

A PROPOSED SIMPLIFICATION OF THE ADAPTIVE LOCAL DECONVOLUTION METHOD

S. HICKEL¹ AND N.A. ADAMS

Abstract. The adaptive local deconvolution method (ALDM) [9] provides a systematic framework for the implicit large-eddy simulation (ILES) of turbulent flows. Exploiting numerical truncation errors, the subgrid scale (SGS) model of ALDM is implicitly contained within the discretization. An explicit computation of model terms therefore becomes unnecessary. Subject of the present paper is a modification of the numerical algorithm that allows for reducing the amount of computational operations without affecting the quality of the results. Computational results for isotropic turbulence and plane channel flow show that the simplified adaptive local deconvolution (SALD) method performs similarly to the original method ALDM and at least as well as established explicit models.

1. INTRODUCTION

In Large-Eddy Simulation (LES) of turbulent flows the evolution of non-universal larger scales is computed, whereas their interaction with unresolved subgrid scales (SGS) is modeled. SGS effects are modeled *explicitly* if the underlying conservation law is modified and subsequently discretized. The filtering concept of Leonard [14] is commonly employed for deriving explicit SGS models without reference to a computational grid and without taking into account a discretization scheme. However, numerically computed SGS stresses are strongly affected by the truncation error of the discretization scheme [6]. This interference can result in strange results such as the lack of grid convergence. Employing an explicit SGS model, Schumann [19] already argued that discretization effects should be taken into account within the SGS model formulation.

As *Implicit Large-Eddy Simulation* (ILES) we denote the situation when the unmodified conservation law is discretized and the numerical truncation error acts as SGS model. Since the SGS model is *implicitly* contained within the discretization scheme an explicit computation of model terms becomes unnecessary.

One recent approach, the Adaptive Local Deconvolution Method (ALDM) [9], provides a systematic framework for implicit SGS modeling in turbulent flows. With ALDM numerical discretization and SGS modeling are merged entirely. A local reconstruction of the solution is obtained from a solution-adaptive combination of deconvolution polynomials. Free parameters, involved in the non-linear weight functions of the respective contributions, allow for SGS modeling. Optimal model parameters were determined by systematically minimizing a cost function which measures the difference between spectral numerical viscosity of ALDM and the eddy viscosity from EDQNM theory for isotropic turbulence so that the truncation error has physical significance.

The efficiency of this approach was demonstrated in Ref. [1] for 1D conservation laws on the example of the viscous Burgers equation. The extension to three spatial dimensions and the incompressible Navier-Stokes equations is detailed in Ref. [9]. ALDM has proven itself as a reliable, accurate, and efficient method for LES of three-dimensional homogeneous isotropic turbulence [9], for plane channel flow [11], and for the complex

¹ Institute of Aerodynamics, Technische Universität München, D-85747 Garching, Germany. E-mail: sh@tum.de

flow in a channel with periodic constrictions [2, 13]. All simulations show good agreement with theory and experimental data and thus demonstrate the good performance of the implicit model. It has been shown that ALDM performs at least as well as established explicit models.

In the present paper we revisit the numerical formulation of ALDM for the incompressible 3D Navier-Stokes equations. A simplification of the method is proposed. The good performance of the implicit model is preserved while computational costs are reduced significantly.

2. SOLUTION-ADAPTIVE LOCAL DECONVOLUTION REVISITED

ALDM is a general nonlinear, i.e. solution adaptive, discretization scheme designed for implicit LES. It is based on standard approaches which were modified in such a way that the resulting truncation error functions as implicit SGS model. In the following, all relevant parts of the algorithm are reviewed. We demonstrate how the original formulation of ALDM [9] can be further simplified to accelerate computation. The reader is provided with a self-contained paper on the *simplified* adaptive local deconvolution (SALD) method.

The incompressible Navier-Stokes equations in non-dimensional form read

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{u}) + \nabla p - \nu \nabla \cdot \nabla \mathbf{u} = \mathbf{0} , \quad (1a)$$

$$\nabla \cdot \mathbf{u} = 0 , \quad (1b)$$

where \mathbf{u} is the velocity vector, p is the pressure, and ν is the kinematic viscosity. For implicit modeling we only consider the nonlinear term $\nabla \cdot \mathbf{F} + \nabla p$ in the momentum equation (1a), whereas the linear terms, i.e. the diffusive flux, are approximated by a standard centered discretization. ALDM applies to the convective flux $\mathbf{F} = \mathbf{u}\mathbf{u}$. For incompressible flows a fractional-step approach is taken where the normal stresses due to the pressure p are subsequently computed by solving a Poisson equation.

Even if filtering is not performed explicitly we can use the filtered formulation of Leonard [14] as analytical tool when designing and analyzing discrete operators

$$\frac{\partial \bar{\mathbf{u}}_N}{\partial t} + \mathbf{G} * \nabla \cdot \mathbf{F}_N(\mathbf{G}^{-1} * \bar{\mathbf{u}}_N) + \nabla \bar{p} - \nu \nabla \cdot \nabla \bar{\mathbf{u}}_N = -\mathbf{G} * \nabla \cdot \boldsymbol{\tau}_{SGS} , \quad (2a)$$

$$\nabla \cdot \bar{\mathbf{u}}_N = 0 . \quad (2b)$$

Note that with implicit LES the subgrid-stress tensor $\boldsymbol{\tau}_{SGS}$ is not computed explicitly but modeled by numerical truncation errors.

A finite-volume discretization is based on the top-hat filter kernel

$$\mathbf{G}(\mathbf{x}_{i,j,k}, \mathbf{x}) = \frac{1}{\Delta x_i \Delta y_j \Delta z_k} \begin{cases} 1 & , (\mathbf{x}_{i,j,k} + \mathbf{x}) \in I_{i,j,k} \\ 0 & , \text{otherwise} \end{cases} , \quad (3)$$

which returns the cell average of a function

$$\bar{\varphi}(\mathbf{x}_{i,j,k}, t) = \frac{1}{\Delta x_i \Delta y_j \Delta z_k} \iiint_{I_{i,j,k}} \varphi(\mathbf{x}_{i,j,k} - \mathbf{x}, t) d\mathbf{x} , \quad (4)$$

where the integration domain $I_{i,j,k} = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}] \times [z_{k-\frac{1}{2}}, z_{k+\frac{1}{2}}]$ is equivalent to a cell of the underlying Cartesian computational grid so that the filter width corresponds to the local grid size

$$\Delta \mathbf{x}_{i,j,k} = \begin{bmatrix} \Delta x_i \\ \Delta y_j \\ \Delta z_k \end{bmatrix} = \begin{bmatrix} x_{i+1/2} - x_{i-1/2} \\ y_{j+1/2} - y_{j-1/2} \\ z_{k+1/2} - z_{k-1/2} \end{bmatrix}. \quad (5)$$

Here and in the following half-integer indices denote cell faces and the coordinate system $\{x, y, z\}$ is synonymous with $\{1, 2, 3\}$. The 3D filter kernel \mathbf{G} can be factorized into three 1D operators

$$\mathbf{G}(\mathbf{x}) = \mathbf{G}_x(x) * \mathbf{G}_y(y) * \mathbf{G}_z(z). \quad (6)$$

An inverse-filter operation can be defined as convolution with the inverse kernels

$$\mathbf{G}^{-1}(\mathbf{x}) = \mathbf{G}_x^{-1}(x) * \mathbf{G}_y^{-1}(y) * \mathbf{G}_z^{-1}(z). \quad (7)$$

In the framework of a finite-volume discretization filtering applied to the flux divergence returns the flux through the surface $S_{i,j,k}$ of cell $I_{i,j,k}$. By Gauss' theorem we obtain

$$\begin{aligned} [\mathbf{G} * \nabla \cdot \mathbf{F}(\mathbf{G}^{-1} * \bar{\mathbf{u}})]_{i,j,k} &= \frac{1}{\Delta x_i} \mathbf{G}_y * \mathbf{G}_z * \left(\left. {}^1 \mathbf{f}(\mathbf{G}^{-1} * \bar{\mathbf{u}}) \right|_{i+\frac{1}{2},j,k} - \left. {}^1 \mathbf{f}(\mathbf{G}^{-1} * \bar{\mathbf{u}}) \right|_{i-\frac{1}{2},j,k} \right) \\ &+ \frac{1}{\Delta y_j} \mathbf{G}_z * \mathbf{G}_x * \left(\left. {}^2 \mathbf{f}(\mathbf{G}^{-1} * \bar{\mathbf{u}}) \right|_{i,j+\frac{1}{2},k} - \left. {}^2 \mathbf{f}(\mathbf{G}^{-1} * \bar{\mathbf{u}}) \right|_{i,j-\frac{1}{2},k} \right) \\ &+ \frac{1}{\Delta z_k} \mathbf{G}_x * \mathbf{G}_y * \left(\left. {}^3 \mathbf{f}(\mathbf{G}^{-1} * \bar{\mathbf{u}}) \right|_{i,j,k+\frac{1}{2}} - \left. {}^3 \mathbf{f}(\mathbf{G}^{-1} * \bar{\mathbf{u}}) \right|_{i,j,k-\frac{1}{2}} \right), \end{aligned} \quad (8)$$

where the flux vector ${}^l \mathbf{f} = u_l \mathbf{u}$ denotes the l -direction component of \mathbf{F} . As a consequence of this identity (8), finite-volume schemes require a reconstruction of data at the faces of the computational volumes.

The numerical computation of eq. (8) necessarily involves approximations

$$\left[\tilde{\mathbf{G}} * \tilde{\nabla} \cdot \mathbf{F}(\tilde{\mathbf{G}}^{-1} * \bar{\mathbf{u}}) \right]_{i,j,k} \approx [\mathbf{G} * \nabla \cdot \mathbf{F}(\mathbf{G}^{-1} * \bar{\mathbf{u}})]_{i,j,k} \quad (9)$$

The original formulation of ALDM [9] employs a standard Gaussian quadrature rule with kernel \mathbf{C}_k and order $\mathcal{O}(\Delta x^k)$ to approximate the filter operation $\mathbf{G}_l * \mathbf{G}_m$ over the cell face transversal to the flux direction whereas a solution adaptive deconvolution scheme is proposed to approximate \mathbf{G}^{-1} . In Ref. [9] a simple second-order accurate filtering scheme is used for $\mathbf{G}_l * \mathbf{G}_m$. The benefits of higher-order schemes were found to be negligible since second-order accurate interpolants contribute to the deconvolution operator.

Recent numerical experiments presented in the following section have shown that also the deconvolution scheme can be simplified significantly without affecting the prediction power of the implicit SGS model. Based on these results we recommend as simplification

$$\begin{aligned} \left[\tilde{\mathbf{G}} * \tilde{\nabla} \cdot \mathbf{F}(\tilde{\mathbf{G}}^{-1} * \bar{\mathbf{u}}) \right]_{i,j,k} &= \frac{1}{\Delta x_i} \left(\left. \mathbf{f}(\tilde{\mathbf{G}}_x^{-1} * \bar{\mathbf{u}}) \right|_{i+\frac{1}{2},j,k} - \left. \mathbf{f}(\tilde{\mathbf{G}}_x^{-1} * \bar{\mathbf{u}}) \right|_{i-\frac{1}{2},j,k} \right) \\ &+ \frac{1}{\Delta y_j} \left(\left. \mathbf{f}(\tilde{\mathbf{G}}_y^{-1} * \bar{\mathbf{u}}) \right|_{i,j+\frac{1}{2},k} - \left. \mathbf{f}(\tilde{\mathbf{G}}_y^{-1} * \bar{\mathbf{u}}) \right|_{i,j-\frac{1}{2},k} \right) \\ &+ \frac{1}{\Delta z_k} \left(\left. \mathbf{f}(\tilde{\mathbf{G}}_z^{-1} * \bar{\mathbf{u}}) \right|_{i,j,k+\frac{1}{2}} - \left. \mathbf{f}(\tilde{\mathbf{G}}_z^{-1} * \bar{\mathbf{u}}) \right|_{i,j,k-\frac{1}{2}} \right), \end{aligned} \quad (10)$$

where no filtering $\mathbf{G}_l * \mathbf{G}_m$ is applied. Furthermore, approximate deconvolution $\tilde{\mathbf{G}}_l^{-1}$ is performed only in those directions l for which interpolation at cell faces is necessary using the 1D reconstruction (i.e. deconvolution and interpolation) operator \mathcal{X}_x^λ of ALDM proposed in Ref. [1] and adapted in Ref. [9]. In the simplified adaptive local deconvolution method the fully 3D scheme of ALDM is replaced by a single 1D step at the target cell face. Deconvolution is not performed in the transverse directions.

In the following we repeat the derivation of the algorithm as given in Ref. [9] and point out the differences between full ALDM and the SALD method. The 1D adaptive deconvolution operator, denoted by \mathcal{X}_x^λ , is defined on a 1D grid $x_N = \{x_i\}$. Applied to the filtered grid function $\bar{\varphi}_N = \{\bar{\varphi}(x_i)\}$ it returns the approximately deconvolved grid function $\tilde{\varphi}_N^\lambda \doteq \{\varphi(x_{i+\lambda})\}$ on the shifted grid $x_N^\lambda = \{x_{i+\lambda}\}$

$$\mathcal{X}_x^\lambda \bar{\varphi}_N = \{\varphi(\bar{\varphi}, x_{i+\lambda}) + \mathcal{O}(\Delta x_i^\kappa)\} = \tilde{\varphi}_N^\lambda. \quad (11)$$

Following eq. (7), a 3D reconstruction is obtained by successive 1D operations

$$\tilde{\varphi}_N^\lambda = \mathcal{X}^\lambda \bar{\varphi}_N = \mathcal{X}_z^{\lambda_3} (\mathcal{X}_y^{\lambda_2} (\mathcal{X}_x^{\lambda_1} \bar{\varphi}_N)) . \quad (12)$$

The filtered data are given at the cell centers $\{x_i\}$. Reconstruction at the left cell faces $\{x_{i-\frac{1}{2}}\}$ is indicated by $\lambda = -1/2$ and at the right faces $\{x_{i+\frac{1}{2}}\}$ by $\lambda = +1/2$. The ALDM operators which perform deconvolution at the cell centers, denoted by $\lambda = 0$, are replaced by identity in the simplified algorithm.

Deconvolution and interpolation are done simultaneously using Lagrangian interpolation polynomials as proposed by Harten *et al.* [8]. Given a generic k -point stencil ranging from x_{i-r} to $x_{i-r+k-1}$ the 1D ansatz for the top-hat kernel (3) reads

$$\varphi(x_{i+\lambda}) = \sum_{l=0}^{k-1} c_{k,r,l}^\lambda(x_i) \bar{\varphi}_N(x_{i-r+l}) + \mathcal{O}(\Delta x_i^k), \quad (13)$$

with $r \in \{0, \dots, k\}$. The grid-dependent coefficients are

$$c_{k,r,l}^\lambda(x_i) = \left(x_{i-r+l+\frac{1}{2}} - x_{i-r+l-\frac{1}{2}} \right) \sum_{m=l+1}^k \frac{\sum_{\substack{p=0 \\ p \neq m}}^k \prod_{\substack{n=0 \\ n \neq p,m}}^k x_{i+\lambda} - x_{i-r+n-\frac{1}{2}}}{\prod_{\substack{n=0 \\ n \neq m}}^k x_{j-r+m-\frac{1}{2}} - x_{i-r+n-\frac{1}{2}}}, \quad (14)$$

as given by Shu [21]. They apply for grids with variable mesh width and for arbitrary target positions $x_{i+\lambda}$. In case of a staggered grid the values of x_i are different for each velocity component and the coefficients $c_{k,r,l}^\lambda$ have to be specified accordingly.

Selecting a particular interpolation stencil would return a linear discretization with a fixed, solution independent functional expression of the k -th order truncation error, provided the function is sufficiently smooth on the interpolation stencil. ALDM adopts the idea of the Weighted-Essentially-Non-Oscillatory (WENO) scheme of Shu [21] where interpolation polynomials of a single order $k \equiv K$ are selected and combined nonlinearly. The essential difference between ALDM and WENO is that we superpose all interpolants of order $k = 1, \dots, K$

$$\tilde{\varphi}_N^\lambda(x_{i+\lambda}) = \sum_{k=1}^K \sum_{r=0}^{k-1} \omega_{k,r}^\lambda(\bar{\varphi}_N, x_i) \sum_{l=0}^{k-1} c_{k,r,l}^\lambda(x_i) \bar{\varphi}_N(x_{i-r+l}) \quad (15)$$

to allow for lower-order contributions to the truncation error for implicit SGS modeling [1]. Usually it is $K = 3$. The weights $\omega_{k,r}^\lambda(\bar{\varphi}_N, x_i)$ can be constructed as to yield an accurate approximation of order $\kappa = 2K - 1$ in

smooth regions [21]. For our purpose, however, we do not need highest possible order of accuracy. Rather the superposition (15) introduces free discretization parameters which allow to control error cancelations. The sum of all weights is constrained to be unity for consistency. More restrictively we require

$$\omega_{k,r}^\lambda(\bar{\varphi}_N, x_i) = \frac{1}{K} \frac{\gamma_{k,r}^\lambda \beta_{k,r}(\bar{\varphi}_N, x_i)}{\sum_{s=0}^{k-1} \gamma_{k,s}^\lambda \beta_{k,s}(\bar{\varphi}_N, x_i)}, \quad (16)$$

with $r = 0, \dots, k-1$ for each $k = 1, \dots, K$. Note that λ denotes a superscript index and not a power. The solution-adaptive behavior of ALDM is controlled by the functional

$$\beta_{k,r}(\bar{\varphi}_N, x_i) = \left(\varepsilon_\beta + \sum_{m=-r}^{k-r-2} (\bar{\varphi}_{i+m+1} - \bar{\varphi}_{i+m})^2 \right)^{-2}, \quad (17)$$

where ε_β is a small number to prevent division by zero in eq. (16). $\beta_{k,r}$ measure the smoothness of the grid function on the respective stencil to obtain a non-linear adaptation of the deconvolution. The advantage of definition (17) over smoothness measures proposed by Liu *et al.* [17] and by Jiang & Shu [12] is that

$$\beta_{k,r}(\bar{\varphi}_N, x_i) = \beta_{k,r-1}(\bar{\varphi}_N, x_{i-1}). \quad (18)$$

can be exploited to improved computational efficiency. The parameters $\gamma_{k,r}^\lambda$ represent a stencil-selection preference that would become effective in the statistically homogeneous case. The requirement of an isotropic discretization for this case implies symmetries on the parameters

$$\gamma_{k,r}^{-1/2} = \gamma_{k,k-1-r}^{+1/2}. \quad (19)$$

The number of independent parameters is further reduced by the consistency requirement $\sum_{r=0}^{k-1} \gamma_{k,r}^{+1/2} = 1$. Using the simplified procedure with $K = 3$, three parameters $\{ \gamma_{2,0}^{+1/2}, \gamma_{3,0}^{+1/2}, \gamma_{3,1}^{+1/2} \}$ remain available for modeling. The original formulation [9] provides one additional parameter $\gamma_{3,1}^0$. Optimal parameters for the original scheme were found by means of an evolutionary optimization algorithm [1, 9, 10]. These parameters, reproduced in table 1, result in an excellent spectral match of the numerical viscosity of ALDM with prediction of EDQNM theory [3, 15]. We have refrained from repeating the parameter optimization with the simplified scheme because the change by applying the transversal deconvolution operation was found to be negligible in terms of the effective numerical viscosity. For further computational details of evaluation and optimization of the spectral numerical viscosity of ALDM the reader should refer to Ref. [9].

TABLE 1. Optimized values for discretization parameters [9].

parameter	$\gamma_{2,0}^{+1/2}$	$\gamma_{2,1}^{+1/2}$	$\gamma_{3,0}^{+1/2}$	$\gamma_{3,1}^{+1/2}$	$\gamma_{3,2}^{+1/2}$	σ
optimal value	1.00000	0.00000	0.01902	0.08550	0.89548	0.06891

It is interesting to note that the evolutionary optimization finally selected $\gamma_{2,0}^{+1/2} = 1$ and $\gamma_{2,1}^{+1/2} = 0$. Consequently, solution adaptivity is ruled out for all but the third-order stencils. It is therefore not necessary to compute $\beta_{2,r}$.

Another tool which is exploited is the choice of an appropriate and consistent numerical flux function that approximates the physical flux function

$$\tilde{\mathbf{F}}_N \approx \mathbf{F} = \mathbf{u}\mathbf{u} \quad \text{and} \quad \tilde{\mathbf{f}} \approx \mathbf{f} = u_l \mathbf{u}. \quad (20)$$

A review of common numerical flux functions can be found, e.g., in LeVeque's textbook [16]. During construction of ALDM the modified differential equations (MDE) of various flux functions were analyzed [1]. Based on these findings the following modification of a Lax-Friedrichs flux function was proposed [9] for the three-dimensional Navier-Stokes equations and collocated Cartesian grids

$$\tilde{\mathbf{f}}_{i+\frac{1}{2},j,k} = \frac{1}{4} (\tilde{u}_{i+1,j,k}^L + \tilde{u}_{i,j,k}^R) (\tilde{u}_{i+1,j,k}^L + \tilde{u}_{i,j,k}^R) - {}^1\sigma_{i,j,k} \begin{bmatrix} |\bar{u}_{i+1,j,k} - \bar{u}_{i,j,k}| (\tilde{u}_{i+1,j,k}^L - \tilde{u}_{i,j,k}^R) \\ |\bar{v}_{i+1,j,k} - \bar{v}_{i,j,k}| (\tilde{v}_{i+1,j,k}^L - \tilde{v}_{i,j,k}^R) \\ |\bar{w}_{i+1,j,k} - \bar{w}_{i,j,k}| (\tilde{w}_{i+1,j,k}^L - \tilde{w}_{i,j,k}^R) \end{bmatrix}, \quad (21a)$$

$$\tilde{\mathbf{f}}_{i,j+\frac{1}{2},k} = \frac{1}{4} (\tilde{v}_{i,j+1,k}^B + \tilde{v}_{i,j,k}^F) (\tilde{v}_{i,j+1,k}^B + \tilde{v}_{i,j,k}^F) - {}^2\sigma_{i,j,k} \begin{bmatrix} |\bar{u}_{i,j+1,k} - \bar{u}_{i,j,k}| (\tilde{u}_{i,j+1,k}^B - \tilde{u}_{i,j,k}^F) \\ |\bar{v}_{i,j+1,k} - \bar{v}_{i,j,k}| (\tilde{v}_{i,j+1,k}^B - \tilde{v}_{i,j,k}^F) \\ |\bar{w}_{i,j+1,k} - \bar{w}_{i,j,k}| (\tilde{w}_{i,j+1,k}^B - \tilde{w}_{i,j,k}^F) \end{bmatrix}, \quad (21b)$$

$$\tilde{\mathbf{f}}_{i,j,k+\frac{1}{2}} = \frac{1}{4} (\tilde{w}_{i,j,k+1}^D + \tilde{w}_{i,j,k}^U) (\tilde{w}_{i,j,k+1}^D + \tilde{w}_{i,j,k}^U) - {}^3\sigma_{i,j,k} \begin{bmatrix} |\bar{u}_{i,j,k+1} - \bar{u}_{i,j,k}| (\tilde{u}_{i,j,k+1}^D - \tilde{u}_{i,j,k}^U) \\ |\bar{v}_{i,j,k+1} - \bar{v}_{i,j,k}| (\tilde{v}_{i,j,k+1}^D - \tilde{v}_{i,j,k}^U) \\ |\bar{w}_{i,j,k+1} - \bar{w}_{i,j,k}| (\tilde{w}_{i,j,k+1}^D - \tilde{w}_{i,j,k}^U) \end{bmatrix}. \quad (21c)$$

A further simplification of this flux function is not possible without affecting the performance of the model. However, the computational efficiency can be improved by merging convective and diffusive flux computations. The extension to a staggered grid arrangement is straight forward and leads to

$$\tilde{\mathbf{f}}_{i+\frac{1}{2},j,k} = \frac{1}{4} \begin{bmatrix} (\tilde{u}_{i+1,j,k}^L + \tilde{u}_{i,j,k}^R) (\tilde{u}_{i+1,j,k}^L + \tilde{u}_{i,j,k}^R) \\ (\tilde{u}_{i,j+1,k}^B + \tilde{u}_{i,j,k}^F) (\tilde{v}_{i+1,j,k}^L + \tilde{v}_{i,j,k}^R) \\ (\tilde{u}_{i,j,k+1}^D + \tilde{u}_{i,j,k}^U) (\tilde{w}_{i+1,j,k}^L + \tilde{w}_{i,j,k}^R) \end{bmatrix} - {}^1\sigma_{i,j,k} \begin{bmatrix} |\bar{u}_{i+1,j,k} - \bar{u}_{i,j,k}| (\tilde{u}_{i+1,j,k}^L - \tilde{u}_{i,j,k}^R) \\ |\bar{v}_{i+1,j,k} - \bar{v}_{i,j,k}| (\tilde{v}_{i+1,j,k}^L - \tilde{v}_{i,j,k}^R) \\ |\bar{w}_{i+1,j,k} - \bar{w}_{i,j,k}| (\tilde{w}_{i+1,j,k}^L - \tilde{w}_{i,j,k}^R) \end{bmatrix}, \quad (22a)$$

$$\tilde{\mathbf{f}}_{i,j+\frac{1}{2},k} = \frac{1}{4} \begin{bmatrix} (\tilde{v}_{i+1,j,k}^B + \tilde{v}_{i,j,k}^F) (\tilde{u}_{i,j+1,k}^B + \tilde{u}_{i,j,k}^F) \\ (\tilde{v}_{i,j+1,k}^B + \tilde{v}_{i,j,k}^F) (\tilde{v}_{i,j+1,k}^B + \tilde{v}_{i,j,k}^F) \\ (\tilde{v}_{i,j,k+1}^D + \tilde{v}_{i,j,k}^U) (\tilde{w}_{i,j+1,k}^B + \tilde{w}_{i,j,k}^F) \end{bmatrix} - {}^2\sigma_{i,j,k} \begin{bmatrix} |\bar{u}_{i,j+1,k} - \bar{u}_{i,j,k}| (\tilde{u}_{i,j+1,k}^B - \tilde{u}_{i,j,k}^F) \\ |\bar{v}_{i,j+1,k} - \bar{v}_{i,j,k}| (\tilde{v}_{i,j+1,k}^B - \tilde{v}_{i,j,k}^F) \\ |\bar{w}_{i,j+1,k} - \bar{w}_{i,j,k}| (\tilde{w}_{i,j+1,k}^B - \tilde{w}_{i,j,k}^F) \end{bmatrix}, \quad (22b)$$

$$\tilde{\mathbf{f}}_{i,j,k+\frac{1}{2}} = \frac{1}{4} \begin{bmatrix} (\tilde{w}_{i+1,j,k}^D + \tilde{w}_{i,j,k}^U) (\tilde{u}_{i,j,k+1}^D + \tilde{u}_{i,j,k}^U) \\ (\tilde{w}_{i,j+1,k}^D + \tilde{w}_{i,j,k}^U) (\tilde{v}_{i,j,k+1}^D + \tilde{v}_{i,j,k}^U) \\ (\tilde{w}_{i,j,k+1}^D + \tilde{w}_{i,j,k}^U) (\tilde{w}_{i,j,k+1}^D + \tilde{w}_{i,j,k}^U) \end{bmatrix} - {}^3\sigma_{i,j,k} \begin{bmatrix} |\bar{u}_{i,j,k+1} - \bar{u}_{i,j,k}| (\tilde{u}_{i,j,k+1}^D - \tilde{u}_{i,j,k}^U) \\ |\bar{v}_{i,j,k+1} - \bar{v}_{i,j,k}| (\tilde{v}_{i,j,k+1}^D - \tilde{v}_{i,j,k}^U) \\ |\bar{w}_{i,j,k+1} - \bar{w}_{i,j,k}| (\tilde{w}_{i,j,k+1}^D - \tilde{w}_{i,j,k}^U) \end{bmatrix}. \quad (22c)$$

Please note that in the latter equation the indices are defined using the specific (staggered) coordinate system of the respective velocity component. Abbreviations are summarized in table 2. The first term of the numerical flux function corresponds to the physical Navier-Stokes flux. For maximum order of consistency it is computed from the mean of both interpolants of the deconvolved velocity at the considered cell face. The difference between them is exploited as an estimate of the local truncation error. In the second term on the right-hand side it is multiplied with the magnitude of a filtered velocity increment which is related to the first-order structure

TABLE 2. Interpolation directions for 3D reconstruction.

Direction	Example
(R) rightward	$\tilde{\mathbf{u}}_{i,j,k}^R \approx \mathbf{u}(x_{i+\frac{1}{2}}, y_j, z_k)$
(L) leftward	$\tilde{\mathbf{u}}_{i,j,k}^L \approx \mathbf{u}(x_{i-\frac{1}{2}}, y_j, z_k)$
(F) forward	$\tilde{\mathbf{u}}_{i,j,k}^F \approx \mathbf{u}(x_i, y_{j+\frac{1}{2}}, z_k)$
(B) backward	$\tilde{\mathbf{u}}_{i,j,k}^B \approx \mathbf{u}(x_i, y_{j-\frac{1}{2}}, z_k)$
(U) upward	$\tilde{\mathbf{u}}_{i,j,k}^U \approx \mathbf{u}(x_i, y_j, z_{k+\frac{1}{2}})$
(D) downward	$\tilde{\mathbf{u}}_{i,j,k}^D \approx \mathbf{u}(x_i, y_j, z_{k-\frac{1}{2}})$

function. For developed turbulence the Kolmogorov theory predicts a scaling with a 1/3 power of the two-point separation [5]. The coefficients

$${}^1\sigma_{i,j,k} = \sigma \left(\frac{L_0}{\Delta_0} \frac{x_{i+1,j,k} - x_{i,j,k}}{L_{i,j,k}} \right)^{-s}, \quad (23a)$$

$${}^2\sigma_{i,j,k} = \sigma \left(\frac{L_0}{\Delta_0} \frac{y_{i,j+1,k} - y_{i,j,k}}{L_{i,j,k}} \right)^{-s}, \quad (23b)$$

$${}^3\sigma_{i,j,k} = \sigma \left(\frac{L_0}{\Delta_0} \frac{z_{i,j,k+1} - z_{i,j,k}}{L_{i,j,k}} \right)^{-s}, \quad (23c)$$

were introduced to compensate for varying grid-size effects. The scalar factor σ adds another free parameter to ALDM, see table 1. For isotropic turbulence it was found that $s = 1/3$ ensures a resolution-independent numerical viscosity, as long as the numerical cutoff is located within the inertial range of the kinetic-energy spectrum. $\Delta_0 = 2\pi/32$ is a reference grid size as in Ref. [9]. The original scheme for isotropic turbulence is recovered if $L_{i,j,k} = L_0$. For the purpose of wall modeling a van Driest damping is introduced

$$L_{i,j,k} = L_0 - L_0 \exp \left(- \left(\frac{l_W u_\tau}{a \nu} \right)^d \right), \quad (24)$$

where l_W is the wall distance and u_τ is the friction velocity at the closest wall, d and a are free parameters. During computational experimentation only a weak dependence between these parameters and the prediction quality could be found. We estimate optimal values as $d = s^{-1}$ and $a = 25$ based on these computational experiments. The proposed van Driest damping reduces the SGS dissipation at wall distances below $50\nu/u_\tau$, whereas the scheme remains unchanged above.

The friction velocity is defined as

$$u_\tau(\mathbf{x}) = \sqrt{\nu \left| \frac{\partial \langle \mathbf{u}(\mathbf{x}) \rangle}{\partial \mathbf{n}_W(\mathbf{x})} \right|}. \quad (25)$$

The derivative is evaluated at the wall using the wall-normal unit vector \mathbf{n}_W . In plane channel flow it is convenient to compute the friction velocity from

$$u_\tau = \frac{\nu}{h} Re_\tau \quad (26)$$

approximating the unknown Reynolds number Re_τ by its nominal value.

In order to allow for the treatment of arbitrarily shaped walls and moving obstacles a level-set based approach is pursued. The distance $l_W(\mathbf{x})$ from the point \mathbf{x} to the closest point at a solid surface \mathbf{x}_W is a well-defined

scalar quantity within the entire fluid domain. The interior of solid obstacles being marked by a negative sign of l_W . The wall-normal unit vector is obtained from the gradient of the level-set function

$$\mathbf{n}_W = \frac{\nabla l_W}{|\nabla l_W|} \quad (27)$$

presuming the level-set function l_W to be smooth. Since it is $\lim_{l_W \rightarrow 0} |\nabla l_W| = 1$, the local wall-tangential friction velocity can be evaluated as

$$u_\tau(\mathbf{x}) = \sqrt{\nu |\nabla l_W(\mathbf{x}) \cdot \nabla \langle \mathbf{u} \rangle(\mathbf{x})|} \quad (28)$$

in the near-wall region.

3. NUMERICAL RESULTS

In this section computational results for LES of decaying isotropic turbulence and turbulent channel flow are presented for validation of the proposed method. SALD and ALDM are implemented in an experimental flow solver, INCA, which solves the incompressible Navier-Stokes equations in Cartesian coordinates. This solver is our workhorse for numerical-method development and covers various linear and nonlinear discretizations on collocated and staggered grids as well as spectral methods.

For time advancement, we employ the explicit 3rd-order Runge-Kutta scheme of Shu [7, 20]. This time-discretization scheme is total-variation diminishing (TVD) for Courant numbers $\text{CFL} \leq 1$, provided the underlying spatial discretization is TVD. We use $\text{CFL} = 1.0$ for all simulations. Pressure-Poisson equation and diffusive term are discretized by 2nd-order centered differences. In order to enhance computational efficiency, the discretized diffusive term is merged with the convective-flux approximation, eq. (21-22). Periodic boundary conditions allow to use a fast FFT-based Poisson solver with appropriate modified wave numbers for the first test case, where oddball modes are removed in spectral space. The pressure-Poisson solver for the channel-flow test case employs FFT in streamwise and spanwise direction and performs direct tridiagonal-matrix inversions in wall-normal direction. It should be noted that the oddball modes remain part of the solution and a staggered velocity arrangement is used for the second test case.

3.1. Homogeneous isotropic turbulence

As a first test case the numerical schemes are applied to decaying grid-generated turbulence. The computations are initialized with energy spectrum and Reynolds numbers adapted to the wind-tunnel experiments of Comte-Bellot and Corrsin [4], denoted hereafter as CBC. Among other space-time correlations CBC provides streamwise energy spectra for grid-generated turbulence at three positions downstream of a mesh. Table 3 of Ref. [4] gives corresponding 3D energy spectra which were obtained under the assumption of isotropy.

In the simulation this flow is modeled as decaying turbulence in a $(2\pi)^3$ -periodic computational domain, discretized on a collocated grid with 64^3 cells. Based on the Taylor hypothesis the temporal evolution in the simulation corresponds to a downstream evolution in the wind-tunnel experiment with the experimental mean-flow speed which is approximately constant. The energy distribution of the initial velocity field is matched to the first measured 3D energy spectrum of CBC. The SGS model is verified by comparing computational and experimental 3D energy spectra at later time instants which correspond to the other two measuring stations. For more details, particularly with regard to non-dimensionalization and initial-data generation, we refer to [9].

We present numerical results from implicit LES with the original ALDM scheme of Hickel *et al.* [9] and with successively simplified formulations proposed in the present paper. Figure 1a compares results obtained by 4th-order (\mathcal{C}_4) with those obtained by a second order (\mathcal{C}_2) Gaussian quadrature rule for the approximation of the filter operation $\mathbf{G}_l * \mathbf{G}_m$. It is found that the choice of the integration kernel has negligible effects on the computed energy spectra. The effect of omitting deconvolution in the transverse directions, i.e. the difference between full and simplified ALDM, is even smaller, see Fig. 1b.

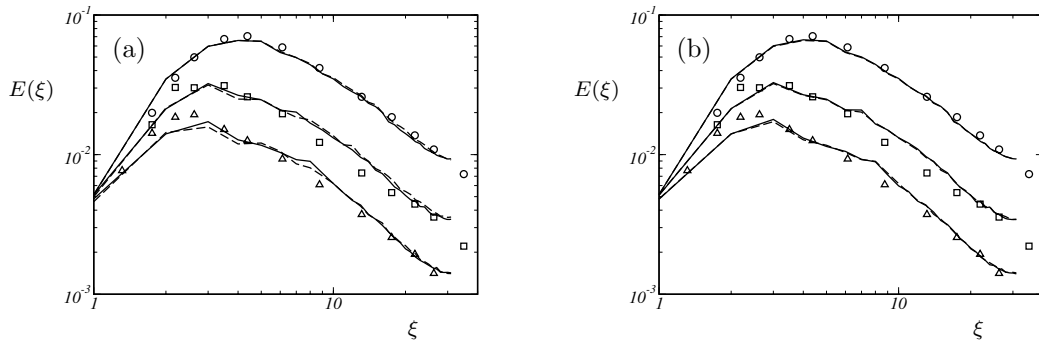


FIGURE 1. Instantaneous 3D energy spectra for LES with 64^3 cells of the Comte-Bellot – Corrsin experiment. (a) ——— ALDM with full deconvolution and C_2 , - - - - ALDM with full deconvolution and C_4 . (b) ——— SALD, - - - - ALDM with full deconvolution and C_2 . Symbols represent experimental data of Ref. [4].

3.2. Turbulent channel flow

As an example for anisotropic wall bounded turbulence, we simulate channel flow at $Re_{bulk} = 6875$ ($Re_\tau = 395$). Reference DNS data are provided by Moser *et al.* [18]. The computational domain measures $2\pi h \times 2h \times \pi h$ (streamwise \times wall-normal \times spanwise), where h is the channel half width. The spectral DNS of Moser *et al.* required $256 \times 193 \times 192$ grid points, whereas the computational grid of the present LES consists of $64 \times 68 \times 64$ cells. To accommodate the change of turbulence structure in the vicinity of solid walls, the grid is stretched in wall-normal direction

$$y(j) = -\frac{1}{\tanh(C_G)} \tanh\left(C_G - 2C_G \frac{j}{N_y + 1}\right), \quad (29)$$

where N_y is the number of cells in the wall-normal direction and $C_G = 2.0$ is the grid-stretching parameter.

Figure 2 shows profiles of mean velocity, Reynolds stresses, mean pressure, and pressure fluctuations from LES with the original and the simplified scheme. We observe a close match between both simulations and a good agreement with the reference DNS data. Inevitable differences between DNS and LES become most evident in the pressure fluctuations. Note the correct wall-asymptotic behavior of the Reynolds stresses of the LES. These results are encouraging and show that the implicit modeling approach gives good results for low-Reynolds-number wall-bounded flows although the model parameters were derived for isotropic turbulence and high Reynolds numbers.

4. CONCLUSION

In the present paper we have revised the numerical algorithm of the Adaptive Local Deconvolution Method (ALDM) for the incompressible 3D Navier-Stokes equations. We have given recommendations for the efficient implementation of ALDM. A simplification of the algorithm is proposed leading to the simplified adaptive local deconvolution (SALD) method. The resulting SALD method allows for considerable savings of computational resources. It is about twice as fast as the original ALDM, whereas the prediction power of the implicit model is preserved. The method is more efficient than a second-order central scheme with a dynamic Smagorinsky SGS model, in particular with regard to CPU time. Note that ALDM gives at least as good predictions as established explicit models [9].

For incompressible flows a fractional-step approach is pursued where a pressure correction is subsequently computed solving a Poisson equation. Flows in complex geometries with non-periodic directions require the use of iterative Poisson solvers, consuming typically about 60% and sometimes up to 95% of the computational time. It is recommended to use the SALD method whenever the CPU-time consumption of the discretization

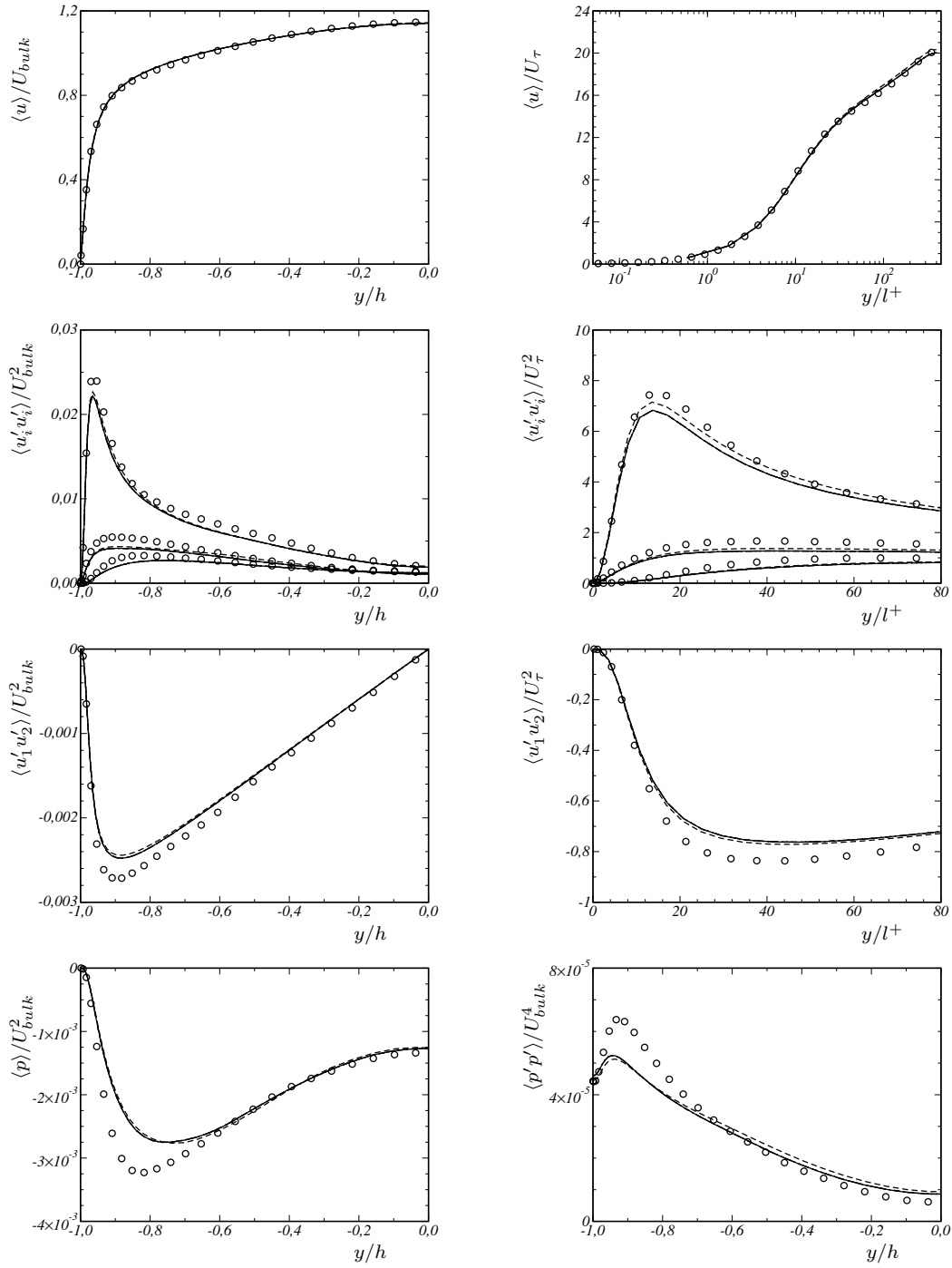


FIGURE 2. Mean profiles of velocity and pressure, Reynolds stresses, and pressure fluctuations for LES of turbulent channel flow at $Re_\tau = 395$. — SALD, - - - ALDM with full deconvolution and C_2 , \circ DNS of Moser *et al.* [18].

of the convective term is relevant. This holds in particular for flow configurations where efficient FFT-based solvers for the pressure-Poisson equation are available, such as homogeneous turbulence and channel flow.

REFERENCES

- [1] N.A. Adams, S. Hickel, and S. Franz. Implicit subgrid-scale modeling by adaptive deconvolution. *J. Comp. Phys.*, 200:412–431, 2004.
- [2] N.A. Adams, S. Hickel, T. Kempe, and J.A. Domaradzki. On the relation between subgrid-scale modelling and numerical discretization in large-eddy simulation. In *Proceedings of the Cyprus International Symposium on Complex Effects in Large Eddy Simulations (CY-LES2005)*, Limassol, Cyprus, 2005.
- [3] J.-P. Chollet. Two-point closures as a subgrid-scale modeling tool for large-eddy simulations. In F.Durst and B.E. Launder, editors, *Turbulent Shear Flows IV*, pages 62–72, Heidelberg, 1984. Springer.
- [4] G. Comte-Bellot and S. Corrsin. Simple Eulerian time correlation of full and narrow-band velocity signals in grid-generated ‘isotropic’ turbulence. *J. Fluid Mech.*, 48:273–337, 1971.
- [5] U. Frisch. *Turbulence*. Cambridge University Press, 1995.
- [6] S. Ghosal. An analysis of numerical errors in large-eddy simulations of turbulence. *J. Comput. Phys.*, 125:187–206, 1996.
- [7] S. Gottlieb and C.-W. Shu. Total variation diminishing Runge-Kutta schemes. *Math. Comput.*, 67:73–85, 1998.
- [8] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *J. Comp. Phys.*, 71:231–303, 1987.
- [9] S. Hickel, N.A. Adams, and J.A. Domaradzki. An adaptive local deconvolution method for implicit LES. *J. Comp. Phys.*, 213:413–436, 2006.
- [10] S. Hickel, S. Franz, N.A. Adams, and P.D. Koumoutsakos. Optimization of an implicit subgrid-scale model for LES. In *Proceedings of the 21st International Congress of Theoretical and Applied Mechanics (ICTAM)*, Warsaw, Poland, 2004.
- [11] S. Hickel, T. Kempe, and N.A. Adams. On implicit subgrid-scale modeling in wall-bounded flows. In *Proceedings of the EUROMECH Colloquium 469*, pages 36–37, Dresden, Germany, 2005.
- [12] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202–228, 1996.
- [13] T. Kempe, S. Hickel, and N.A. Adams. LES of the flow over periodic hills using implicit subgrid scale modeling. In *ERCRAFTAC Fluid Dynamics Colloquium*, Prague, Czech Republic, 2005.
- [14] A. Leonard. Energy cascade in large eddy simulations of turbulent fluid flows. *Adv. Geophys.*, 18A:237–248, 1974.
- [15] M. Lesieur. *Turbulence in Fluids*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 3 edition, 1997.
- [16] R. J. LeVeque. *Numerical methods for conservation laws*. Birkhäuser, Basel, Switzerland, 1992.
- [17] S. W. Liu, C. Meneveau, and J. Katz. On the properties of similarity subgrid-scale models as deduced from measurements in a turbulent jet. *J. Fluid Mech.*, 275:83–119, 1994.
- [18] R. D. Moser, J. Kim, and N.N. Mansour. Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$. *Phys. Fluids*, 11:943–945, 1999.
- [19] U. Schumann. Subgrid scale model for finite-difference simulations of turbulence in plane channels and annuli. *J. Comp. Phys.*, 18:376–404, 1975.
- [20] C.-W. Shu. Total-variation-diminishing time discretizations. *SIAM J. Sci. Stat. Comput.*, 9(6):1073–1084, 1988.
- [21] C.-W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. Technical Report 97-65, ICASE, NASA Langley Research Center, Hampton, Virginia, 1997.