Flow Turbulence Combust (2008) 81:77–96
DOI 10.1007/s10494-007-9124-0

POD Based Reconstruction of Subgrid Stresses for Wall Bounded Flows Using Neural Networks

Christian Wollblad · Lars Davidson

Received: 18 March 2007 / Accepted: 26 October 2007 / Published online: 7 December 2007

Abstract A zonal hybrid method for the computation of wall bounded flows was developed. Data from a direct numerical simulation of channel flow at Reynolds number 500 were filtered and the resulting subgrid stresses expanded in a series using proper orthogonal decomposition. The series was truncated. A feed forward neural network was found to be superior to linear stochastic estimation for estimating the coefficient of the series. The neural network and the orthonormal base from the expansion were shown by a priori tests to be suitable as a subgrid model for the innermost part of a boundary layer. The system was applied together with a Smagorinsky subgrid model to channel flow at Reynolds number 500 with good results. Possible improvements and generalization to other types of flows and higher Reynolds numbers are discussed.

Keywords LES · POD · Neural network · Subgrid stresses

1 Introduction

The high computational cost of large eddy simulations (LES) for unsteady wall bounded flows makes alternative approaches attractive. A common technique is to use one computational method close to the wall and another in the outer region. These so called hybrid methods, or zonal methods, have been explored for example by Davidson and Peng [1], Hamba [2] and Tucker and Davidson [3], all of whom used some RANS model close to the wall up to some matching line and LES outside that line. The argument that these methods will work is that the LES and the RANS

C. Wollblad (✉) · L. Davidson
Division of Fluid Dynamics, Department of Applied Mechanics,
Chalmers University of Technology, Hörsalvägen 7b,
SE41296 Gothenburg, Sweden
e-mail: wollblad@chalmers.se

formulations of the Navier–Stokes equations are the same when the stress terms are expressed in terms of the turbulent viscosity, ν_t. Hence any model can be used for ν_t. LES is used away from walls since it provides good accuracy at a reasonable computational cost everywhere except for regions close to walls. Close to walls, RANS is the only feasible method for calculations at high Reynolds number and is thus applied there.

There are however conceptual problems with this approach. For plane channel flow, hybrid methods give a sudden increase in the mean streamwise velocity somewhere outside the matching line [1]. The reason is that RANS gives much higher ν_t values and will therefore affect a much larger part of the turbulent spectrum, while LES has ν_t levels that damp only the smallest resolved scales. This will manifest itself as a jump in resolved turbulent scales at the matching line, as demonstrated in [4].

Some remedies have been suggested. Tucker and Davidson [3] used a one-equation $k - l$ model in both the RANS and LES regions, and the regions differed in how the filter length scale was chosen. The results for plane channel flow improved if the transition from RANS to LES was made in a smooth manner instead of abruptly changing the filter length scale definition at the matching line. Several authors have used forcing at the matching line to reintroduce resolved LES scales from modeled RANS scales. See for example [4–6] and [7]. Hybrid methods with forcing produce good mean velocity profiles for plane channel flows, but the forcing conditions are rather arbitrary. This is to some extent due to the problem pointed out by Temmerman et al. [8]: resolved turbulence is transported from the LES region into the RANS region, which responds by increasing the turbulent viscosity, thus diminishing the effect of the forcing. This is however not always the case. In the work of Davidson and Billson [6], forcing resulted in a lower ν_t values.

An alternative approach to the problem at the matching line is given by Hamba [2]. The value of ν_t on the RANS side of the matching line is much higher than on the LES side. To obtain the same level of ν_t on the LES side, the filter width has to be increased by approximately a factor 5, depending on the exact location of the matching line. However, the filter operator and the spatial derivatives in the Navier–Stokes equations do not commute in the mathematical sense but only to a second-order approximation in terms of the spatial derivative of the filter [9]. Hence, this rapid change in filter width will give rise to serious errors unless accounted for in some way. Hamba adopted a scheme with additional filtering of the data on the LES side of the matching line and obtained an improvement for computations of plane channel flow.

As none of the suggested remedies works without a substantial amount of *ad hoc* adjustment, either the RANS method or the LES method has to be replaced. Since the main objective is to be able to make LES without completely resolving the boundary layer, it would be directly counterproductive to remove the LES part. In this work, it is therefore the RANS model that is discarded and replaced. Data from a direct numerical simulation (DNS) are filtered to give resolved velocities and subgrid stresses. The subgrid stresses close to walls are expanded using proper orthogonal decomposition (POD), which gives an orthonormal base for the subgrid stresses. The series will be truncated to give a low-order base for representing those subgrid stresses. Two different methods for recombining the base elements given some LES data are investigated, viz. linear stochastic estimation (LSE) and neural networks (NN).

For completeness, it should be mentioned that there exist methods that cannot be considered strictly zonal but are still related to the zonal methods in the sense that they use a different concept close to walls than in freestream regions. For example, a RANS model can be used in the whole domain, but the ν_t level is given an explicit dependence on the grid size. As the grid becomes coarse relative to the flow structures, for example in boundary layers, the method becomes equivalent to RANS and, when the grid becomes very fine compared to the flow structures, the method goes towards DNS, at least in theory. See for example [7] and [10]. Another alternative approach is to use a large eddy formulation in the whole domain but, instead of resolving the inner parts of the boundary layer, the first cell is made several hundred viscous units high. The wall is then accounted for by shear stress boundary conditions, often called approximate boundary conditions, which are computed in some way from the resolved velocity field. See for example [11] and [12].

The organization of the rest of this article is intended as a walkthrough on the construction of the method. Results will be presented along the way to narrow down the alternatives so that it is always clear how the theory is intended to be used.

2 Filtering of DNS Data

Data are taken from a DNS of turbulent channel flow. The Reynolds number $Re = u_\tau H/\nu$ is equal to 500. u_τ is the friction velocity, H is channel half height and ν is the kinematic viscosity. The grid is Cartesian. The finite volume code is described in [1] and [13]. The resolution of the DNS is $\Delta x^+ = 49.1$, $\Delta z^+ = 12.3$ and $\min(\Delta y^+) = 0.3$. The wall normal stretching is 17%. A sketch of the geometry is shown in Fig. 1.

A box filter of streamwise and spanwise size $\Delta x^+ = 196$ and $\Delta z^+ = 49$ is used to obtain filtered data $\bar{u}_i$ and $\bar{p}$ and subgrid stresses $\tau_{ij}^{\text{tot}} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j$. No filtering is done in the wall normal direction. The filter is thus 4×4 DNS cells and is 2×2 times larger than recommended for a well resolved LES [14]. A larger filter of 8×8 DNS cells was also tested but was discarded since the LES then became underresolved in terms of $\Delta x/H$.

In incompressible flow, the spherical part of the subgrid stress tensor $1/3\tau_{kk}^{\text{tot}}\delta_{ij}$ is often included in the pressure. The current method is fully capable of modeling the total subgrid stress tensor, but, to avoid a pressure transformation at the matching

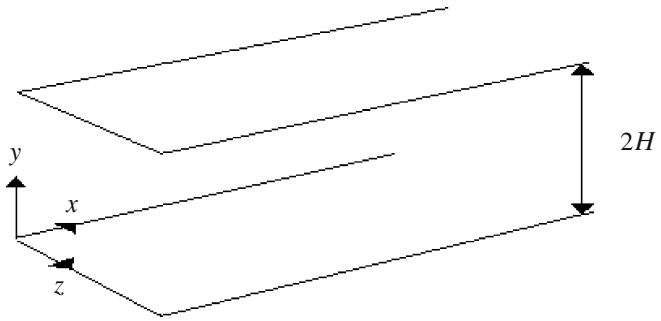

Fig. 1 Channel geometry

line, only the deviatoric part of the subgrid stress tensor, $\tau_{ij} = \tau_{ij}^{\text{tot}} - 1/3\tau_{kk}^{\text{tot}}\delta_{ij}$, will be modeled.

3 Proper Orthogonal Decomposition

POD is a method for expanding an arbitrary vector field, υ_i, defined on a domain Ω in an orthonormal $\mathcal{L}^2$ base $\{\phi_i^n\}_{n=1}^{\infty}$. Unlike Fourier expansions, for example, the basis function for the expansion is not known in advanced but is instead found by the variational problem "find functions ϕ_i that maximize $\langle|(\upsilon_i, \phi_i)|\rangle$ subject to the constraint $||\phi_i||^2 = 1$." $(\cdot, \cdot)$ is the $\mathcal{L}^2$ inner product and $< \cdot >$ is the assemble average over all functional forms of υ_i. The solution is given by the eigenvalue problem

$$\int_{\Omega} \langle \upsilon_i(x)\upsilon_j(x')\rangle \phi_j(x')dx' = \lambda\phi_i(x) \tag{1}$$

According to the Hilbert–Schmidt theorem [15], the eigenvectors will be an orthonormal $\mathcal{L}^2$ base on Ω. The basis functions, $\phi_i^n(x)$, are often referred to as POD modes, and any realization of the vector field $\upsilon_i(x)$ can be expressed as

$$\upsilon_i = \sum_{n=1}^{\infty} a^n \phi_i^n \tag{2}$$

where

$$a^n = \int_{\Omega} \upsilon_i(x)\phi_i^n(x)dx \tag{3}$$

are known as POD coefficients. An important property of POD can be derived by starting from the Spectral theorem [15]. After some manipulation, it can be shown that

$$\langle \upsilon_i(x)\upsilon_j(x')\rangle = \sum_{n=1}^{\infty} \lambda^n \phi_i^n(x)\phi_j^n(x') \tag{4}$$

where λ^n is the eigenvalue belonging to POD mode n. Combining this result with (2), it is obvious that $\langle a^n a^m\rangle = \delta_{nm}\lambda^n$. If υ_i is, for example, a turbulent velocity field, this expresses that λ^n is twice the average kinetic energy in POD mode n. Since the Hilbert–Schmidt theorem states that the eigenvalues are ordered with λ^1 being the largest eigenvalue, it follows that the truncated projection

$$\hat{\upsilon}_i = \sum_{n=1}^{N} a^n \phi_i^n \tag{5}$$

is better (measured in $\mathcal{L}^2$ norm) than any other representation of $\hat{\upsilon}_i$ using the same number of basis functions.

More on POD can be found in [16] and [17].

The theory is here applied using

$$\upsilon_i = [\tau_{11}\ \tau_{12}\ \tau_{13}\ \tau_{21}\ \tau_{22}\ \tau_{23}\ \tau_{31}\ \tau_{23}\ \tau_{33}] \tag{6}$$

The homogeneous directions x and z and the time are used for averaging and Ω is chosen to be $\{y^+ : y^+ \in [0, y^+_{\max}]\}$. In this work $y^+_{\max}$ is equal to 62. After reordering

the resulting nine-dimensional POD modes to a tensor of rank two, the above equations give the exact representation

$$\tau_{ij}(x, y, z, t) = \sum_{n=1}^{\infty} a^n(x, z, t)\phi_{ij}^n(y) \tag{7}$$

for all $y^+ \leq y^+_{\max}$. The upper limit $y^+_{\max}$ is chosen with possible future generalization in mind. Only the innermost 10% of a boundary layer is approximately universal [18]. Ten percent of the boundary layer thickness corresponds in this case to $y^+ = 50$, and hence Ω should not extend far beyond that limit.

To make the representation effective in terms of computational cost, the series must be truncated. The accuracy of the truncated POD representation of τ_{ij}, denoted $\hat{\tau}_{ij}$, is measured using the correlation coefficient

$$C_{ij} = \frac{\langle(\hat{\tau}_{ij} - \langle\hat{\tau}_{ij}\rangle)(\tau_{ij} - \langle\tau_{ij}\rangle)\rangle}{\hat{\tau}_{ij,\mathrm{rms}}\tau_{ij,\mathrm{rms}}} \tag{8}$$

Figure 2 shows C_{12} for three different choices of N. As can be seen, truncation after 30 POD modes gives an accurate enough representation with very little improvement when more modes are added. The same trend can be seen for all elements of τ_{ij}. Thirty modes is a relatively small number and can be compared to the number used in the work of Johansson [19]. He created a low-dimensional POD system to compute plane channel flow and used between 90 and 180 modes.

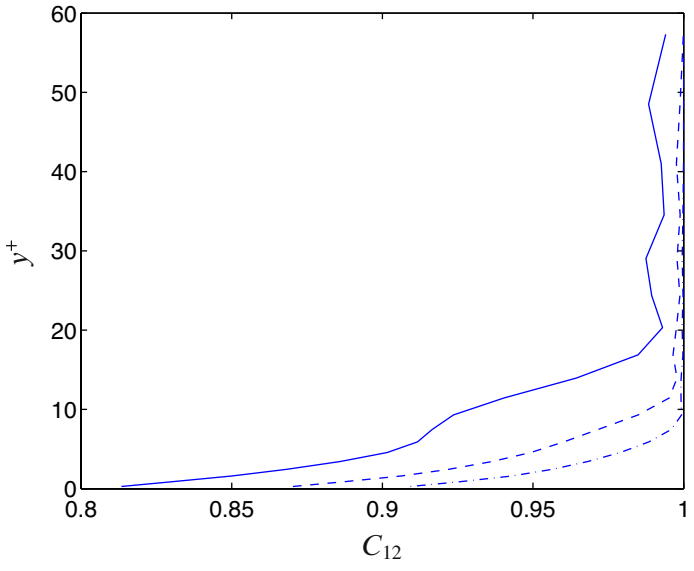

Fig. 2 Correlation coefficient for τ_{12} and $\hat{\tau}_{12}$. *solid line* $N = 20$, *broken line* $N = 30$, *dotted broken line* $N = 40$

4 SGS Reconstruction

To turn the (truncated) POD representation into a subgrid model, we ask 'what values do the POD coefficients assume for a given set of events $[h_1, \ldots, h_M]$'. The events are such that they can be calculated from filtered data, for example resolved velocity gradients. Two methods for calculating the most probable values of a^n given $[h_1, \ldots, h_M]$ are investigated, linear stochastic estimation (LSE) and neural networks (NN).

LSE was used in the work of Taylor and Glauser to reconstruct POD coefficients [20]. LSE assumes a linear relationship between the fluctuations of the coefficients a^n and the values of $[h_1, \ldots, h_M]$:

$$a^n \approx \vartheta^n = b_i^n h_i + \langle a^n \rangle \tag{9}$$

with summation over index i. For each n, the solution to the linear equation system $\langle h_i h_j \rangle b_i^n = \langle (a^n - \langle a^n \rangle) h_j \rangle$ minimizes the average square error, $\langle (\vartheta^n - a^n)^2 \rangle$. The vectors b_i^n can hence be calculated exactly. The simplicity in terms of construction and implementation and the low computational cost are the great advantages of LSE.

A small feed forward NN consisting of three layers is sketched in Fig. 3. This is the most common type of network for the reproduction of functions. The most important feature of NN is that they assume no functional form of the sought after relation. The first layer, the "input" layer, has two nodes that "send" their input data to each of the neurons in the next layer. Each arrow is known as a "connection" and the connection going from input node j to node k in the next layer has an associated weight, w_{kj}. The next layer, the open circles, is a so called "hidden" layer. Hidden node number k will produce an output given by

$$y_k = \phi \left(w_{kj} h_j + b_k \right) \tag{10}$$

where b_k is the bias of the neuron. The function ϕ is the activation function. For hidden neurons, the activation functions normally take values in the interval $[0, 1]$ or $[-1, 1]$. The neuron in the last layer, the triangle, is an output neuron and has the same functionality as those in the hidden layer, although the activation function might differ.

The training of a network refers to the optimization process in which the weights and the biases are decided. Available input-output relations are often split into three parts: A training set, a validation set and a cross-validation set. The training set is used in the optimization process to decide the weights and biases of the network. The validation set is used to control that the network does not become over-fitted

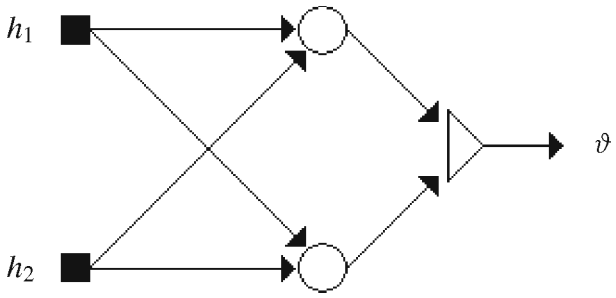

Fig. 3 A small feed forward neural network with two input nodes taking the scalar arguments h_1 and h_2, one hidden layer and one output node (giving the scalar ϑ)

to the training set. The cross-validation set is used to compare the performance of different network designs, e.g. different numbers of nodes, different activation functions etcetera.

A more complete description of neural networks can be found in textbooks on the subject, such as [21].

The NN used in this work, a feed forward network with biases and two hidden layers, is in theory capable of reproducing any continuous function. In practice, however, the number of nodes needed in each hidden layer for an exact reproduction can become very large and the training an impossible task. The tiny network in Fig. 3 has six weights and three biases to be determined, and the dimension of the optimization process quickly grows with the number of nodes. This puts severe constraints on which optimization methods can be used.

The network is constructed using Matlab's neural network toolbox [22]. The nodes of the hidden layers use activation functions of the form *tanh* while the output layer uses pure linear activation functions. The mean is removed from both the input and

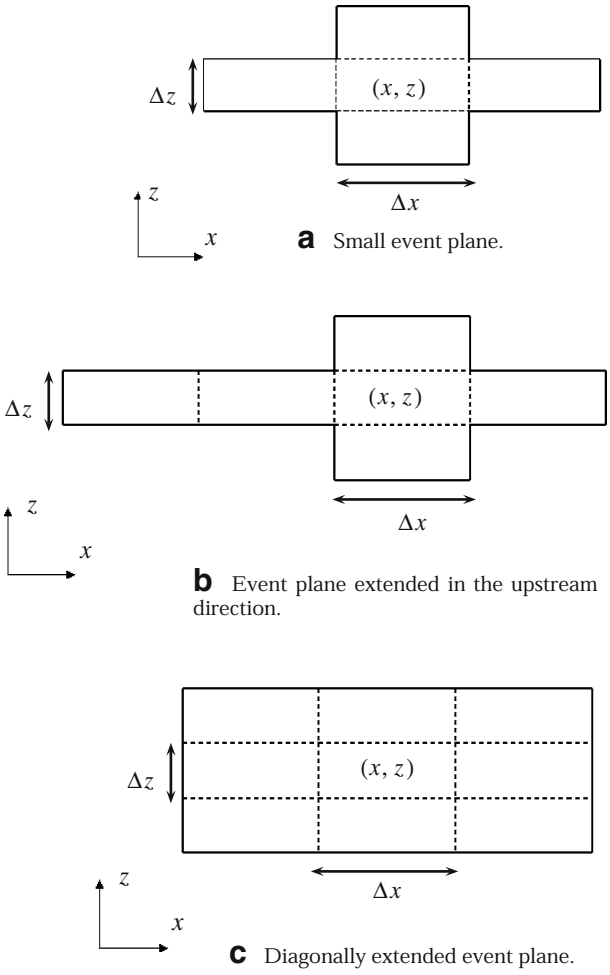

Fig. 4 The three tested alternatives of spatial extent of the event planes in the $x - z$ plane

a Small event plane.

b Event plane extended in the upstream direction.

c Diagonally extended event plane.

output data of the network, but only the input data are normalized. Half the available data set from the filtered DNS is used as the training set and the rest is split into a validation set and a cross-validation set. Matlab's neural network toolbox provides four conjugate gradient methods. These are the only methods recommended for training large networks intended to reproduce functions. Other training algorithms available are either intended for other types of networks or are too costly in terms of computer memory. For this application, none of the conjugate gradient methods could be concluded to be superior to the others. The results are presented for networks trained by the conjugate gradient algorithm with Polak–Ribiére updates.

Several different sets of events were tested. The sets were inspired by the work of Bagwell [23], Nicoud et al. [11] and Cole et al. [24]. Bagwell [23] utilized LSE to construct approximate boundary conditions for LES. He used an entire plane of the channel at constant y. Such a choice makes the method unsuitable for generalization to more complex geometries. Nicoud et al. [11], who used LSE for the same purpose, showed that more local events sufficed and that expanding the event field far in space did not give more accurate results. Cole et al. [24]. applied LSE on a jet mixing layer and found that "single-point estimates do not yield adequate representations of the instantaneous velocity field, but that two reference points located on opposite sides of the shear layer yield realistic estimates, with little gained by adding more reference points."

Contrary to the findings of Cole et al., it was found in the present study that both LSE and NN performed much better if events calculated in a plane at a distance $y^+ = y^+_{\max}/2 = 31$ from the walls were included. Three different shapes of the "event planes" were tested. These are shown in Fig. 4. Recall the coordinate system shown in Fig. 1. The larger planes (Fig. 4b and c) did not give any improvements compared to that shown in Fig. 4a. With reference to the results of Nicoud et al. [11] even larger planes were not tested. The mathematical formulations of the events were taken to be part of the terms in the finite volume formulation of the Navier–Stoke's equations:

$$
\begin{aligned}
u_i^+(x, y, z) \qquad & \text{for } y^+ = y^+_{\max}/2,\ y^+_{\max} \\
u_i^+(x \pm \Delta x, y, z) \qquad & \text{for } y^+ = y^+_{\max}/2,\ y^+_{\max} \\
u_i^+(x, y, z \pm \Delta z) \qquad & \text{for } y^+ = y^+_{\max}/2,\ y^+_{\max} \\
\partial P^+/\partial z^+(x, y, z \pm \Delta z/2) \qquad & \text{for } y^+ = y^+_{\max}/2,\ y^+_{\max} \\
\partial P^+/\partial y^+(x, y, z) \qquad & \text{for } y^+ = 0,\ y^+_{\max}/2,\ y^+_{\max} \\
\partial P^+/\partial y^+(x \pm \Delta x, y, z) \qquad & \text{for } y^+ = 0,\ y^+_{\max}/2,\ y^+_{\max} \\
\partial P^+/\partial y^+(x, y, z \pm \Delta z) \qquad & \text{for } y^+ = 0,\ y^+_{\max}/2,\ y^+_{\max} \\
\partial u^+/\partial y^+(x, y = 0, z) \qquad & \text{for } y^+ = 0 \\
\partial u^+/\partial y^+(x \pm \Delta x, y = 0, z) \qquad & \text{for } y^+ = 0 \\
\partial u^+/\partial y^+(x, y = 0, z \pm \Delta z) \qquad & \text{for } y^+ = 0 \\
\partial w^+/\partial y^+(x, y = 0, z) \qquad & \text{for } y^+ = 0 \\
\partial w^+/\partial y^+(x \pm \Delta x, y = 0, z) \qquad & \text{for } y^+ = 0 \\
\partial w^+/\partial y^+(x, y = 0, z \pm \Delta z) \qquad & \text{for } y^+ = 0
\end{aligned}
$$

where Δx and Δz are the constant cell length and width. The pressure terms deserve some extra attention. Since $1/3\tau_{kk}^{\text{tot}}\delta_{ij}$ has been subtracted from the subgrid stress tensor, the term $1/3\rho\tau_{kk}^{\text{tot}}$ has to be added to the pressure when the LSE matrix and NN are created, i. e. $P = \bar{p} + 1/3\rho\tau_{kk}^{\text{tot}}$. Observant readers can also see that the events do not include any streamwise pressure gradients. This is to facilitate generalization. In calculations of pressure driven channel flow, the pressure gradient is often replaced by a force term, and the value of the streamwise pressure gradient will thus depend on the implementation. It is tempting to simply exclude the events based on pressure, but our investigation shows that both LSE and NN perform much better when events based on pressure are included than when they are not.

The number of events, M, is 59. Sixty neurons in the first hidden layer and 40 neurons in the second hidden layer was the best configuration of those tested. It was found that two hidden layers give better results than a single hidden layer. It was also observed that the number of nodes in the hidden layers could not be decreased much without loss of performance. Neither a third hidden layer, nor more neurons in the hidden layers did improve the performance of the network.

Both LSE and NN recover the mean values of the subgrid stresses almost perfectly. In Fig. 5 the instantaneous untruncated subgrid stresses τ_{12} are compared with the reconstructed (and truncated) subgrid stresses calculated using coefficients estimated from LSE and from NN. The stresses estimated using LSE do not feature an especially higher correlation than subgrid stresses calculated with a Smagorinsky model. Stresses estimated from the neural network, however, feature a much higher correlation with the real stresses. Correlation coefficients C_{11}, C_{22} and C_{33} are all larger than C_{12} while the coefficients for the much less important τ_{13} and τ_{23} are somewhat smaller.

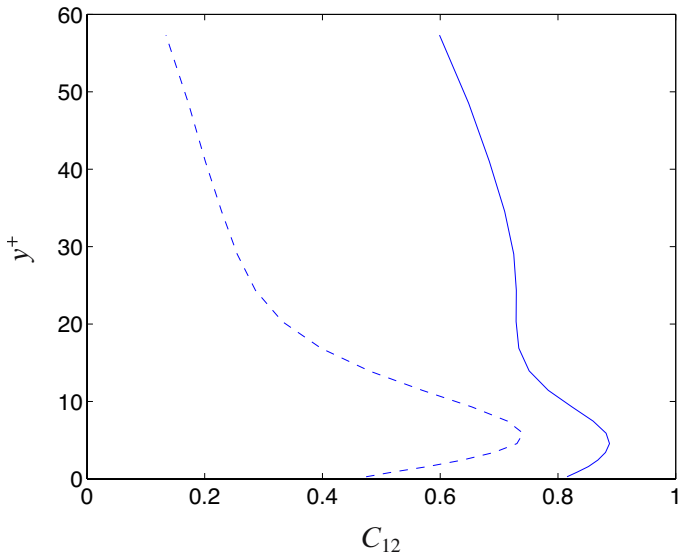

Fig. 5 Correlation coefficient for τ_{12} and $\hat{\tau}_{12}^{NN}$ (*solid line*) and for τ_{12} and $\hat{\tau}_{12}^{LSE}$ (*broken line*)

A possible explanation for the failure of LSE can be that $\hat{\tau}_{ij,\mathrm{rms}}^{LSE} \ll \tau_{ij,\mathrm{rms}}$ for all important components for the SGS tensor. This demonstrates that a rather small part of the POD coefficients' dependence on the events is linear. That NN is a superior choice for reconstruction of the POD coefficients was confirmed by implementing LSE as described in Section 6. As will be seen, a system based on NN will work while a system using LSE gives very poor results. Hence, LSE will not be discussed further.

5 Subgrid Model – A Priori Tests

As a first step we consider simulation of the same flow from which the system was constructed, i.e. turbulent channel flow at Reynolds number 500 but with cells that are four times larger in the x and z directions than in the DNS calculation. A zonal approach is chosen where the POD-NN system is used as a subgrid model close to a wall while a LES subgrid model is used everywhere else. This could be made in each iteration in any numerical scheme.

There are several necessary conditions that the system must fulfill in order to function as a subgrid model. One is to reproduce the averages of the subgrid stresses, and it has already been mentioned that it does. Another is to give a correct level of dissipation of turbulent kinetic energy to the subgrid scales, $-\tau_{ij}\bar{S}_{ij}$, where $\bar{S}_{ij}$ is the resolved strain-rate tensor. Figure 6 shows $\langle \tau_{ij}\bar{S}_{ij} \rangle$ calculated using both the real τ_{ij} and using $\hat{\tau}_{ij}^{NN}$. As can be seen, the dissipation is well reproduced. This is a key requirement since the only reason that the Smagorinsky model works is that it reproduces the roughly correct level of dissipation to subgrid scales [25].

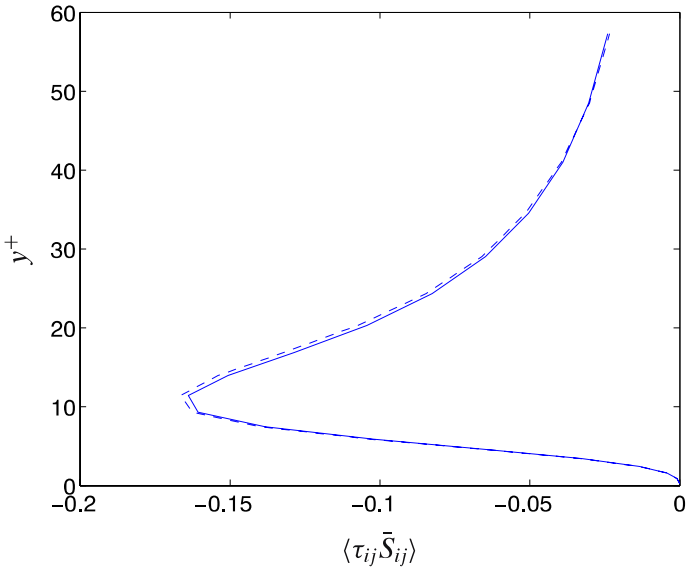

Fig. 6 Transfer of turbulent kinetic energy to subgrid scales calculated using real stresses (*solid line*) and modelled stresses (*broken line*)

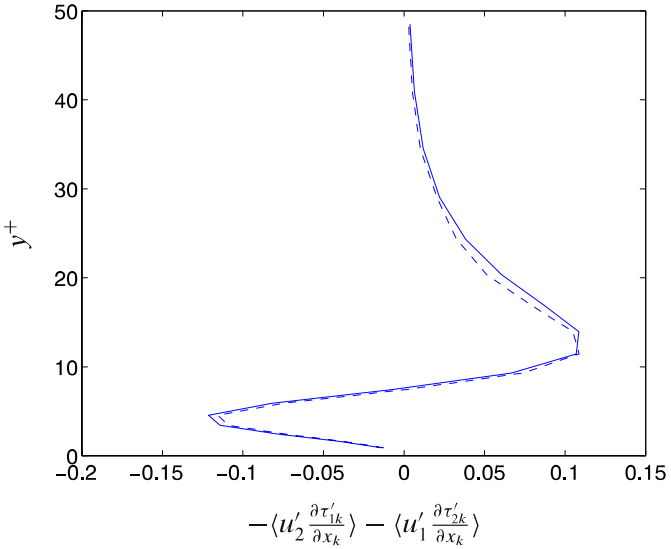

Fig. 7 Equation 11 for $\langle u'_1 u'_2 \rangle$ calculated using the real subgrid stresses (*solid line*) and modelled subgrid stresses (*broken line*)

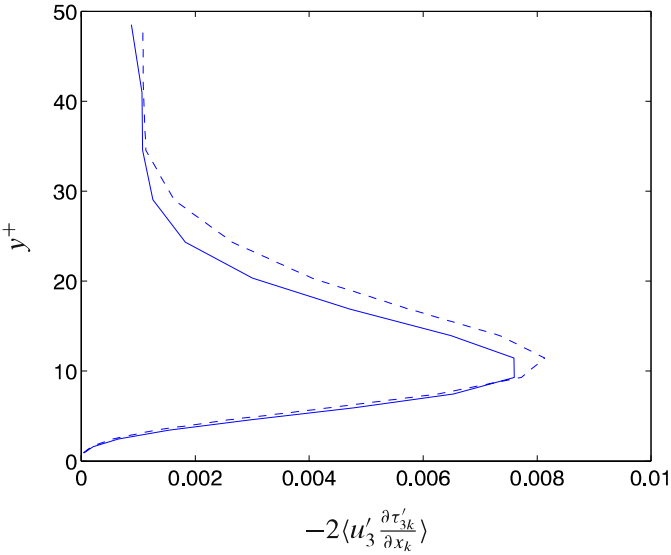

Fig. 8 Equation 11 for $\langle u'_3 u'_3 \rangle$ calculated using the real subgrid stresses (*solid line*) and modelled subgrid stresses (*broken line*)

A subgrid model must also affect the resolved Reynolds stresses in a correct way. The spatially filtered velocity field, $\bar{u}_i$, can be decomposed into a time averaged component, $\langle \bar{u}_i \rangle = U_i$, and a deviation from the time average, $\bar{u}_i - U_i = u'_i$. The transport equations for the resolved Reynolds stresses $\langle u'_i u'_j \rangle$ will contain the terms

$$-\langle u'_j \frac{\partial \tau'_{ik}}{\partial x_k} \rangle - \langle u'_i \frac{\partial \tau'_{jk}}{\partial x_k} \rangle \tag{11}$$

which represent all the effects of the subgrid stress tensor on the resolved Reynolds stresses [26]. Figures 7 and 8 show (11) for $\langle u'_1 u'_2 \rangle$ and $\langle u'_3 u'_3 \rangle$ calculated using the real stresses and the modeled stresses. Figure 7 is representative of the quality of equation (11) for $\langle u'_1 u'_1 \rangle$, $\langle u'_2 u'_2 \rangle$ and $\langle u'_1 u'_2 \rangle$, which are all strikingly good. Equation 11 is less well reproduced for $\langle u'_3 u'_3 \rangle$ but is still good enough. This test shows that the POD-NN system reproduces the subgrid stress tensors well enough to be able to calculate its first-order spatial derivatives.

A priori tests like these shown so far can however only give indications of the system's ability to function as a subgrid model.

6 Subgrid Model – Implementation and Results

An incompressible, finite volume code with a non-staggered grid arrangement is used [1]. For space discretization, central differencing is used for all terms. The Crank–Nicolson scheme is used for time discretization of all equations. The numerical procedure is based on an implicit, fractional step technique with a multi-grid pressure Poisson solver [27].

The Smagorinsky model is used in the center of the channel, i.e. further than $y^+_{\max} = 62$ viscous units from the walls. The model is given by

$$\tau_{ij}^{Smag} = -2\,(\Delta C_s)^2 \sqrt{2\bar{S}_{nm}\bar{S}_{nm}}\bar{S}_{ij} f_\mu \tag{12}$$

where $\bar{S}_{ij}$ is the resolved strain rate tensor, Δ is the filter width given by $(\Delta x \Delta y \Delta z)^{1/3}$ and $f_\mu = 1 - exp(-l^+/26)$ is the van Driest damping term. l^+ is the distance to the closest wall in viscous units. The model constant, c_S, is set to the standard value of 0.09.

The POD-NN system is applied to regions $y^+ \leq y^+_{\max}$ and $2H^+ - y^+ \leq y^+_{\max}$ where H is the channel half height (see Fig. 1). The divergences of the reconstructed subgrid stresses $\hat{\tau}_{ij}^{NN}$ are added as source terms to the discretized Navier–Stokes equations.

There are a few things that must be done to get a stable system. First, $\hat{\tau}_{ij}^{NN}$ needs to be underrelaxed to smooth its time history. If $\hat{\tau}_{ij}^{NN*}$ is the value from last iteration, the value used as a source term for the next iteration, $\hat{\tau}_{ij}^{NN**}$, is given by

$$\hat{\tau}_{ij}^{NN**} = c_r \hat{\tau}_{ij}^{NN}(\bar{u}_i^*, P^*) + (1 - c_r)\hat{\tau}_{ij}^{NN*} \tag{13}$$

where $\hat{\tau}_{ij}^{NN}(\bar{u}^*, P^*)$ is the value given by the POD-NN using the last known values of the velocity and pressure, $\bar{u}_i^*$ and P^*. Several values between 0.5 and 0.95 were tested for c_r, and the results seem to be independent of the exact value. The only noticeable difference is that the numerical scheme becomes more unstable with higher values

of c_r. The results shown later are calculated using $c_r = 0.8$. For consistency, the same amount of underrelaxation was applied to the Smagorinsky viscosity.

It is important that at least two iterations are made in each time step. If only one iteration is used, the lag between the velocity and the source terms created by the POD-NN system will create pressure fluctuations that grow in an unlimited fashion. Neither extremely short time steps nor underrelaxation of $\hat{\tau}_{ij}^{NN}$ can make the scheme stable if only one iteration per time step is used.

If $\hat{\tau}_{ij}^{NN}$ is used to account for 100% of the subgrid stresses in the regions close to the walls, the calculations eventually diverge. The measures described above lengthen the time before divergence and can even make the solution quasi-steady at solutions not far from the correct one. Stability can however be achieved by calculating the subgrid stresses by

$$\hat{\tau}_{ij} = c_b \hat{\tau}_{ij}^{NN} + (1 - c_b)\tau_{ij}^{Smag} \tag{14}$$

The blending coefficient, c_b, will be zero in the center region of the channel and non-zero in regions $y^+ \leq y^+_{\max}$ and $2H^+ - y^+ \leq y^+_{\max}$. It is possible to have c_b equal to a constant close to the walls and, since small values of c_b make the solution tend toward a pure Smagorinsky solution, the desire is to have c_b as close to one as possible. On the other hand, if c_b is constant close to one, there will be a sharp change in the approximation of τ_{ij} at the matching line, which creates a local velocity minimum there that can eventually destabilize the calculation. Therefore, c_b must be reduced in some way a few nodes before the matching line. Two different alternatives for c_b are shown in Fig. 9. Each marker represents a cell center, and the distributions will be referred to as distributions 1 and 2 as described in the caption to the figure.

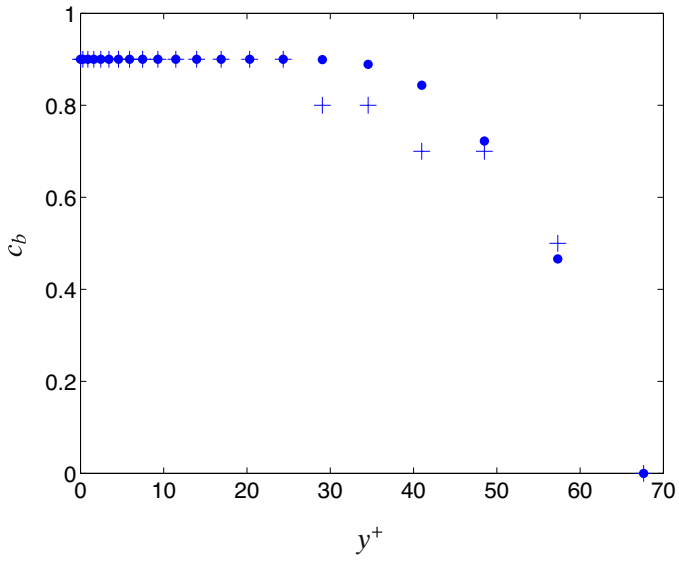

Fig. 9 Two distributions of the blending coefficient, c_b. *Solid dots* mark distribution 1 and plus marks distribution 2

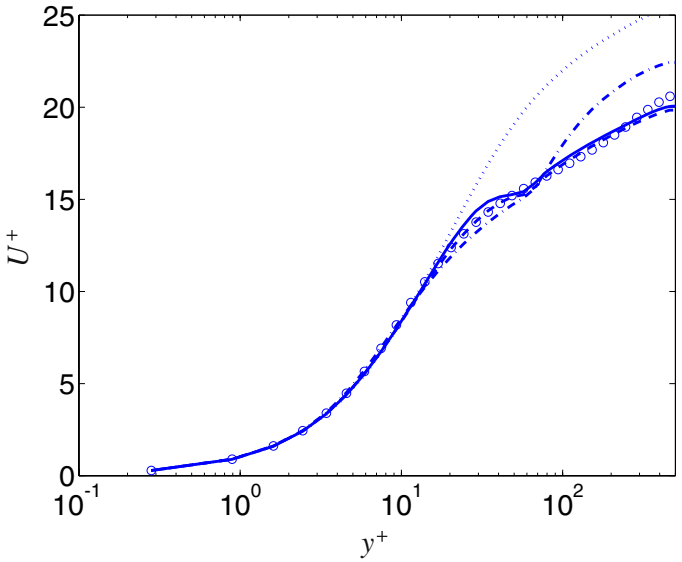

Fig. 10 Velocity profiles for DNS (*circle series*), pure LES (*dotted line*), hybrid LES-RANS (*dotted dash line*) and POD-NN using distributions 1 (*solid line*) and 2 (*broken line*)

Figure 10 shows some velocity profiles, all with $u_\tau = 1.00$ and Reynolds number equal to 500. Shown for reference are a solution using the Smagorinsky model in the whole domain, a DNS solution and a zonal RANS-LES solution with a one-equation $k - \ell$ model and the same matching line. The RANS-LES hybrid solution suffers from the deficiencies mentioned in the introduction, which results in too low resolved $\langle u'v' \rangle$ stresses. The pure Smagorinsky solution overshoots the velocity profile. Two profiles calculated using the POD-NN system are shown, one for each distribution of c_b shown in Fig. 9. Both of these calculations give better results than the pure LES and the LES-RANS hybrid method. Distribution 2 gives a slightly smoother solution than distribution 1, but both profiles display a local retardation of the velocity at the matching line. Neither distribution 1 nor distribution 2 can be claimed to be optimal, but the results indicate that the method can be tuned to obtain an almost exact velocity profile.

Figures 11 and 12 compare RMS values of resolved normal Reynolds stresses for the pure LES solution and the for the POD-NN solution using distribution 2. The stresses in Fig. 11 are typical for a poorly resolved LES with too high levels of streamwise stresses and too low levels of $\langle v'v' \rangle$ and $\langle w'w' \rangle$ stresses. Note that the DNS markers are the unfiltered Reynolds stresses, and the LES stresses should hence fall below these curves, but not to the degree displayed in Fig. 11. The levels displayed in Fig. 12 are much closer to what a filtered DNS solution looks like. The peak in the $\langle u'u' \rangle$ stress in Fig. 12 is located in the cell just outside the matching line. Despite the reduction of the blending coefficient, c_b, there will still be a huge gradient in the approximation of τ_{ij}, which can be seen in (11) to give rise to a nonphysically large production term in the $\langle u'u' \rangle$ equation and hence the extra peak in the $\langle u'u' \rangle$ stresses. This extra peak could have been removed by using a less aggressive reduction of c_b,

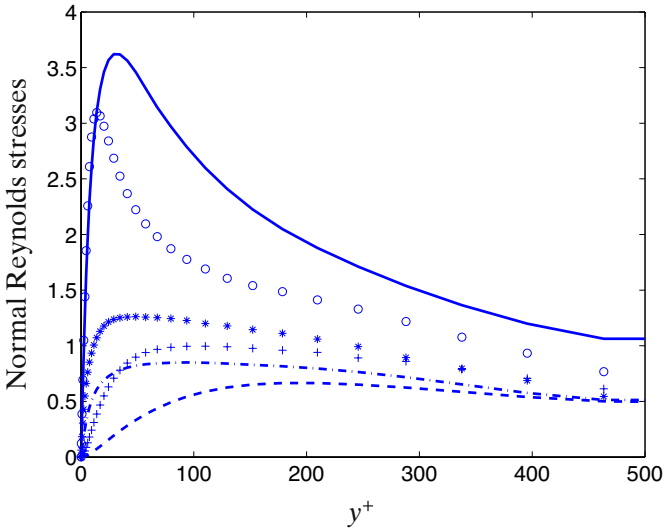

Fig. 11 *Circle series*, *asterisk series* and *plus series* are the RMS values of the $\langle u'u' \rangle$, $\langle w'w' \rangle$ and $\langle v'v' \rangle$ Reynolds stresses from DNS. *Solid line*, *dotted dash line* and *broken line* are the RMS values of the resolved $\langle u'u' \rangle$, $\langle w'w' \rangle$ and $\langle v'v' \rangle$ Reynolds stresses from the pure LES solution

but that would have been at the cost of a poorer velocity profile. The only way to get around this problem is to move the matching line further from the wall. This however cannot be done using the current DNS at Reynolds number 500 since, as already

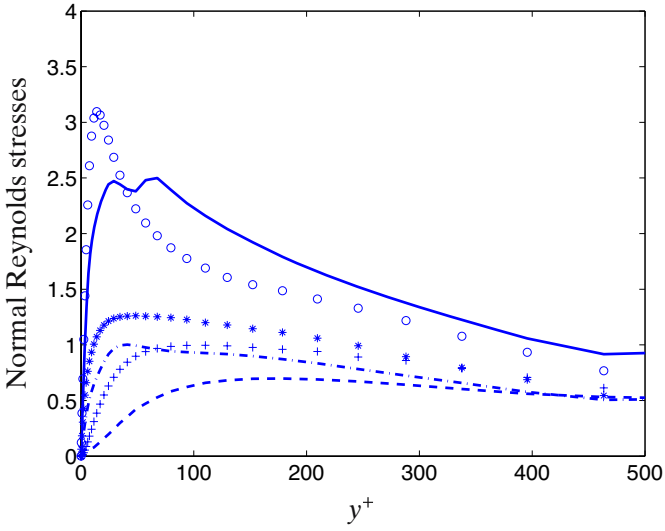

Fig. 12 *Circle series*, *asterisk series* and *plus series* are the RMS values of the $\langle u'u' \rangle$, $\langle w'w' \rangle$ and $\langle v'v' \rangle$ Reynolds stresses from DNS and *solid line*, *dotted dash line* and *broken line* are the RMS values of the resolved $\langle u'u' \rangle$, $\langle w'w' \rangle$ and $\langle v'v' \rangle$ Reynolds stresses from the POD-NN calculation using distribution 2

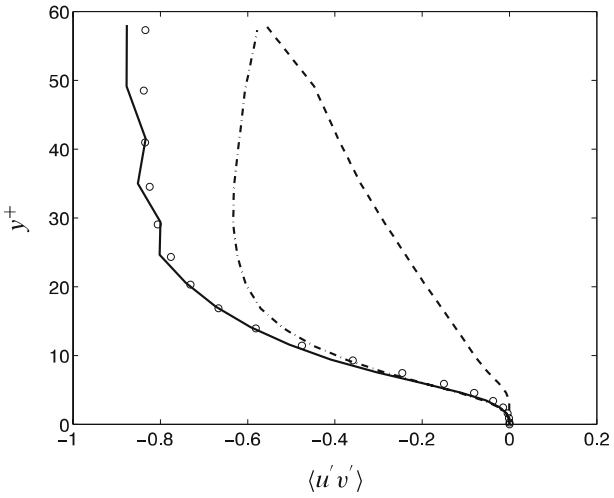

Fig. 13 Shear stresses from DNS (*circle series*), resolved Reynolds stresses $\langle u'v' \rangle$ (*broken line*) and $\langle \tau_{12}^{NN} \rangle$ (*dotted dash line*). The *solid line* is the total stresses

mentioned, it is only the innermost 10% of a boundary layer that is approximately universal [18]. This will be discussed further in Section 7.

Figure 13 shows the resolved $\langle u'v' \rangle$ Reynolds stresses and $\langle \hat{\tau}_{12}^{NN} \rangle$ along with the total shear stress given by

$$\langle u'v' \rangle + c_b \langle \hat{\tau}_{12}^{NN} \rangle + (1 - c_b) \langle \tau_{12}^{Smag} \rangle + \langle \tau_{12}^{visc} \rangle \tag{15}$$

For comparison, shear stresses from the DNS computation are also shown. Two features are especially prominent in Fig. 13. First, due to the large filter, $\hat{\tau}_{12}^{NN}$ is the dominant term in (15). Second, the fact that $\hat{\tau}_{12}^{NN}$ is not small near the matching line motivates the reduction of c_b in that region.

Figure 14 shows the resolved $\langle u'u' \rangle$ Reynolds stresses from DNS and POD-NN calculations using distribution 2 together with $\langle \hat{\tau}_{11}^{NN} \rangle$. The total streamwise stress is also shown. Note that $\hat{\tau}_{11}^{NN}$ approximates only the deviatoric part of the full subgrid

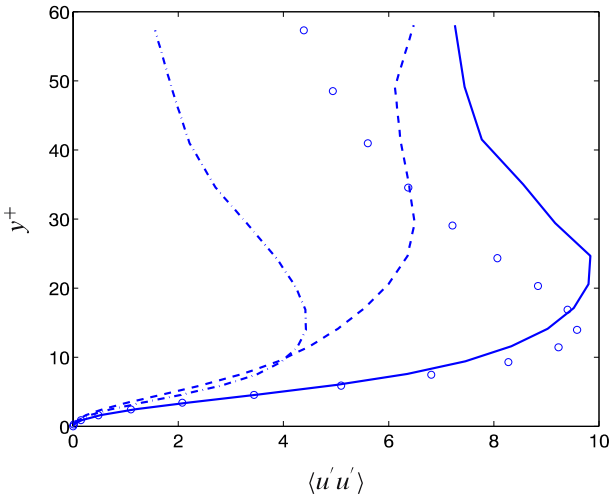

Fig. 14 $\langle u'u' \rangle$ stresses from DNS (*circle series*), resolved Reynolds stresses $\langle u'u' \rangle$ (*broken line*) and $\langle \tau_{11}^{NN} \rangle$ (*dotted dash line*). The *solid line* is the total stresses

stress element τ_{11}^{tot}. It can be seen that the current method does not suffer as heavily as other hybrid methods from the so called *double counting* phenomenon, except in regions where the POD-NN system is not dominant. Double counting is effectively that resolved plus modeled turbulent kinetic energy is much higher than a DNS at the same Reynolds number and is more the standard than the exception in hybrid methods (see for example [6]).

The Reynolds stresses obtained using distribution 1 are very similar to those obtained using distribution 2 and are hence not shown.

The additional cost of calculating and adding $\hat{\tau}_{ij}^{NN}$, compared to using the Smagorinsky model only, is 17%. This figure is of course heavily dependent on the size of the network and can probably be decreased as discussed in the next section.

7 Possible Improvements

The number of input nodes in the NN is given by the number of events and the number of output nodes by the number of POD modes used. The optimal number of nodes in the hidden layers and the number of hidden layers are however open questions.

As already mentioned, networks with one hidden layer performed strictly worse than networks with two hidden layers. This suggests that two hidden layers are indeed superior to one hidden layer. It was also mentioned that a third hidden layer did not give any improvement. However, this observation cannot be used to draw the conclusion that a third hidden layer cannot give an improvement. The third layer adds more degrees of freedom, which makes the training process more difficult. Along the same line of reasoning, it can be concluded that much fewer nodes in the hidden layers give the NN poorer performance. Nevertheless, a greater number of nodes can perhaps be beneficial if a more elaborate training method is used. One possible alternative to the gradient based methods is evolutionary methods. D. Knight used neural networks in his work with biomaterials and found that gradient based methods have a tendency to find a local minimum while evolutionary methods are much more likely to find an optimum close to the global one (Knight, private communication).

If no nodes can be removed without loss of performance, there might still be connections that are not needed, and some can even hamper the performance of the network. This is especially true for large networks such as the one used in this work. There are methods, for example brain surgery, for identifying and removing unnecessary and/or hampering connections [21]. Besides giving the network better performance, removing connections can also speed up the network considerably.

As mentioned in Section 6, if the POD-NN is used without underrelaxation and some blending with the Smagorinsky model, it will diverge. The constraints put on the implementation of the POD-NN system can hopefully be relaxed with a better trained network.

As mentioned in the last section, the need for blending with the Smagorinsky model and the "jump" in the mean velocity profile seem to be due to the positioning of the matching line. The model should hence be based on DNS simulations that allow the matching line to be placed further from the wall.

8 Notes About Generalization

For the model to be of any practical use, the POD modes must display some kind of Reynolds number independence, at least in a homeomorphic sense, as must the events used as input to the neural network. The viscous scaling used in this work is not generally useful. It decreases the Reynolds number dependence of the stresses, but, as can be seen in Fig. 15, there is still some dependence. The data are taken from the work of Jimenez et al. [28]. The viscous scaling is furthermore not defined in separation and reattachment points. Hence, a more suitable scaling method must be found.

In order to apply this technique to a general geometry, a DNS must be made for each type of wall bounded flow expected, one for attached flow, one for impinging flow, one for separation, one for reattachment and so on. There are then two alternatives. The first is to take all of the data sets and make one POD of them all and an NN that predicts the POD coefficients. This is relatively simple but can require very many POD modes to retain accuracy in the truncated representation. The second option is to make separate proper orthogonal decompositions with associated neural networks for each of the direct numerical simulations. This will keep down the number of POD modes needed in each case and hence the size of each NN. To decide which POD-NN should be used, a so called competitive network can be trained. A competitive network takes input and gives one of a number of possible discrete outputs. A common example is networks designed for item recognition. Such a competitive network has two outputs: either yes, the object is in the picture, or no, the object is not in the picture. A competitive network suitable for a general subgrid model of the kind discussed here should take the events as input and then give the POD-NN that is most suitable of those available to use as output.

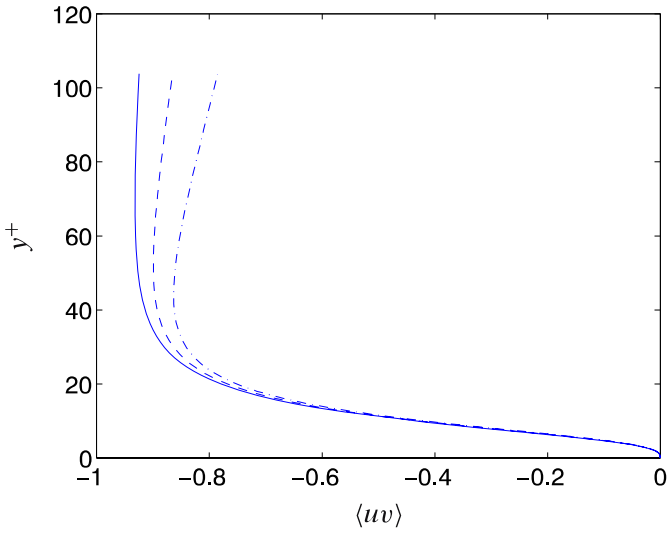

Fig. 15 Inner scaled $\langle uv \rangle$ Reynolds stresses from DNSfor $Re = 550$ (*dotted dash line*), $Re = 950$ (*broken line*) and $Re = 2{,}000$ (*solid line*)

A question that currently lacks an answer is the generalization to different filter widths. Mistakes are always made during work, and one of them was to run the model with $\Delta x^+ = 100$ and $\Delta z^+ = 25$ compared to $\Delta x^+ = 200$ and $\Delta z^+ = 50$ which are the filter widths for which the model is intended. The result was that the flow came more or less to a halt, probably because of too large subgrid shear stresses.

9 Conclusions

It has been shown that POD expansion of the deviatoric part of the subgrid stress tensor τ_{ij}^{tot} provides an orthonormal base that can be truncated at a relatively low dimension. The POD coefficients are much better reproduced by a neural network than by linear stochastic estimation.

A priori tests show that the POD-NN system should work as a subgrid model for cells close to a solid wall. Implementation in a finite volume code gave a system with some numerical difficulties, but very satisfactory results were obtained in the reproduction of the same channel flow from which the POD-NN system was created.

Possible improvements in the design of the NN were discussed as were some issues regarding generalization to higher Reynolds number and general geometries.

Acknowledgements This project is sponsored by the Swedish Energy Agency.

References

1. Davidson, L., Peng, S.-H.: Hybrid LES-RANS: a one-equation SGS model combined with a $k-\omega$ model for predicting recirculating flows. Int. J. Numer. Methods Fluids **43**, 1003–1018 (2003)
2. Hamba, F.: A hybrid RANS/LES simulation of turbulent channel flow. Theor. Comput. Fluid Dyn. **16**, 387–403 (2003)
3. Tucker, P.G., Davidson, L.: Zonal $k-l$ based large eddy simulations. Comput. Fluids **33**, 267–287 (2004)
4. Larsson, J., Lien, F.S., Yee, E.: Large eddy simulation of high Reynolds number channel flow on coarse grids. In: 13th Annual Conference of the Computational Fluid Dynamics Society of Canada, pp. 61–68. St. Johns, Newfoundland, Canada (2005)
5. Davidson, L., Dahlström, S.: Hybrid LES-RANS: an approach to make LES applicable at high Reynolds number. Int. J. Comput. Fluid Dyn. **19**, 415–427 (2005)
6. Davidson, L., Billson, M.: Hybrid LES-RANS using synthesized turbulent fluctuations for forcing in the interface region. Int. J. Heat Fluid Flow **27**(6), 1028–1042 (2006)
7. Piomelli, U., Balaras, E., Pasinato, H., Squires, K.D., Spalart, P.R.: The inner-outer layer interface in large-eddy simulations with wall-layer models. Int. J. Heat Fluid Flow **24**, 538–550 (2003)
8. Temmerman, L., Hadžiabdić, M., Leschziner, M.A., Hanjalić, K.: A hybrid two-layer URANS-LES approach for large eddy simulation at high Reynolds number. Int. J. Heat Fluid Flow **26**, 173–190 (2005)
9. Ghosal, S.: Mathematical and physical constraints on large-eddy simulation of turbulence. AIAA J. **37**, 425–433 (1999)
10. Speziale, C.G.: A combined large-eddy simulation and time-dependent RANS capability for high-speed compressible flows. J. Sci. Comput. **13**, 253–274 (1998)
11. Nicoud, F., Bagget, J.S., Moin, P., Cabot, W.: Large eddy simulation wall-modelling based on suboptimal control theory and linear stochastic estimation. Phys. Fluids **13**, 2968–2984 (2001)
12. Piomelli, U., Ferzieger, J., Moin, P., Kim, J.: New approximate boundary conditions for large eddy simulations of wall-bounded flows. Phys. Fluids **1**, 1061 (1989)
13. Davidson, L., Farhanieh, B.: CALC-BFC A Finite-Volume Code Employing Collocated Variable Arrangement and Cartesian Velocity Components for Computation of Fluid Flow and

Hyeat Transfer in Complex Three-Dimensional Geometries. Internal report 95/11, Chalmers University of Technology, Department of Thermo and Fluid Dynamics, Gothenburg, Sweden (1995)
14. Piomelli, U., Chasnov, J.R.: Large-eddy simulations: theory and applications. In: Henningson, D., Hallbaeck, M., Alfredsson, H., Johansson, A. (eds.) Transition and Turbulence Modelling, pp. 269–336. Kulwer (1996)
15. Debnath, L., Mikusiński, P.: Introduction to Hilbert Spaces with Applications. Academic Press, 2nd edition (1999)
16. Holmes, P., Lumley, J.L., Berkooz, G.: Turbulence, coherent structures, dynamical systems and symmetry. Cambridge Monographs on Mechanics (1996)
17. George, W.K.: Some thoughts on similarity, the POD, and finite boundaries. In: Gyr, A., Tsinober, A. (eds.) Trends in Mathematics, pp. 117–128 (1998)
18. Pope, S.B.: Turbulent Flows. Cambridge University Press, Cambridge, UK (2000)
19. Johansson, P.S.: Generation of inlet boundary conditions for numerical simulations of turbulent flows. Ph.D. thesis, Norwegian University of Science and Technology (2004)
20. Taylor, J.A., Glauser, M.N.: Towards practical flow sensing and control via POD and LSE based low-dimensional tools. J. Fluids Eng. **126**, 337–345 (2004)
21. Haykin, S.: Neural Networks. Prentice Hall International, Inc., international edition (1999)
22. Matlab documentation web page, http://www.mathworks.com/access/helpdesk/help/toolbox/nnet/ (2006)
23. Bagwell, T.G.: Stochastic estimation of near-wall closure in turbulence models. Ph.D. thesis, Department of Nulcear Engineering, University of Illinois at Urbana-Champaign (1994)
24. Cole, D.R., Glausser, M.N., Guezzennec, Y.G.: An application of stochastic estimation to the jet mixing layer. Phys. Fluids **4**, 192–194 (1991)
25. Liu, S., Katz, J., Meneveau, C.: Evolution and modelling of subgrid scales during rapid straining of turbulence. J. Fluid Mech. **387**, 281–320 (1999)
26. Davidson, L.: Transport equations in incompressible URANS and LES. Internal Report, 2006/01, Chalmers University of Technology, Department of Applied Mechanics, Division of Fluid Dynamics, Sweden (2006)
27. Emvin, P.: The full multigrid method applied to turbulent flow in ventilated enclosures using structured and unstructured grids. Ph.D. thesis, Dept. of Thermo and Fluid Dynamics, Chalmers University of Technology, Göteborg (1997)
28. DNS turbulent channel data, http://torroja.dmt.upm.es/ftp/channels/ (2006)