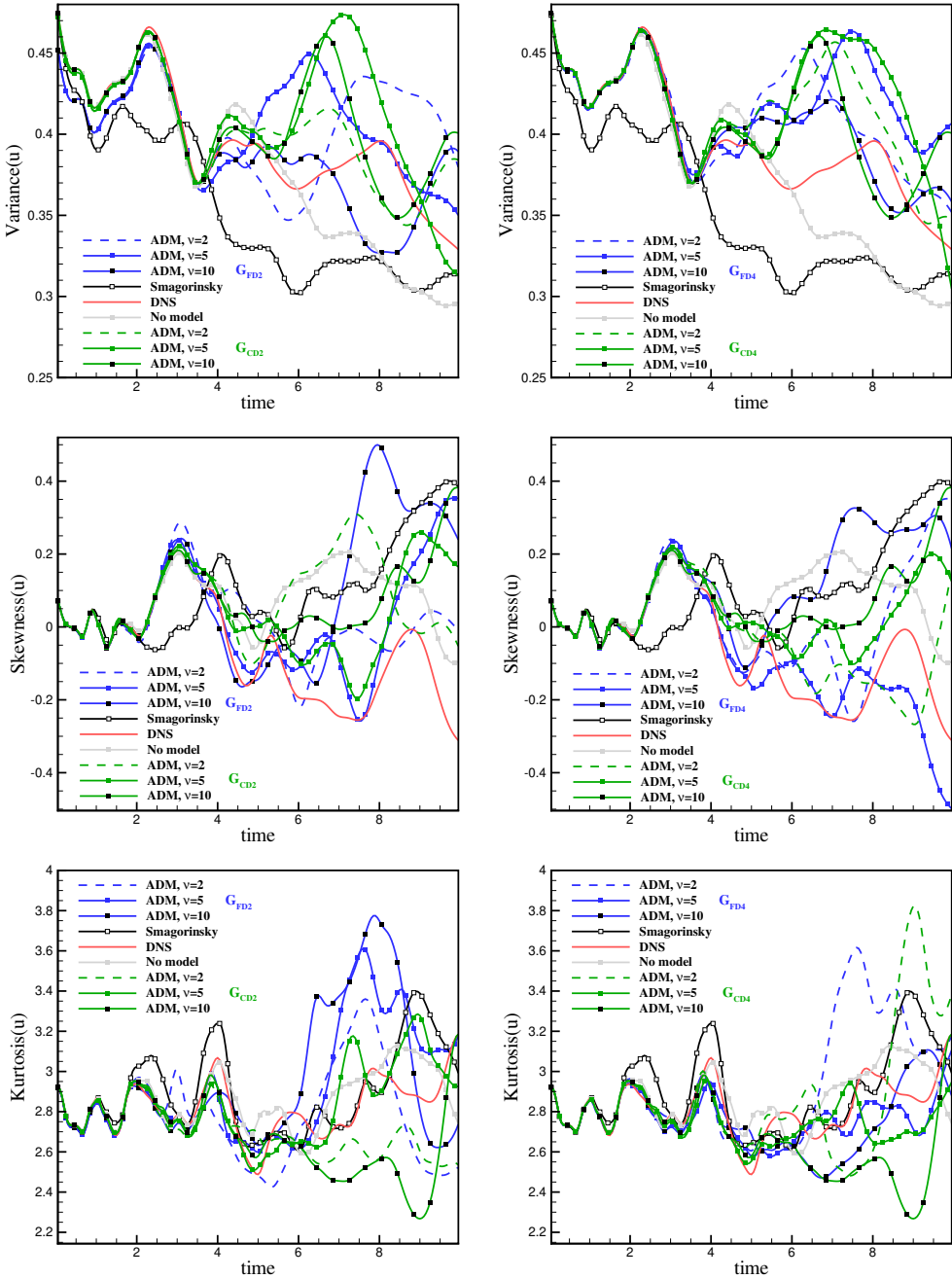
Fig. 19. Evolution of the variance, skewness and kurtosis of the u velocity component for $Re = 3000$. DNS results, no-model and LES results with the Smagorinsky model and ADM with various parameter ν and the filters $\mathbf{G}_{\mathrm{FD2}}$, $\mathbf{G}_{\mathrm{FD4}}$, $\mathbf{G}_{\mathrm{CD2}}$ and $\mathbf{G}_{\mathrm{CD4}}$.

For Re = 1000 they appear from around $t = 5$ and for Re = 3000 from approximately $t = 3$ and in this case the discrepancies are much larger. This is because for Re = 3000, the differences in the solution accuracy of small scales (see $\widehat{E}(k,t)$ in Fig. 15) play a more important role. The solutions obtained with the Smagorinsky model differ the most from the DNS data already at $t = 1$. The results obtained with the no-model approach and ADM are much better. In this case, for Re = 1000, the variance is correctly predicted almost over the entire simulation time. The solutions obtained with the explicit filters are more accurate, especially with $\mathbf{G}_{\text{FD4}}$. The sensitivity of ADM to the number of van Cittert iterations ν seems to be different than expected. One could presume that increasing ν will improve the results, though, in [51] it was said that taking $\nu > 5$ does not affect the solutions. Here, it turns out not always to be the case. For the explicit filters, the change $\nu = 2 \to 5 \to 10$ moves the solution closer to DNS. However, for the compact filters, only the change $\nu = 2 \to 5$ improves the accuracy, while $\nu = 5 \to 10$ makes it even worse than for $\nu = 2$. Tables 6 and 7 show the values of the solution error (4.25) calculated for the velocity components and vorticity at $t = 10$ for Re = 1000 and at $t = 5$ for Re = 3000. ADM solutions with $\nu = 5$ are taken into account. The numbers corresponding to the smallest and largest error for a given quantity are underlined by single and double lines. The no-model solutions presented so far in the figures seemed to be as good as the ones obtained using ADM. In fact, at some time moments, they were even better. However, this is not the case if one takes the overall error. The no-model solutions are more accurate than those obtained with the Smagorinsky model but ADM performs better. In this case, taking into account the majority of the results, the use of the filter $\mathbf{G}_{\text{FD4}}$ leads to the most accurate solutions for

TABLE 6. Comparison of the error (4.25) for Re = 1000. Results for no-model approach and LES with the Smagorinsky model and ADM with $\nu = 5$. The most accurate and the worst results are underlined by a single and double line.

Moment	No-model	Smagorinsky	$\mathbf{G}_{\text{FD2}}$	$\mathbf{G}_{\text{FD4}}$	$\mathbf{G}_{\text{CD2}}$	$\mathbf{G}_{\text{CD4}}$
$\mu_2(u, t = 10)$	0.00036	<u>0.00180</u>	0.00043	<u>0.00030</u>	0.00058	0.00062
$\mu_2(v, t = 10)$	0.00037	<u>0.00191</u>	0.00045	<u>0.00031</u>	0.00056	0.00060
$\mu_2(\omega, t = 10)$	<u>0.18896</u>	0.11024	<u>0.07648</u>	0.14921	0.14666	0.17758
$\mu_3(u, t = 10)$	0.00235	<u>0.01871</u>	0.00324	<u>0.00123</u>	0.00245	0.00271
$\mu_3(v, t = 10)$	<u>0.01479</u>	0.01233	0.00130	<u>0.00088</u>	0.00392	0.00423
$\mu_3(\omega, t = 10)$	0.00101	<u>0.00404</u>	0.00091	<u>0.00021</u>	0.00058	0.00059
$\mu_4(u, t = 10)$	0.00393	<u>0.01434</u>	0.00368	<u>0.00225</u>	0.00750	0.00803
$\mu_4(v, t = 10)$	0.00412	<u>0.01607</u>	0.00350	<u>0.00121</u>	0.00908	0.00993
$\mu_4(\omega, t = 10)$	0.00235	<u>0.02465</u>	0.00225	<u>0.00070</u>	0.00188	0.00207

Table 7. Comparison of the error (4.25) for Re = 3000. Results for no-model approach and LES with the Smagorinsky model and ADM with $\nu = 5$. The most accurate and the worst results are underlined by a single and double line.

Moment	No-model	Smagorinsky	$\mathbf{G}_{\text{FD2}}$	$\mathbf{G}_{\text{FD4}}$	$\mathbf{G}_{\text{CD2}}$	$\mathbf{G}_{\text{CD4}}$
$\mu_2(u, t = 5)$	0.000947	0.003950	0.001221	0.000297	0.000677	0.000544
$\mu_2(v, t = 5)$	0.000950	0.004282	0.001225	0.000305	0.000736	0.000572
$\mu_2(\omega, t = 5)$	0.278206	0.813487	0.955202	0.262578	0.188827	0.075727
$\mu_3(u, t = 5)$	0.005196	0.012857	0.001857	0.001771	0.005291	0.005570
$\mu_3(v, t = 5)$	0.004264	0.011412	0.002199	0.005808	0.004527	0.004250
$\mu_3(\omega, t = 5)$	0.000895	0.003450	0.001621	0.000793	0.001256	0.001538
$\mu_4(u, t = 5)$	0.006150	0.010762	0.006827	0.005295	0.005629	0.005667
$\mu_4(v, t = 5)$	0.006118	0.025810	0.004519	0.002641	0.004030	0.003956
$\mu_4(\omega, t = 5)$	0.005150	0.018319	0.022696	0.015865	0.011288	0.009230

both Reynolds numbers. One can presume that this filter approximates the filter induced by the discretisation in the best way.

5. Conclusions

The paper presented a comprehensive analysis of the approximate deconvolution method based on the van Cittert series and high-order explicit and compact filters. It has been shown that the van Cittert deconvolution method converges, i.e., it leads to the inverse filter that recovers the original function/signal from the filtered one, significantly faster when the high-order filters characterised by a flatter transfer function are used. This is attributed to the fact that the filtered function resulting from the filtering is less different from the original one, and thus, easier to be deconvolved. The analysis of the deconvolution process in spectral space showed that as long as the full convergence is not reached, the amplitudes of the spectrum of the deconvolved function are suppressed. The main focus of the paper was on the accuracy of ADM in the cases in which the inverse filter $\mathbf{G}^{-1}$ was calculated based on the filter different than the basic one $\mathbf{G}$ used to create the filtered function. This was motivated by the fact that in practice, the precise form of $\mathbf{G}$ is rarely known and needs to be assumed. The analysis performed for different complexity of the test functions in 1D and 2D cases showed that inappropriate 'guess' of the filter $\mathbf{G}$ for ADM can have a significant and opposite than expected impact on the result of the deconvolution. It has been shown that if the test function was filtered using $\mathbf{G}$ with the transfer function, which largely suppressed the spectrum of the filtered function, as in the case of explicit filters, the deconvolution should be performed using $\mathbf{G}^{-1}$ based on the filter of the order higher than the basic $\mathbf{G}$. Otherwise, the deconvolved

function differed more from the original function than the filtered one. In the case of compact filters, which alter the spectrum of the filtered function only moderately, the deconvolution process always brought the deconvolved function closer to the original one. Hence, it can be said that the use of high-order filters in ADM minimises a risk of an artificial increase of the energy of small scales. Most often, such a situation should be avoided. However, there are also applications in which the amplification of tiny function components can be needed. As an example, one can imagine numerical simulations performed using a highly dissipative upwind discretisation method that significantly suppresses small-scale phenomena. Having defined an appropriate deconvolution method applied at the postprocessing step, it would be possible to recover the magnitude of these small scales, at least partially.

In the last section of the paper, ADM was used to model a 2D turbulent flow by the LES method. The results were compared with DNS data and also with the solutions obtained using the Smagorinsky model and the no-model approach. The role of ADM was to recover sub-filter scales in the resolved part of the flow scales. The comparisons were performed for various quantities including energy spectra, TKE evolution, variance, skewness, kurtosis of the velocity components, and vorticity. In general, it has been shown that ADM provides more accurate results than the no-model approach and the Smagorinsky model. In view of the analysis performed for the 1D and 2D linear problems and also contrary to [59], it was slightly surprising that the use of ADM with the simple 4th order explicit filter turned out to be more accurate than ADM employing the compact filters. In [59], however, the 2nd order discretisation schemes were used. Hence, it can be concluded that the ADM accuracy depends on the applied discretisation methods and the use of the filter that better approximates the filter induced by the discretisation scheme leads to more accurate results. The induced filter reflecting simultaneously the discretisation of all terms in the Navier–Stokes equations cannot be explicitly defined, and hence, the conclusions on which filter should be used for a particular solution algorithm cannot be univocally formulated. Parametric studies employing various discretisation methods and various filters are planned for future research.

Acknowledgements

References

1. R. Vichnevetsky, J.B. Bowles, *Fourier Analysis of Numerical Approximations of Hyperbolic Equations*, SIAM, Philadelphia, 1982.

2. R. Bracewell, J. Roberts, *Aerial smoothing in radio astronomy*, Australian Journal of Physics, **7**, 4, 615, 1954.

3. P.H. Van Cittert, *Zum Einflußder Spaltbreite auf die Intensitätsverteilung in Spektrallinien*, II. Zeitschrift Für Physik, **69**, 5-6, 298–308, 1931.

4. H.C. Burger, P.H. van Cittert, *Wahre und scheinbare Intensitätsverteilung in Spektrallinien*, Zeitschrift Für Physik, **79**, 11-12, 722–730, 1932.

5. M.G. Smith, A.N. Walker, *Correction of observational profiles of spectral lines*, Astrophysics and Space Science, **1**, 2, 151–165, 1968.

6. A.R. Stokes, *A numerical Fourier-analysis method for the correction of widths and shapes of lines on X-ray powder photographs*, Proceedings of the Physical Society, **61**, 4, 382–391, 1948.

7. M.S. Paterson, *Calculation of the correction for instrumental broadening in X-ray diffraction lines*, Proceedings of the Physical Society, Section A, **63**, 5, 477–482, 1950.

8. S. Ergun, *Direct method for unfolding convolution products – its application to X-ray scattering intensities*, Journal of Applied Crystallography, **1**, 1, 19–23, 1968.

9. P.A. Jansson, *Method for determining the response function of a high-resolution infrared spectrometer*, Journal of the Optical Society of America, **60**, 2, 184–191, 1970.

10. P. Jansson, *Deconvolution with Applications in Spectroscopy*, Academic Press, New York, 1984.

11. F. Yano, S. Nomura, *Deconvolution of scanning electron microscopy images*, Scanning, **15**, 1, 19–24, 1993.

12. E.J. Rapperport, *Deconvolution: A Technique to Increase Electron Probe Resolution*, [in:] *Electron Probe Microanalysis*, A.J. Tousimis and L. Marton [eds.], Academic Press, New York, p. 117, 1969.

13. Z. Mencik, *Iterative deconvolution of smeared data functions*, Journal of Applied Crystallography, 7, 1, 44–50, 1974.

14. M. Bertero, P. Boccacci, *Introduction to Inverse Problems in Imaging*, Bristol, U.K.: IOP, 1998.

15. L. Wang, Y. Huang, X. Luo, Z. Wang, S. Luo, *Image deblurring with filters learned by extreme learning machine*, Neurocomputing, **74**, 16, 2464–2474, 2011.

16. S. Kawata, Y. Ichioka, *Iterative image restoration for linearly degraded images I Basis*, Journal of the Optical Society of America, **70**, 7, 762, 1980.

17. S. Kawata, Y. Ichioka, *Iterative image restoration for linearly degraded images II Reblurring procedure*, Journal of the Optical Society of America, **70**, 7, 768, 1980.

18. H.J. Scudder, *Introduction to computer aided tomography*, Proceedings of the IEEE, **66**, 6, 628–637, 1978.

19. Z. Danovich, Y. Segal, *Laminogram reconstruction through regularizing Fourier filtration*, NDT and E International, **27**, 3, 123–130, 1994.

20. Y. GE, Q. ZHANG, Z. HU, J. CHEN, W. SHI, H. ZHENG, D. LIANG, *Deconvolution-based backproject-filter (BPF) computed tomography image reconstruction method using deep learning technique*, arXiv preprint: arXiv:1807.01833, 2018.

21. Y. LECUN, Y. BENGIO, G. HINTON, *Deep learning*, Nature, **521**, 436–444, 2015.

22. K.H. JIN, M.T. MCCANN, E. FROUSTEY, M. UNSER, *Deep convolutional neural network for inverse problems in imaging*, IEEE Transactions on Image Processing, **26**, 9, 4509–4522, 2017.

23. S. HAYKIN, *Blind Deconvolution*, PTR Prentice Hall, 1994.

24. P. CAMPISI, K. EGIAZARIAN, *Blind Image Deconvolution. Theory and Applications*, CRC Press, 2007.

25. M.S.C. ALMEIDA, L.B. ALMEIDA, *Blind and Semi-Blind Deblurring of Natural Images*, IEEE Transactions on Image Processing, **19**, 1, 36–52, 2010.

26. S. JEFFERIES, K. SCHULZE, C. MATSON, K. STOLTENBERG, E.K. HEGE, *Blind deconvolution in optical diffusion tomography*, Optics Express, **10**, 1, 46, 2002.

27. C. YU, C. ZHANG, L. XIE, *A blind deconvolution approach to ultrasound imaging*, IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, **59**, 2, 271–280, 2012.

28. T. TAXT, G.V. FROLOVA, *Noise robust one-dimensional blind deconvolution of medical ultrasound images*, IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, **46**, 2, 291–299, 1999.

29. T. TAXT, J. STRAND, *Two-dimensional noise-robust blind deconvolution of ultrasound images*, IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, **48**, 4, 861–866, 2001.

30. T. TAXT, *Three-dimensional blind deconvolution of ultrasound images*, IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, **48**, 4, 867–871, 2001.

31. R. GRÜNER, T. TAXT, *Iterative blind deconvolution in magnetic resonance brain perfusion imaging*, Magnetic Resonance in Medicine, **55**, 4, 805–815, 2006.

32. J. MA, F.-X. LE DIMET, *Deblurring from highly incomplete measurements for remote sensing*, IEEE Transactions on Geoscience and Remote Sensing, **47**, 3, 792–802, 2009.

33. M.J. ROAN, M.R. GRAMANN, J.G. ERLING, L.H. SIBUL, *Blind deconvolution applied to acoustical systems identification with supporting experimental results*, The Journal of the Acoustical Society of America, **114**, 4, 1988–1996, 2003.

34. T.G. STOCKHAM, T.M. CANNON, R.B. INGEBRETSEN, *Blind deconvolution through digital signal processing*, Proceedings of the IEEE, **63**, 4, 678–692, 1975.

35. G.T. ZHENG, M.A. BUCKLEY, G. KISTER, G.F. FERNANDO, *Blind deconvolution of acoustic emission signals for damage identification in composites*, AIAA Journal, **39**, 6, 1198–1205, 2001.

36. T. YAMAGUCHI, T. MIZUTANI, M. TARUMI, D. SU, *Sensitive damage detection of reinforced concrete bridge slab by "Time-Variant Deconvolution" of SHF-Band Radar Signal*, IEEE Transactions on Geoscience and Remote Sensing, 1–11, 2018.

37. A.K. TAKAHATA, E.Z. NADALIN, R. FERRARI, L.T. DUARTE, R. SUYAMA, R.R. LOPES, J.M.T.ROMANO, M. TYGEL, *Unsupervised processing of geophysical signals: a review of some key aspects of blind deconvolution and blind source separation*, IEEE Signal Processing Magazine, **29**, 4, 27–35, 2012.

38. Y. Li, G. Zhang, *A seismic blind deconvolution algorithm based on bayesian compressive sensing*, Mathematical Problems in Engineering, **2015**, 1–11, 2015.

39. M. Mirel, I. Cohen, *Multichannel semi-blind deconvolution (MSBD) of seismic signals*, Signal Processing, **135**, 253–262, 2017.

40. H.-Z. Wu, L.-Y. Fu, X.-H. Meng, *Blind deconvolution of seismic signals with non-white reflectivities*, Exploration Geophysics, **38**, 4, 235–241, 2007.

41. H.H. Lari, A. Gholami, *Nonstationary blind deconvolution of seismic records*, Geophysics, **84**, 1, V1–V9, 2019.

42. P. Castaldi, R. Diversi, R.P. Guidorzi, U. Soverini, *Blind estimation and deconvolution of communication channels with unbalanced noise*, IFAC Proceedings Volumes, **33**, 15, 325–330, 2000.

43. S. Cang, J. Swärd, X. Sheng, A. Jakobsson, *Toeplitz-based blind deconvolution of underwater acoustic channels using wideband integrated dictionaries*, Signal Processing, 107812, 2020.

44. W.J. Layton, L.G. Rebholz, *Approximate Deconvolution Models of Turbulence*, Lecture Notes in Mathematics, Springer-Verlag, Berlin, Heidelberg, 2012.

45. F.K. Chow, R.L. Street, M. Xue, J.H. Ferziger, *Explicit filtering and reconstruction turbulence modeling for large-eddy simulation of neutral boundary layer flow*, Journal of the Atmospheric Sciences, **62**, 7, 2058–2077, 2005.

46. J. Mathew, R. Lechner, H. Foysi, J. Sesterhenn, R. Friedrich, *An explicit filtering method for large eddy simulation of compressible flows*, Physics of Fluids, **15**, 8, 2279–2289, 2003.

47. J. Gullbrand, F.K. Chow, *The effect of numerical errors and turbulence models in large-eddy simulations of channel flow, with and without explicit filtering*, Journal of Fluid Mechanics, **495**, 323–341, 2003.

48. D. Carati, G. S. Winckelmans, H. Jeanmart, *On the modelling of the sub-grid-scale and filtered-scale stress tensors in large-eddy simulation*, Journal of Fluid Mechanics, **441**, 119–138, 2001.

49. B.J. Geurts, F. van der Bos, *Numerically induced high-pass dynamics in large-eddy simulation*, Physics of Fluids, **17**, 12, 125103, 2005.

50. S. Stolz, N.A. Adams, *An approximate deconvolution procedure for large-eddy simulation*, Physics of Fluids, **11**, 7, 1699–1701, 1999.

51. S. Stolz, N.A. Adams, L. Kleiser, *An approximate deconvolution model for large-eddy simulation with application to incompressible wall-bounded flows*, Physics of Fluids, **13**, 4, 997–1015, 2001.

52. J.A. Domaradzki, N.A. Adams, *Direct modelling of sub-grid scales of turbulence in large eddy simulations*, Journal of Turbulence, **3**, N24, 2002.

53. P. Schlatter, S. Stolz, L. Kleiser, *LES of transitional flows using the approximate deconvolution model*, International Journal of Heat and Fluid Flow, **25**, 3, 549–558, 2004.

54. O.A. Mahfoze, S. Laizet, *Non-explicit large eddy simulations of turbulent channel flows from* $Re_\tau = 180$ *up to* $Re_\tau = 5200$, Computers and Fluids, **228**, 105019, 2021.

55. S. Stolz, N.A. Adams, L. Kleiser, *The approximate deconvolution model for large-eddy simulations of compressible flows and its application to shock-turbulent-boundary-layer interaction*, Physics of Fluids, **13**, 10, 2985–3001, 2001.

56. R. Von Kaenel, N.A. Adams, L. Kleiser, J. Vos, *An approximate deconvolution model for large-eddy simulation of compressible flows with finite-volume schemes*, 40th AIAA Aerospace Sciences Meeting and Exhibit, 2002.

57. N.A. Adams, A. Stolz, *A sub-grid-Scale Deconvolution Approach for Shock Capturing*, Journal of Computational Physics, **178**, 391–426, 2002.

58. L. Goodfriend, F.K. Chow, M. Vanella, E. Balaras, *Large-eddy simulation of decaying isotropic turbulence across a grid refinement interface using explicit filtering and reconstruction*, Journal of Turbulence, **14**, 12, 58–76, 2013.

59. O. San, A.E. Staples, T. Iliescu, *A posteriori analysis of low-pass spatial filters for approximate deconvolution LES of homogeneous incompressible flows*, International Journal of Computational Fluid Dynamics, **29**, 1, 40–66, 2014.

60. O. San, *Analysis of low-pass filters for approximate deconvolution closure modelling in one-dimensional decaying Burgers turbulence*, International Journal of Computational Fluid Dynamics, **30**, 1, 20–37, 2016.

61. O. San, P. Vedula, *Generalized Deconvolution Procedure for Structural Modeling of Turbulence*, Journal of Scientific Computing, **75**, 2, 1187–1206, 2017.

62. O. San, A.E. Staples, Z. Wang, T. Iliescu, *Approximate deconvolution large eddy simulation of a barotropic ocean circulation model*, Ocean Modelling, **40**, 2, 120–132, 2011.

63. O. San, A.E. Staples, T. Iliescu, *Approximate deconvolution large eddy simulation of a stratified two-layer quasigeostrophic ocean model*, Ocean Modelling, **63**, 1–20, 2013.

64. F.K. Chow, R.L. Street, M. Xue, J.H. Ferziger, *Explicit filtering and reconstruction turbulence modeling for large-eddy simulation of neutral boundary layer flow*, Journal of the Atmospheric Sciences, **62**, 7, 2058–2077, 2005.

65. F.K. Chow, R.L. Street, *Evaluation of turbulence closure models for large-eddy simulation over complex terrain: flow over Askervein hill*, Journal of Applied Meteorology and Climatology, **48**, 5, 1050–1065, 2009.

66. B. Zhou, F.K. Chow, *Large-eddy simulation of the stable boundary layer with explicit filtering and reconstruction turbulence modeling*, Journal of the Atmospheric Sciences, **68**, 9, 2142–2155, 2011.

67. P. Domingo, L. Vervisch, *Large eddy simulation of premixed turbulent combustion using approximate deconvolution and explicit flame filtering*, Proceedings of the Combustion Institute, **35**, 2, 1349–1357, 2015.

68. Q. Wang, M. Ihme, *Regularized deconvolution method for turbulent combustion modeling*, Combustion and Flame, **176**, 125–142, 2017.

69. C. Mehl, J. Idier, B. Fiorina, *Evaluation of deconvolution modelling applied to numerical combustion*, Combustion Theory and Modelling, **22**, 1, 38–70, 2017.

70. C.D. Pruett, B.C. Thomas, C.E. Grosch, T.B. Gatski, *A temporal approximate deconvolution model for large-eddy simulation*, Physics of Fluids, **18**, 2, 028104, 2006.

71. J. Borggaard, T. Iliescu, *Approximate deconvolution boundary conditions for large eddy simulation*, Applied Mathematics Letters, **19**, 8, 735–740, 2006.

72. P. Schlatter, *Large-eddy simulation of transition and turbulence in wall-bounded shear flow*, PhD Thesis, ETH Zurich, 2005.

73. A.A. Dunca, R. Lewandowski, *Modeling error in approximate deconvolution models*, Communications in Mathematical Sciences, **12**, 4, 757–778, 2014.

74. A.N. Tikhonov, V.Y. Arsenin, *Solutions of Ill-Posed Problems*, V.H. Winston and Sons, Washington, DC, 1977.

75. C. Eckart, *The correction of continuous spectra for the finite resolution of the spectrometer*, Physical Review, **51**, 9, 735–738, 1937.

76. T. Sarkar, D. Weiner, V. Jain, *Some mathematical considerations in dealing with the inverse problem*, IEEE Transactions on Antennas and Propagation, **29**, 2, 373–379, 1981.

77. M. Ekstrom, *A spectral characterization of the ill-conditioning in numerical deconvolution*, IEEE Transactions on Audio and Electroacoustics, **21**, 4, 344–348, 1973.

78. R. Gold, *An Iterative Unfolding Method for Response Matrices*, Mathematics and Computer Research and Development Rep. ANL-6984, Argonne National Laboratory, Argonne, III, 1964.

79. S. Singh, S.N Tandon, H.M. Gupta, *An iterative restoration technique*, Signal Processing, **11**, 1, 1–11, 1986.

80. G. Thomas, *An improvement of the Van-Cittert's method*, ICASSP'81, IEEE International Conference on Acoustics, Speech, and Signal Processing, 1981.

81. N.R. Hill, G.E. Ioup, *Convergence of the van Cittert iterative method of deconvolution*, Journal of the Optical Society of America, **66**, 5, 487, 1976.

82. B. Parruck, S.M. Riad, *An optimization criterion for iterative deconvolution*, IEEE Transactions on Instrumentation and Measurement, **32**, 1, 137–140, 1983.

83. A. Bennia, S.M. Riad, *Filtering capabilities and convergence of the Van-Cittert deconvolution technique*, IEEE Transactions on Instrumentation and Measurement, **41**, 2, 246–250, 1992.

84. S.K. Lele, *Compact finite difference schemes with spectral-like resolution*, Journal of Computational Physics, **103**, 1, 16–42, 1992.

85. D.V. Gaitonde, J.S. Shang, J.L. Young, *Practical aspects of higher-order numerical schemes for wave propagation phenomena*, Internationa Journal of Numerical Methods in Engeeniring, **45**, 1849, 1999.

86. P. Sagaut, *Large Eddy Simulation for Incompressible Flows*, Springer-Verlag, Berlin, Heidelberg, 2001.

87. B.J. Geurts, *Elements of Direct and Large-Eddy Simulation*, Edwards Publishing, Dordrecht, 2004.

88. O. San, A.E. Staples, *High-order methods for decaying two-dimensional homogeneous isotropic turbulence*, Computers and Fluids, **63**, 105–127, 2012.

89. L. Caban, A. Tyliszczak, *High-order compact difference schemes on wide computational stencils with a spectral-like accuracy*, Computers and Mathematics with Applications, **108**, 123–140, 2022.

90. W. LAYTON, R. LEWANDOWSKI, *A simple and stable scale-similarity for large eddy simulation: energy and existence of weak solutions*, Applied Mathematics Letters, **16**, 1205–1209, 2003.

91. A. DUNCA, Y. EPSHTEYN, *On the Stolz-Adams deconvolution model for large eddy simulation of turbulent flows*, SIAM Journal of Mathematical Analysis, **37**, 6, 1890–1902, 2006.

92. L.C. BERSELLI, R. LEWANDOWSKI, *Convergence of approximate deconvolution models to the mean Navier–Stokes equations*, L'Association Publications de l'Institut Henri Poincaré, **29**, 171–198, 2012.

93. C. CANUTO, M.Y. HUSSAINI, A. QUARTERONI, T. ZANG, *Spectral Methods in Fluid Dynamics*, Springer-Verlag, Berlin, Heidelberg, 1998.