Turbulent scalar flux in inclined jets in crossflow: counter gradient transport and deep learning modelling

Pedro M. Milani[1],†, **Julia Ling**[2] **and John K. Eaton**[1]

[1]Mechanical Engineering Department, Stanford University, Stanford, CA 94305, USA
[2]Citrine Informatics, Redwood City, CA 94061, USA

A cylindrical and inclined jet in crossflow is studied under two distinct velocity ratios, $r = 1$ and $r = 2$, using highly resolved large eddy simulations. First, an investigation of turbulent scalar mixing sheds light onto the previously observed but unexplained phenomenon of negative turbulent diffusivity. We identify two distinct types of counter gradient transport, prevalent in different regions: the first, throughout the windward shear layer, is caused by cross-gradient transport; the second, close to the wall right after injection, is caused by non-local effects. Then, we propose a deep learning approach for modelling the turbulent scalar flux by adapting the tensor basis neural network previously developed to model Reynolds stresses (Ling *et al.*, *J. Fluid Mech.*, vol. 807, 2016*a*, pp. 155–166). This approach uses a deep neural network with embedded coordinate frame invariance to predict a tensorial turbulent diffusivity that is not explicitly available in the high-fidelity data used for training. After ensuring analytically that the matrix diffusivity leads to a stable solution for the advection diffusion equation, we apply this approach in the inclined jets in crossflow under study. The results show significant improvement compared to a simple model, particularly where cross-gradient effects play an important role in turbulent mixing. The model proposed herein is not limited to jets in crossflow; it can be used in any turbulent flow where the Reynolds averaged transport of a scalar is considered.

Key words: turbulent mixing, turbulence modelling, jets

1. Introduction

The jet in crossflow is a canonical geometry in fluid mechanics, whereby fluid is ejected from an orifice and interacts with main flow passing over that orifice (Mahesh 2013). This configuration has several practical applications. In the turbomachinery community, film cooling systems eject cooler air through rows of holes in the blade surface, which ideally stays close to the surface and protects it against the hot mainstream (Bogard & Thole 2006). In atmospheric flows some types of volcanic plumes rise from the vent while interacting with prevailing winds. This interaction helps determine mixing and deposition of ash (e.g. Gardner, Burgisser & Stelling 2007). Most interesting jets in crossflow involve mixing because the jet carries a scalar contaminant, such as heat or particles, into

† Email address for correspondence: pmmilani@stanford.edu

the crossflow. In these problems it is imperative to analyse the resulting scalar concentration field, denoted by c, in addition to the velocity field u_i.

Different parameters determine the resulting flow in a jet in crossflow, including geometry (such as hole shape and angle), Reynolds number and crossflow characteristics (like incoming boundary layer thickness and turbulence). One key dimensionless parameter in the incompressible regime is the velocity ratio r, defined as

$$r = U_j/U_c, \tag{1.1}$$

where U_j is the bulk velocity of the jet and U_c is the bulk velocity of the crossflow. In general, low velocity ratio jets stay close to the wall, while high velocity ratio jets penetrate deep into the crossflow before turning.

The literature on jets in crossflow is extensive (Mahesh 2013). Fric & Roshko (1994) used flow visualization and qualitatively described four types of vortices that appear in transverse jets in crossflow: the counter-rotating vortex pair, jet-shear layer vortices, horseshoe vortices and wake vortices. Su & Mungal (2004) and Schreivogel *et al.* (2016) used different techniques to simultaneously measure velocity and concentration, and then reported some turbulent mixing statistics. Computational studies have applied the Reynolds averaged Navier–Stokes (RANS) equations, but scalar transport is not computed accurately due to deficiencies in the available turbulence models (Acharya, Tyagi & Hoda 2001). High-fidelity simulations, on the other hand, have been shown to be much more predictive. Yuan, Street & Ferziger (1999) performed one of the first successful large eddy simulations of a jet in crossflow. Muppidi & Mahesh (2008) reported results from a direct numerical simulation (DNS) that agreed well with the experiment of Su & Mungal (2004), a transverse jet in crossflow at $r = 5.7$. They analysed entrainment rates and reported on inadequacies of standard turbulent scalar mixing models, including regions of perceived counter gradient transport.

The turbulence community has dedicated much effort to the modelling of the Reynolds stress, and multiple models are available and commonly employed in jets in crossflow (Hoda & Acharya 1999). Previous research also shows that errors in the turbulent mixing models contribute decisively to poor mean scalar field predictions in these flows (Ling *et al.* 2016*b*). Therefore, the present paper focuses solely on modelling the turbulent scalar flux, $\overline{u_i'c'}$, which represents the effect of turbulence on the mean scalar concentration field and is unclosed in the Reynolds averaged scalar transport equation.

The most commonly used model, called the gradient diffusion hypothesis (GDH), assumes that turbulence acts like isotropic diffusion on the mean concentration gradient as shown in

$$\overline{u_i'c'} = -\alpha_t \frac{\partial \bar{c}}{\partial x_i}. \tag{1.2}$$

The spatially varying turbulent diffusivity field in (1.2) is α_t, which is usually specified by means of a turbulent Schmidt number, $Sc_t \equiv \alpha_t/\nu_t$, where the eddy viscosity ν_t is given by a baseline momentum model (Kays 1994; Combest, Ramachandran & Dudukovic 2011).

More advanced models have been proposed, but have not gained much traction outside of research laboratories (Combest *et al.* 2011). Batchelor (1949) recognized that a tensor diffusivity D_{ij} is more appropriate:

$$\overline{u_i'c'} = -D_{ij} \frac{\partial \bar{c}}{\partial x_j}. \tag{1.3}$$

Different authors (e.g. Tavoularis & Corrsin 1981) measured diagonal and off-diagonal elements of D_{ij} in simple flows. To actually solve the scalar equation, Daly &

Harlow (1970) and Abe & Suga (2001) proposed algebraic models where D_{ij} is specified based on the Reynolds stress. Such formulations can theoretically improve results, but recent work in jets in crossflow has shown that they are not necessarily superior to simpler models and struggle with convergence issues (Ling *et al.* 2016*b*; Ryan *et al.* 2017).

Previous work highlighted some deficiencies of the GDH of (1.2) in jets in crossflow (e.g. Kohli & Bogard 2005; Weatheritt *et al.* 2020). In the present paper we focus our attention to a particular observation that has been reported before: regions of negative turbulent diffusivity, or counter gradient transport. This means that at least one component of the $\overline{u_i'c'}$ vector has the same sign as the equivalent component of $\partial \bar{c}/\partial x_i$, which directly contradicts (1.2) with a positive turbulent diffusivity. This was observed in jets in crossflow by several authors, both numerically (Muppidi & Mahesh 2008; Salewski, Stankovic & Fuchs 2008; Bodart *et al.* 2013; Milani & Eaton 2018) and experimentally (Salewski *et al.* 2008; Schreivogel *et al.* 2016). These workers reported the phenomenon without discussing its causes, so the present paper expands their analysis by explaining the observed counter gradient transport. This also inspired us to devise a deep learning approach for modelling the turbulent scalar flux that does not rely on the simple GDH of (1.2).

Machine learning tools have been rising in popularity in the turbulence closure literature, as evidenced by the review of Duraisamy, Iaccarino & Xiao (2019). Ling, Kurzawski & Templeton (2016*a*) used deep neural networks to predict coefficients in a basis expansion for the Reynolds stress anisotropy tensor, which became known as tensor basis neural network (TBNN). Parish & Duraisamy (2016) proposed a modelling strategy that couples machine learning with field inversion through adjoint optimization to develop new models. Sandberg *et al.* (2018) used a genetic optimization algorithm to generate data-driven analytical expressions for turbulent anisotropy and turbulent diffusivity, and Maulik *et al.* (2019) explored the use of a neural network towards subgrid scale modelling in large eddy simulations. The bulk of the aforementioned literature focuses on Reynolds stress modelling, but some authors have had success in applying machine learning techniques for closure of the passive scalar equation (Milani *et al.* 2018; Sandberg *et al.* 2018; Sotgiu, Weigand & Semmler 2018; Milani, Ling & Eaton 2020; Weatheritt *et al.* 2020). In the present paper a different scalar closure model is proposed, leveraging the tensor basis neural network concept of Ling *et al.* (2016*a*) to generate a more general model form.

The contribution of the present paper is twofold. First, we analyse and classify the regions of counter gradient transport found in an inclined jet in crossflow. These results suggest that an anisotropic model is needed to better capture the turbulent transport in part of the flow. Second, we adapt the popular TBNN methodology, which originally predicts the Reynolds stress tensor, to model the turbulent scalar flux vector. We analyse the results of this approach, denoted TBNN-s (where the 's' stands for scalar flux modelling), with the counter gradient transport regions in mind. The paper is organized as follows: § 2 describes the flow under consideration and the high-fidelity simulations used to generate the data; § 3 identifies and analyses the counter gradient transport regions; § 4 contains the description and the results of the deep learning model employed; and § 5 presents conclusions and suggestions for future work.

2. Numerical set-up

Two highly resolved large eddy simulations (LES) are used in the present work. The simulations were described in detail and validated against experimental data in
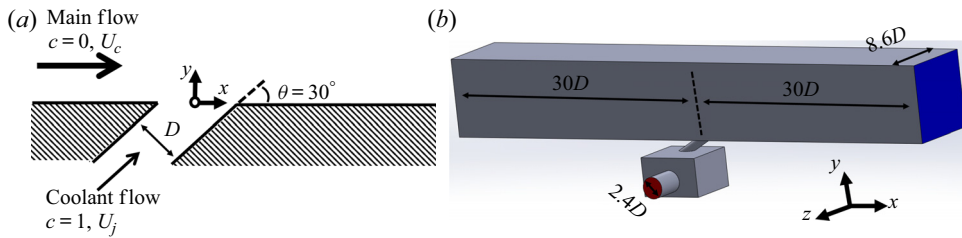
FIGURE 1. (*a*) Schematic of the inclined jet in crossflow. The coordinate frame origin is located at the centre of the hole as it meets the bottom wall. (*b*) Three-dimensional view of the complete simulation domain, with plenum feed inlet plane shown in red, outlet plane in blue and walls in grey. Figures adapted from Milani *et al.* (2019*a*).

Milani *et al.* (2019*a*). The present paper employs those results for analysis, thus, this section provides only a brief summary.

The geometry considered is a fully three-dimensional (3-D) inclined jet in crossflow, shown in figure 1(*a*). The hole has a circular cross-section of diameter D and is inclined 30° with respect to the streamwise direction. The mainstream consists of a turbulent developing flow in a square channel, whose boundary layer thickness just upstream of the hole location is $\delta_{99}/D = 1.5$. The free stream turbulence level is roughly 1 %. The total simulation domain is shown in figure 1(*b*). Note that the injection hole is short, with length 4.1D, and is fed from a rectangular plenum underneath the channel. This configuration was chosen partly because it is representative of film cooling applications; it also means that the flow inside the hole is highly non-uniform and contains secondary flows (Milani *et al.* 2019*a*).

The incompressible, filtered continuity and Navier–Stokes equations are solved in three dimensions as shown in (2.1*a,b*). The tilde over a variable is shown explicitly to denote filtered quantities; u_i are the Cartesian components of the velocity and p is the pressure. The unresolved scales are accounted for in the subgrid scale stress, $\tilde{\tau}_{ij} = \widetilde{u_i u_j} - \tilde{u}_i \tilde{u}_j$, which is closed using the Vreman model (Vreman 2004). The fluid density ρ and kinematic viscosity ν are constant. The two simulations use the same geometry and main channel inlet condition, and their difference lies in the velocity ratio r, which is varied by changing the flow rate into the plenum. The resulting jet Reynolds number is $Re_D = U_j D/\nu = 2900$ for the $r = 1$ case and $Re_D = 5800$ for $r = 2$. The transport equations are

$$\frac{\partial \tilde{u}_k}{\partial x_k} = 0; \quad \frac{\partial \tilde{u}_i}{\partial t} + \frac{\partial \left(\tilde{u}_j \tilde{u}_i \right)}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \tilde{p}}{\partial x_i} + \nu \frac{\partial^2 \tilde{u}_i}{\partial x_j \partial x_j} - \frac{\partial}{\partial x_j} \tilde{\tau}_{ij}. \quad (2.1a,b)$$

The filtered equation for a passive scalar c is also solved as shown in (2.2). The molecular Schmidt number is $Sc = 1$ and the subgrid scale mixing is represented by $\tilde{\sigma}_j = \widetilde{c u_j} - \tilde{c} \tilde{u}_j$, which is modelled employing the Reynolds analogy with a constant turbulent Schmidt number, $Sc_t = 0.85$. The scalar is injected with concentration $c = 1$ in the jet and $c = 0$ in the mainstream; all walls are adiabatic. The passive scalar transport equation is

$$\frac{\partial \tilde{c}}{\partial t} + \frac{\partial \left(\tilde{u}_j \tilde{c} \right)}{\partial x_j} = \frac{\nu}{Sc} \frac{\partial^2 \tilde{c}}{\partial x_j \partial x_j} - \frac{\partial}{\partial x_j} \tilde{\sigma}_j. \quad (2.2)$$

The equations are solved with the incompressible solver Vida from Cascade Technologies. It employs the method developed by Ham (2007), which consists of second-order accurate spatial discretization and explicit time advancement, based on
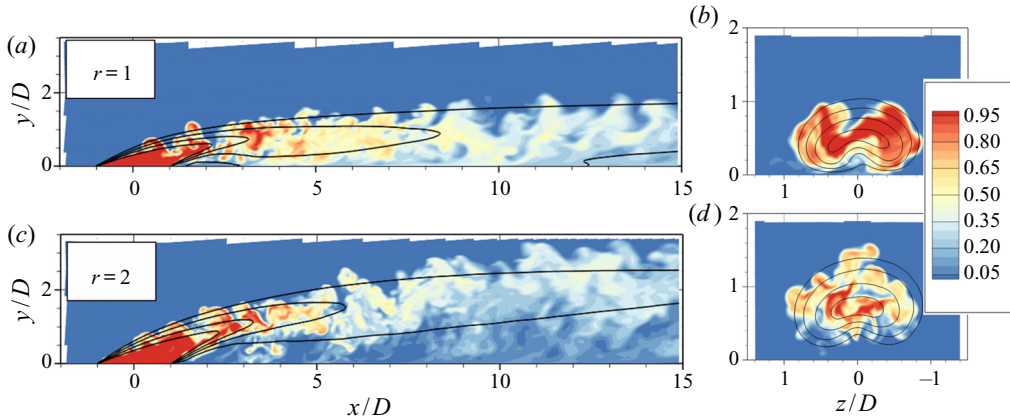
FIGURE 2. Scalar concentration results c on the centreplane ($z = 0$, a,c) and in an axial plane located at $x/D = 2$ (b,d). Colour contours show an instantaneous snapshot of c and lines indicate isocontours of mean concentration, $\bar{c} = 0.2, 0.4, 0.6, 0.8$.

second-order Adams–Bashforth. The mesh is block-structured and contains exclusively hexahedral elements, either $40.1M$ ($r = 1$) or $48.3M$ ($r = 2$). The y^+ values in the hole and bottom walls are around 1, so no wall models are employed there. Throughout the whole jet-crossflow interaction, the subgrid scale viscosity is significantly smaller than the molecular viscosity, so the present large eddy simulations have almost DNS-like resolution in the regions of interest. Therefore, we subsequently omit the tilde over grid-filtered quantities. Thus, u_i and c will refer to the (filtered) results from the LES, overbars will represent time averages of filtered quantities, and primes will denote fluctuating quantities in the filtered sense (so $c' \equiv c - \bar{c}$). The inlet conditions are generated using a method based on that from Xie & Castro (2008) and are chosen to match a concomitant experimental set-up. The validation is performed against 3-D magnetic resonance imaging (MRI) data for both velocity and scalar concentration and the agreement is excellent. More details including grid and time convergence studies are in Milani *et al.* (2019*a*).

Figure 2 shows contour plots of scalar concentration c in the $r = 1$ and $r = 2$ cases to provide some physical insight into the flows. In both cases the jet detaches from the bottom wall as soon as it is injected; it re-attaches in the lower velocity ratio case, but remains detached under the higher velocity ratio. As the y–z planes show, the mean profile assumes a typical kidney shape due to the influence of the counter rotating vortex pair (CVP), a distinguishing feature of this flow (Mahesh 2013). More detailed results are in Milani *et al.* (2019*a*).

3. Counter gradient transport

In this section we use the validated high-fidelity simulations to study turbulent mixing. The focus is two distinct regions of negative turbulent diffusivity (or counter gradient transport) which we classify as type 1 and type 2.

3.1. *Type 1*

Figure 3(*a*) shows vertical profiles of the streamwise components $\overline{u'c'}$ and $\partial \bar{c}/\partial x$ for $r = 1$. These are located at $x/D = 5$ on the centreline ($z/D = 0$). There is a region, marked in grey, where both $\overline{u'c'} < 0$ and $\partial \bar{c}/\partial x < 0$. Figure 3(*b*) presents profiles of the mean
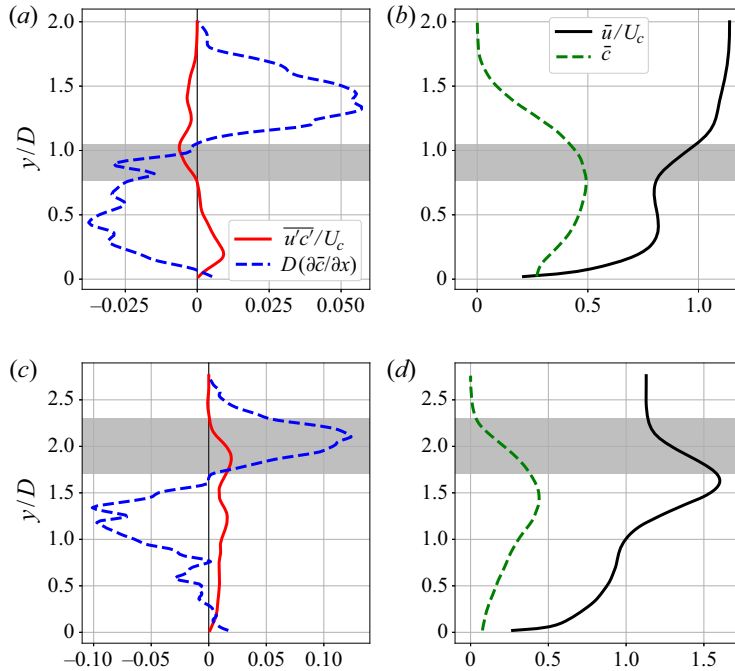
P. M. Milani, J. Ling and J. K. Eaton

FIGURE 3. Vertical profiles at $x/D = 5$ and $z/D = 0$ from the $r = 1$ (*a,b*) and $r = 2$ (*c,d*) LES. (*a,c*) Streamwise turbulent scalar flux and mean concentration gradient. (*b,d*) Mean streamwise velocity and concentration. The grey area shows a region of implied negative diffusivity in the streamwise direction. (*a*) $r = 1$. (*b*) $r = 1$. (*c*) $r = 2$. (*d*) $r = 2$.

streamwise velocity and concentration, $\bar{u}$ and $\bar{c}$, in the same location. It shows that the region with implied streamwise counter gradient transport is located just above the jet centreline, which is a similar observation to that of Schreivogel *et al.* (2016) and Bodart *et al.* (2013). Here, the likely cause of that phenomenon is not turbulent motion in the x-direction, but instead turbulent motion in the y-direction. In the grey area turbulent eddies of sufficiently small size that bring fluid from above ($v' < 0$) tend to bring weaker concentration ($c' < 0$) and higher streamwise velocity ($u' > 0$) since locally $\partial\bar{c}/\partial y < 0$ and $\partial\bar{u}/\partial y > 0$. Similarly, eddies that bring fluid from below ($v' > 0$) correlate $c' > 0$ with $u' < 0$. That causes $\overline{u'c'}$ to be negative, which implies negative diffusivity since $\partial\bar{c}/\partial x < 0$.

The same phenomenon manifests itself in higher velocity ratio jets, with slight differences. Figure 3(*c*) shows vertical profiles at $x/D = 5$ and $z/D = 0$ from the $r = 2$ LES. As is clear in figure 3(*d*), the jet core is faster than the free stream flow, so the y gradients of mean streamwise velocity and mean scalar gradient are roughly aligned in the top half of the jet, unlike when $r = 1$. Through a similar argument as before, vertical turbulent transport produces positive $\overline{u'c'}$ in that region. Thus, the perceived negative diffusivity region should occur when $\partial\bar{c}/\partial x > 0$. The streamwise mean scalar gradient is controlled by two main, competing effects: the spreading of the jet (which causes the concentration in the core to decrease), and the fact that the jet as a whole is moving up, away from the wall. Positive mean concentration streamwise gradients do not happen immediately above the jet centreline, but instead are present further up in the windward shear layer, as seen in figure 3(*c*). So, the grey band is located higher up and is wider when compared to the one shown in figure 3(*a*) for $r = 1$.
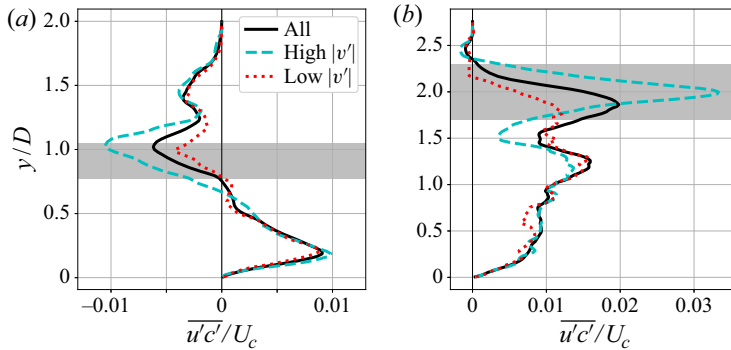
FIGURE 4. Vertical profiles at $x/D = 5$ and $z/D = 0$ of $\overline{u'c'}$ from the LES. Different averages are performed: unconditional or conditioned on high/low values of $|v'|$ at the centre of the grey band. (*a*) $r = 1$. (*b*) $r = 2$.

To support these explanations, consider figures 4(*a*) and 4(*b*). They show three differently averaged profiles of $\overline{u'c'}$ in the same location as shown in figure 3 for $r = 1$ and $r = 2$. For the solid line, the quantity $u'c'$ is averaged over all available time steps to produce the same quantity shown in figures 3(*a*) and 3(*c*). For the other two lines, the average is conditioned on the local value of the magnitude of the vertical velocity fluctuation, $|v'|$, at the centre of the grey band ($y/D = 0.91$ and $y/D = 2$, respectively). The average is taken either when $|v'|$ is high (top 20 % of samples) or low (bottom 20 % of samples). Away from the shaded region, the three lines converge as expected since the value of v' is uncorrelated with $u'c'$ at long distances. However, in the grey band, the three lines are significantly different: $\overline{u'c'}$ has much higher magnitude when $|v'|$ is high, and much smaller magnitude when $|v'|$ is low. This shows that vertical transport indeed plays an important role in setting the value of $\overline{u'c'}$ in this region and that events of high $|v'|$ accentuate the perceived negative diffusivity in the streamwise direction.

To investigate the behaviour of $\overline{u_i'c'}$ in more detail, it is useful to examine its transport equation, shown in (3.1). This equation is exact, and can be derived by multiplying u_i' to the (unaveraged) scalar transport equation and c' to the (unaveraged) Navier–Stokes equation, adding the results, and performing a time average. Capital letters U and C stand for time averages exactly like overbars (thus, $\bar{c} = C$), but are used here for a leaner notation. Here δ_{ij} is the Kronecker delta and α is the molecular diffusivity of the scalar, so $\alpha = \nu/Sc$. Note that these equations are mostly analogous to the Reynolds stress transport equations. Term A is the advection due to the mean velocity. Term I is the production term and shows how mean velocity and scalar gradients can locally generate $\overline{u_i'c'}$. Term II is the viscous destruction term. Term III consists of different mechanisms of turbulent transport, and term IV is a source term due to the pressure-scalar gradient correlation. For more details and modelling ideas, consult Younis, Speziale & Clark (2005). The equation is

$$
\frac{\partial \overline{u_i'c'}}{\partial t} + \overbrace{\frac{\partial \left(U_j \overline{u_i'c'} \right)}{\partial x_j}}^{A} = \overbrace{-\overline{u_j'c'} \frac{\partial U_i}{\partial x_j} - \overline{u_j'u_i'} \frac{\partial C}{\partial x_j}}^{I} \overbrace{-(\nu + \alpha) \overline{\frac{\partial c'}{\partial x_j} \frac{\partial u_i'}{\partial x_j}}}^{II}
$$

$$
\overbrace{-\frac{\partial}{\partial x_j} \left(\overline{u_j'u_i'c'} + \frac{\overline{p'c'}}{\rho} \delta_{ij} - \alpha \overline{u_i' \frac{\partial c'}{\partial x_j}} - \nu \overline{c' \frac{\partial u_i'}{\partial x_j}} \right)}^{III} + \overbrace{\frac{\overline{p'}}{\rho} \frac{\partial c'}{\partial x_i}}^{IV}. \quad (3.1)
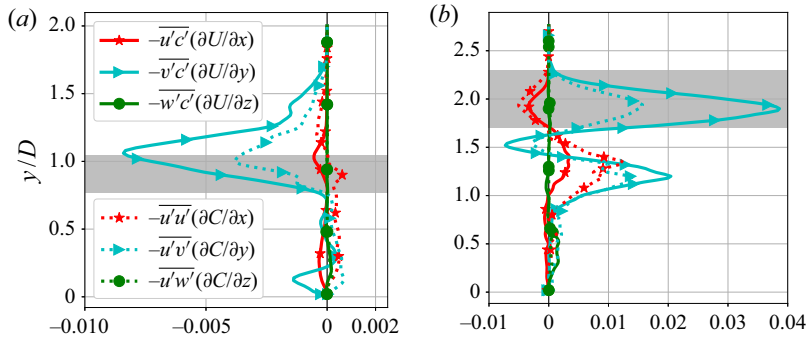$$

FIGURE 5. Vertical profiles of different components of the $\overline{u'c'}$ production term (marked as I in (3.1)). The profiles are shown at $x/D = 5$, $z/D = 0$ and the grey band represents the area of negative diffusivity. Terms are non-dimensionalized by U_c and D. (a) $r = 1$. (b) $r = 2$.

As further evidence that vertical velocity and concentration gradients are responsible for the streamwise negative diffusivity shown in figure 3, we consider the production terms of (3.1) for $\overline{u'c'}$. In figure 5 vertical profiles of the six different components of term I are shown. Note that the GDH considers only the effects of the mean concentration gradient in the x-direction on $\overline{u'c'}$, which is reflected on the production term $-\overline{u'u'}(\partial C/\partial x)$. However, in the regions where counter gradient transport is observed, the production term is dominated by the components containing the vertical gradients: $-\overline{v'c'}(\partial U/\partial y)$ and $-\overline{u'v'}(\partial C/\partial y)$. These terms are indeed negative in figure 5(a) and positive in figure 5(b), which explains the sign of $\overline{u'c'}$ in the grey regions.

There is one important caveat in our discussion of type 1 counter gradient transport. In shear flows the resulting mean scalar concentration is not a strong function of the modelled $\overline{u'c'}$ since the mean advection typically dominates the scalar transport in the streamwise direction. This point is discussed in more detail in appendix A. So, a turbulence model that fails to capture this phenomenon might still produce acceptable mean scalar field results in jets in crossflow. We believe the present analysis is still relevant for two reasons. First, it builds physical understanding of the turbulent transport in 3-D jets in crossflow by showing a concrete instance of the failure of the GDH and explaining the underlying cause. Such understanding is necessary to design improved and robust mixing models, and could be useful to interpret turbulence results in other flows as well. Second, in different contexts the correct prediction of these type 1 transport regions might be more relevant. For example, cross-gradient transport could be more important in different 3-D turbulent flows; or the quantity of interest might not be the mean scalar field, and instead be more sensitive to the streamwise scalar flux.

3.2. *Type 2*

A second region where counter gradient transport is present is near the wall, right after injection. In this case, negative diffusivity is observed in the vertical component: both $\overline{v'c'}$ and $\partial\bar{c}/\partial y$ are positive there. Figures 6(a) and 6(b) show vertical profiles at $x/D = 2$ and $x/D = 5$ for both $r = 1$ and $r = 2$, with grey bands indicating counter gradient transport regions. In general, it seems that the simple GDH could be useful: the solid lines approximately track the negated dashed lines throughout most of the plot, so a moderately non-uniform diffusivity could model $\overline{v'c'}$. However, this pattern breaks down
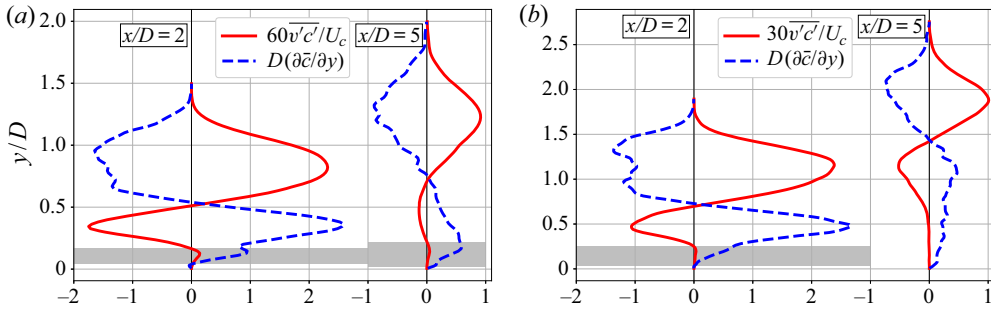
FIGURE 6. Vertical profiles at the centreplane of the y-component of scalar gradient and turbulent scalar flux. The grey band indicates regions where both have the same sign; for $r = 2$ and $x/D = 5$, no consistent region of counter gradient transport is observed (thus, no grey band), but $\overline{v'c'}$ is very near zero close to the wall. (a) $r = 1$. (b) $r = 2$.

close to the wall in all cases, where $\overline{v'c'}$ either has the same sign as $\partial \bar{c}/\partial y$ or is much smaller in magnitude than expected.

The first hypothesis to explain this instance of counter gradient transport is that it is similar in nature to the one described before, i.e. that gradients in directions orthogonal to y cause a positive correlation between v' and c'. However, this is not the case. Averages that are conditional on u' or w' are inconclusive, and the $\overline{v'c'}$ production term is dominated by mean gradients in the y-direction. So, the negative diffusivity observed in figure 6 is caused by a different physical mechanism. To help explain it, figure 7 shows a complete budget of $\overline{v'c'}$, with vertical profiles of all terms of (3.1).

As the plots show, the production term is consistently negative in the grey band, which means that local effects act as a sink and, therefore, favour a negative value of $\overline{v'c'}$. Since the resulting scalar flux is positive, other effects are overwhelming the production. In particular, close to injection ($x/D = 2$), term III is the most positive term. This is the turbulent transport of $\overline{v'c'}$, which unlike the production is inherently non-local. The breakdown of term III (not shown) shows that the molecular terms are negligible throughout the domain, and close to the wall the pressure-scalar correlation is dominant. This suggests that non-local turbulent effects, chiefly through fluctuating pressure, are responsible for generating a positive correlation between v' and c' in this region. Physically, this could be explained by the large scale stirring mentioned by Bodart *et al.* (2013): turbulent eddies with much larger length scales than the length scales over which $\partial \bar{c}/\partial y$ varies act in those regions, meaning that they can induce turbulent fluctuations that cannot be explained by local information. Note, however, that Bodart *et al.* (2013) pointed out locations of type 1 counter gradient transport in their data but provided a physical reasoning that is more appropriate for type 2.

Another interesting observation is on the role of mean flow advection (term A) in the balance. Since A is on the left-hand side of (3.1), a positive value represents a net outflow and a negative value represents a net inflow. Even though term A is not as important as some other terms in the grey band, it seems to enter the balance as a net outflow at $x/D = 2$ and as a net inflow at $x/D = 5$ for both velocity ratios. Since this is mostly due to mean streamwise advection, it suggests that non-local effects generate a positive $\overline{v'c'}$ close to the wall right after injection, which then is advected downstream. In other words, those non-local effects are not as strong at $x/D = 5$, but $\overline{v'c'}$ remains positive there partly because it is being advected from upstream (i.e. due to memory effects).
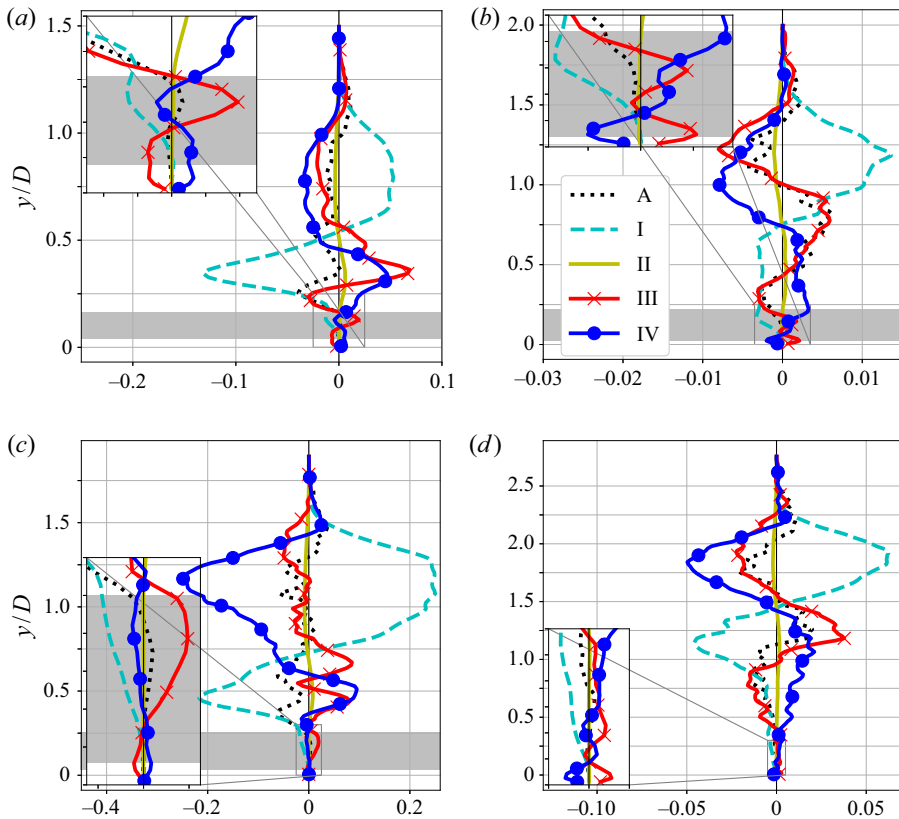
FIGURE 7. Terms from (3.1) in the channel centreline, non-dimensionalized using D and U_c. For $r = 1$, the zoomed in version shows $0 < y/D < 0.25$. For $r = 2$, it shows $0 < y/D < 0.3$. The grey band indicates counter gradient transport in the vertical direction. (a) $r = 1$, $x/D = 2$. (b) $r = 1$, $x/D = 5$. (c) $r = 2$, $x/D = 2$. (d) $r = 2$, $x/D = 5$.

This explanation is consistent with figure 6: the positive $\overline{v'c'}$ close to the wall is more intense and concentrated at $x/D = 2$, and it is more diffuse at $x/D = 5$.

3.3. *Three-dimensional perspective*

In the previous subsections we highlighted locations where a particular component of the turbulent scalar flux had the same sign as the equivalent component of the mean scalar gradient. We did so for two reasons. First, that is how many previous authors reported counter gradient transport in this flow. For example, Muppidi & Mahesh (2008), Bodart *et al.* (2013) and Schreivogel *et al.* (2016) noted regions where $\overline{u'c'}$ and $\partial \bar{c}/\partial x$ have the same sign, while Salewski *et al.* (2008) and Milani & Eaton (2018) highlighted regions where $\overline{v'c'}$ and $\partial \bar{c}/\partial y$ have the same sign; their results are consistent with the locations and mechanisms we explained as type 1 and type 2, respectively. Second, starting with a one-dimensional perspective permits an easier visualization of the turbulent scalar flux transport equation budgets as done in figures 5 and 7. However, despite providing useful insights, that methodology is dependent on the particular choice of axis; in fact, unless the mean scalar gradient and the negative turbulent scalar flux vectors are perfectly aligned, it is possible to construct a coordinate frame where one of the components will indicate

apparent counter gradient transport. So, in this subsection we study the same locations by considering the angle between the vectors $-\overline{u_i'c'}$ and $\partial\bar{c}/\partial x_i$, given by θ in (3.2). This measure takes all of the 3-D information into account and is independent of the choice of axis. In places where the simple GDH of (1.2) is valid exactly, we should have $\theta = 0$; in places of 'true' counter gradient transport, $\theta > 90°$. The angle θ is given by

$$\theta = \arccos\left(\frac{-\overline{u_i'c'}\dfrac{\partial\bar{c}}{\partial x_i}}{\sqrt{\overline{u_i'c'}\,\overline{u_i'c'}}\sqrt{\dfrac{\partial\bar{c}}{\partial x_i}\dfrac{\partial\bar{c}}{\partial x_i}}}\right). \tag{3.2}$$

Figure 8(*a,b*) shows colour contours of the angle θ in the $r = 1$ LES. Figures 8(*c,d*) and 8(*e,f*) contain the same planes showing the regions of type 1 and type 2, respectively, as described previously. In general, the angle θ is highest close to the wall, whose presence induces strong anisotropy into the turbulent mixing. Interestingly, there are key differences between type 1 and type 2 regions identified previously. On the windward shear layer, where cross-gradient effects give rise to type 1 transport, the overall misalignment indicated by θ is not too high, with angles between 20° and 50° being common. On the other hand, regions of type 2 transport shown in figure 8(*e,f*) coincide with severe misalignment of $-\overline{u_i'c'}$ and $\partial\bar{c}/\partial x_i$, and in such areas the angle θ is frequently higher than 90°. This shows that type 1 transport, caused mainly by local effects, actually translates to a consistent but small misalignment between the vectors when seen in three dimensions. The non-equilibrium, non-local causes of type 2 transport actually lead to a more than 90° misalignment between the flux and gradient vectors in three dimensions, which can be seen as a 'true' counter gradient transport that is independent of the coordinate frame. Figure 9, which presents the same plots for the $r = 2$ dataset, leads to similar conclusions.

From a modelling perspective, regions where θ is significantly above zero cannot be well captured with the simple GDH of (1.2) irrespective of the diffusivity chosen. Moving to a tensor diffusivity D_{ij}, as shown in (1.3), can capture cross-gradient effects (since turbulent transport in the x-direction can be a function of the y gradient) and would thus be expected to model well the turbulent mixing in regions of type 1. This is further supported by the fact that θ in those regions is less than 90°, so the matrix diffusivity D_{ij} is only required to rotate the concentration gradient vector by an acute angle. Thus, a positive semi-definite matrix D_{ij} can be used, guaranteeing numerical stability. In locations where type 2 transport is found, the non-local effects on $\overline{u_i'c'}$ preclude a purely local model, such as (1.3), from capturing all the relevant physics. Even if a diffusivity D_{ij} where chosen such that $\overline{u_i'c'}$ is matched, that matrix would not be positive semi-definite since $\theta > 90°$ in those areas. Therefore, the equation would be numerically unstable, akin to using a negative value of α_t in (1.2). For locations of type 2 counter gradient transport, only a non-local model, based potentially on solving separate transport equations for $\overline{u_i'c'}$, could hope to reproduce the appropriate physics.

4. Deep learning modelling

In this section we transition to the question of modelling the $\overline{u_i'c'}$ vector using deep learning. For simplicity, this section makes heavy use of matrix notation instead of index notation: a lower case bold letter indicates a vector (first-order tensor) and an upper case bold letter indicates a 3×3 matrix (second-order tensor).
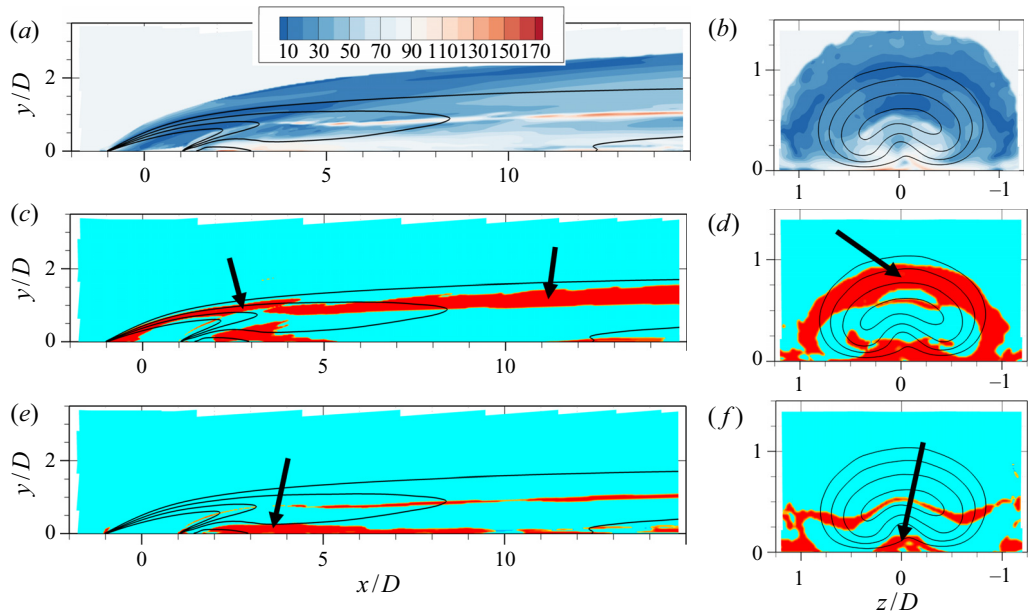
FIGURE 8. Counter gradient transport in the $r = 1$ case. (a,c,e) Centreplanes $(z = 0)$ and (b,d,f) axial planes at $x/D = 2$. Lines indicate isocontours of $\bar{c} = 0.2, 0.4, 0.6, 0.8$. (a,b) Contains colour contours of θ in degrees. (c,d) Shows, in red, places where $\overline{u'c'}(\partial\bar{c}/\partial x) > 0$ (with arrows highlighting type 1 transport). (e,f) Identifies regions where $\overline{v'c'}(\partial\bar{c}/\partial y) > 0$ (with arrows highlighting type 2 transport).
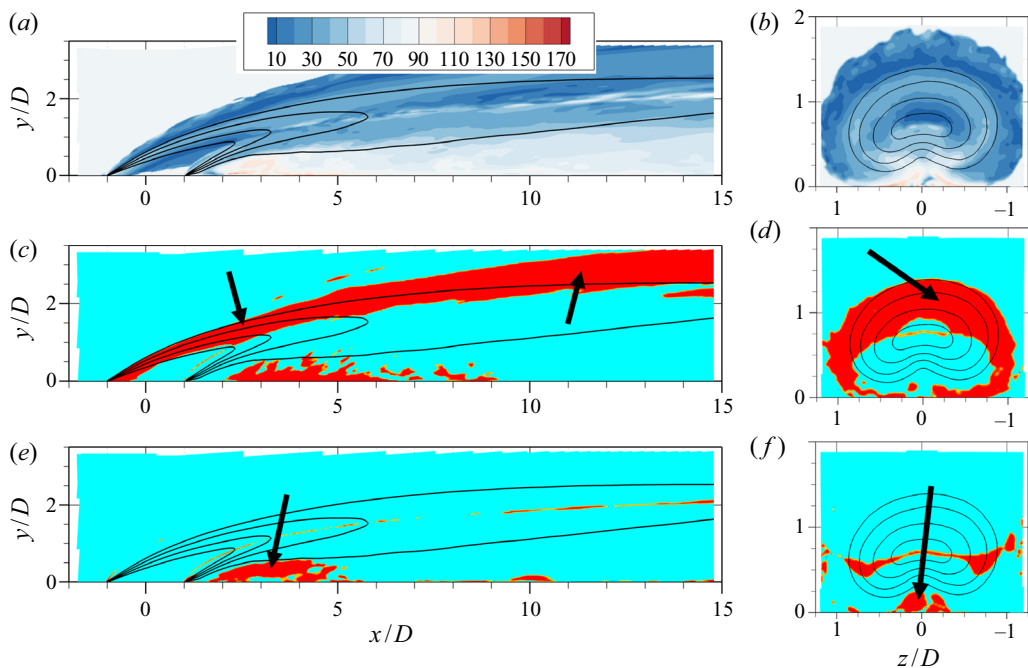

FIGURE 9. Counter gradient transport in the $r = 2$ case. Same description as figure 8.

4.1. *Tensor basis neural network for scalar flux modelling*

Given the promise of the tensor basis neural network proposed by Ling *et al.* (2016a), we would like to use a similar method to model the vector $\overline{u'c'}$. In summary, we aim to employ a neural network architecture that is designed to respect rotational invariance while predicting a tensorial output. However, the architecture described in Ling *et al.* (2016a) is designed to prescribe the Reynolds stress anisotropy tensor **B**, which is a symmetric and traceless second-order tensor. The approach must be modified for the turbulent scalar flux vector $\overline{u'c'}$, which does not have those characteristics.

We start by assuming that the turbulent scalar flux is a general vector-valued function of the mean velocity and scalar gradients, $\nabla \bar{u}$ and $\nabla \bar{c}$. The velocity gradient tensor is split into a symmetric and an anti-symmetric part, **S** and **R**, respectively, such that $\nabla \bar{u} = \mathbf{S} + \mathbf{R}$. As is done in previous work that models turbulent scalar fluxes (Milani *et al.* 2018), two additional dimensionless quantities are also used to regress $\overline{u'c'}$: the Reynolds number based on wall distance, $Re_d = \sqrt{k}d/\nu$, and the eddy viscosity ratio, ν_t/ν, where d is the distance to the nearest wall, k is the turbulent kinetic energy and ν_t is the eddy viscosity from a baseline turbulence model. So, the functional dependence is

$$- \overline{u'c'} = f(\mathbf{S}, \mathbf{R}, \nabla \bar{c}, \nu_t/\nu, Re_d). \tag{4.1}$$

Note that the model in (4.1) depends on a baseline turbulence model for the momentum equations, which provides field quantities such as k and ν_t. This is a design choice that makes the model more usable in practice, since one would not have access to high-fidelity simulation values for these quantities in a practical RANS simulation. A similar approach was taken by other authors who worked in scalar flux modelling with machine learning techniques (e.g. Sandberg *et al.* 2018; Sotgiu *et al.* 2018). As is done in Ling *et al.* (2016a), (4.1) is cast as a summation of an appropriate tensor basis, where each basis element is a vector and is multiplied by a scalar factor $g^{(n)}$. This quantity, in turn, can be an arbitrary function of invariants λ_j, which are scalar quantities derived from the tensors **S**, **R** and $\nabla \bar{c}$ that are reference frame independent. Thus, $\overline{u'c'}$ becomes

$$- \overline{u'c'} = \sum_{n=1}^{6} g^{(n)}(\lambda_1, \lambda_2, \ldots, \lambda_{15})t^{(n)}. \tag{4.2}$$

When the function is written in this form, it is guaranteed to be unchanged under any coordinate frame rotation and reflection as explained in Ling *et al.* (2016a). The neural network's eventual goal will be to approximate the functions $g^{(n)}(\lambda_1, \lambda_2, \ldots, \lambda_{15})$.

For this problem, we cannot use the same basis and invariants used in Pope (1975) or Ling *et al.* (2016a), since the quantity of interest is a vector ($\overline{u'c'}$) that depends upon one symmetric matrix (**S**), one anti-symmetric matrix (**R**) and one vector ($\nabla \bar{c}$). Turning to the review of Zheng (1994), one can find the appropriate vector basis, given as

$$\left. \begin{aligned} \mathbf{t}^{(1)} &= \nabla \bar{c}, \quad \mathbf{t}^{(2)} = \mathbf{S}\nabla \bar{c}, \quad \mathbf{t}^{(3)} = \mathbf{R}\nabla \bar{c}, \\ \mathbf{t}^{(4)} &= \mathbf{S}^2\nabla \bar{c}, \quad \mathbf{t}^{(5)} = \mathbf{R}^2\nabla \bar{c}, \quad \mathbf{t}^{(6)} = (\mathbf{S}\mathbf{R} + \mathbf{R}\mathbf{S})\nabla \bar{c}, \end{aligned} \right\} \tag{4.3}$$

and the appropriate invariants for the incompressible case, given as

$$\left.\begin{array}{c} \lambda_1 = \mathrm{tr}(\boldsymbol{S}^2), \quad \lambda_2 = \mathrm{tr}(\boldsymbol{S}^3), \quad \lambda_3 = \mathrm{tr}(\boldsymbol{R}^2), \quad \lambda_4 = \mathrm{tr}(\boldsymbol{S}\boldsymbol{R}^2), \quad \lambda_5 = \mathrm{tr}(\boldsymbol{S}^2\boldsymbol{R}^2), \\ \lambda_6 = \mathrm{tr}(\boldsymbol{S}^2\boldsymbol{R}^2\boldsymbol{S}\boldsymbol{R}), \quad \lambda_7 = \nabla\bar{c}^T\nabla\bar{c}, \quad \lambda_8 = \nabla\bar{c}^T\boldsymbol{S}\nabla\bar{c}, \quad \lambda_9 = \nabla\bar{c}^T\boldsymbol{S}^2\nabla\bar{c}, \\ \lambda_{10} = \nabla\bar{c}^T\boldsymbol{R}^2\nabla\bar{c}, \quad \lambda_{11} = \nabla\bar{c}^T\boldsymbol{S}\boldsymbol{R}\nabla\bar{c}, \quad \lambda_{12} = \nabla\bar{c}^T\boldsymbol{S}^2\boldsymbol{R}\nabla\bar{c}, \\ \lambda_{13} = \nabla\bar{c}^T\boldsymbol{R}\boldsymbol{S}\boldsymbol{R}^2\nabla\bar{c}, \quad \lambda_{14} = Re_d, \quad \lambda_{15} = \nu_t/\nu. \end{array}\right\} \quad (4.4)$$

Note that the first 13 invariants are obtained from the tensor inputs, and the last two are purely the scalar quantities Re_d and ν_t/ν (which are automatically invariant to the choice of coordinate frame). In (4.3) and (4.4) products denote standard matrix-matrix or matrix-vector products, and tr() denotes the trace of a matrix.

Under this formulation, given a vector basis and the invariants, the neural network directly prescribes the vector $\overline{u'c'}$ whose divergence acts as a source term in the Reynolds averaged scalar transport equation. As discussed in Wu *et al.* (2019), such an approach can lead to an ill conditioned equation, which is difficult to solve numerically. To mitigate this, they suggest that whenever possible data-driven closures should aim to predict diffusivity coefficients instead of source terms. With that in mind, we note that all basis elements shown in (4.3) consist of a matrix multiplying the mean concentration gradient. So, we can define a different basis $\boldsymbol{T}^{(n)}$ where $\boldsymbol{t}^{(n)} = \boldsymbol{T}^{(n)}\nabla\bar{c}$. Thus, we rewrite (4.2) as

$$-\overline{u'c'} = \overbrace{\left[\sum_{n=1}^{6} g^{(n)}(\lambda_1, \lambda_2, \ldots, \lambda_{15})\boldsymbol{T}^{(n)}\right]}^{\boldsymbol{D}}\nabla\bar{c}, \quad (4.5)$$

with the matrix part of the basis given by

$$\left.\begin{array}{c} \boldsymbol{T}^{(1)} = \boldsymbol{I}, \quad \boldsymbol{T}^{(2)} = \boldsymbol{S}, \quad \boldsymbol{T}^{(3)} = \boldsymbol{R}, \\ \boldsymbol{T}^{(4)} = \boldsymbol{S}^2, \quad \boldsymbol{T}^{(5)} = \boldsymbol{R}^2, \quad \boldsymbol{T}^{(6)} = \boldsymbol{S}\boldsymbol{R} + \boldsymbol{R}\boldsymbol{S}, \end{array}\right\} \quad (4.6)$$

where $\boldsymbol{I}$ is the 3×3 identity matrix.

In (4.5) the term in square brackets is naturally interpreted as a 3×3 turbulent diffusivity matrix $\boldsymbol{D}$. Thus, we recover an equation of the same form as (1.3) that is amenable to be learned via a neural network with embedded rotational invariance.

Before the full model architecture is discussed, there remains the issue of non-dimensionalization. To obtain relevant scales and values for the eddy viscosity ν_t, a baseline turbulence model is run on the original mesh utilizing the frozen mean velocity field from the LES calculation. Any model that yields length and time scales and a value for the eddy viscosity could be employed with the current framework; in the present paper we use the realizable $k - \epsilon$ model of Shih, Zhu & Lumley (1995). So, given the mean LES velocity and pressure fields, the transport equations for k and ϵ are solved, which also generates ν_t as a by-product. This approach to non-dimensionalization has been used by others such as Sandberg *et al.* (2018) and Milani *et al.* (2020). Then, $\boldsymbol{S}$, $\boldsymbol{R}$ and $\nabla\bar{c}$ in (4.4) and (4.6) are non-dimensionalized using local values of k and ϵ before being fed as inputs to the network. To guarantee that the neural network maps from dimensionless inputs to a dimensionless output, a non-dimensional diffusivity tensor $\boldsymbol{D}^*$ is explicitly introduced as $\boldsymbol{D} = \nu_t\boldsymbol{D}^*$.

The architecture proposed, henceforth referred to as tensor basis neural network for scalar flux modelling, or TBNN-s, is presented in figure 10. The network receives as
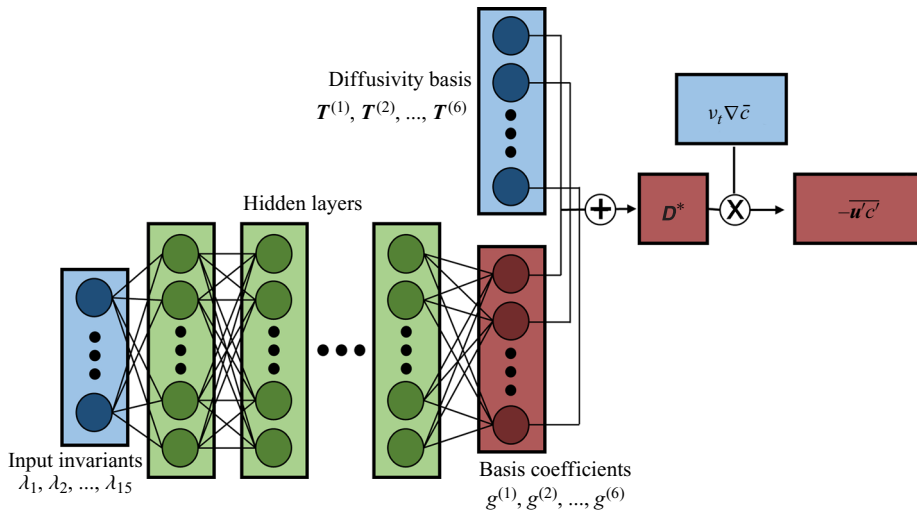
FIGURE 10. Schematic of TBNN-s architecture. Blue cells represent inputs to the network, green cells represent the hidden layers containing tuneable parameters and red cells denote different levels of outputs.

inputs the mean flow invariants (listed in (4.4)) non-dimensionalized by k and ϵ and processes them through a sequence of hidden layers. These are fully connected layers, which sequentially perform a linear transformation (matrix multiplication plus an offset) followed by a nonlinear activation function (the simple rectified linear unit, or ReLU, is employed). The last hidden layer does not use a ReLU function and transforms its inputs into a six-dimensional vector containing the values of $g^{(n)}$, the dimensionless basis coefficients. Note that the hidden layers are responsible for explicitly providing the functions $g^{(n)}(\lambda_1, \ldots, \lambda_{15})$ and contain all the tuneable parameters of the model, which are learned from training data. Then, the coefficients $g^{(n)}$ multiply elementwise their respective dimensionless basis tensor $\boldsymbol{T}^{(n)}$ shown in (4.6) and the result is summed producing the dimensionless diffusivity matrix $\boldsymbol{D}^*$. Finally, the diffusivity is made dimensional via multiplication by ν_t and its matrix-vector product with the mean scalar gradient produces $-\overline{\boldsymbol{u}'c'}$.

Note that this modelling framework is similar to that of Weatheritt *et al.* (2020), who also expanded the turbulent scalar flux vector as a sum of tensor basis that maintains rotational invariance. Our model has two key differences. First, we use a deep neural network to approximate the functions $g^{(n)}(\lambda_i)$, while Weatheritt *et al.* (2020) use gene expression programming, a different regression tool (see Weatheritt *et al.* (2017) for a discussion of their differences). Second, their basis also involves the Reynolds stress tensor. In theory this allows for more accurate representation of the scalar flux, as they confirm. However, in a typical RANS framework one does not have access to the true values of the Reynolds stress; as such, we preferred to develop models that are based solely on the mean velocity gradient. In future work it would be useful to compare their model to ours under the same conditions in order to analyse strengths and weaknesses of different data-driven turbulence modelling strategies.

4.2. *Loss function and implementation*

Neural networks are trained via gradient descent-like algorithms. A loss function J is defined to capture how inadequate the present solution is compared to the known results

from training data. Given multiple batches of training data, the loss function is calculated together with its gradient with respect to each trainable parameter. The gradient, a key component for training neural networks, is calculated using backpropagation, which is sequential application of the chain rule starting at the loss function J and progressing back through the layers all the way to the first hidden layer. For backpropagation to work, every operation defined in the network must be differentiable which is the case for figure 10. Note that the loss function is defined based on the mismatch between the predicted value of the turbulent scalar flux, $\overline{u'c'}_{PRED}$, and the turbulent scalar flux given in the training data, $\overline{u'c'}_{LES}$. We experimented with a few different options, including the common sum of squared differences and sum of absolute differences:

$$J_{L1} = \frac{1}{N_{data}} \sum_{i=1}^{N_{data}} \|\overline{u'c'}_{PRED}^{(i)} - \overline{u'c'}_{LES}^{(i)}\|_1, \tag{4.7}$$

$$J_{L2} = \frac{1}{N_{data}} \sum_{i=1}^{N_{data}} \|\overline{u'c'}_{PRED}^{(i)} - \overline{u'c'}_{LES}^{(i)}\|_2^2. \tag{4.8}$$

Here $\| \ \|_1$ indicates vector 1-norm, $\| \ \|_2$ indicates vector 2-norm, N_{data} is the number of data points over which the loss is computed and a superscript i indicates the i-th training point. These definitions of the loss have the disadvantage that they depend on a particular non-dimensionalization and scaling to compare points from different datasets and different locations from the same dataset. Instead, if we desire a loss function whose gradient with respect to model parameters is independent of the scaling used, the mean of the natural logarithms of the vector norms is a possible choice:

$$J_{log} = \frac{1}{N_{data}} \sum_{i=1}^{N_{data}} \log \left(\frac{\|\overline{u'c'}_{PRED}^{(i)} - \overline{u'c'}_{LES}^{(i)}\|_2}{\|\overline{u'c'}_{LES}^{(i)}\|_2} \right). \tag{4.9}$$

In (4.9) the factor that scales the loss at each point is the $\|\overline{u'c'}_{LES}^{(i)}\|_2$ in the denominator. Because the logarithm of the ratio becomes a difference of logarithms, this term vanishes when the gradient of J_{log} with respect to the neural network parameters is calculated. This scaling is chosen so that the loss attains readily interpretable values (for example, a loss around 0 indicates that the vectorial discrepancy between predicted and given scalar fluxes has roughly the same norm as the LES scalar flux itself).

We experimented with the three loss definitions shown in (4.7), (4.8) and (4.9). The results on the mean scalar field $\bar{c}$ were similar among the three, so a comparison will not be shown for brevity. For the results in the present paper, the function J_{log} of (4.9) is employed. We also added an L2 regularization term to the loss, which penalizes high magnitude parameters in the model. This is commonly done to prevent overfitting. The loss is optimized with a widely used algorithm based on stochastic gradient descent, called adaptive moment estimation (Adam) from Kingma & Ba (2014). Part of the reason for the current popularity of this method is that it has been shown to be reasonably robust to the choice of hyperparameters, especially the learning rate.

To choose the different hyperparameters for the TBNN-s, we used validation points that were not used during training. Milani *et al.* (2019*a*) presents data on the geometry of figure 1 for a third value of velocity ratio, $r = 1.5$. In the current paper the TBNN-s is trained exclusively on the $r = 2$ dataset and the hyperparameters are chosen such that good values of the loss function are obtained on points taken from the $r = 1.5$ case. We observed

that the quality of the model generated is not a strong function of the hyperparameters considered within a reasonable range. Besides, the validation aims to find hyperparameters that minimize the loss in (4.9), when the true test for the model is how well it predicts the mean scalar field $\bar{c}$ after the Reynolds averaged equations are solved. Thus, not much emphasis should be placed on the specific choice of hyperparameters, and we picked a set that performed well on the validation set after manual tuning. In the current work we use a learning rate of 10^{-3} and other default parameters for Adam; we employ 10 hidden layers with 30 nodes each, roughly in line with the network size of Ling *et al.* (2016*a*); we use an L2 regularization strength of 10^{-2}; we use an additional regularization method called dropout (Srivastava *et al.* 2014) with a drop probability of 0.2; and the batch size in which training data is fed into the Adam optimizer is 50. Training a neural network is not a deterministic exercise because results depend on a random initialization and on the stochastic nature of the optimization algorithm. To mitigate this, several training runs were executed with the same hyperparameters and the network that was saved and employed in this paper is the one that, at any point during each training procedure, produced the minimum loss in the validation set.

The network as shown in figure 10 is implemented in Python using the Tensorflow package (Abadi *et al.* 2016) and the code can be found online in the following GitHub repository: https://github.com/pmmilani/tbnns. The TBNN-s discussed in following subsections was trained on 100k randomly sample points from the inclined jet in crossflow with $r = 2$. Only points where the mean scalar gradient is significant ($\|\nabla\bar{c}\|_2/(\epsilon/k^{1.5}) > 10^{-3}$) are considered at training time.

4.3. *Results*

The architecture described in § 4.1 is used in its entirety when the network is being trained and a correct value of $\overline{u'c'}$ is available. Once the network has been trained, it is applied up to the prediction of **D**. In other words, the network learns to predict a turbulent diffusivity tensor that best matches the turbulent scalar flux presented at training time even though it has never seen an input turbulent diffusivity tensor. When applied to a test set, it predicts a **D** field over the whole domain which is then used to solve the Reynolds averaged scalar equation

$$\frac{\partial}{\partial x_i}(\bar{u}_i\bar{c}) = \frac{\partial}{\partial x_i}\left(\frac{\nu}{Sc}\frac{\partial\bar{c}}{\partial x_i}\right) + \frac{\partial}{\partial x_i}\left(D_{ij}\frac{\partial\bar{c}}{\partial x_j}\right), \tag{4.10}$$

where the last term on the right-hand side consists of the model for the turbulent scalar flux involving the turbulent diffusivity matrix. To test the model independent of any velocity field errors, (4.10) is solved with the mean velocity field provided by the LES and with the matrix turbulent diffusivity **D** provided by the TBNN-s. Note that the TBNN-s is run only once, using LES results with k and ϵ fields from the baseline turbulence model (realizable $k - \epsilon$) as inputs, and then the diffusivity matrix is fixed and (4.10) is solved on the LES mesh. We employ ANSYS Fluent for the baseline turbulence model and for solving the scalar transport equation.

Before discussing the results, there is one important consideration to be made about the diffusivity matrix **D**. An arbitrary scalar (isotropic) diffusivity could be numerically unstable, namely if it were ever negative. Analogously, a matrix diffusivity **D** produced by the TBNN-s could also lead to an unstable partial differential equation under some conditions, so it must be post processed before we use it in (4.10). In particular, we require that the diffusivity **D** be a positive semi-definite matrix. Note that this is a requirement for stability for any closure of the form given in (1.3), including higher-order
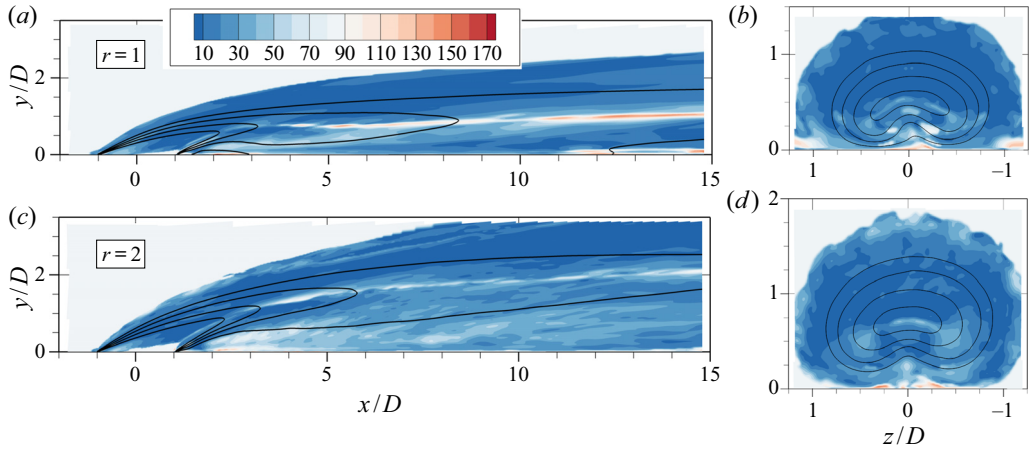
FIGURE 11. Angle between modelled and true turbulent scalar flux in degrees, given in $\theta^{TBNN\text{-}s}$. (*a*,*c*) Centreplane ($z = 0$) and (*b*,*d*) axial planes at $x/D = 2$; black lines indicate isocontours of $\bar{c} = 0.2, 0.4, 0.6, 0.8$.

algebraic closures. See appendix B for a more detailed explanation of this point. Therefore, we post process the diffusivity $\boldsymbol{D}^*$ output: in locations where the predicted matrix is not positive semi-definite, it is replaced by a diagonal matrix that uses the GDH with fixed $Sc_t = 0.85$ as the model.

4.3.1. *A priori results*

A priori results directly test the TBNN-s predictions of the scalar flux. All the results shown are for a network trained on the $r = 2$ dataset only.

Figure 11 shows the alignment between the predicted turbulent scalar flux vector and the ground truth scalar flux vector from the LES for both datasets. The angle in degrees is calculated as

$$\theta^{TBNN\text{-}s} = \arccos \left(\frac{-\overline{u_i'c'} \left(D_{ij} \dfrac{\partial \bar{c}}{\partial x_j} \right)}{\sqrt{\overline{u_i'c'}\, \overline{u_i'c'}} \sqrt{\left(D_{ij} \dfrac{\partial \bar{c}}{\partial x_j} \right) \left(D_{ij} \dfrac{\partial \bar{c}}{\partial x_j} \right)}} \right), \tag{4.11}$$

where D_{ij} is the matrix turbulent diffusivity predicted by the TBNN-s.

Note that these plots should be compared to the misalignment that would arise from a baseline GDH model shown in § 3 in figures 8(*a*,*b*) and 9(*a*,*b*). In figure 11 we see significant improvement in the full domain for both cases, $r = 2$ (where the model was trained) and $r = 1$ (where the model's generality is tested). In the windward shear layer, where type 1 counter gradient transport was observed, the alignment is now excellent, with values below $10°$ being prevalent. This supports the argument made in § 3 that this physical process would be amenable to be modelled with a local, but anisotropic model. The alignment near the wall is also significantly superior to that shown in figures 8(*a*,*b*) and 9(*a*,*b*). However, some higher values of $\theta^{TBNN\text{-}s}$ are still present there. First, this reflects the fact that whenever the original misalignment, between $-\overline{u_i'c'}$ and $\partial \bar{c}/\partial x_j$, is more than $90°$ it cannot be fully fixed with a positive semi-definite diffusivity matrix. Second, as argued in § 3, the turbulent scalar flux in this near-wall region is affected by non-equilibrium, non-local effects which cannot be satisfactorily captured with a purely local model.
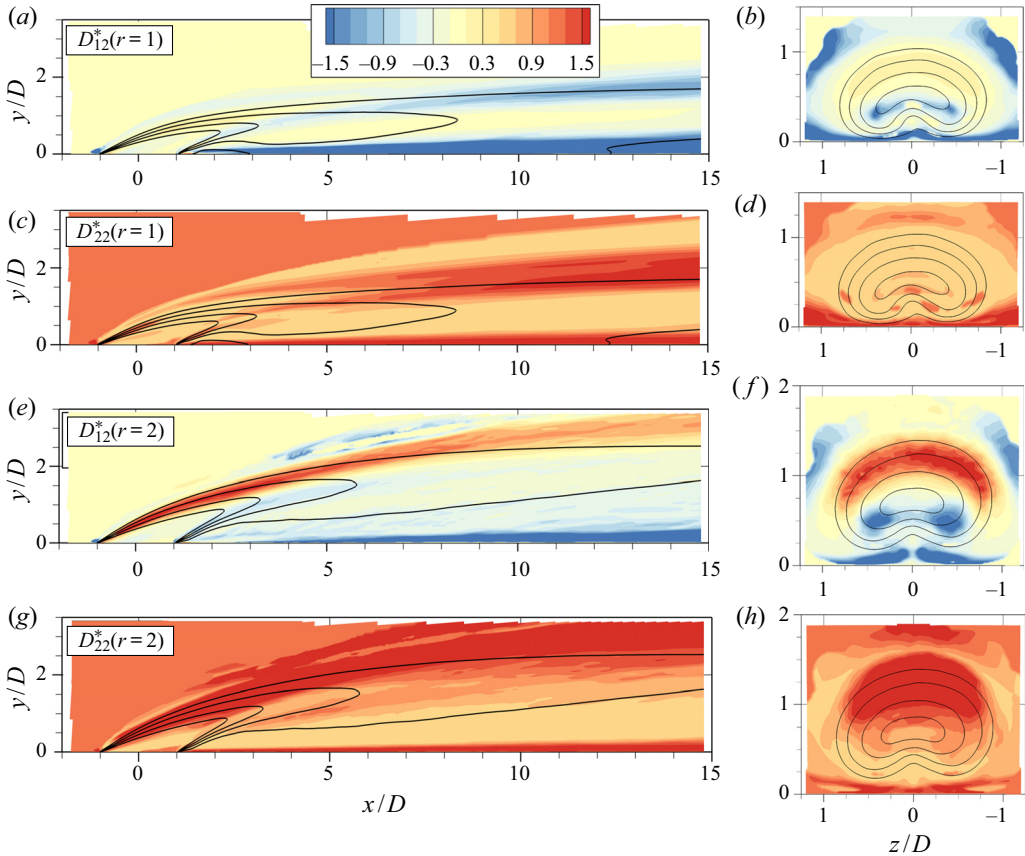
FIGURE 12. Dimensionless components of the diffusivity matrix **D** predicted by the TBNN-s in $r = 1$ and $r = 2$ cases. (a,c,e,g) Centreplane $(z = 0)$ and (b,d,f,h) axial planes at $x/D = 2$; black lines indicate isocontours of $\bar{c} = 0.2, 0.4, 0.6, 0.8$.

Figure 12 shows two different components of the turbulent diffusivity matrix predicted by the TBNN-s in the $r = 1$ and $r = 2$ cases. Here D^*_{12} is an off-diagonal component that controls the effect of wall-normal gradients on the streamwise transport (and, thus, is relevant for prediction of type 1 transport as described in § 3), and D^*_{22} is the wall-normal diagonal element. Note that a standard GDH model would have uniform components, with D^*_{12} being zero and D^*_{22} being $1/Sc_t$.

The component D^*_{12} has low magnitude in most regions of the flow, showing that cross-gradient effects are not important everywhere. But it is particularly active where the wall-normal mean velocity gradient is significant. This is to be expected, since strong vertical gradients in $\bar{u}$ would cause vertical eddies to correlate u' with c' in the presence of vertical concentration gradients. Close to the wall, this term is largest in magnitude because of the high velocity gradient due to the no-slip condition. It is also active in the windward shear layer, in locations of type 1 transport. While interesting from a physical perspective, these improvements by themselves do not lead to a significant improvement in mean scalar concentration, as discussed in appendix A. Note that the TBNN-s prediction of D^*_{12} in the windward shear layer has different signs in the $r = 1$ and $r = 2$ cases, which is the correct behaviour since $\partial\bar{u}/\partial y$ is positive for $r = 1$ and negative for $r = 2$ in that region. This showcases the capacity of the model to learn physics rather than just

regurgitate the training data; in this case, this is due to the basis expansion in (4.5), which explicitly multiplies the data-driven functions, $g^{(i)}$, by terms such as **S** which are sensitized to the local mean flow gradients.

The component D_{22}^* is positive everywhere as it should be, due to the positive semi-definite requirement on **D**. It does contain some spatial variation. In general, it is lower in the bottom half on the jet and higher in the top half, except for very near the wall where it attains the highest values. These results are consistent with the experimental results of Kohli & Bogard (2005) who reported smaller vertical heat diffusivity below the jet's centreline compared to above it. Relatively high values of D_{22}^* right above the wall and after injection appear in both the training set and in the test set, which is inconsistent with the regions of type 2 transport. This is because from the argument in § 3.2, we would expect D_{22}^* to be physically very small or even negative in this region since the counter gradient turbulent transport in the wall-normal direction is not caused by cross-gradients in the mean concentration. A possible reason for this inconsistency is that in those regions both the turbulent transport and the mean scalar gradients have very low magnitude, so the diffusivity entries are, in a sense, a ratio between two very small numbers. Thus, the TBNN-s which attempts to predict this ratio might fail to obtain satisfactory results.

4.3.2. *A posteriori results*

Finally, we show *a posteriori* results which test the ability of the model integrated into a computational fluid dynamics code to predict the mean concentration distribution. Appendix A includes a discussion on the relative importance of different components of the turbulent scalar flux on the *a posteriori* results discussed here. We compare results from two different models. The baseline is a simple, isotropic GDH where the turbulent diffusivity is prescribed with a fixed turbulent Schmidt number, $Sc_t = 0.85$ (Kays 1994). The second is the TBNN-s model which is proposed in the current paper and trained on the $r = 2$ dataset. Both modelled results use the mean velocity from the LES and depend on a baseline turbulence model for the momentum equations, which is the two-equation realizable $k - \epsilon$ model in the present work.

Figures 13 and 14 contain two-dimensional contours of the mean scalar concentration $\bar{c}$ in the $r = 1$ and $r = 2$ cases, respectively. In figure 13 the LES results indicate that the $r = 1$ jet detaches from the wall after injection and reattaches at approximately $x/D = 4$. The mean profile is distorted by the presence of the counter-rotating vortex pair as the axial (y–z) plane shows. The isotropic model, shown in figure 13(*c,d*), broadly overestimates mixing which transports scalar towards the wall immediately downstream of injection and speeds up the decay of the jet. In the axial view, the extent of the jet is larger in the spanwise direction due to the increased mixing. The TBNN-s model, as shown in figure 13(*e,f*), converges to a smooth solution, which is already an achievement for a tensorial and data-driven model. This solution is also significantly more accurate than the baseline model of figure 13(*c,d*): the streamwise decay of the concentration profile is predicted much more accurately and the extent of the jet's shear layer is more consistent with the LES data. As hinted by *a priori* results, the near-wall behaviour of the TBNN-s is still lacking, even though it represents an improvement over the simple GDH. This warrants future work to adapt the TBNN-s model to regions where non-local effects dominate and lead to counter gradient transport.

Figure 14 shows the same results for the $r = 2$ dataset. Here, since the jet is further from the wall, non-equilibrium effects are less significant so both models perform relatively better. The standard GDH model seems to slightly underpredict the streamwise decay of the jet, opposite from what was observed in $r = 1$. The TBNN-s model in figure 14(*e,f*)
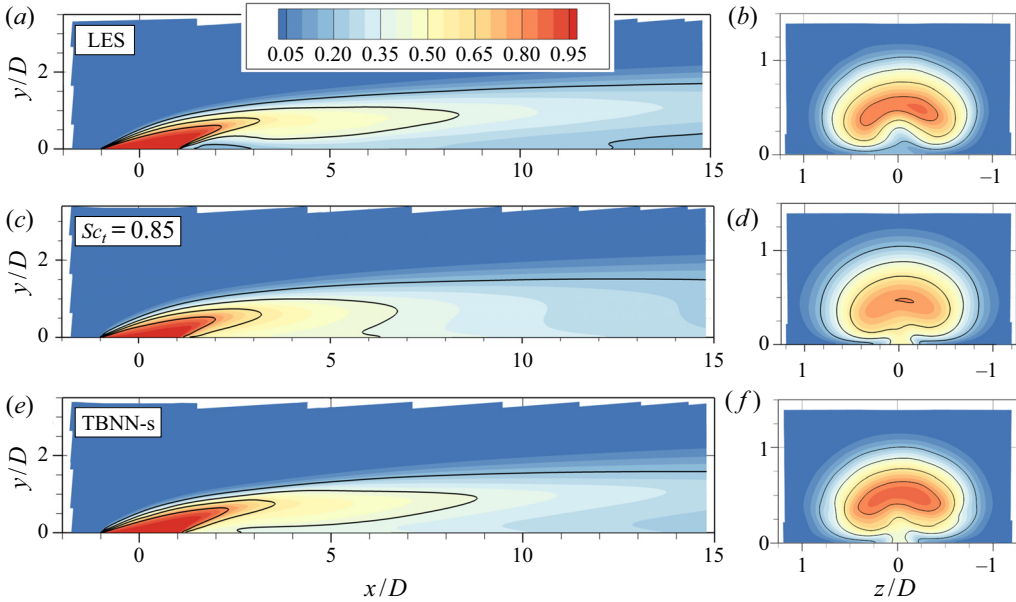
FIGURE 13. Mean scalar concentration contours in the $r = 1$ case, with (a,c,e) showing centreplane $(z = 0)$ and (b,d,f) showing axial $(y$–$z)$ planes at $x/D = 2$. Isocontours at $\bar{c} = 0.2, 0.4, 0.6, 0.8$ are also shown. (a,b) Contains LES results, (c,d) shows results from baseline GDH model, and (e,f) presents results from the TBNN-s.
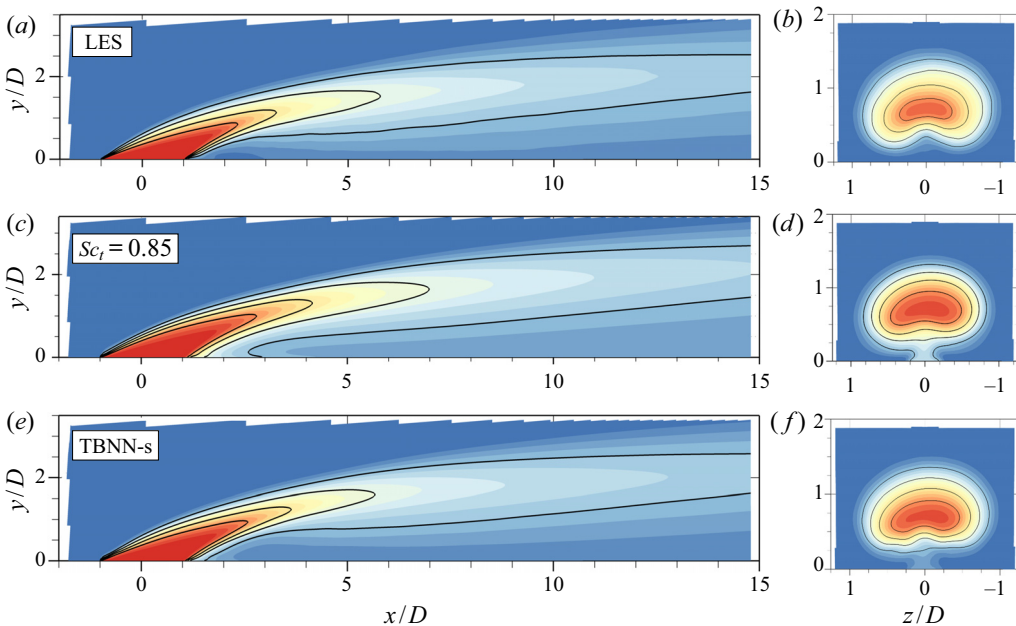

FIGURE 14. Mean scalar concentration contours in the $r = 2$ case, with same legend as figure 13.

manages to correct for that. This is encouraging because by employing a tensorial diffusivity, the data-driven model manages to correct the baseline model in both directions (by broadly decreasing the decay rate for $r = 1$ and increasing the decay rate in $r = 2$).
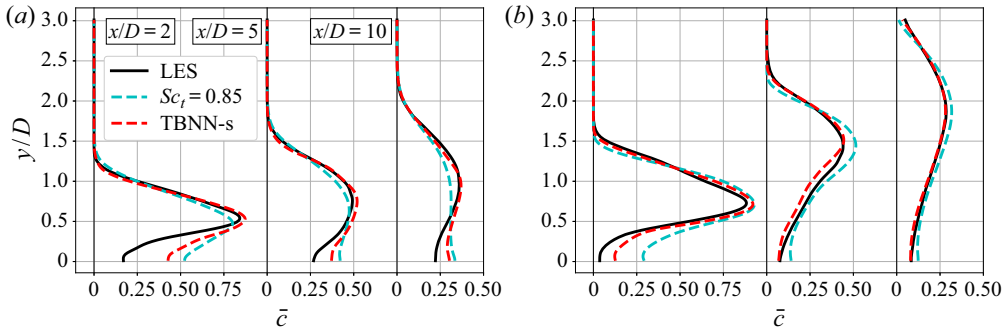
FIGURE 15. Vertical profiles of $\bar{c}$ at the centre spanwise position ($z = 0$) calculated in three distinct streamwise locations. (*a*) $r = 1$. (*b*) $r = 2$.
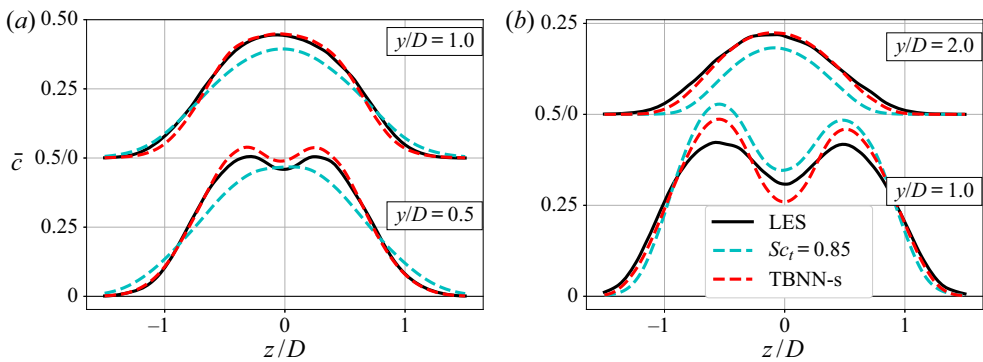

FIGURE 16. Horizontal profiles of $\bar{c}$ at $x/D = 5$ plotted for two distinct vertical stations. (*a*) $r = 1$. (*b*) $r = 2$.

This suggests that generalizable physics are indeed learned by the model during training and carry over to a different velocity ratio. As was observed in $r = 1$, the near-wall transport, particularly right after injection, is not captured accurately. Physically, as was discussed in § 3, this is due to the prevalence of non-equilibrium effects in this area, which cannot be modelled with a local diffusivity-based model (as evidenced by type 2 counter gradient transport).

For a more quantitative comparison between the same results, figure 15 shows vertical profiles of mean scalar concentration at three different streamwise locations. Despite its shortcomings, it is worth noting that the GDH with $Sc_t = 0.85$ does a reasonably good job at capturing broad qualitative features of the scalar field given its simplicity. The TBNN-s model, however, presents clear improvements: it does a good job at predicting the streamwise decay of the peak jet concentration and the vertical concentration gradient in the windward shear layer. The highest discrepancy between the TBNN-s model and LES data is exactly where type 2 gradient transport was found. Figure 16 shows spanwise profiles of mean scalar concentration at $x/D = 5$ and further reinforces the points made in this section, especially the accuracy with which the data-driven tensorial diffusivity can predict the jet shear layer profiles.

4.3.3. *Basis coefficients results*

Lastly, we would like to gain some further understanding of the TBNN-s model. Deep neural networks are notoriously powerful but hard to interpret (Goodfellow, Bengio & Courville 2016) and physical understanding of machine-learned turbulence models
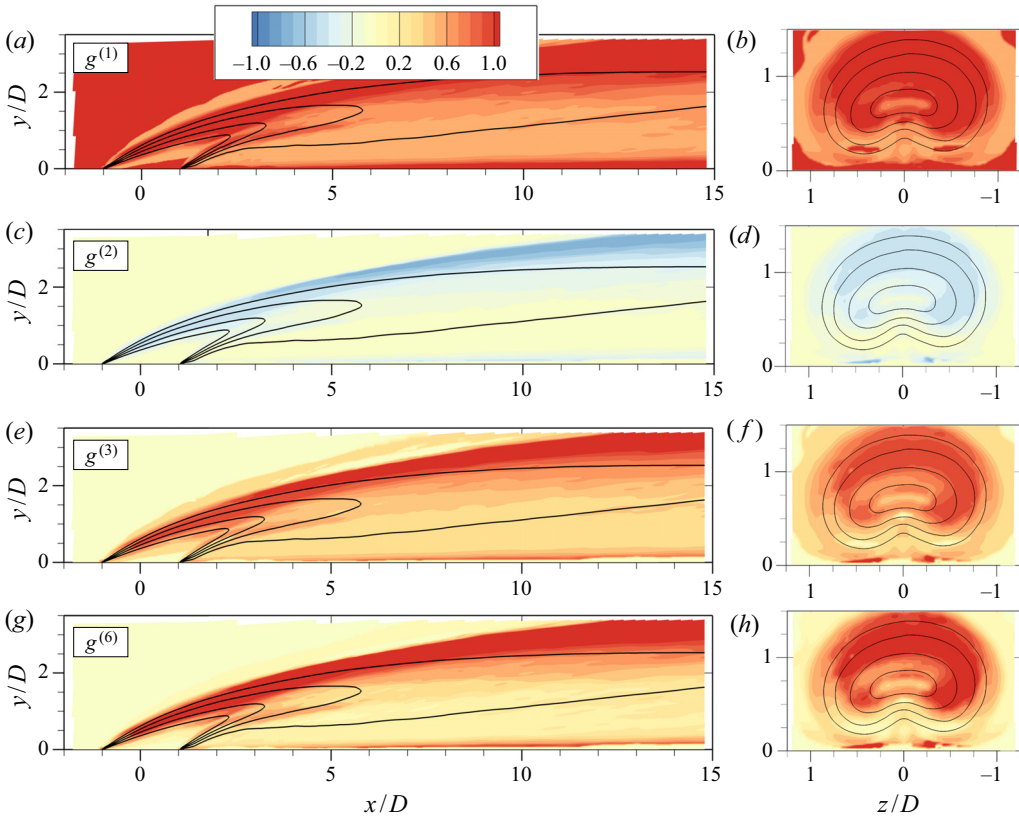
FIGURE 17. Colour contours of the basis coefficients $g^{(i)}$ predicted by the TBNN-s model in the $r = 2$ jet. (*a,c,e,g*) Centreplane ($z = 0$) and (*b,d,f,h*) axial planes at $x/D = 2$; black lines indicate isocontours of $\bar{c} = 0.2, 0.4, 0.6, 0.8$.

remains an important open question (Duraisamy *et al.* 2019), so we do not intend to exhaust that question here. One method to interpret the model takes advantage of the TBNN-s structure which writes the turbulent diffusivity matrix as a weighted sum of simpler tensor basis elements. By analysing the relative contributions of such simpler elements, one can gain useful insight into the model. Thus, in the current subsection we look at the tensor basis weights $g^{(i)}$ prescribed by the TBNN-s in these flows. Recall that the structure in figure 10 outputs a dimensionless turbulent diffusivity matrix $\boldsymbol{D}^*$ which has $O(1)$ entries, and the diffusivity basis elements $\boldsymbol{T}^{(n)}$ (see (4.6)) are also non-dimensionalized such that in practice they have $O(1)$ entries. Thus, the predicted values of tensor basis weights $g^{(i)}$ should be dimensionless, $O(1)$ quantities.

Figure 17 contains contours of four different basis coefficients ($g^{(1)}$, $g^{(2)}$, $g^{(3)}$ and $g^{(6)}$) in the $r = 2$ jet in crossflow. Since the TBNN-s was trained on the $r = 2$ data, this is the machine-learned model's attempt at reproducing its training data within the tensor basis decomposition. The coefficients $g^{(4)}$ and $g^{(5)}$ are not shown because the TBNN-s assigns very low values for both of these compared to the other weights. This suggests that the quadratic and symmetric basis elements, $\boldsymbol{T}^{(4)} = \boldsymbol{S}^2$ and $\boldsymbol{T}^{(5)} = \boldsymbol{R}^2$, are not employed by the network to match the training data. All the coefficients shown have well-defined signs as learned by the deep neural network: $g^{(1)}$, $g^{(3)}$ and $g^{(6)}$ are positive, while $g^{(2)}$ is negative. Also note that while $g^{(1)}$ and $g^{(2)}$ multiply symmetric tensors, $g^{(3)}$ and $g^{(6)}$ multiply anti-symmetric basis elements. This suggests that the strictly symmetric diffusivity matrix
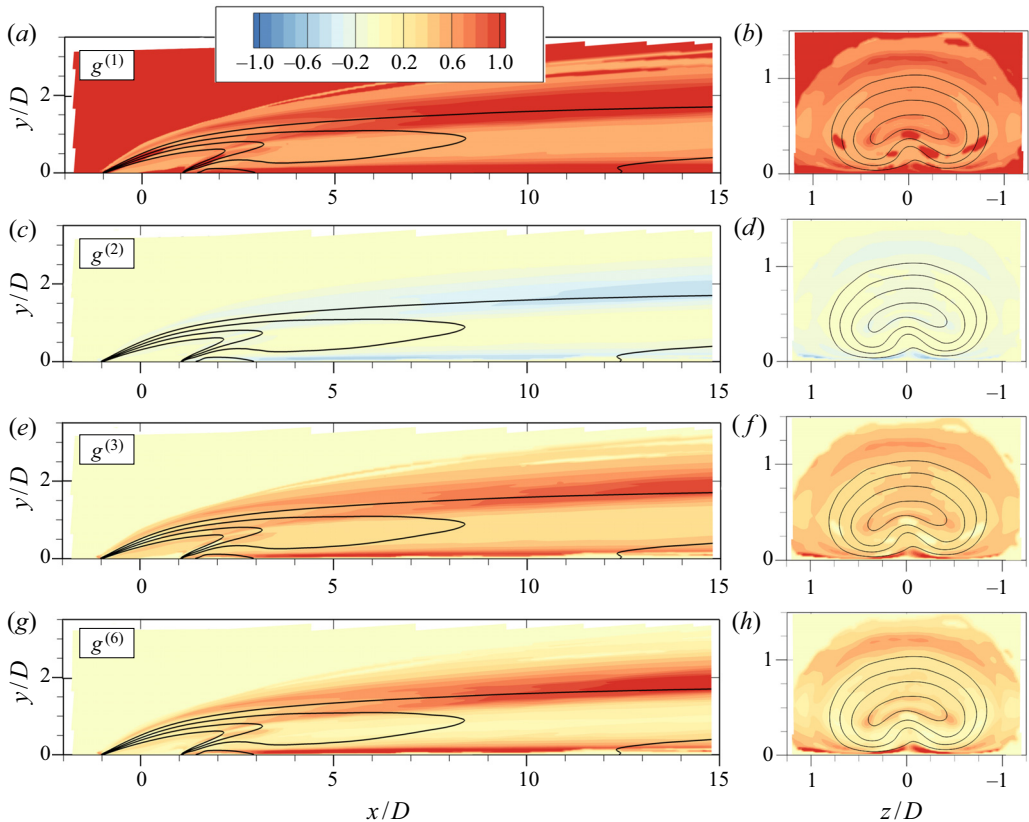
FIGURE 18. Colour contours of the basis coefficients $g^{(i)}$ predicted by the TBNN-s model in the $r = 1$ jet. (a,c,e,g) Centreplane ($z = 0$) and (b,d,f,h) axial planes at $x/D = 2$; black lines indicate isocontours of $\bar{c} = 0.2, 0.4, 0.6, 0.8$.

hypothesized by some previous authors (e.g. Daly & Harlow 1970) is insufficient to capture the data in this case, even in locations where the turbulent scalar flux is mostly locally determined such as in the windward shear layer.

Other features can be observed in figure 17. For example, the basis coefficients seem to attain generally higher magnitudes in locations of highest shear, such as right above the wall and in the windward shear layer. The coefficient $g^{(1)}$ determines the isotropic contribution and is the only coefficient allowed in the simple GDH of (1.2); the TBNN-s prescribes values roughly between 0.5 and 1.5 to that coefficient, with lower magnitudes in locations of type 2 transport (compare figures 9e,f and 17a,b) which would be the expected behaviour if the turbulent transport in those locations is to be captured. On the other hand, $g^{(1)}$ is highest in the windward shear layer, indicating that the standard GDH underestimates the isotropic mixing in those locations.

Figure 18 shows equivalent results for the $r = 1$ jet in crossflow, where the model is tested outside of its training data. The overall patterns are very similar from before; the coefficients $g^{(4)}$ and $g^{(5)}$ are significantly smaller and thus not shown. The similarity in the coefficients between the calibration data and the test data indicates that the modelling framework, in particular the tensor basis expansion, is indeed sound. One difference between the $r = 1$ and $r = 2$ results is that the former seem to have slightly lower

magnitudes, indicating that differences in the modelled eddy viscosity ν_t are not enough to capture the increased turbulent mixing from the $r = 1$ to $r = 2$ jet.

This opportunity for interpretation naturally arises from the tensor basis expansion described in §4.1. The patterns indicated here can be used to further improve and simplify the models. For example, future work could remove the $\boldsymbol{T}^{(4)}$ and $\boldsymbol{T}^{(5)}$ bases, or leverage the overall structure of figures 17 and 18 to prescribe uniform ratios $g^{(2)}/g^{(1)}$, $g^{(3)}/g^{(1)}$ and $g^{(6)}/g^{(1)}$. We also believe that interpretation work should proceed in the direction of understanding the functional form $g^{(n)}(\lambda_i)$ so that one can answer the question of why the deep neural network is predicting its results. The approaches exemplified in Ling *et al.* (2017) and Milani, Ling & Eaton (2019*b*) could provide a useful starting point for this endeavour. As mentioned before, this is a major open question in the field of deep learning particularly for physical applications.

5. Concluding remarks

Two highly resolved large eddy simulations of inclined jets in crossflow are used to analyse turbulent scalar transport. In §3 we discuss physical origins and modelling implications of counter gradient turbulent transport. Counter gradient turbulent transport is classified into two types. Type 1 arises in the streamwise component of the turbulent scalar flux in the windward shear layer, and is caused by cross-gradient effects, whereby gradients in the wall-normal and spanwise direction control the scalar flux. This phenomenon is mostly local and is consistent with the reports of previous authors such as Muppidi & Mahesh (2008), Bodart *et al.* (2013) and Schreivogel *et al.* (2016). Type 2 transport, on the other hand, is governed chiefly by non-local contributions to the scalar flux budget. It is observed close to the wall, after injection, and in the wall-normal component of the turbulent transport, consistent with the observations of Salewski *et al.* (2008) and Milani & Eaton (2018). Despite multiple reports of counter gradient transport in these flows, authors have not carefully examined its causes. For example, Bodart *et al.* (2013) reported observations consistent with type 1 transport in their windward shear layer, but erroneously alluded to non-local effects as possible causes (which actually cause type 2 transport as shown presently). These discussions show that type 1 and type 2 phenomena would require models of different complexities. The former could be modelled with a local formulation, but would require a tensorial diffusivity to capture cross-gradient effects. The latter cannot be modelled adequately with any diffusivity-based model since it is inherently non-local; it might require, for example, solving separate transport equations for the scalar flux components.

Given the implementation difficulties associated with non-local models (Combest *et al.* 2011), we choose to use a matrix turbulent diffusivity with a machine learning approach. We adapt the tensor basis neural network of Ling *et al.* (2016*a*) to model the scalar flux, and denote this approach TBNN-s (tensor basis neural network for scalar flux modelling). Our deep learning model obeys rotational invariance through a tensor basis expansion, and is able to predict a tensorial diffusivity $\boldsymbol{D}$ that can be used in the Reynolds averaged scalar transport equations even though the training data do not contain this matrix explicitly. As expected, the turbulent diffusivity matrix improves the alignment of the turbulent scalar flux vector, and also increases the accuracy of the mean scalar concentration field. This happens not just in the training set, but also in the test flow. The TBNN-s model proposed herein can be applied in any other problem where scalar transport is of interest given adequate training data. We expect that this modelling approach could be particularly useful to practitioners who are interested in one specific class of turbulent flow and have

a few high-fidelity datasets to train their models. Note that unlike many other machine learning works in the recent fluid mechanics literature, this model can be directly applied to arbitrarily complex 3-D turbulent flows and computational meshes. The results in this paper show that our framework is a good candidate to be employed in turbulent shear flows, where cross-gradient transport effects are important. As discussed in § 4.3.3, future workers might focus on further developing the model by simplifying it; or on further interpreting deep neural networks used in physical applications, to understand how the input invariants are mapped to output turbulence quantities of interest.

Acknowledgements

Declaration of interests

The authors report no conflict of interest.

Appendix A. Relative magnitude of the turbulent scalar flux

One noteworthy aspect of modelling the turbulent scalar flux is that its contribution to the overall transport of the passive scalar varies with location and direction within the flow. In general, the passive scalar equation balances the mean advection and the turbulent scalar flux (assuming that molecular diffusion is negligible in such flows) as show in

$$\frac{\partial}{\partial x_i}(\bar{u}_i \bar{c}) = \frac{\partial}{\partial x_i}\left(-\overline{u_i' c'}\right). \tag{A 1}$$

In shear flows such as the jet in crossflow, length scales and mean velocities in the streamwise (x) direction are much larger than in the transverse directions (y and z), so streamwise mean advection is usually thought as balancing turbulent mixing in the transverse direction. This means that the computation of the mean scalar field through the RANS (A 1) is more sensitive to errors in the transverse components of the turbulent scalar flux ($\overline{v'c'}$ and $\overline{w'c'}$) than it is to errors in the streamwise component $\overline{u'c'}$. For a visualization, figure 19 shows the magnitude of the turbulent scalar flux components relative to the equivalent mean advection component from the $r = 1$ jet LES. As the conceptual discussion suggests, $\overline{v'c'}$ is relatively more important than $\overline{u'c'}$; its magnitude is particularly high in the shear layer all around the jet and in some places close to the wall and right after injection.

This suggests that accurately capturing type 1 transport does not impact significantly the predicted mean scalar field in this particular flow; on the other hand, capturing type 2 is much more important. The goal of discussing type 1 transport in § 3.1 is to explain the exact observations of previous authors regarding counter gradient transport (Muppidi & Mahesh 2008; Bodart *et al.* 2013; Schreivogel *et al.* 2016) and show a specific physical mechanism that causes qualitatively incorrect predictions in the turbulent scalar flux in jets in crossflow (namely, cross-gradient transport). This physical analysis leads us to investigate machine-learned models that incorporate cross-gradient transport effects through a tensorial diffusivity, which is the focus of § 4. This improved model is not only able to capture specifically type 1 transport (as shown in figure 20*a*), but it also improves
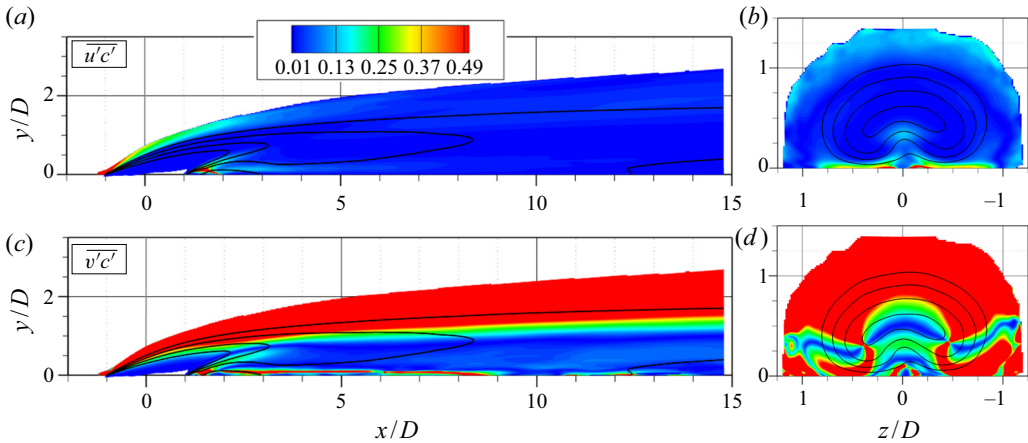
FIGURE 19. Magnitude of turbulent scalar flux components relative to the equivalent mean advection component in the $r = 1$ jet. Black lines showing isocontours at $\bar{c} = 0.2, 0.4, 0.6, 0.8$ are also shown. (a,b) The streamwise component $|\overline{u'c'}/(\bar{u}\bar{c})|$ and (c,d) the wall-normal component $|\overline{v'c'}/(\bar{v}\bar{c})|$.
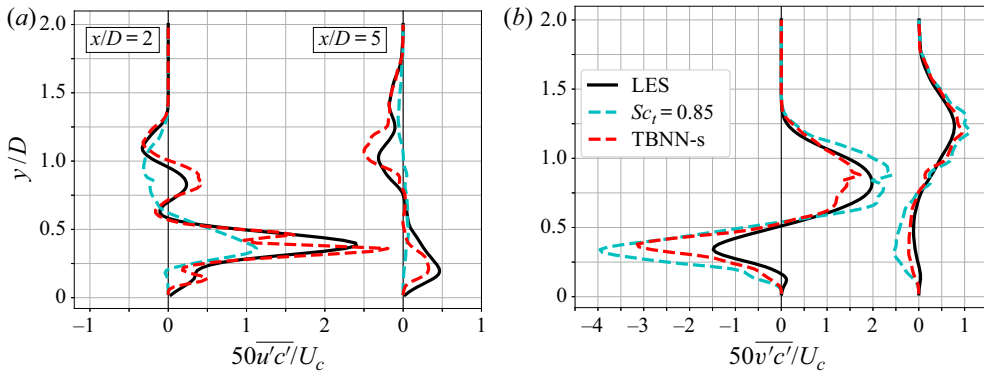

FIGURE 20. Vertical profiles of the turbulent scalar flux in the $r = 1$ jet at the centreplane $(z/D = 0)$ and two different streamwise locations. Lines compare LES results to the baseline model (GDH with $Sc_t = 0.85$) and to the TBNN-s model trained on the $r = 2$ jet (as discussed in § 4.3). (a) The streamwise component $\overline{u'c'}$ of the flux, and (b) the wall-normal component $\overline{v'c'}$.

the prediction of the more important components such as $\overline{v'c'}$, as shown in figure 20(b). This explains its significant improvement over the baseline model.

Appendix B. Stability of a tensorial diffusivity

In this appendix we discuss the requirement for stability of a matrix turbulent diffusivity used to model the scalar flux in the Reynolds averaged advection diffusion equation. If the diffusivity were a scalar, it would have to be non-negative; for a matrix diffusivity, there is an analogous requirement: that the 3×3 matrix be positive semi-definite. To see this, we multiply both sides of (4.10) by $\bar{c}$ to derive a transport equation for the squared

concentration $\phi = \bar{c}^2/2$, as shown in

$$\bar{c} \times \left[\frac{\partial}{\partial x_i}(\bar{u}_i \bar{c}) = \frac{\partial}{\partial x_i}\left(\frac{\nu}{Sc}\frac{\partial \bar{c}}{\partial x_i} \right) + \frac{\partial}{\partial x_i}\left(D_{ij}\frac{\partial \bar{c}}{\partial x_j} \right) \right]. \tag{B 1}$$

The first term in (B 1) can easily be manipulated since we assume the mean velocity field is divergence free, and simply becomes an advection of $\phi = \bar{c}^2/2$. The two terms on the left-hand side require a slightly more sophisticated application of the chain rule of differentiation, namely

$$\bar{c}\frac{\partial}{\partial x_i}\left(\frac{\nu}{Sc}\frac{\partial \bar{c}}{\partial x_i} \right) = \frac{\partial}{\partial x_i}\left(\frac{\nu}{Sc}\frac{\partial \left(\frac{1}{2}\bar{c}^2\right)}{\partial x_i} \right) - \frac{\nu}{Sc}\left(\frac{\partial \bar{c}}{\partial x_i}\frac{\partial \bar{c}}{\partial x_i} \right) \tag{B 2}$$

and

$$\bar{c}\frac{\partial}{\partial x_i}\left(D_{ij}\frac{\partial \bar{c}}{\partial x_j} \right) = \frac{\partial}{\partial x_i}\left(D_{ij}\frac{\partial \left(\frac{1}{2}\bar{c}^2\right)}{\partial x_j} \right) - D_{ij}\left(\frac{\partial \bar{c}}{\partial x_i}\frac{\partial \bar{c}}{\partial x_j} \right). \tag{B 3}$$

Now, using (B 2) and (B 3) we rewrite (B 1) as

$$\frac{\partial}{\partial x_i}(\bar{u}_i \phi) = \frac{\partial}{\partial x_i}\left(\frac{\nu}{Sc}\frac{\partial \phi}{\partial x_i} \right) + \frac{\partial}{\partial x_i}\left(D_{ij}\frac{\partial \phi}{\partial x_j} \right) - \frac{\nu}{Sc}\left(\frac{\partial \bar{c}}{\partial x_i}\frac{\partial \bar{c}}{\partial x_i} \right) - D_{ij}\left(\frac{\partial \bar{c}}{\partial x_i}\frac{\partial \bar{c}}{\partial x_j} \right). \tag{B 4}$$

Note that the first three terms of (B 4) are the regular mean advection and diffusion terms of quantity ϕ. The last two act as source terms. For the diffusion of c to be stable, it must act to dissipate the quantity ϕ (the same way that momentum diffusion acts to dissipate kinetic energy), thus, the source terms must be non-positive. For the molecular diffusion, this is true if the diffusion coefficient ν/Sc is non-negative. The last term of (B 4) is a quadratic form of matrix $\mathbf{D}$ and will be non-positive if $\nabla \bar{c}^T \mathbf{D} \nabla \bar{c} \geq 0$. For this to hold for any mean concentration field, we require that $\mathbf{D}$ be positive semi-definite (which by definition means that the quadratic form of $\mathbf{D}$ with any vector is non-negative).

The remaining question is, given an arbitrary matrix diffusivity $\mathbf{D}$, how to test whether it is positive semi-definite? For a symmetric 3×3 matrix, the simplest test lies in its eigenvalues: a symmetric matrix is positive semi-definite if and only if all its eigenvalues are non-negative. For a general matrix, with complex eigenvalues, one extra step is needed. First, we split $\mathbf{D}$ into symmetric and anti-symmetric components $\mathbf{A}$ and $\mathbf{W}$, respectively:

$$\mathbf{D} = \overbrace{\tfrac{1}{2}(\mathbf{D} + \mathbf{D}^T)}^{\mathbf{A}} + \overbrace{\tfrac{1}{2}(\mathbf{D} - \mathbf{D}^T)}^{\mathbf{W}}. \tag{B 5}$$

Then, we recognize that the quadratic form of an anti-symmetric matrix with any real vector is zero. Thus, the quadratic form of $\mathbf{D}$ with an arbitrary vector $\mathbf{v}$ is identical to that of its symmetric part $\mathbf{A} = (\mathbf{D} + \mathbf{D}^T)/2$ with $\mathbf{v}$, as shown in (B 6). From that, we see that $\mathbf{D}$ is positive semi-definite if and only if its symmetric part $\mathbf{A}$ is positive semi-definite, since

$$\mathbf{v}^T \mathbf{D} \mathbf{v} = \mathbf{v}^T \mathbf{A} \mathbf{v} + \mathbf{v}^T \mathbf{W} \mathbf{v}.^{\!\!\!0} \tag{B 6}$$

In summary, to check whether an arbitrary turbulent diffusivity matrix $\mathbf{D}$ is positive semi-definite, we look at its symmetric part: $\mathbf{D}$ will be positive semi-definite if and only if $(\mathbf{D} + \mathbf{D}^T)/2$ contains only non-negative eigenvalues. This criterium is used to post process the output of the TBNN-s, which leads to a stable advection diffusion equation with a matrix turbulent diffusivity. Note that having only non-negative diagonal entries is a necessary but not sufficient condition for $\mathbf{D}$ to be positive semi-definite. On the other hand, a positive semi-definite matrix can have positive, zero or negative off-diagonal entries.

REFERENCES

ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., CORRADO, G. S., DAVIS, A., DEAN, J., DEVIN, M., *et al.* 2016 Tensorflow: large-scale machine learning on heterogeneous distributed systems. arXiv:1603.04467.

ABE, K. & SUGA, K. 2001 Towards the development of a Reynolds-averaged algebraic turbulent scalar-flux model. *Intl J. Heat Fluid Flow* **22** (1), 19–29.

ACHARYA, S., TYAGI, M. & HODA, A. 2001 Flow and heat transfer predictions for film cooling. *Ann. N. Y. Acad. Sci.* **934** (1), 110–125.

BATCHELOR, G. K. 1949 Diffusion in a field of homogeneous turbulence. I. Eulerian analysis. *Austral. J. Chem.* **2** (4), 437–450.

BODART, J., COLETTI, F., BERMEJO-MORENO, I. & EATON, J. K. 2013 High-fidelity simulation of a turbulent inclined jet in a crossflow. *Center Turbul. Res. Annu. Res. Briefs* **19**, 263–275.

BOGARD, D. G. & THOLE, K. A. 2006 Gas turbine film cooling. *J. Propul. Power* **22** (2), 249–270.

COMBEST, D. P., RAMACHANDRAN, P. A. & DUDUKOVIC, M. P. 2011 On the gradient diffusion hypothesis and passive scalar transport in turbulent flows. *Ind. Engng Chem. Res.* **50** (15), 8817–8823.

DALY, B. J. & HARLOW, F. H. 1970 Transport equations in turbulence. *Phys. Fluids* **13** (11), 2634–2649.

DURAISAMY, K., IACCARINO, G. & XIAO, H. 2019 Turbulence modeling in the age of data. *Annu. Rev. Fluid Mech.* **51**, 357–377.

FRIC, T. F. & ROSHKO, A. 1994 Vortical structure in the wake of a transverse jet. *J. Fluid Mech.* **279**, 1–47.

GARDNER, J. E., BURGISSER, A. & STELLING, P. 2007 Eruption and deposition of the Fisher Tuff (Alaska): evidence for the evolution of pyroclastic flows. *J. Geol.* **115** (4), 417–435.

GOODFELLOW, I., BENGIO, Y. & COURVILLE, A. 2016 *Deep Learning*. MIT Press.

HAM, F. 2007 An efficient scheme for large eddy simulation of low-Ma combustion in complex configurations. *Center Turbul. Res. Annu. Res. Briefs* **4**, 41–46.

HODA, A. & ACHARYA, S. 1999 Predictions of a film coolant jet in crossflow with different turbulence models. *J. Turbomach.* **122** (3), 558–569.

KAYS, W. M. 1994 Turbulent Prandtl number – where are we? *J. Heat Transfer* **116** (2), 284–295.

KINGMA, D. P. & BA, J. 2014 Adam: a method for stochastic optimization. arXiv:1412.6980.

KOHLI, A. & BOGARD, D. G. 2005 Turbulent transport in film cooling flows. *J. Heat Transfer* **127** (5), 513–520.

LING, J., HUTCHINSON, M., ANTONO, E., DECOST, B., HOLM, E. A. & MEREDIG, B. 2017 Building data-driven models with microstructural images: generalization and interpretability. *Mater. Discovery* **10**, 19–28.

LING, J., KURZAWSKI, A. & TEMPLETON, J. 2016*a* Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *J. Fluid Mech.* **807**, 155–166.

LING, J., RYAN, K. J., BODART, J. & EATON, J. K. 2016*b* Analysis of turbulent scalar flux models for a discrete hole film cooling flow. *J. Turbomach.* **138** (1), 011006.

MAHESH, K. 2013 The interaction of jets with crossflow. *Annu. Rev. Fluid Mech.* **45**, 379–407.

MAULIK, R., SAN, O., JACOB, J. D. & CRICK, C. 2019 Sub-grid scale model classification and blending through deep learning. *J. Fluid Mech.* **870**, 784–812.

MILANI, P. M. & EATON, J. K. 2018 Magnetic resonance imaging, optimization, and machine learning to understand and model turbulent mixing. In *21st Australasian Fluid Mechanics Conference* (ed. I. Marusic), Australasian Fluid Mechanics Society.

MILANI, P. M., GUNADY, I. E., CHING, D. S., BANKO, A. J., ELKINS, C. J. & EATON, J. K. 2019*a* Enriching MRI mean flow data of inclined jets in crossflow with large eddy simulations. *Intl J. Heat Fluid Flow* **80**, 108472.

MILANI, P. M., LING, J. & EATON, J. K. 2019*b* Physical interpretation of machine learning models applied to film cooling flows. *J. Turbomach.* **141** (1), 011004.

MILANI, P. M., LING, J. & EATON, J. K. 2020 Generalization of machine-learned turbulent heat flux models applied to film cooling flows. *J. Turbomach.* **142** (1), 011007.

MILANI, P. M., LING, J., SAEZ-MISCHLICH, G., BODART, J. & EATON, J. K. 2018 A machine learning approach for determining the turbulent diffusivity in film cooling flows. *J. Turbomach.* **140** (2), 021006.

MUPPIDI, S. & MAHESH, K. 2008 Direct numerical simulation of passive scalar transport in transverse jets. *J. Fluid Mech.* **598**, 335–360.

PARISH, E. J. & DURAISAMY, K. 2016 A paradigm for data-driven predictive modeling using field inversion and machine learning. *J. Comput. Phys.* **305**, 758–774.

POPE, S. B. 1975 A more general effective-viscosity hypothesis. *J. Fluid Mech.* **72** (2), 331–340.

RYAN, K. J., BODART, J., FOLKERSMA, M., ELKINS, C. J. & EATON, J. K. 2017 Turbulent scalar mixing in a skewed jet in crossflow: experiments and modeling. *Flow Turbul. Combust.* **98** (3), 781–801.

SALEWSKI, M., STANKOVIC, D. & FUCHS, L. 2008 Mixing in circular and non-circular jets in crossflow. *Flow Turbul. Combust.* **80** (2), 255–283.

SANDBERG, R. D., TAN, R., WEATHERITT, J., OOI, A., HAGHIRI, A., MICHELASSI, V. & LASKOWSKI, G. 2018 Applying machine learnt explicit algebraic stress and scalar flux models to a fundamental trailing edge slot. *J. Turbomach.* **140** (10), 101008.

SCHREIVOGEL, P., ABRAM, C., FOND, B., STRAUSSWALD, M., BEYRAU, F. & PFITZNER, M. 2016 Simultaneous kHz-rate temperature and velocity field measurements in the flow emanating from angled and trenched film cooling holes. *Intl J. Heat Mass Transfer* **103**, 390–400.

SHIH, T. H., ZHU, J. & LUMLEY, J. L. 1995 A new Reynolds stress algebraic equation model. *Comput. Methods Appl. Mech. Engng* **125** (1), 287–302.

SOTGIU, C., WEIGAND, B. & SEMMLER, K. 2018 A turbulent heat flux prediction framework based on tensor representation theory and machine learning. *Intl Commun. Heat Mass Transfer* **95**, 74–79.

SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I. & SALAKHUTDINOV, R. 2014 Dropout: a simple way to prevent neural networks from overfitting. *J. Machine Learning Res.* **15** (1), 1929–1958.

SU, L. K. & MUNGAL, M. G. 2004 Simultaneous measurements of scalar and velocity field evolution in turbulent crossflowing jets. *J. Fluid Mech.* **513**, 1–45.

TAVOULARIS, S. & CORRSIN, S. 1981 Experiments in nearly homogenous turbulent shear flow with a uniform mean temperature gradient. Part 1. *J. Fluid Mech.* **104**, 311–347.

VREMAN, A. W. 2004 An eddy-viscosity subgrid-scale model for turbulent shear flow: algebraic theory and applications. *Phys. Fluids* **16** (10), 3670–3681.

WEATHERITT, J., SANDBERG, R. D., LING, J., SAEZ, G. & BODART, J. 2017 A comparative study of contrasting machine learning frameworks applied to RANS modeling of jets in crossflow. In *ASME Turbo Expo 2017: Turbomachinery Technical Conference and Exposition*. American Society of Mechanical Engineers Digital Collection.

WEATHERITT, J., ZHAO, Y., SANDBERG, R. D., MIZUKAMI, S. & TANIMOTO, K. 2020 Data-driven scalar-flux model development with application to jet in cross flow. *Intl J. Heat Mass Transfer* **147**, 118931.

WU, J., XIAO, H., SUN, R. & WANG, Q. 2019 Reynolds-averaged Navier–Stokes equations with explicit data-driven Reynolds stress closure can be ill-conditioned. *J. Fluid Mech.* **869**, 553–586.

XIE, Z. T. & CASTRO, I. P. 2008 Efficient generation of inflow conditions for large eddy simulation of street-scale flows. *Flow Turbul. Combust.* **81** (3), 449–470.

YOUNIS, B. A., SPEZIALE, C. G. & CLARK, T. T. 2005 A rational model for the turbulent scalar fluxes. *Proc. R. Soc.* A **461** (2054), 575–594.

YUAN, L. L., STREET, R. L. & FERZIGER, J. H. 1999 Large-eddy simulations of a round jet in crossflow. *J. Fluid Mech.* **379**, 71–104.

ZHENG, Q. S. 1994 Theory of representations for tensor functions – a unified invariant approach to constitutive equations. *Appl. Mech. Rev.* **47** (11), 545–587.