Improving accuracy of turbulence models by neural network

Satoshi Miyazaki
Graduate School of Information Sciences, Tohoku University, Sendai 980-8579, Japan

Yuji Hattori*
Institute of Fluid Sciences, Tohoku University, Sendai 980-8577, Japan
(Dated: December 4, 2020)

Neural networks of simple structures are used to construct a turbulence model for large-eddy simulation (LES). Data obtained by direct numerical simulation (DNS) of homogeneous isotropic turbulence are used to train neural networks. It is shown that two methods are effective for improvement of accuracy of the model: weighting data for training and addition of the second-order derivatives of velocity to the input variables. As a result, high correlation between the exact subgrid scale stress and the prediction by the neural network is obtained for large filter width; the correlation coefficient is about 0.9 and 0.8 for filter widths 48.8\eta and 97.4\eta, respectively, where \eta is the Kolmogorov scale. The models established by neural networks are close to but not identical with the gradient models. LES with the neural network model is performed for the homogeneous isotropic turbulence and the initial-value problem of the Taylor-Green vortices. The results obtained with the neural network model are in reasonable agreement with those of the filtered DNS. However, symmetry in the latter problem is broken since the neural network model does not possess rigorous symmetry under orthogonal transformations.

* hattori@ifs.tohoku.ac.jp; corresponding author
I. INTRODUCTION

Machine learning has been extensively applied to a wide range of problems in fluid dynamics, now forming an active area in data-driven fluid dynamics [1, 2]. Turbulence modelling is one of the important topics in this area since more accurate and robust turbulence models are wanted to predict and control turbulent flows and to optimize and design devices affected by turbulence. Many efforts have been devoted not only to RANS (Reynolds-Averaged Navier-Stokes) modeling [3–10] but also to LES (Large-Eddy Simulation) modeling [11–15] seeking better models beyond the human knowledge.

In the LES modeling, several works have shown that machine learning is a promising tool for improving the sub-grid scale (SGS) model beyond the existing models like the Smagorinsky model [16], the similarity model [17], the gradient model [18], and their variants [19–22]. Gamahara and Hattori [11] used a simple neural network (NN) to find a new model of the subgrid-scale (SGS) stress; the relation between the velocity gradient tensor as the input variables and the SGS stress as the output variable was established by neural networks. Data required for training and test of the model were provided by direct numerical simulation (DNS) of a turbulent channel flow. They showed that neural networks can establish models similar to the gradient model. \textit{A posteriori} test using the NN model was performed to show that neural networks are a promising tool for establishing a new subgrid model, although further improvement is required. Zhou et al. [13] also took an approach similar to Gamahara and Hattori [11]; they showed that addition of the velocity gradient tensor at points in the neighborhood to the input variables significantly improves the results including the correlation between the correct SGS stress and the prediction by the neural network.

Beck et al. [14] used a deep neural network with DNS data of weakly compressible isotropic turbulence. Xie et al. [15] also took an approach similar to Gamahara and Hattori [11]; they showed that addition of the velocity gradient tensor to points in the neighborhood to the input variables significantly improves the results including the correlation between the correct SGS stress and the prediction by the neural network.

There are two goals of different levels in this line of approach. One is to establish a turbulence model by machine learning as a black box; in this approach one does not pay attention to what is happening in e.g. neural networks pursuing just better prediction for a particular flow. Nowadays it can be done easily for a particular flow chosen for training with powerful tools of machine learning. However, whether the model gives accurate results for other flows is unknown since machine learning may have used some characteristics specific to the flow. The other goal is to establish an accurate and robust turbulence model which has an explicit expression as a function of resolved-scale variables and physical interpretation; this should be the ultimate goal.

Although recent works including those mentioned above showed some promising results, several important problems should be yet overcome to achieve our ultimate goal. They can be classified into three: (i) \textit{reliability} which implies that the SGS model obtained by machine learning should give accurate results within a reasonable range of errors; it should be numerically stable in actual LES calculations; (ii) \textit{universality} which implies that the obtained SGS model should be applicable not only to a wide range of the Reynolds numbers but also to flows of which types and/or geometries are different from the flow used in training process; and (iii) \textit{usability} which implies that the model can be implemented without difficulties and heavy numerical costs. Although the recent works mentioned above showed promising results, there has been no model that satisfies the three properties. Some NN models lack numerical stability in LES calculations. Applicability to flows other than the flow used in training is shown in some works but still limited. Moreover, no explicit expression of the NN model has been cultivated; in this sense no NN model has a firm physical basis, although our previous work pointed out similarity to the gradient model [11].

In order to solve the above problems, we should proceed step by step. Although a number of excellent tools of machine learning are available in these days, more knowledges and experiences should be explored to use them efficiently; we should clarify what are most useful in development of turbulence models. In particular, more should be investigated with simple tools like shallow neural networks which have a smaller number of parameters and options than deep neural networks and can allow us to infer an explicit expression of the turbulence model.

In this paper, we pursue methods for improving prediction accuracy of neural networks following the above line. Our objectives are to show that weighting training data and choice of the input variables can improve the prediction accuracy of neural networks and to check the accuracy in the actual LES calculations using the turbulence model constructed by neural networks.

This paper is organized as follows. Numerical methods are described in Sec. II. The turbulence models constructed by neural networks are evaluated by \textit{a priori} test in Sec. III: it is shown that the prediction accuracy of the NN model is improved by weighting training data and choice of the input variables. The NN models are used in actual LES calculations in Sec. IV; they are applied to two problems: the homogeneous isotropic turbulence and the initial value problem of the Taylor-Green vortices; the results are compared to LES using the existing models. Summary and future works are given in Sec. V.
II. NUMERICAL METHODS

A. Outline

The numerical procedure is similar to that of Gamahara and Hattori [11]. Our task consists of three steps: (i) development of a SGS model by a neural network, (ii) a priori test in which the SGS stress is compared between the developed model and the true values, and (iii) a posteriori test in which LES using the developed model is performed and evaluated.

In LES small fluctuations of a flow variable \( f \) are filtered out and we are concerned with the resolved-scale or grid-scale (GS) flow field \( \bar{f} = \int G(x') f(x - x') dx' \), where \( G \) is a filter function. The filtered Navier-Stokes equations for the GS flow field read

\[
\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\bar{u}_i \bar{u}_j) = -\frac{\partial \bar{p}}{\partial x_i} + \frac{1}{Re} \frac{\partial^2 \bar{u}_i}{\partial x_k \partial x_k} - \frac{\partial \tau_{ij}}{\partial x_j},
\]

(1)

\[
\frac{\partial \bar{u}_j}{\partial x_j} = 0,
\]

(2)

where the residual or SGS stress tensor

\[
\tau_{ij} = \bar{u}_i \bar{u}_j - \bar{u}_i \bar{u}_j
\]

(3)

depends not only on the GS flow field but also on the fluctuations. The Gaussian filter is chosen as a filter function except that the top-hat filter is also used for comparison in Sec. III B. The filter width \( \Delta \) is the same in the \( x \), \( y \), and \( z \) directions.

Our first step is to express the SGS stress tensor using the GS flow field

\[
\tau_{ij} = F[\bar{u}_i].
\]

(4)

In the expression above, \( \tau_{ij}(x) \) can depend on the GS velocity components \( \{\bar{u}_i\} \) and their derivatives \( \{\bar{u}_i, \partial_j \bar{u}_i, \partial_j \partial_k \bar{u}_i, \ldots\} \) at any point in general. In the present study, however, we seek a pointwise relation

\[
\tau_{ij}(x) = F(\{\partial_j \bar{u}_i(x), \partial_j \partial_k \bar{u}_i(x), \ldots\});
\]

(5)

since it can be widely applied to flows in complex geometry and physical interpretation will be easier. The actual choice of the independent (or input) variables will be explained in Sec. II E. In the present study we use neural networks to establish a functional relation between the GS flow field and the SGS stress tensor.

B. Training data and direct numerical simulation

In Gamahara and Hattori [11], the training data for neural networks were provided by DNS of a turbulent channel flow. It turned out that the near-wall region requires careful treatment both in a priori and a posteriori tests. Thus, in the present study, we use the data obtained by DNS of homogeneous isotropic turbulence, which is not affected by walls, as the training data. DNS was performed in the same way as in Ishihara et al. [28] and Hattori and Ishihara [29]. The three-dimensional incompressible Navier-Stokes equations were solved by the Fourier spectral method. Forcing was introduced at low wavenumbers to keep the kinetic energy constant. See Ishihara et al. [28] and Hattori and Ishihara [29] for the details. The number of the Fourier modes was \( N^3 = 512^3 \) or \( 1024^3 \). The values of simulation parameters and the turbulence characteristics are listed in Table I, where \( R_\lambda, k_{\text{max}}, \Delta t, \nu, \varepsilon, \eta, \) and \( T \) denote the Reynolds number based on the Taylor microscale, the effective maximum wavenumber, the size of the time step, the kinematic viscosity, the averaged rate of energy dissipation, the Kolmogorov scale, and the eddy turnover time.

| Case | \( N^3 \) | \( R_\lambda \) | \( k_{\text{max}} \) | \( 10^3 \Delta t \) | \( 10^4 \nu \) | \( \varepsilon \) | \( 10^4 \eta \) | \( T \) |
|------|----------|-----------|-----------|----------------|-------------|--------|-------------|-----|
| Case 1 | 512^3 173 | 214 1.0 | 7.0 | 0.0795 | 8.10 | 2.10 |
| Case 2 | 1024^3 268 | 483 0.625 | 2.8 | 0.0829 | 4.03 | 1.94 |

The training data were calculated and extracted from a snapshot of the DNS data. The number of data \( n_d \) was chosen from the range \( 5000 \leq n_d \leq 50000 \). How to choose the data is described in section III A. The trained neural networks were tested using data extracted from snapshots different from those used for training.
C. Neural network

As discussed in the introduction, we employ a shallow feed-forward neural network for training to establish a functional relation between the GS flow field and the SGS stress tensor. Our neural network consists of three layers: the input, hidden, and output layers. A single neuron of the $l$-th layer receives a set of inputs $\{X_{(l-1)}^j\}$ and then outputs $X_{i}^{(l)}$ which is calculated as

\begin{align}
X_{i}^{(l)} &= B \left( s_{i}^{(l)} + b_{i}^{(l)} \right), \\
s_{i}^{(l)} &= \sum_{j} W_{ij}^{(l)} X_{j}^{(l-1)},
\end{align}

where $B(z) = 1/(1 + e^{-\alpha z})$ is the activation function, $b_{i}^{(l)}$ is the bias parameter, and $W_{ij}^{(l)}$ is the weight. The bias parameters and the weights are corrected iteratively so that the final output $X_{i}^{(L)}$ approximates well the given SGS stress. The data of the first layer $\{X_{(1)}^{j}\}$ are given by the GS flow field. The back propagation is used as a method for training to minimize the difference between the output and the given SGS stress $\sum |X_{i}^{(3)} - \tau_{ij}|^2$.

The neural network is trained for one diagonal component $\tau_{11}$ and one off-diagonal component $\tau_{12}$ separately. The other components can be predicted by permutation of the indices: e.g. $\tau_{23}$ can be predicted by the network trained for $\tau_{12}$ by permutation of indices $(1, 2, 3) \rightarrow (2, 3, 1)$.

D. Sampling training data

In this study a sufficient amount of training data are available since the number of data is $N^3$ times the number of the instantaneous field data. The training data are normally chosen without any preference from the available data; they are chosen randomly or on particular parts (lines, sections etc.). However, these methods of data sampling can be disadvantageous since large values of the SGS stress, which are important in the filtered equation, are rarely encountered and training can be insufficient (Fig. 1). Therefore, we consider two methods of data sampling: One is the continuous sampling, in which the data on several lines are chosen; and the other is the uniform sampling, in which the data are chosen so that the probability density function (p.d.f.) of the sampled SGS stress becomes a uniform distribution within a range containing large values. In other words, the sampled data are weighted in the uniform sampling. We expect that the uniform sampling improves the efficiency of training.

Figure 1 shows the probability density function of $\tau_{11}$ obtained by two different methods of sampling. The total number of data is 5000. The data obtained by the continuous sampling are concentrated at small values. On the other hand, the probability density function of the uniform sampling is constant by definition.

E. Input variables

The choice of the input variables is one of the most important points for successful learning. The base set of the input variables is the velocity gradient tensor $\{\partial_j u_i\}$ as in Gamahara and Hattori [11], while the distance from the...
wall is absent for the homogeneous isotropic turbulence. In the present study, we add the second-order derivative of velocity to the input variables and see how the prediction is improved. There are variants of the above two sets in which some components are excluded. The sets of the input variables are listed below:

- Set S: (the Smagorinsky type) the components of the velocity gradient tensor sufficient to reproduce the Smagorinsky model

\[ \tau_{11} = F^{(S)} \left( \left\{ \overline{S}_{ij} \right\} \right), \]  

where \( \overline{S}_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \)

- Set D1: the first-order derivative of velocity

\[ \tau_{11} = F^{(D1)} \left( \left\{ \frac{\partial u_i}{\partial x_j} \right\} \right). \]  

- Set G1: the first-order gradient of the velocity components whose indices appear in the corresponding SGS stress component

\[ \tau_{11} = F^{(G1)} \left( \left\{ \frac{\partial u_i}{\partial x_j} \right\} \right), \]

\[ \tau_{12} = F^{(G1)} \left( \left\{ \frac{\partial u_i}{\partial x_j}, \frac{\partial u_j}{\partial x_j} \right\} \right). \]

- Set D2: the first-order and the second-order derivatives of velocity

\[ \tau_{11} = F^{(D2)} \left( \left\{ \frac{\partial u_i}{\partial x_j}, \frac{\partial^2 u_i}{\partial x_j \partial x_k} \right\} \right). \]

- Set G2: the first-order and the second-order derivatives of the velocity components whose indices appear in the corresponding SGS stress component

\[ \tau_{11} = F^{(G2)} \left( \left\{ \frac{\partial u_i}{\partial x_j}, \frac{\partial^2 u_i}{\partial x_j \partial x_k} \right\} \right), \]

\[ \tau_{12} = F^{(G2)} \left( \left\{ \frac{\partial u_i}{\partial x_j}, \frac{\partial^2 u_i}{\partial x_j \partial x_k}, \frac{\partial^2 u_i}{\partial x_j \partial x_k \partial x_k} \right\} \right). \]

The numbers of the input variables in the sets above are listed in Table IV.

F. Large-Eddy Simulation

In a posteriori test LES was performed using the trained neural networks. The equations (1) and (2) were solved numerically by the same method as DNS, while the SGS stress \( \tau_{ij} \) was calculated by the trained neural networks. LES with existing models such as the Smagorinsky model

\[ \tau^{(SM)}_{ij} - \frac{1}{3} \delta_{ij} \tau^{(SM)}_{kk} = -2(C_S \Delta)^2 (2 \overline{S}_{ij} \overline{S}_{ij})^{1/2} \overline{S}_{ij}, \]

where \( C_S \) is the Smagorinsky coefficient, and the Bardina model

\[ \tau_{ij}^{(B)} = \overline{u_i \overline{u_j}} - \overline{u_i} \overline{u_j} + \tau_{ij}^{(SM)}, \]

where the value of \( C_S \) can be different from that of the Smagorinsky model, was also performed for comparison. The gradient model

\[ \tau_{ij}^{(GM)} = \frac{\overline{S}^2}{12} \frac{\partial u_i}{\partial x_k} \frac{\partial u_j}{\partial x_k}. \]
which can be derived by the Taylor expansion, is compared to neural networks in a priori test. The next-order terms can be derived by the Taylor expansion and can be included in the above model [23–25]

\[
\tau_{ij}^{(EGM)} = \frac{\Delta^2}{12} \frac{\partial u_i}{\partial x_k} \frac{\partial u_j}{\partial x_k} + \frac{\Delta^4}{288} \frac{\partial^2 u_i}{\partial x_k \partial x_l} \frac{\partial^2 u_j}{\partial x_k \partial x_l}.
\] (18)

It is called the extended gradient model in this paper; this is also used in LES and in a priori test.

### III. RESULTS OF A PRIORI TEST

#### A. Effects of data sampling method

Table II shows the correlation coefficient between the exact SGS stress obtained by filtering DNS data of Case 1 with filter width \(\Delta = 8\Delta_{DNS} = 12.2\eta\) and the prediction by a trained neural network with the input variables of Set D1, which is hereafter called NN-D1 (similar abbreviation is used for the other Sets). The two methods of data sampling are compared. The uniform sampling gives better correlation than the continuous sampling as expected. The improvement of correlation is more pronounced for the off-diagonal components. Table III shows the error evaluated by the \(L^2\)-norm between the exact SGS stress and the prediction by the neural networks obtained by the two methods of data sampling. For the off-diagonal components the uniform sampling gives smaller errors than the continuous sampling, while the errors in the diagonal components are slightly larger for the uniform sampling than for the continuous sampling; the latter result is due to that the uniform sampling slightly overpredicts the magnitude of \(\tau_{11}\) as shown later in Fig. 3. This over-prediction, however, disappears when Sets D2 and G2 are used as the input variables. Thus, we use the uniform sampling in the following. Tables II and III also confirm that the neural networks trained with \(\tau_{11}\) and \(\tau_{12}\) give similar correlation for the other diagonal components (\(\tau_{22}\) and \(\tau_{33}\)) and off-diagonal components (\(\tau_{23}\) and \(\tau_{31}\)), respectively. Therefore, we show the results for \(\tau_{11}\) and \(\tau_{12}\) in the rest of this section.

| Sampling   | \(\tau_{11}\) | \(\tau_{22}\) | \(\tau_{33}\) | mean | \(\tau_{12}\) | \(\tau_{23}\) | \(\tau_{31}\) | mean |
|------------|----------------|----------------|----------------|------|----------------|----------------|----------------|------|
| continuous | 0.745          | 0.755          | 0.751          |      | 0.575          | 0.592          | 0.578          | 0.582 |
| uniform    | 0.821          | 0.818          | 0.825          | 0.822| 0.872          | 0.878          | 0.877          | 0.876 |

| Sampling   | \(\tau_{21}\) | \(\tau_{22}\) | \(\tau_{23}\) | \(\tau_{23}\) | \(\tau_{31}\) | \(\tau_{31}\) | mean |
|------------|----------------|----------------|----------------|---------------|----------------|----------------|------|
| continuous | 0.534          | 0.525          | 0.526          | 0.528         | 0.888          | 0.868          | 0.887 | 0.881 |
| uniform    | 0.664          | 0.655          | 0.642          | 0.654         | 0.556          | 0.541          | 0.543 | 0.547 |

#### B. Dependence on input variables

Next we seek a set of the input variables which maximizes the correlation between the exact SGS stress and the prediction by neural networks. In particular, the effects of adding the second-order derivatives of velocity on the prediction accuracy are investigated. The results of prediction by the neural networks with the sets of the input variables introduced in Sec. II E are also compared to the gradient model (GM) and the extended gradient model (EGM). The DNS data and the filter width are the same as in Sec. III A: Case 1 and \(\Delta = 8\Delta_{DNS} = 12.2\eta\).

Table IV shows correlation between the exact SGS stress and the prediction by the neural network for each set of the input variables. In addition to the Gaussian filter, the top-hat filter is also used as filter function for a limited number of sets. The table shows that correlation is low for the Smagorinsky type (Set S), while high correlation is obtained
for the other sets which use derivatives of velocity in accordance with Gamahara and Hattori [11]. NN-G1 gives better correlation than NN-D1, although the number of the input variables is smaller. It implies that the derivatives of \( u_i \) and \( u_j \) are important for estimate of \( \tau_{ij} \); it also implies that it is not always advantageous to increase the number of the input variables.

The most important point in Table IV is that addition of the second-order derivatives increases the correlation coefficients for the Gaussian filter significantly: e.g. 0.814 for \( \tau_{11} \) with NN-D1 increases to 0.946 with NN-D2, while 0.901 for \( \tau_{12} \) with NN-G1 increases to 0.975 with NN-G2. This is not the case for the top-hat filter as 0.746 for \( \tau_{11} \) with NN-D1 increases only a little to 0.783 with NN-D2. This difference in the improvement of correlation is closely related with the fact that the extended gradient model (18) is derived for the Gaussian filter, while it cannot be derived for the top-hat filter which is the product of the filters in the three directions.

The table also shows correlation for the gradient and the extended gradient models. The correlation is higher than the prediction by the neural networks; it is not striking as high correlation between the exact SGS stress and the gradient-type models has been reported in previous works [26].

Table IV also shows the number \( N_h \) of the neurons in the hidden layer used for each set of the input variables. These values were optimized by investigating the performance of the neural networks as shown in Fig. 2 for NN-D2 and NN-G2; the filter size is set to \( \Delta = 32\Delta_{DNS} = 48.7\eta \), which is four times the value used for obtaining correlation, to elucidate the dependence on \( N_h \). Compelling effects of changing \( N_h \) are observed in Fig. 2: the prediction accuracy increases with \( N_h \) for small values of \( N_h \), while overfitting can lower the regression ability of the neural network with Set G2 for large \( N_h \). The dependence on \( N_h \) is weak for \( N_h \geq 50 \). Since these effects depend on the set of the input variables, the optimized value of \( N_h \) is different between the cases in general.

![FIG. 2. (a) Correlation between exact SGS stress and prediction by NN-D2 and NN-G2 and (b) error. Dependence on the number \( N_h \) of neurons in the hidden layer is shown.](image)

Table IV. Correlation coefficient between exact SGS stress and prediction by neural network. Dependence on the set of the input variables is shown. The input variables in parentheses are used only for \( \tau_{12} \). Correspondingly, the values of \( N_i \) and \( N_h \) in parentheses are those for \( \tau_{12} \) which are different from \( \tau_{11} \). Case1, \( \Delta = 8\Delta_{DNS} = 12.2\eta \).

| Set | \( N_i \) | \( N_h \) | input variables | Corr (Gaussian) | Corr (top-hat) |
|-----|------|------|-----------------|-----------------|-----------------|
|     |      |      | \( \tau_{11} \) | \( \tau_{12} \)  | \( \tau_{11} \) | \( \tau_{12} \)  |
| S   | 6    | 20   | \( S_{ij} \)    | 0.227           | 0.180           | 0.118           | 0.140           |
| D1  | 9    | 20   | \( \partial_i \vec{u} \), \( \partial_j \vec{u} \) | 0.814           | 0.878           | 0.746           | 0.811           |
| G1  | 3 (6)| 10   | \( \partial_i \vec{u} \), \( \partial_j \vec{u} \) | 0.922           | 0.901           | 0.894           | 0.893           |
| D2  | 27   | 32   | \( \partial_i \vec{u} \), \( \partial_j \vec{u} \) | 0.946           | 0.972           | 0.783           | 0.838           |
| G2  | 9 (18)| 20 (32)| \( \partial_i \vec{u} \), \( \partial_j \vec{u} \) | 0.981           | 0.975           | 0.783           | 0.838           |
| GM  | 3 (6)| –    | \( \partial_i \vec{u} \) | 0.962           | 0.964           | –               | –               |
| EGM | 9 (18)| –    | \( \partial_i \vec{u} \), \( \partial_j \vec{u} \) | 0.993           | 0.994           | –               | –               |

Figures 3 and 4 compare joint probability density functions (joint p.d.f.s) of the exact SGS stress and the prediction by neural networks for different sets of the input variables. The joint p.d.f.s of the exact SGS stress and the gradient and extended gradient models are also shown. The figures show that addition of the second-order derivatives makes the distributions more concentrated near the diagonal lines of perfect match. The joint p.d.f.s for NN-G1 and NN-G2...
are more concentrated than those for NN-D1 and NN-D2, respectively, in accordance with the results on correlation. The distributions of $\tau_{11}$ for the gradient and extended gradient models are below the diagonal lines showing that these models underestimate $\tau_{11}$; the similar but weaker trend is observed for the magnitude of $\tau_{12}$. This underestimate is attributed to absence of the higher-order terms which are positive for the diagonal components $\tau_{ii}$. On the other hand, the distributions for the neural networks are nearly symmetric with respect to the diagonal lines.

Figures 5 and 6 show instantaneous distributions of the SGS stress components $\tau_{11}$ and $\tau_{12}$ on $z = 0$. The exact SGS stress, the prediction by the neural networks (NN-D1, NN-G1, NN-D2, and NN-G2), and the gradient and the extended gradient models are compared. All distributions are similar to the exact distribution; however, the prediction by NN-D1 involves small-scale structures which do not exist in the exact distribution, while similar structures are also visible for NN-G1. The distribution of the gradient model looks quite similar to the exact distribution, although the values are smaller than those of the exact one as the gradient model underestimates the SGS stress. On the other hand, NN-D2 and NN-G2 and the extended gradient model give nearly perfect prediction as the distributions are difficult to distinguish with each other. These results are in accordance with those on the correlation and the joint p.d.f.s.

C. Dependence on filter width

Next we investigate how the filter width $\Delta$ affects the regression ability of the neural network. The neural networks of Sets D1, G1, D2, and G2 are considered, while the gradient and the extended gradient models are included for comparison. The range of the filter width is $8\Delta_{\text{DNS}} = 12.2\eta \leq \Delta \leq 64\Delta_{\text{DNS}} = 97.4\eta$. In Fig. 7 the correlation coefficients and the error are plotted against the filter width. Naturally correlation becomes weak as the filter width becomes large, while the errors increase with the filter width. The correlation coefficients decrease rapidly for NN-D1 and NN-G1 which use only the first derivative of velocity. For the gradient and the extended gradient models, correlation is still high for the largest filter width, Corr being 0.94 at $\Delta = 64\Delta_{\text{DNS}}$ for the latter.

On the other hand, the error of $\tau_{11}$ is smaller for NN-D2 and NN-G2 than for the extended gradient model; on the whole, the error is not much different between NN-D2, NN-G2, the gradient model, and the extended gradient model.
As observed in previous subsections, the gradient and the extended gradient models underestimate the magnitude of the SGS stress, which explains why the error is not so small in spite of the high correlation.

As we see in the next subsection, it is most likely that the model constructed by the neural networks (NN models) is close to (but not identical with) the gradient and the extended gradient models; one of the advantages of the neural networks is to correct the trend of underestimation of the gradient models by training.

D. Comparison between NN model and gradient model

The results so far suggest that the models constructed by neural networks are close to the gradient and the extended gradient models. There are three points which support it: (i) the prediction by NN-G1 and NN-G2 is better than NN-D1 and NN-D2, respectively, which implies that the prediction is improved by removing the input variables which do not appear in the gradient or the extended gradient models from Set D1 or Set D2; (ii) adding second-order derivatives to the input variables of the neural network improves prediction significantly, which is the case for the extended gradient model compared to the gradient model; (iii) it is most likely that the trained neural network has simple structures since the number of neurons in the hidden layer required for high correlation is not so large in spite of the shallow structure of the neural network.

In order to check the relation between the models constructed by neural networks and the gradient models, cross correlation between them is investigated for the filter width larger than that in Sec. III B: $\Delta = 16\Delta_{DNS} = 24.4\eta$ (Table V). The correlation between NN-G2 and the extended gradient model is higher than that between NN-G2 and the exact SGS stress, supporting that the trained neural network is close to the extended gradient model. However, there is difference in the magnitude as observed in Fig. 8, which shows the joint p.d.f.s of the pairs of the neural network, the extended gradient model, and the exact SGS stress. Therefore, the trained neural network is close to but not identical with the extended gradient model.

If the trained neural network is close to the extended gradient model, the network trained for $\tau_{12}$ can be also used for $\tau_{11}$ by giving appropriate input variables according to eq. (18). Figure 9 shows the joint p.d.f. of the exact value of $\tau_{11}$ and the prediction by NN-G2 trained for $\tau_{12}$. The distribution is slightly wider than Fig. 9(c); the correlation coefficient is 0.915, while the error is 0.316, which are smaller and larger, respectively, than those between DNS and
FIG. 5. Spatial distribution of SGS stress $\tau_{11}$ on $z = 0$. Case 1, $\overline{\Delta} = 12.2\eta$. (a) (Filtered) DNS, (b) NN-D1, (c) NN-G1, (d) GM, (e) NN-D2, (f) NN-G2, (g) EGM.

Thus, correlation is slightly weaker.

E. Results on SGS production term

In LES the SGS production term $P = -\tau_{ij}\overline{S}_{ij}$, which is the energy transfer from the GS component to the SGS component, is an important quantity. In this subsection we focus on the SGS production term; correlation between the exact value of $P$ and that evaluated using the NN models and the existing models is investigated. In addition, we also train a neural network to predict the SGS production term directly and compare the results to those obtained in the preceding subsections. The DNS data of Case 1 are used, while the filter width is set to $\overline{\Delta} = 8\Delta_{\text{DNS}} = 12.2\eta$. Figure 10 shows the p.d.f. of $P$ calculated by filtering the DNS data. It shows that there is backscatter $P < 0$, although it is much smaller than $P > 0$.

Table VI shows the correlation of $P$ between the exact value calculated by filtering the DNS data and that calculated by the NN models and the existing models. The ratio of the spatial average of $P$ calculated as $\langle P_{\text{model}} \rangle / \langle P_{\text{DNS}} \rangle$ is also shown; it is not shown for the Smagorinsky model since the ratio depends on the Smagorinsky coefficient; actually,
FIG. 6. Spatial distribution of SGS stress $\tau_{12}$ on $z = 0$. Case 1, $\Delta = 12.2\eta$. (a) (Filtered) DNS, (b) NN-D1, (c) NN-G1, (d) GM, (e) NN-D2, (f) NN-G2, (g) EGM.

the ratio is unity for $C_s = 0.10$ which is smaller than the value $C_s = 0.164$ based on the Kolmogorov theory. In Table VI, NN-D1p and NN-D2p are the neural networks trained to predict $P$ with the input variables of Sets D1 and D2, respectively. For the NN models, the correlation coefficients are slightly smaller than those for $\tau_{ij}$ (Table IV). For the gradient and extended gradient models, the values are similar in Tables IV and VI. It is noteworthy that the Smagorinsky model shows high correlation of $\text{Corr} = 0.816$, although correlation is weak for the SGS stress (Table IV). NN-D1p and NN-D2p give higher or similar correlation than NN-D1 and NN-D2, respectively. The ratio $\langle P_{\text{model}} \rangle / \langle P_{\text{DNS}} \rangle$ is close to unity for NN-D1p and NN-D2p, while it is smaller than unity for NN-D1 and NN-D2. This implies that setting the SGS production term as the target function can improve LES modeling. This point will be further addressed in the future work. It is pointed out that the gradient and the extended gradient models also underestimate $P$.

Figure 11 shows the joint p.d.f.s of the exact SGS production term calculated using the DNS data and the prediction by the NN models and the existing models. In the joint p.d.f. of NN-D2, the distribution is concentrated below the diagonal line implying that it underestimates $P$ on average. This trend is not observed for the other models. It is also observed that $P \geq 0$ for the Smagorinsky model, which is regarded as one of the drawbacks of the Smagorinsky model. Figure 12 compares spatial distributions of $P$ on $z = 0$. All of the NN models and the existing models
FIG. 7. (a,b) Correlation coefficients between exact SGS stress and prediction by neural network and (c,d) error; those between the exact SGS stress and the gradient and the extended gradient models are included. Dependence on the filter size is shown. (a,c) $\tau_{11}$, (b,d) $\tau_{12}$.

TABLE V. Correlation between NN-G2 and extended gradient model (EGM) and the error. Those with the exact SGS stress (DNS) are included for comparison.

|            | (DNS,EGM) | (DNS,NN-G2) | (EGM,NN-G2) |
|------------|-----------|-------------|-------------|
| Corr       | $\tau_{11}$ | $\tau_{12}$ | $\tau_{11}$ | $\tau_{12}$ | $\tau_{11}$ | $\tau_{12}$ |
| error      | 0.282     | 0.221       | 0.223       | 0.346       | -           | -           |

reproduce successfully the exact distribution on the whole, although negative values are absent in the Smagorinsky model and NN-D2 and NN-D2p overestimate backscatter in several regions. Based on the results in this subsection, we use NN-G2 in a posteriori test in the next section.

TABLE VI. Correlation of SGS production term $P = -\tau_{ij}\overline{S_{ij}}$ between exact value and prediction by NN models and existing models. The ratio of the spatial average of $P$ is also shown except for the Smagorinsky model. Case 1, $\Delta = 12\eta$.

| Model      | NN-D1 | NN-G1 | NN-D2 | NN-G2 | GM | EGM | Smag. | NN-Dp | NN-D2p |
|------------|-------|-------|-------|-------|----|-----|-------|-------|-------|
| Corr       | 0.521 | 0.792 | 0.890 | 0.923 | 0.949 | 0.992 | 0.816 | 0.826 | 0.902 |
| $\langle P_{\text{model}} \rangle / \langle P_{\text{DNS}} \rangle$ | 0.577 | 0.803 | 0.639 | 1.028 | 0.886 | 0.984 | - | 0.928 | 1.085 |
FIG. 8. Joint p.d.f. of SGS stress. Case 1, $\Delta = 24.4\eta$. (a) $\tau_{11}$, (b) $\tau_{12}$. (a,d) DNS vs. NN-G2, (b,e) DNS vs. EGM, (c,f) NN-G2 vs. EGM.

FIG. 9. Joint p.d.f. of exact value of $\tau_{11}$ and prediction by NN-G2 trained for $\tau_{12}$. Case 1, $\Delta = 24.4\eta$.

IV. RESULTS OF A POSTERIORI TEST

A. Stabilization

In this section we implement the neural network with the input variables of Set G2 (NN-G2) to actual LES (a posteriori test) and investigate the accuracy by comparing the results with those obtained with two existing models: the Smagorinsky and Bardina models. An important remark here is that the NN model should be stabilized. Figure 13 shows the energy spectrum of homogeneous isotropic turbulence obtained by LES using the NN model without stabilization. The energy at small scales grows rapidly which is unphysical; the calculation diverges eventually. Similar divergence has been reported by Beck et al. [14]. Thus we stabilize the NN model by clipping. Namely, the SGS stress
FIG. 10. P.d.f. of SGS production term $P$. Case $1, \Delta = 12.2\eta, 24\eta, 48.7\eta$.

FIG. 11. Joint p.d.f. of SGS production term $P$. Case $1, \Delta = 12.2\eta$. The horizontal axis is the exact value obtained by filtering the DNS data. The vertical axes are predictions by (a) NN-D2, (b) NN-G2, (c) gradient model, (d) extended gradient model, (e) Smagorinsky model, (f) NN-D2p.

of the NN model $\tau_{ij}$ is clipped as

$$\tau_{ij}^* = \begin{cases} \tau_{ij}, & \tau_{ij} S_{ij} \leq 0 \\ 0, & \text{otherwise}, \end{cases}$$

which is introduced in Lu et al. [27] for stabilizing the gradient model. By this clipping procedure the SGS stress is forced to be zero to prevent backscatter. The stabilized NN model is successfully used in LES as shown below. The same clipping procedure was applied to the extended gradient model.
FIG. 12. Spatial distribution of SGS production term $P$. Case 1, $\Delta = 12.2\eta$. (a) DNS, (b) NN-D2, (c) NN-G2, (d) extended gradient model, (e) Smagorinsky model, (f) NN-D2p.

FIG. 13. Time evolution of energy spectrum of homogeneous isotropic turbulence. LES with the NN model without stabilization.

B. Homogeneous isotropic turbulence

First, we show the results of LES of homogeneous isotropic turbulence. Four cases shown in Table VII are considered. The initial conditions were obtained by filtering the DNS data of the homogeneous isotropic turbulence (Table I). In Table VII, $N_{LES}^3$, $R_\lambda$, $\Delta_{DNS}$, $\eta$, $k_{max}$, and $E$ are the number of modes used in LES, the Reynolds number based on the Taylor microscale, the grid size of the corresponding DNS, the Kolmogorov scale, the magnitude of the largest effective wavevector, and the energy of the corresponding (unfiltered) DNS data; $E$ is the energy after filtering. LES with the existing models were also performed for comparison; the Smagorinsky coefficient $C_s$ was adjusted by try and error to give better results. The filter width is the same for Cases HIT1-1 and HIT2-1, while it is the same for Cases HIT1-2 and HIT2-2; the ratio of the filter width to the Kolmogorov scale is the same for Cases HIT1-2 and HIT2-1. The grid size in LES is set to $\Delta_{LES} = \Delta/2$. The size of the time step is fixed to $10^{-3}$. Forcing was introduced at low
TABLE VII. Simulation parameters of LES of isotropic homogeneous turbulence.

| Case  | \(N_{\text{LES}}^3\) | \(R_A\) | \(\Delta / \Delta_{\text{DNS}}\) | \(\Delta / \eta\) | \(k_{\text{max}}\) | \(E / E^\text{s}\) | \(C_I\) |
|-------|-------------------|------|------------------|-------|----------|--------|------|
| HIT1-1| 128\(^3\) | 173  | 8        | 12.2  | 60       | 0.942  | 0.10 |
| HIT1-2| 64\(^3\)  | 173  | 16       | 24.4  | 30       | 0.865  | 0.10 |
| HIT2-1| 128\(^3\) | 268  | 16       | 24.4  | 60       | 0.911  | 0.12 |
| HIT2-2| 64\(^3\)  | 268  | 32       | 48.7  | 30       | 0.831  | 0.12 |

wavenumbers to keep the kinetic energy constant as in DNS. The simulation was stopped at \(t = 6\), which is about three times the eddy turnover time based on the integral length scale; the statistics below were calculated using the data at the final time. For the NN model, the neural network was trained using the corresponding DNS data with the filter width shown in Table VII.

The rate of energy dissipation, the integral length scale, and the Taylor microscale are compared in Table VIII; also included are the skewness and the flatness of the p.d.f.s of \(\partial_1 \overline{u_1}\) and \(\partial_1 \overline{u_2}\), which will be discussed later. The values of DNS are included for reference. The results obtained by LES should be compared to the values obtained for the filtered DNS data, which are denoted by 'Filtered' in the table. The rate of energy dissipation in Table VIII is the viscous dissipation of the filtered velocity field; it is smaller than the rate of dissipation of the total energy since energy is transferred to the subgrid scales through the production term \(P_1\), of which ratio increases with the filter width; 67% and 93% of the energy dissipation are due to the energy transfer to the subgrid scales in Cases HIT1-1 and HIT2-2, respectively.

Table VIII shows that the Bardina model overpredicts the energy dissipation in comparison to the other models; the parameters should be carefully tuned by e.g. a dynamic procedure to make the Bardina model accurate. The integral length scale \(\lambda_I\) calculated by

\[
\lambda_I = \frac{3\pi}{4} \frac{\int_0^{k_{\text{max}}} k^{-1} E(k) dk}{\int_0^{k_{\text{max}}} E(k) dk}
\]

(20)
does not change very much by filtering: \(\lambda_I\) of DNS data increases \(5 \sim 16\%\) by filtering (Table VIII). The integral length scale obtained with the NN model is not far from that of the filtered data, although it is slightly larger than those obtained with the other models. The longitudinal Taylor microscale \(\lambda_\parallel\) and the transverse Taylor microscale \(\lambda_\perp\) are calculated by

\[
\lambda_\parallel^2 = \frac{\langle u_1^2 \rangle}{\left< \left( \frac{\partial u_1}{\partial x_1} \right)^2 \right>}, \quad \lambda_\perp^2 = \frac{\langle u_2^2 \rangle}{\left< \left( \frac{\partial u_2}{\partial x_2} \right)^2 \right>}
\]

(21)

In contrast to the integral length scale the Taylor microscales increase by filtering since small-scale fluctuations below the filter width are removed: \(\lambda_\parallel\) of the filtered data is about \(1.5 \sim 3.4\) times that of DNS (Table VIII). The relation \(\lambda_\parallel = \sqrt{2} \lambda_\perp\), which is known for the homogeneous isotropic turbulence, is also confirmed in Table VIII. On the whole the Taylor microscales obtained with the NN model are in good agreement with those of the filtered data.

Figure 14 compares the energy spectrum between DNS, filtered DNS, and LES. The spectrum of the filtered DNS deviates from that of DNS at a wavenumber within the inertial subrange except for Case HIT1-1, showing that the wavenumber corresponding to the filter width is within the inertial subrange for the other three cases. All spectra of LES are in reasonable agreement with the filtered DNS, although the energy spectrum of the NN model is slightly smaller than that of the filtered DNS.

Figure 15 shows the distributions of the magnitude of vorticity on \(z = 0\) for the filtered DNS data and LES of Case HIT2-1. We cannot expect agreement between the distributions since small differences are amplified exponentially. However, some features of the distributions can be compared: tube-like structures are observed in all cases; the magnitude of vorticity is comparable between all cases. In particular, it is confirmed that the NN model successfully gives vorticity distributions whose features are close to the filtered DNS.

Figure 16 compares the p.d.f.s of the longitudinal derivative of velocity \(\partial_1 \overline{u_1}\). They are not normalized so that the difference in the standard deviation can be observed. In all cases p.d.f.s are skewed but less intermittent than those of DNS since the small scales are removed. All p.d.f.s except those of the Bardina model nearly collapse. The skewness and flatness factors are compared in Table VIII; the flatness factors of the p.d.f.s of the transversal derivative \(\partial_2 \overline{u_2}\) are also included. The values of the NN model and the Smagorinsky model are in reasonable agreement with those of the filtered DNS data.
TABLE VIII. Comparison of turbulence characteristics.

| Case     | Model | $10^{2} \pi$ | $\lambda_1$ | $\lambda_2$ | $\sqrt{2} \lambda_1$ | $S_\parallel$ | $F_\parallel$ | $F_\perp$ |
|----------|-------|---------------|--------------|--------------|-----------------------|---------------|--------------|-----------|
| HIT1-1   | DNS   | 8.123         | 1.248        | 0.208        | 0.207                 | -0.53         | 5.81         | 8.77      |
|          | Filtered | 3.495         | 1.315        | 0.307        | 0.306                 | -0.50         | 4.35         | 5.29      |
|          | NN    | 3.303         | 1.283        | 0.322        | 0.325                 | -0.32         | 4.32         | 5.73      |
|          | Smagorinsky | 3.601         | 1.239        | 0.311        | 0.311                 | -0.36         | 4.11         | 5.91      |
|          | Bardina | 4.577         | 1.202        | 0.274        | 0.275                 | -0.86         | 5.91         | 7.34      |
|          | EGM   | 3.286         | 1.288        | 0.323        | 0.325                 | -0.41         | 4.26         | 5.50      |
| HIT1-2   | DNS   | 8.123         | 1.248        | 0.208        | 0.207                 | -0.53         | 5.81         | 8.77      |
|          | Filtered | 1.523         | 1.409        | 0.445        | 0.443                 | -0.39         | 3.68         | 4.19      |
|          | NN    | 1.488         | 1.497        | 0.440        | 0.446                 | -0.22         | 4.01         | 4.27      |
|          | Smagorinsky | 1.593         | 1.266        | 0.432        | 0.430                 | -0.31         | 3.64         | 5.91      |
|          | Bardina | 2.030         | 1.221        | 0.376        | 0.384                 | -0.69         | 4.52         | 7.34      |
|          | EGM   | 1.469         | 1.448        | 0.447        | 0.451                 | -0.30         | 3.80         | 5.50      |
| HIT2-1   | DNS   | 8.270         | 1.096        | 0.130        | 0.130                 | -0.56         | 6.96         | 8.76      |
|          | Filtered | 1.630         | 1.181        | 0.279        | 0.279                 | -0.42         | 3.98         | 5.06      |
|          | NN    | 1.654         | 1.227        | 0.278        | 0.277                 | -0.33         | 4.05         | 4.42      |
|          | Smagorinsky | 1.609         | 1.140        | 0.281        | 0.280                 | -0.45         | 4.03         | 5.59      |
|          | Bardina | 1.726         | 1.212        | 0.270        | 0.272                 | -0.73         | 4.88         | 5.80      |
| HIT2-2   | DNS   | 8.270         | 1.096        | 0.130        | 0.130                 | -0.56         | 6.96         | 8.76      |
|          | Filtered | 0.590         | 1.271        | 0.441        | 0.440                 | -0.40         | 3.83         | 4.33      |
|          | NN    | 0.550         | 1.354        | 0.471        | 0.447                 | -0.35         | 3.73         | 3.94      |
|          | Smagorinsky | 0.631         | 1.179        | 0.426        | 0.424                 | -0.39         | 3.60         | 4.71      |
|          | Bardina | 0.686         | 1.275        | 0.408        | 0.406                 | -0.68         | 4.38         | 5.67      |

FIG. 14. Energy spectrum of homogeneous isotropic turbulence. Comparison between DNS, filtered DNS, and LES. The solid lines show $E(k) \propto k^{-5/3}$. (a) Case HIT1-1, (b) Case HIT1-2, (c) Case HIT2-1, (d) Case HIT2-2.
FIG. 15. Distributions of magnitude of vorticity on $z = 0$. Case HIT2-1. (a) Filtered DNS, (b) Smagorinsky model, (c) Bardina model, (d) NN-G2.

FIG. 16. P.d.f.s of longitudinal derivative of velocity $\partial_1 \overline{u}_1$. (a) Case HIT1-1, (b) Case HIT1-2, (c) Case HIT2-1, (d) Case HIT2-2.
C. Initial-value problem of Taylor-Green vortices

The initial-value problem of the three-dimensional Taylor-Green vortices is often used to check the accuracy of a numerical scheme in computational fluid dynamics. In this section we consider this problem in which the initial velocity field is given by

\[ u(0, x) = u_0 \sin(x) \cos(y) \cos(z), \]
\[ v(0, x) = -u_0 \cos(x) \sin(y) \cos(z), \]
\[ w(0, x) = 0. \]

The initial coherent large-scale vortices develop into turbulence which contains fine-scale structures as shown in Fig. 17. The four cases listed in Table IX are considered. For each case LES is performed with two values of the filter width: \( \Delta = 8\Delta_{\text{DNS}} \) and \( 32\Delta_{\text{DNS}} \). The NN models trained in Case HIT-1 (\( \Delta = 8\Delta_{\text{DNS}} \)) is used for LES with grid points \( N_{\text{LES}}^3 = 128^3 \), while the NN model trained in Case HIT-2 (\( \Delta = 32\Delta_{\text{DNS}} \)) is used for LES with grid points \( N_{\text{LES}}^3 = 64^3 \). It is pointed out that the filter width in the physical space is the same for training and LES. LES with the Smagorinsky model and the Bardina model is also performed. The time step in LES is fixed to \( \Delta t = 1.0 \times 10^{-3} \).

| Case   | \( N^3 \times 10^3 \) | \( \nu \) | \( k_{\text{max}} \) | \( 10^3\Delta t \) |
|--------|-------------------------|-----------|----------------------|------------------|
| TGV-1  | 512^3                   | 7.0       | 248                  | 1.0              |
| TGV-2  | 512^3                   | 4.0       | 248                  | 1.0              |
| TGV-3  | 1024^3                  | 2.8       | 483                  | 0.625            |
| TGV-4  | 1024^3                  | 1.1       | 483                  | 0.625            |

Figure 18 shows time evolution of the rate of energy dissipation. The DNS results are included for reference only in Fig. 18(a) and (c) since the values are much larger than the values of filtered DNS and LES in the other cases. The rate of energy dissipation increases until \( t \approx 9 \) and then decreases. The three models (the NN, Smagorinsky, and Bardina models) reproduce this behavior, while the maximum of the rate of the energy dissipation is earlier for the Bardina model than the filtered DNS and the other models; this is because the added eddy viscosity is too large so that the energy dissipation is overpredicted. The Smagorinsky model gives good results for larger filter width \( \Delta = 32\Delta_{\text{DNS}} \) except for Case TGV-4, while the performance is comparable to the NN model for \( \Delta = 8\Delta_{\text{DNS}} \). The results show that the neural network trained for forced homogeneous isotropic turbulence can be used for decaying turbulence.

Figure 19 shows the energy spectrum for Case TGV-2 at three instants: (i) \( t = 6 \) at which the turbulent structures are being created, (ii) \( t = 9 \) at which the rate of energy dissipation is maximum, and (iii) \( t = 12 \) at which turbulence is decaying. The inertial subrange in which \( E(k) \propto k^{-5/3} \) is established at \( t = 9 \) and 12. For \( N_{\text{LES}}^3 = 128^3 \) the NN model and the Smagorinsky model reproduce the spectrum at \( t = 6 \) which retains the features of the initial
FIG. 18. Time evolution of rate of energy dissipation. Comparison between DNS, filtered DNS, LES. (a) Case TGV-1, $N_{LES}^3 = 128^3$, (b) Case TGV-1, $N_{LES}^3 = 64^3$, (c) Case TGV-2, $N_{LES}^3 = 128^3$, (d) Case TGV-2, $N_{LES}^3 = 64^3$, (e) Case TGV-3, $N_{LES}^3 = 128^3$, (f) Case TGV-3, $N_{LES}^3 = 64^3$, (g) Case TGV-4, $N_{LES}^3 = 128^3$, (h) Case TGV-4, $N_{LES}^3 = 64^3$. 
Taylor-Green vortices successfully. At $t = 12$, however, the energy spectrum of the NN model deviates from the filtered DNS at low wavenumbers, while they are in reasonable agreement at large wavenumbers.

The reason for the difference between the NN model and the filtered DNS can be understood by Fig. 20, which shows the vorticity distribution on $z = \pi/2$ at $t = 9$. Symmetry with respect to $x = \pi/2$, $y = \pi/2$, and $x = \pm y$ satisfied by the initial velocity field is preserved for the filtered DNS and the Smagorinsky model, while it is broken for the NN model. This is because the symmetry under the orthogonal transformations is not incorporated in the present NN model, although it respects the symmetry under a limited number of transformations such as $(x, y, z) \to (y, z, x)$ by its construction. Figure 21 showing the results of a priori test at $t = 9$ confirms that the symmetry is slightly broken by the NN model. The small asymmetry in the SGS stress grows in a chaotic manner leading to the broken symmetry observed in Fig. 20. It may be possible to keep the symmetry using e.g. the approach by Ling et al. [3]. However, the number of tensors and scalars which express a general form of the SGS stress would be large when the second-order derivatives of the velocity are included in the input variables; it is not evident that training is successful in this case.

FIG. 19. Comparison of energy spectrum between DNS, filtered DNS, and LES. Case TGV-2. The solid lines show $E(k) \propto k^{-5/3}$. (a–c) $N_3^{LES} = 128^3$, (d–f) $N_3^{LES} = 64^3$. (a,d) $t = 6$, (b,e) $t = 9$, (c,f) $t = 12$.

FIG. 20. Distributions of $\omega_z$ on $z = \pi/2$ at $t = 9$. Case TGV-2, $N_3^{LES} = 128^3$. (a) Filtered DNS, (b) Smagorinsky model, (c) NN model.
V. CONCLUDING REMARKS

The turbulence models for the SGS stress in LES were developed by neural networks which consist of three layers. Two methods were shown to be effective for improvement of regression accuracy of the neural network: one is to introduce weight into data sampling so that the SGS stress of large magnitude contributes to training; the other is to include the second-order derivatives of velocity in the input variables. As a result strong correlation of the SGS stress between the exact values and those predicted by the NN models was observed; the correlation coefficient is about 0.9 and 0.8 for large filter widths $\Delta = 48.8\eta$ and $97.4\eta$, respectively, although correlation is slightly lower than the gradient and extended gradient models. It was also shown that the NN models are close to but not identical with the gradient and extended gradient models. The NN model was used in LES of the homogeneous isotropic turbulence and the initial-value problem of the Taylor-Green vortices. The NN model should be stabilized to assure numerical stability. The results obtained with the stabilized NN model were in good agreement with those of the filtered DNS and LES with the Smagorinsky model. However, the NN model could not keep the symmetry of the flow in the initial-value problem of the Taylor-Green vortices.

We emphasize that neural networks of simple structure can predict the SGS stress accurately using methods for improvement. The NN model developed in the present study would allow us to infer a formula which is established by neural networks and has physical interpretation. Similar approach in Zhou et al. [13] using only the velocity gradient tensor (and the filter width) gave correlation coefficients of about 0.9 for $\Delta \approx 70\eta$ (estimated by their simulation parameters); their results may be further improved by using our methods. In Xie et al. [15] high correlation with $\text{Corr} \approx 0.99$ has been achieved for $\Delta \approx 33\eta$ (also estimated) by using higher-order derivatives of velocity at the points in the neighborhood; although their approach seems effective for accurate regression of the SGS stress, it would be difficult to be implemented into e.g. a body-fitted coordinate system.

One important task which should be done before pursuing an explicit form of the SGS stress is to establish a numerically stable NN model; our NN model should be stabilized in a posteriori test, which is also the case in other recent works [13–15]. It would be worth trying to seek a method different from addition of eddy viscosity and clipping.

ACKNOWLEDGMENTS

Numerical calculations were performed on the Altix UV1000, UV2000, and the Supercomputer system “AFI-NITY” at the Advanced Fluid Information Research Center, Institute of Fluid Science, Tohoku University.

[1] Perspective on machine learning for advancing fluid mechanics, Phys. Rev. Fluids 4, 100501 (2019).
[2] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, Machine Learning for Fluid Mechanics, Annu. Rev. Fluid Mech. 52, 477 (2020).
[3] J. Ling, A. Kurzawski, and J. Templeton, Reynolds averaged turbulence modelling using deep neural networks with embedded invariance, J. Fluid Mech. 807, 155 (2016).
[4] E. J. Parish, K. Duraisamy, A paradigm for data-driven predictive modeling using field inversion and machine learning, *J. Comput. Phys.* **305**, 758 (2016).

[5] J.-L. Wu, J.-X. Wang, H. Xiao, and J. Ling, A Priori Assessment of Prediction Confidence for Data-Driven Turbulence Modeling, *Flow Turbulence Combust.* **99**, 25 (2017).

[6] A. P. Singh, S. Medida, and K. Duraisamy, Machine-Learning-Augmented Predictive Modeling of Turbulent Separated Flows over Airfoils, *AIAA J.* **55**, 2215 (2017).

[7] J.-X. Wang, J.-L. Wu, and H. Xiao, Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data, *Phys. Rev. Fluids* **2**, 034603 (2017).

[8] R. Matai and P. A. Durbin, Zonal Eddy Viscosity Models Based on Machine Learning, *Flow Turbulence Combust.* **103**, 93 (2019).

[9] L. Zhu, W. Zhang, J. Kou, and Y. Liu, Machine learning methods for turbulence modeling in subsonic flows around airfoils, *Phys. Fluids* **31**, 015105 (2019).

[10] K. Duraisamy, G. Iaccarino, and H. Xiao, Turbulence Modeling in the Age of Data, *Annu. Rev. Fluid Mech.* **51**, 357 (2019).

[11] M. Gamahara and Y. Hattori, Searching for turbulence models by artificial neural network, *Phys. Rev. Fluids* **2**, 054504 (2017).

[12] R. Maulik and O. San, A neural network approach for the blind deconvolution of turbulent flows, *J. Fluid Mech.* **831**, 151 (2017).

[13] Z. Zhou, G. He, S. Wang and G. Jin, Subgrid-scale model for large-eddy simulation of isotropic turbulent flows using an artificial neural network, *Comput. Fluids* **195**, 104319 (2019).

[14] A. Beck, D. Flad and C.-D. Munz, Deep neural networks for data-driven LES closure models, *J. Comput. Phys.* **398**, 108910 (2019).

[15] C. Xie, J. Wang and W. E, Modeling subgrid-scale forces by spatial artificial neural networks in large eddy simulation of turbulence, *Phys. Rev. Fluids* **5**, 054606 (2020).

[16] J. Smagorinsky, General circulation experiments with the primitive equations, *Mon. Weather Rev.* **91**, 99 (1963).

[17] J. Bardina, J. H. Ferziger, and W. C. Reynolds, Improved turbulence models based on LES of homogeneous incompressible turbulent flows, Rep. TF-19. Department of Mechanical Engineering, Stanford (1984).

[18] R. A. Clark, J. H. Ferziger, and W. C. Reynolds, Evaluation of subgrid-scale models using an accurately simulated turbulent flow, *J. Fluid Mech.* **91**, 1 (1979).

[19] M. Germano, Turbulence: the filtering approach, *J. Fluid Mech.* **238**, 325 (1992).

[20] B. Vreman, B. Geurts, and H. Kuerten, Large eddy simulation of the temporal mixing layer using the Clark model, *Theor. Comput. Fluid Dyn.* **8**, 309 (1996).

[21] Y. Zang, R. L. Street, and J. R. Koseff, A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows, *Phys. Fluids A* **5**, 3186 (1993).

[22] K. Horiuti, A new dynamic two-parameter mixed model for large-eddy simulation, *Phys. Fluids* **9**, 3443 (1997).

[23] W. Yeo, A generalized high pass/low pass filtering procedure for deriving and solving turbulent flow equations, PhD thesis, Ohio State University (1987).

[24] W. Yeo and K. Bedford, Closure-free turbulence modeling based upon a conjunctive higher order averaging procedure, In Computational Methods in Flow Analysis (ed. H. Niki & M. Kawahara), 844 (1988).

[25] A. Leonard, Large-eddy simulation of chaotic convection and beyond. AIAA Paper 97-0204 (1997).

[26] V. Borue and S. A. Orszag, Local energy flux and subgrid-scale statistics in three-dimensional turbulence, *J. Fluid Mech.* **366**, 1 (1998).

[27] H. Lu and F. Porte-Agel, A modulated gradient model for large-eddy simulation: Application to a neutral atmospheric boundary layer, *Phys. Fluids B* **22**, 015109 (2010).

[28] T. Ishihara, Y. Kaneda, M. Yokokawa, K. Itakura, and A. Uno, Small-scale statistics in high-resolution direct numerical simulation of turbulence: Reynolds number dependence of one-point velocity gradient statistics *J. Fluid Mech.* **592**, 335 (2007).

[29] Y. Hattori and T. Ishihara, 4D visualization of isotropic turbulence and dynamics of high-enstrophy structures, Proc. JSST 2012 (International Conference on Simulation Technology) 170 (2012).