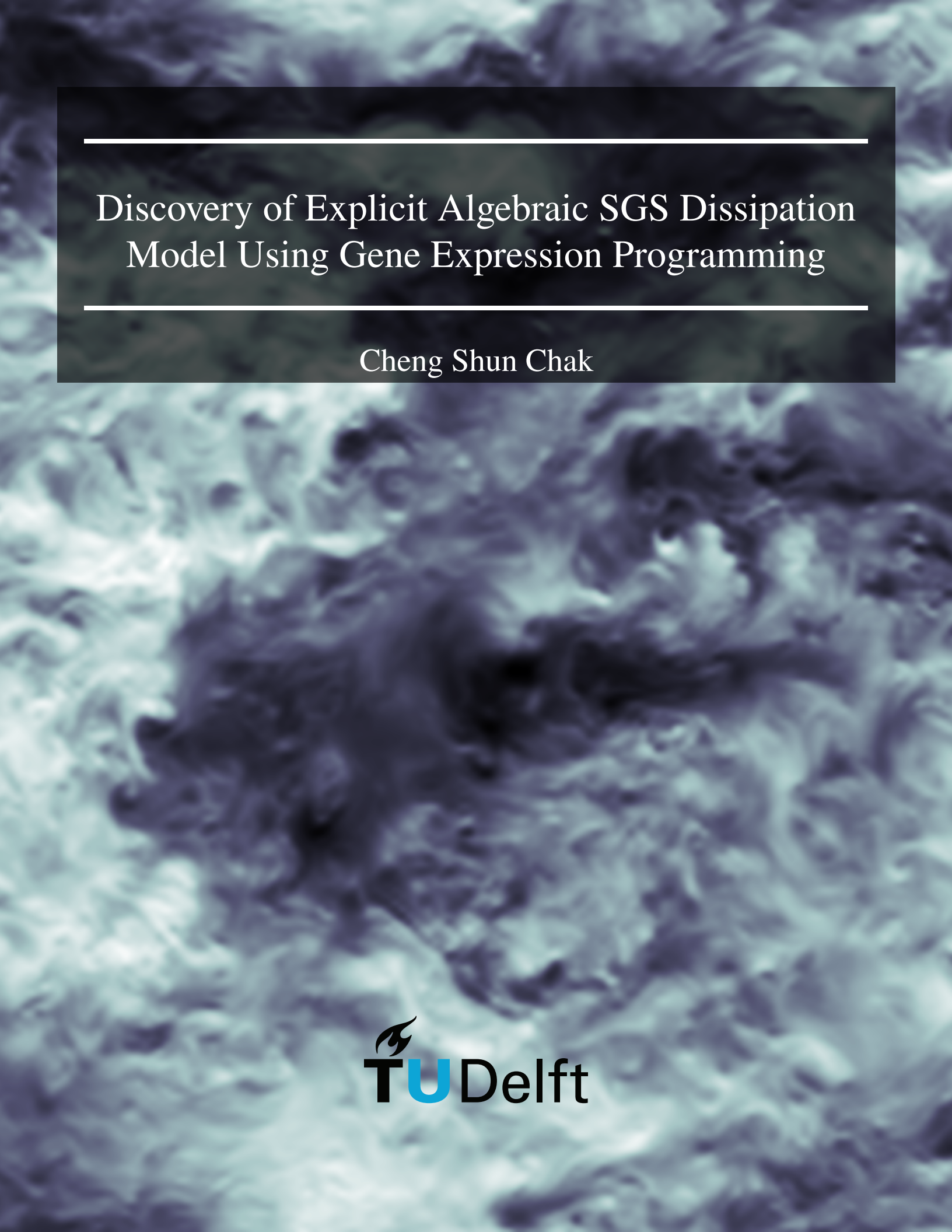
Discovery of Explicit Algebraic SGS Dissipation Model Using Gene Expression Programming

Cheng Shun Chak



Discovery of Explicit Algebraic SGS Dissipation Model Using Gene Expression Programming

by

Cheng Shun Chak

to obtain the degree of Master of Science
at the Delft University of Technology,

to be publicly defended on Thursday, 18th December at 13:30

Student number:	4835549
Project duration:	February 2025 – December 2025
Thesis committee:	Dr.ir. M.I. Gerritsma, TU Delft, Chair
	Prof. Dr.-Ing. habil S. Hickel, TU Delft, Supervisor
	Dr.ir. F. de Prenter, TU Delft, External member
Faculty:	Faculty of Aerospace Engineering, Delft
MSc Track:	Aerodynamics & Wind Energy

TU Delft

Discovery of Explicit Algebraic SGS Dissipation Model Using Gene Expression Programming

Cheng Shun Chak
Technische Universiteit Delft, Faculty of Aerospace Engineering, Aerodynamics group

Explicit algebraic subgrid-scale models (EASSMs) have shown better performance compared to eddy viscosity models (EVMs) due to the improved tensorial alignment to the subgrid-scale (SGS) stress tensor. Under the EASSM framework, the dissipation term requires closure. A number of models exist but some are unsuitable to be applied under the EASSM framework, due to their non-linearity in the SGS anisotropy tensor. The isotropic dissipation model is commonly applied model but studies have contradicted the isotropic assumption. We propose to discover a closure model suitable to be implemented under the EASSM framework using gene expression programming (GEP) as a symbolic regression tool. Techniques such as wall scaling constraints and numerical constant optimization from other works are also incorporated in the present work. In addition, we attempt to mix training data from homogeneous isotropic turbulence (HIT) and turbulent channel flow (TCF) within the training process in order to obtain a more robust model. Discovery of a scalar dissipation model is an intermediate step towards a dissipation tensor model, a priori evaluation shows better performance relative to a reference model. Validation of the scalar model was performed on a channel flow case which also showed improvements. Tensor model discovery produced a few models which provide better tensorial alignment only in selected cases relative to there being no model at all. A critical review of the present work recommended the investigation of implementing an alternate timescale for non-dimensionalization and a study on the dissipation effects that are contained within the pressure-strain model implemented in the framework.

I. Introduction

Large Eddy Simulation (LES) is an increasingly popular method of simulating turbulent flows in industrial applications. The result of applying a filter operation on the Navier-Stokes (NS) equations is a subgrid-scale (SGS) stress term which requires closure. Eddy viscosity models (EVM) are a simple type of model that is in general robust due to sufficient energy drainage from resolved scales. EVMs are based on the assumption of small (subgrid) scale isotropy given high Reynolds number (Re). Firstly, this would require the LES resolution to be sufficiently fine such that the assumption of SGS isotropy can be approximately valid (Rasam et al., 2011). Secondly, there is significant anisotropy within the viscous sublayer and buffer region of wall bounded flows (Montecchia et al., 2017). Moreover, EVMs assume by construction, an underlying alignment between the SGS stress tensor and strain rate tensor, whereas analysis on eigenvector alignment of said tensors suggest a relative alignment angle of $\sim 45°$.

Explicit algebraic subgrid-scale models (EASSMs), inspired by explicit algebraic Reynolds stress models (EARSMs), are based on the modelled transport equation of the SGS stress. Reduction from a partial differential equation to an explicit algebraic form is made possible by applying the weak equilibrium assumption, which assumes the quick relaxation of turbulence to equilibrium, resulting in the advection and diffusion of SGS anisotropy to be negligible. Studies have shown that EASSM provides more accurate description of individual SGS stresses. At the same time, EASSM outperforms EVM especially in coarse grid resolution cases (Inagaki & Kobayashi, 2023; Montecchia et al., 2017; Rasam et al., 2011), this highlights the importance of a model's ability to accurately capture SGS anisotropies. Pope (1975) argued on dimensional grounds that the Reynolds stress (RS) tensor can be expressed as an infinite polynomial in the strain rate and rotation rate tensors, $\langle S_{ij} \rangle$ and $\langle \Omega_{ij} \rangle$; analogously in LES, the filtered strain $\bar{S}_{ij}$ and rotation rate tensors $\bar{\Omega}_{ij}$. Making use of the Cayley-Hamilton theorem, the infinite polynomial can be simplified to a linear combination of ten basis tensors. In the formulation of EASSM by Marstorp et al. (2009), assumptions of two dimensional resolved velocity and local equilibrium between scalar production and dissipation

rates were made. Hickel et al. (2019) and Pouransari and Rasam (2023) have investigated the validity of the latter assumption and proposed alternative treatments, the present study adopts the method by Hickel et al. (2019). Regarding the two-dimensional resolved velocity assumption, Speziale and Gatski (1997) discarded this restriction and extended the formulation to three-dimensional fields in the Reynolds Averaged Navier-Stokes (RANS) context. In the present study, we perform the same extension for LES.

Returning to the modelled SGS stress transport equation, application of the weak equilibrium assumption eliminates the need for model closure for the turbulent transport terms $\mathcal{T}_{ijk}$. Terms that still require closure are the pressure strain Π_{ij} and dissipation ε_{ij}. For the former, the present study simply adopts the model proposed by Launder et al. (1975), which have been applied in the EASSM of Marstorp et al. (2009) and it's subsequent extension by Hickel et al. (2019) and Pouransari and Rasam (2023).

Hickel et al. (2019), Marstorp et al. (2009), and Pouransari and Rasam (2023) applied an isotropic dissipation model $\varepsilon_{ij} = \frac{2}{3}\varepsilon\delta_{ij}$. The analysis of Lee and Reynolds (1985) contradicted purely isotropic dissipation even in homogenous turbulent flows. The work of Gerolymos and Vallet (2016) analyzed the budget and anisotropy of ε_{ij} in wall turbulence, which also suggests the anisotropic nature of ε_{ij}. Rotta (1951) proposed, for homogeneous turbulence at low Re, a dissipation anisotropy model which is equal to the RS, ie $\varepsilon^a_{ij} = a_{ij}$. A combination of this model with the isotropic model via a blending function was proposed by Hanjalić and Launder (1976), however, it was noted by Perot and Natu (2004) that the model was outperformed by the model of Rotta (1951), without combining the isotropic term.

The model of Hallbäck et al. (1990) takes into account the rapid distortion theory, resulting in a model that is nonlinear in the RS anisotropy tensor a_{ij}. Due to said nonlinearity, it would not be possible to derive an explicit algebraic expression for a_{ij} and thus, the model is unsuitable to be applied within the EASSM context. From the transport equation of the dissipation tensor and assuming homogeneous turbulence in equilibrium, Speziale and Gatski (1997) proposed a model for ε^a_{ij} in algebraic form. The model of Perot and Natu (2004) only assumes a quasi-homogeneous dissipation tensor, it however includes the gradient of RS anisotropy tensor, which implies the use of non-local variables. This use of non-local variables contradicts the weak equilibrium assumption and the assumption that the SGS anisotropy is a function only, of the local strain and rotation rate tensor. As such, the model by Perot and Natu (2004) is also unsuitable to be applied in the explicit algebraic model context. Note that the work on dissipation tensor models discussed above are all developed in the RANS context. In general, models can be translated from RANS to LES by replacing the terms defined in RANS with its corresponding counterpart in LES, an example being the RS anisotropy and SGS stress anisotropy tensors.

Specifically in LES, for explicit algebraic models, little literature on dissipation models could be found and the isotropic model for dissipation is generally applied. Other SGS dissipation models for LES do exist, but some are specifically for the dissipation of passive scalar transport (Kasten et al., 2022) while others fall under the type of scale similar SGS models (Chumakov, 2005) which again, is not compatible to be applied in the explicit algebraic model context. Thus, the modelling aim of the present study is to propose a new explicit algebraic dissipation model that is compatible under an extended EASSM framework. (Such extension is to forgo the two-dimensional velocity field assumption and apply an updated treatment for the ratio between production and dissipation as proposed by Hickel et al. (2019))

A broad range of machine learning (ML) methods have been applied in turbulence modelling, Beck and Kurz (2021) and Duraisamy et al. (2019) offers a comprehensive overview of the current landscape in applying ML in turbulence modelling. Given the need for a model form that is compatible with EASSM, symbolic regression (SR) approaches are preferred over more conventional neural network (NN) approaches such as the work of Gamahara and Hattori (2017). Deterministic SR methods such as FFX, SINDy, SRNN, and SPARTA; see Brunton and Kutz (2019), Kim et al. (2021), McConaghy (2011), and Schmelzer et al. (2019), are a type of sparse regression method which requires a predefined library of basis functions, limiting their search space but offering higher computational efficiency. Gene expression programming (GEP), is an evolutionary algorithm which combines the benefits of genetic algorithms and genetic programming (Ferreira, 2006). No predefined library is required and therefore, allows for a broader search space. This, in conjunction with the method's stochastic nature entails higher computational cost. Seeing that GEP often struggles to regress the appropriate numerical constants, Waschkowski et al. (2022) introduced adaptive symbols in place of constants and optimizes the adaptive symbols with gradient based optimizers. Reissmann et al. (2024) applies constraints to candidate models in order to obtain models with the appropriate wall scaling. The

work also implements GEP in an a-Posteriori sense, coupling the GEP algorithm with actual LES simulations. A range of hybrid methods (eg Li et al. (2023) and Mundhenk et al. (2021)) also exists, however, constrained by the duration of this work, the more complex hybrid methods were excluded. Instead, a variation of utilizing gradient information and enforcing scaling constraints was implemented as inspired, by Hasslberger et al. (2025), Reissmann et al. (2024), and Waschkowski et al. (2022) respectively.

Some notable work of applying GEP in turbulence model discovery are those of Hasslberger et al. (2025), Reissmann et al. (2024), and Schoepplein et al. (2018). In these works, an explicit algebraic SGS stress model is directly inferred from training data. This benefits from not relying on the weak equilibrium assumption which is the key in deriving an explicit algebraic formulation. On the other hand, such a model could potentially lack the dynamics that is contained within the SGS transport equation; albeit simplified to algebraic form. So far as the author is aware, most applications of GEP in turbulence model discovery uses only a single flow case. Pursuing a model with high generality, we incorporate 2 flow cases; homogeneous isotropic turbulence (HIT) and turbulent channel flow (TCF) at 2 friction Reynolds number Re_τ, at 3 different filter sizes, as training data.

In light of the aforementioned information, this study aims to propose a SGS dissipation tensor model under the EASSM framework; with GEP as model discovery method, trained with data from mixed flow cases. First, the EASSM framework under which the model discovery will be conducted is explained in section II, followed by the extensions of it that have been made in this study. section III presents the methodology related to the GEP model discovery algorithm. Results of the model discovery process are shown in section V and section VI, for scalar dissipation rate and dissipation anisotropy tensor respectively. Lastly, the findings are summarized in section VII

II. Theoretical background

Assuming incompressible flow with constant density ρ and kinematic viscosity ν, without body forces, the filtered LES equations; with the filter operator denoted as $\bar{(\cdot)}$, read as follows

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j}, \tag{1a}$$

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0, \tag{1b}$$

where the SGS stress tensor τ_{ij} is defined as

$$\tau_{ij} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j. \tag{2}$$

The transport equation for τ_{ij} reads

$$\frac{\partial \tau_{ij}}{\partial t} + \bar{u}_k \frac{\partial \tau_{ij}}{\partial x_k} + \frac{\partial \mathcal{T}_{ijk}}{\partial x_k} = \Pi_{ij} + \mathcal{P}_{ij} - \varepsilon_{ij}. \tag{3}$$

The transport $\mathcal{T}_{ijk}$, pressure-strain Π_{ij}, production $\mathcal{P}_{ij}$ and dissipation ε_{ij} terms are:

$$\mathcal{T}_{ijk} = \overbrace{\overline{(u_i u_k u_j} - \bar{u}_i \bar{u}_k \bar{u}_j) - \bar{u}_i \tau_{jk} - \bar{u}_j \tau_{ki} - \bar{u}_k \tau_{ij}}^{\text{Turbulent transport}}$$
$$+ \overbrace{\frac{1}{\rho}\overline{p u_j}\delta_{ik} + \frac{1}{\rho}\overline{p u_i}\delta_{jk} - \frac{1}{\rho}\bar{p}\bar{u}_j\delta_{ik} - \frac{1}{\rho}\bar{p}\bar{u}_i\delta_{jk}}^{\text{Pressure transport}}$$
$$- \overbrace{\left(\nu\frac{\partial(\overline{u_i u_j})}{\partial x_k} - \nu\frac{\partial(\bar{u}_i\bar{u}_j)}{\partial x_k}\right)}^{\text{Viscous transport}}, \tag{4a}$$

$$\Pi_{ij} = \overline{\frac{p}{\rho}\frac{\partial u_i}{\partial x_j}} + \overline{\frac{p}{\rho}\frac{\partial u_j}{\partial x_i}} - \frac{\bar{p}}{\rho}\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\bar{p}}{\rho}\frac{\partial \bar{u}_j}{\partial x_i}, \tag{4b}$$

$$\mathcal{P}_{ij} = -\tau_{jk}\frac{\partial \bar{u}_i}{\partial x_k} - \tau_{ik}\frac{\partial \bar{u}_j}{\partial x_k}, \tag{4c}$$

$$\varepsilon_{ij} = 2\nu\left(\overline{\frac{\partial u_i}{\partial x_k}\frac{\partial u_j}{\partial x_k}}\right) - 2\nu\left(\frac{\partial \bar{u}_i}{\partial x_k}\frac{\partial \bar{u}_j}{\partial x_k}\right). \tag{4d}$$

Following the application of the weak equilibrium assumption by Marstorp et al. (2009), we obtain an implicit relation in τ_{ij}

$$\frac{\tau_{ij}}{k_{sgs}}(\mathcal{P} - \varepsilon) = \mathcal{P}_{ij} + \Pi_{ij} - \varepsilon_{ij}, \tag{5}$$

with the definition of SGS kinetic energy being

$$k_{sgs} = \frac{1}{2}\tau_{ii}.$$

$\mathcal{P}$ and ε are the scalar production and dissipation rates which are defined as the half trace of their respective tensors, written as:

$$\mathcal{P} = \frac{1}{2}\mathcal{P}_{ii}, \qquad \varepsilon = \frac{1}{2}\varepsilon_{ii}.$$

For the closure of the pressure-strain term Π_{ij}, we use the LRR-QI model by Launder et al. (1975), which reads

$$\begin{aligned}\Pi_{ij} =& -C_R\varepsilon a_{ij} + C_1 k_{sgs}\bar{S}_{ij} \\ &+ \frac{3}{11}(2+3C_2)k_{sgs}\left(\bar{S}_{ik}a_{kj} + a_{ik}\bar{S}_{kj} - \frac{2}{3}\bar{S}_{kl}a_{lk}\delta_{ij}\right) \\ &+ \frac{1}{11}(10-7C_2)k_{sgs}\left(\bar{\Omega}_{ik}a_{kj} - a_{ik}\bar{\Omega}_{kj}\right),\end{aligned} \tag{6}$$

with $\bar{S}_{ij}$, $\bar{\Omega}_{ij}$ and a_{ij} being the filtered strain rate, filtered rotation rate and SGS stress anisotropy tensor.

$$\bar{S}_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) \tag{7a}$$

$$\bar{\Omega}_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i}\right) \tag{7b}$$

$$a_{ij} = \frac{\tau_{ij}}{k_{sgs}} - \frac{2}{3}\delta_{ij} \tag{7c}$$

Under the assumption of nearly homogeneous flow at high Re and based on dimensional analysis, Pope (1975) suggested that a_{ij} is only a function of the non-dimensional strain and rotation rate tensors. This can be expressed in its most general form, as an infinite tensor polynomial, with the coefficients being a functions of the invariants of $\bar{S}_{ij}$ and $\bar{\Omega}_{ij}$. Making use of the Cayley-Hamilton theorem, the infinite tensor polynomial can be expressed as a linear combination of 10 independent basis tensors $\mathbf{T}^{(n)}$, the coefficients of the basis tensors are functions of 5 invariants (Pope, 1975). The basis tensors and invariants are shown in Table 1 and Table 2, the superscript $(\cdot)^*$ indicates non-dimensionalized quantities; $\bar{S}^* = \bar{S}\frac{k_{sgs}}{\varepsilon}$, $\bar{\Omega}^* = \bar{\Omega}\frac{k_{sgs}}{\varepsilon}$.

$\mathbf{T}^{(1)} = \bar{S}^*$	$\mathbf{T}^{(2)} = \bar{S}^*\bar{\Omega}^* - \bar{\Omega}^*\bar{S}^*$
$\mathbf{T}^{(3)} = \bar{S}^{*2} - \frac{1}{3}II_S\mathbf{I}$	$\mathbf{T}^{(4)} = \bar{\Omega}^{*2} - \frac{1}{3}II_\Omega\mathbf{I}$
$\mathbf{T}^{(5)} = \bar{\Omega}^*\bar{S}^{*2} - \bar{S}^{*2}\bar{\Omega}^*$	$\mathbf{T}^{(6)} = \bar{\Omega}^{*2}\bar{S}^* + \bar{S}^*\bar{\Omega}^{*2} - \frac{2}{3}IV\mathbf{I}$
$\mathbf{T}^{(7)} = \bar{\Omega}^*\bar{S}^*\bar{\Omega}^{*2} - \bar{\Omega}^{*2}\bar{S}^*\bar{\Omega}^*$	$\mathbf{T}^{(8)} = \bar{S}^*\bar{\Omega}^*\bar{S}^{*2} - \bar{S}^{*2}\bar{\Omega}^*\bar{S}^*$
$\mathbf{T}^{(9)} = \bar{S}^{*2}\bar{\Omega}^{*2} + \bar{\Omega}^{*2}\bar{S}^{*2} - \frac{2}{3}V\mathbf{I}$	$\mathbf{T}^{(10)} = \bar{\Omega}^*\bar{S}^{*2}\bar{\Omega}^{*2} - \bar{\Omega}^{*2}\bar{S}^{*2}\bar{\Omega}^*$

Table 1. Tensorial bases for normalized SGS anisotropy tensor a_{ij}

$II_S = tr(\bar{S}^{*2})$	$II_\Omega = tr(\bar{\Omega}^{*2})$
$III = tr(\bar{S*}^3)$	$IV = tr(\bar{S}^*\bar{\Omega}^{*2})$
$V = tr(\bar{S}^{*2}\bar{\Omega}^{*2})$	

Table 2. Independent invariants of $\bar{S}^*_{ij}, \bar{\Omega}^*_{ij}$

With this, an ansatz for the SGS stress anisotropy tensor can be introduced as

$$a_{ij} = \sum_{n=1}^{10} \beta^{(n)} \mathbf{T}^{(n)}. \tag{8}$$

The key to obtaining an EASSM is to find expressions for $\beta^{(n)}$, this is only possible if both expressions on the LHS and RHS of Equation 5 are linear in a_{ij}. Keeping this in mind, we proceed to substitute Equation 8 and Equation 6 into Equation 5, resulting in the following

$$\begin{aligned}\left(\sum_n \beta^{(n)}\mathbf{T}^{(n)}\right)\left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) + \frac{2}{3}\delta_{ij}\left(\frac{\mathcal{P}}{\varepsilon} - 1\right) = \bar{S}^*_{ij}\left(C_1 - \frac{4}{3}\right) + \\ + \left[\bar{S}^*_{ik}\left(\sum_n \beta^{(n)}\mathbf{T}^{(n)}\right) + \left(\sum_n \beta^{(n)}\mathbf{T}^{(n)}\right)\bar{S}^*_{kj}\right]\left(\frac{9C_2 - 5}{11}\right) \\ + \left[\left(\sum_n \beta^{(n)}\mathbf{T}^{(n)}\right)\bar{\Omega}^*_{kj} - \bar{\Omega}^*_{ik}\left(\sum_n \beta^{(n)}\mathbf{T}^{(n)}\right)\right]\left(\frac{1 + 7C_2}{11}\right) \\ + \frac{\mathcal{P}}{\varepsilon}\delta_{ij}\left(\frac{18C_2 + 12}{33}\right) - \frac{\varepsilon_{ij}}{\varepsilon}\end{aligned} \tag{9}$$

The terms $\bar{\mathbf{S}}^*\mathbf{T}^{(n)} + \mathbf{T}^{(n)}\bar{\mathbf{S}}^*$ and $\mathbf{T}^{(n)}\bar{\mathbf{\Omega}}^* - \bar{\mathbf{\Omega}}^*\mathbf{T}^{(n)}$ can be expressed only in terms of $\mathbf{T}^{(n)}$ by repeated application of the Cayley-Hamilton theorem. Such derivation had been done by Gatski and Speziale (1993), we simply adopt the results that has been found, which reads as:

$$\bar{\mathbf{S}}^*\mathbf{T}^{(n)} + \mathbf{T}^{(n)}\bar{\mathbf{S}}^* = \sum_\xi H_{n,\xi}\mathbf{T}^{(\xi)} + \frac{2}{3}tr(\mathbf{T}^{(n)}\mathbf{T}^{(1)})\delta_{ij}, \tag{10a}$$

$$\bar{\mathbf{\Omega}}^*\mathbf{T}^{(n)} - \mathbf{T}^{(n)}\bar{\mathbf{\Omega}}^* = \sum_\xi J_{n,\xi}\mathbf{T}^{(\xi)}, \tag{10b}$$

where H and J are presented in Appendix A. Substitution of the results lead to

$$\begin{aligned}\left(\sum_n \beta^{(n)}\mathbf{T}^{(n)}\right)\left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) + \frac{2}{3}\delta_{ij}\left(\frac{\mathcal{P}}{\varepsilon} - 1\right) = \mathbf{T}^{(1)}\left(C_1 - \frac{4}{3}\right) + \\ + \left(\sum_n \sum_\xi H_{n,\xi}\beta^{(n)}\mathbf{T}^{(\xi)} - \frac{2}{3}\frac{\mathcal{P}}{\varepsilon}\delta_{ij}\right)\left(\frac{9C_2 - 5}{11}\right) \\ + \left(\sum_n \sum_\xi J_{n,\xi}\beta^{(n)}\mathbf{T}^{(\xi)}\right)\left(\frac{1 + 7C_2}{11}\right) \\ + \frac{\mathcal{P}}{\varepsilon}\delta_{ij}\left(\frac{18C_2 + 12}{33}\right) - \frac{\varepsilon_{ij}}{\varepsilon}.\end{aligned} \tag{11}$$

We pause here to reflect that in the original EASSM of Marstorp et al. (2009), local equilibrium between production and dissipation rate ($\mathcal{P} = \varepsilon$), and locally two-dimensional flow were assumed. The latter assumption leads to only $\mathbf{T}^{(1,2,3)}$, II_S and II_Ω being retained in the formulation. The standard approach for modelling ε_{ij} is to assume it's isotropy, the model reads as:

$$\varepsilon_{ij} = \frac{2}{3}\varepsilon\delta_{ij}, \tag{12}$$

this standard model is widely applied, for example, by Launder et al. (1975), Marstorp et al. (2009), and Wallin and Johansson (2000). In order to obtain an explicit model, k_{sgs} and ε remains to be determined. Marstorp et al. (2009) proposed dynamic and non-dynamic procedures for determining k_{sgs} and the time scale $\tau^* = \frac{k_{sgs}}{\varepsilon}$, given as explicit expressions, ε itself is not modelled explicitly. Due to $\mathcal{P}$ being a function of a_{ij}, the expression is nonlinear, assuming $\mathcal{P} = \varepsilon$ removes such nonlinearity and leads to the possibility of expressing said terms explicitly (Marstorp et al., 2009).

Gatski and Speziale (1993) forgoes the two-dimensional flow assumption and split the modelling of ε_{ij}into an isotropic and deviatoric part

$$\varepsilon_{ij} = \frac{2}{3}\varepsilon\delta_{ij} + \varepsilon^d_{ij}, \tag{13}$$

where the latter is absorbed into pressure-strain term. ε and k_{sgs} are then determined by solving their respective modelled transport equation as given by the standard k-ε model from RANS.
This summarizes the framework on EASSM, laid out by Gatski and Speziale (1993) and Marstorp et al. (2009), from here on, we depart from the ground work that has been established.

III. EASSM extension

Recalling the goal of discovering a new dissipation rate tensor model, we define the dissipation rate anisotropy tensor ε^a_{ij} and introduce the ansatz

$$\varepsilon^a_{ij} = \frac{\varepsilon_{ij}}{\varepsilon} - \frac{2}{3}\delta_{ij}, \tag{14a}$$

$$\varepsilon^a_{ij} = \sum_{n=1}^{10} \gamma^{(n)}\mathbf{T}^{(n)}. \tag{14b}$$

The scalar functions $\gamma^{(n)}$ follow from $\beta^{(n)}$ to be functions of the 5 invariants in Table 2.
Substitution of Equation 14 into Equation 11 and simplifying yields

$$\begin{aligned}\left(\sum_n \beta^{(n)}\mathbf{T}^{(n)}\right)\left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) = \mathbf{T}^{(1)}\left(C_1 - \frac{4}{3}\right) + \\ + \left(\sum_n\sum_\xi H_{n,\xi}\beta^{(n)}\mathbf{T}^{(\xi)}\right)\left(\frac{9C_2 - 5}{11}\right) \\ + \left(\sum_n\sum_\xi J_{n,\xi}\beta^{(n)}\mathbf{T}^{(\xi)}\right)\left(\frac{1 + 7C_2}{11}\right) \\ - \sum_n \gamma^{(n)}\mathbf{T}^{(n)}.\end{aligned} \tag{15}$$

The equation above is linear in all basis tensors , which allows an expression to be found for all $\beta^{(n)}$ by equating the coefficients of $\mathbf{T}^{(n)}$. The fully expanded system of equations can be found in Appendix A, Equation 16 shows the same equation, simplified in matrix form,

$$\boldsymbol{\beta} = \frac{1}{\theta_3}(\mathbf{Q}\boldsymbol{\beta} - \boldsymbol{\gamma} + \boldsymbol{\varphi}), \tag{16}$$

where

$$\boldsymbol{\gamma} = \begin{bmatrix}\gamma^{(1)} & \gamma^{(2)} & \gamma^{(3)} & \gamma^{(4)} & \gamma^{(5)} & \gamma^{(6)} & \gamma^{(7)} & \gamma^{(8)} & \gamma^{(9)} & \gamma^{(10)}\end{bmatrix}^T, \tag{17}$$

$$\boldsymbol{\beta} = \begin{bmatrix}\beta^{(1)} & \beta^{(2)} & \beta^{(3)} & \beta^{(4)} & \beta^{(5)} & \beta^{(6)} & \beta^{(7)} & \beta^{(8)} & \beta^{(9)} & \beta^{(10)}\end{bmatrix}^T, \tag{18}$$

$$\boldsymbol{\varphi} = \begin{bmatrix}C_1 - \frac{4}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T, \tag{19}$$

$$
\mathbf{Q} = \begin{bmatrix}
0 & -II_\Omega\theta_2 & \frac{1}{3}II_S\theta_1 & -\frac{2}{3}II_\Omega\theta_1 & 0 & \frac{2}{3}IV\theta_1 & II_\Omega^2\theta_2 & (2V - II_SII_\Omega)\theta_2 & -\frac{1}{3}V\theta_1 & 0 \\
\theta_2 & 0 & 0 & 0 & -\frac{1}{2}II_S\theta_1 & \frac{1}{2}II_\Omega\theta_2 & -IV\theta_1 & \frac{1}{3}III\theta_1 & 0 & \left(-\frac{1}{3}V - \frac{1}{6}II_SII_\Omega\right)\theta_1 \\
2\theta_1 & 0 & 0 & 0 & II_\Omega\theta_2 & 2II_\Omega\theta_1 & 0 & -2IV\theta_2 & IV\theta_1 & II_\Omega^2\theta_2 \\
0 & 0 & 0 & 0 & 2II_S\theta_2 & II_S\theta_1 & -2IV\theta_2 & 0 & \frac{1}{3}III\theta_1 & 2(II_SII_\Omega - V)\theta_2 \\
0 & -\theta_1 & -\theta_2 & 0 & 0 & 0 & II_\Omega\theta_1 & 0 & -\frac{1}{2}II_\Omega\theta_2 & \frac{2}{3}IV\theta_1 \\
0 & 3\theta_2 & 0 & \theta_1 & 0 & 0 & -2II_\Omega\theta_2 & II_S\theta_2 & \frac{1}{2}II_S\theta_1 & 0 \\
0 & 0 & 0 & 0 & 0 & -\theta_2 & 0 & 0 & 0 & -\frac{1}{3}II_S\theta_1 \\
0 & 0 & 0 & 0 & \theta_1 & 0 & 0 & 0 & 0 & \frac{1}{3}II_\Omega\theta_1 \\
0 & 0 & 0 & 0 & -3\theta_2 & -\theta_1 & 0 & 0 & 0 & -2II_\Omega\theta_2 \\
0 & 0 & 0 & 0 & 0 & 0 & -2\theta_1 & 0 & -\theta_2 & 0
\end{bmatrix}, \tag{20}
$$

$$
\theta_1 = \frac{9C_2 - 5}{11}, \quad \theta_2 = \frac{1 + 7C_2}{11}, \quad \theta_3 = \frac{\mathcal{P}}{\varepsilon} - 1 + C_R\ . \tag{21}
$$

It is possible to solve for an explicit expression for $\boldsymbol{\beta}$ from Equation 16, although without any simplifications made, this would result in an overly complicated set of equations which is intractable to be used within the GEP optimization. To obtain simpler expressions for $\boldsymbol{\beta}$, $C_2 = 5/9$ was chosen, leading to a much simplified version of Equation 20 since all θ_1 terms vanish. The value of $C_2 = 5/9$ has been reported by Gnanasundaram (2018), Marstorp et al. (2009), and Wallin and Johansson (2000), others works apply a similar value of $C_2 = 0.4$ (Hickel et al., 2019; Launder et al., 1975). Explicit expressions of $\boldsymbol{\beta}$ can then be solved symbolically with Maple (Maplesoft, 2025), the result of this is shown in Appendix B.

The remaining unknowns to be treated are $\frac{\mathcal{P}}{\varepsilon}$, $\gamma^{(n)}$, ε and k_{sgs}. The scalar production has an alternate expression $\mathcal{P} = -k_{sgs}a_{ik}\bar{S}_{ik}$, substitution into Equation 5 would lead to a non linearity a_{ij}. By assuming $\frac{\mathcal{P}}{\varepsilon} = 1$, this was circumvented by Marstorp et al. (2009). The inadequacies of such assumption was pointed out by Hickel et al. (2019), the authors proposed to treat the ratio as an unknown, taking it's value from a previous timestep during a simulation. We will adopt the method in this study. For $\gamma^{(n)}$ and ε, the former is the subject of optimization by the model discovery algorithm. It was decided that for better consistency, the scalar dissipation should be included in the current framework as a separate model discovery problem, denoted by $\sigma := \varepsilon$.

Lastly, as mentioned in the previous section, an explicit form for k_{sgs} was proposed by Marstorp et al. (2009), since the assumption of $\mathcal{P} = \varepsilon$ is no longer made presently, k_{sgs} is instead determined by solving it's transport equation. We follow the modelled transport equation by Yoshizawa and Horiuti (1985) and substitute σ in place of ε, the transport equation thus reads:

$$
\frac{\partial k_{sgs}}{\partial t} + \bar{u}_i\frac{\partial k_{sgs}}{\partial x_i} - \frac{\partial}{\partial x_i}\left[(\nu_k + \nu)\frac{\partial k_{sgs}}{\partial x_i}\right] = -\tau_{ij}\frac{\partial \bar{u}_j}{\partial x_i} - \sigma \tag{22}
$$

This concludes the model framework , there are however some practicalities to be addressed as follows.

First, recall that $\gamma^{(n)}$ are scalar functions of 5 invariants non-dimensionalized by $\frac{k_{sgs}}{\varepsilon}$. This is not possible when discovering the scalar dissipation model since ε itself is unknown. Instead, following Hasslberger et al. (2025), the invariants are non-dimensionalized with $||S||= \sqrt{S_{mn}S_{mn}}$. Quantities non-dimensionalized with $||S||$ are denoted by $\tilde{(\cdot)}$.

Secondly, in the context of LES, the filter width Δ is an important parameter as the definition of τ_{ij} is such that it describes the effect of subgrid scale eddies on grid scale motion, therefore, scale separation directly influences τ_{ij}. As a result, a non-dimensionalized filter width $\Delta^* = \Delta\frac{||S||}{\sqrt{k_{sgs}}}$ is included as a function variable of $\gamma^{(n)}$.

Thirdly, consider the definition of ε, the material property ν has not been included, a non-dimensionalized viscosity $\nu^* = \nu\frac{||S||}{k_{sgs}}$ is further added as a variable to σ. With all function variables non-dimensionalized, this ensures the

dimensional consistency of the output model by GEP. To re-dimensionalize the scalar dissipation model σ, a multiplier $\nu||S||^2$ is chosen. Equation 23 shows the set of function variables for both $\gamma^{(n)}$ and σ, including re-dimensionalization for ε. Lastly, Figure 1 summarizes the overall theoretical framework that has been discussed.

$$\varepsilon_{ij}^{a} = \sum_{n} \gamma^{(n)}(II_S, II_\Omega, III, IV, V, \Delta^*)\mathbf{T}_{ij}^{(n)} \tag{23a}$$

$$\varepsilon = \sigma(\widetilde{II}_S, \widetilde{II}_\Omega, \widetilde{III}, \widetilde{IV}, \widetilde{V}, \Delta^*, \nu^*)\nu||S||^2 \tag{23b}$$

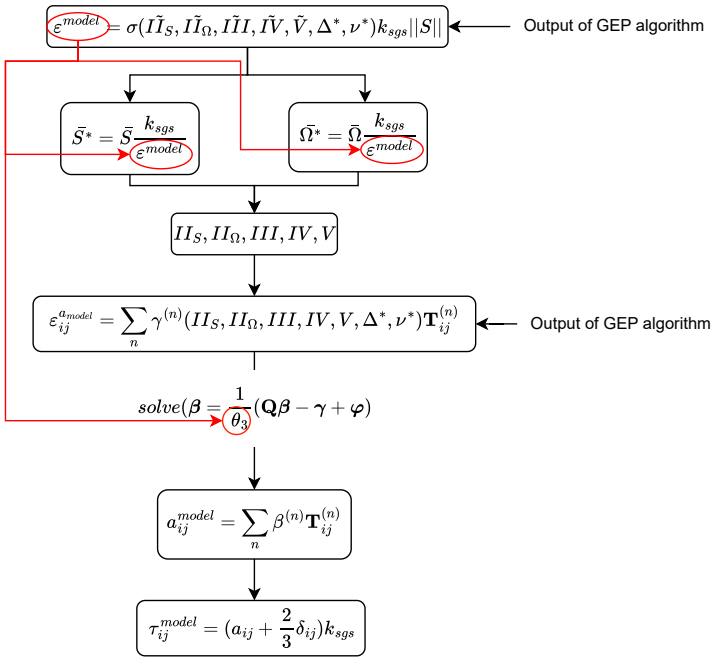


Figure 1. Flowchart of theoretical framework, red lines highlights where scalar dissipation rate is used in the final SGS stress model

IV. Methodology

A. GEP

For a fundamental discussion on the GEP method, readers are referred to the book by Ferreira (2006). To explain briefly, a gene can be encoded as a linear string consisting of terminals and mathematical operators. The string is separated into a head and tail section, the latter may only contain terminals while the former may contain both operators and terminals. The head length h is a user input which determines the possible length of an expression, the tail length can be calculated as $t = h(n_{max} - 1) - 1$, with n being the maximum arity of the functional set. Using the Karva language (Ferreira, 2006), the gene can be translated into an expression tree and mathematical expression. The generated mathematical expressions are candidate solutions to the optimization problem. An example of a single genic chromosome is shown in Figure 2. The set of terminals, operators, and head length used in the scalar and tensorial model discovery cases are tabulated in Table 3.

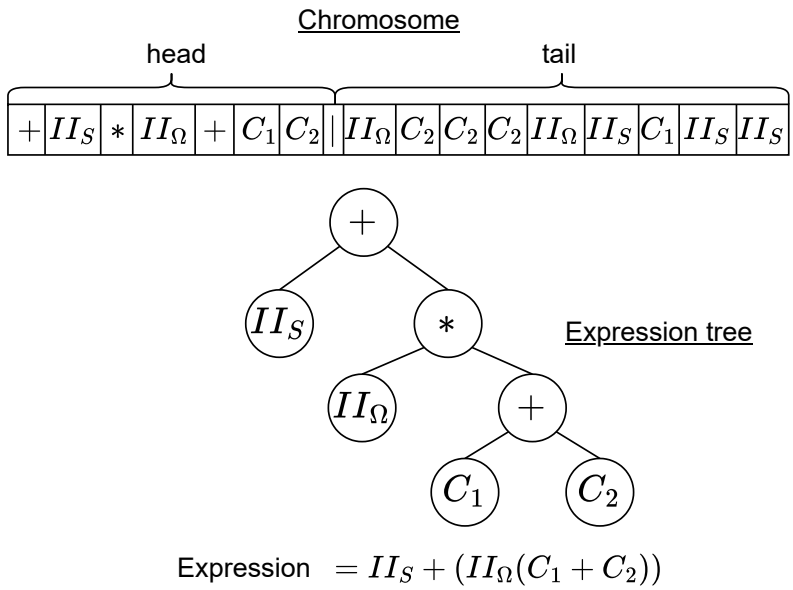


Figure 2. Example of single genic chromosome and corresponding expression tree and mathematical expression

For simpler problem such as in the context of scalar dissipation rate model discovery, the chromosome is made up of a single gene. For complicated problems, chromosomes with multiple genes can be constructed, depending on the application, the genes can either be linked through a linking function to form more complicated expression, or treated separately. For the dissipation anisotropy tensor case, 10 scalar functions,$\gamma^{(n)}$, are required as coefficient functions to the basis tensors, therefore chromosomes with 10 genes, each encoding for one scalar function, are used. No linking function is applied since each gene represents an independent coefficient function to their corresponding basis tensor. There is a subtle difference in using a multi-genic chromosome as opposed to multiple single-genic populations. For multi-genic populations without linking functions, there is still interaction between genes that are within a single chromosome with applying genetic operators. Separating the genes and creating multiple population instead will remove such interactions.

The GEP algorithm starts by generating a random population of individuals of size n_{pop}, subsequently, the fitness of each individual is evaluated. Based on each chromosome's fitness, they are selected to produce offsprings, other genetic operators such as mutation and crossover are then applied. Both tournament and roulette selection had been applied in this study, for a detailed description of the selection schemes, readers are referred to Ferreira (2006). This is repeated until a fixed number of generations, n_{gen}, have been reached. GEP here is applied in the a priori sense, meaning that the numerical value of GEP output is directly compared against DNS data instead of the output being propagated in a CFD solver, and using the output of CFD in the evaluation of the fitness function. A flowchart illustrating the steps within the algorithm is shown in Figure 1.

A number of hyperparameters controls the behaviour of the algorithm, they are tabulated in Table 3. As GEP is an inherently stochastic algorithm, there is no general rule for setting the hyperparameters, they are found by trial and error, loosely following Ferreira (2006) and Reissmann et al. (2021). In this paper, GEP is implemented in python with the package geppy by Gao et al. (2022) which is build on top of DEAP by Fortin et al. (2012), along with some custom additions that are discussed in the following sections.

Table 3. GEP hyperparameters for scalar and tensor model discovery.

Parameter	ε	ε^a_{ij}
number of genes	1	10
head	25	25
Generations	175	500
Population size	350	350
Point mutation rate	0.25	0.2
Inversion rate	0.1	0.1
IS transposition rate	0.1	0.05
RIS transposition rate	0.1	0.05
Gene transposition rate	-	0.05
1-point crossover rate	0.3	0.3
2-point crossover rate	0.2	0.2
Gene crossover rate	-	0.05
Mathematical operator	+, -, ×, ÷	+, -, ×, ÷
terminals	$\widetilde{II}_S$,$\widetilde{II}_\Omega$,$\widetilde{III}$, $\widetilde{IV}$,$\widetilde{V}$,Δ^*, ν^*	II_S,II_Ω,III, IV,V,Δ^*
Numerical constants	C_1, C_2, C_3, C_4, C_5	* (Adaptive symbol)

B. Training data

As have been mentioned, the current method leverages different flow cases as training data to increase generality of the output model. For a first step towards this aim, HIT and TCF at $Re_\tau \approx$ 700,1500 were considered. These cases are hereafter simply referred to as HIT, TCF700 and TCF1500. HIT and TCF DNS data were both sourced internally within the Aerodynamics group of TU Delft. The former being from the work by Pestana and Hickel (2019). A.Chakraborty, also from the Aerodynamics group provided the TCF datasets. The channel dimensions are 3δ, 2δ and 1.5δ in the x (streamwise), y (wall-normal) and z (spanwise) directions respectively, and with δ being the channel half height. Periodic boundary conditions were applied in the x and z directions, in the x direction, a pressure gradient is being adapted to maintain a bulk velocity of unity. The kinematic viscosity ν for the TCF700 and TCF1500 are 7.6923e-5 and 3.1746e-5 respectively.

Filtering operation was implemented as a volumetric average using a non-dimensional filter width $\tilde{\Delta}$, which denotes the number of DNS grid cells spanned in each spatial direction. Each of the flow cases were filtered at $\tilde{\Delta} = 9, 17, 25$, resulting in 9 sets of training data in total. We use the suffix $\_f\langle n\rangle$ to denote a data case filtered with $\tilde{\Delta} = n$. Training data was sampled from each filtered case with a uniform distribution on it's computational grid, thus, for TCF cases, there is a bias in training data towards the channel walls due to grid refinement. The combination of cases examined, along with the amount of sampled data given in percentage of the respective cases, is shown in Table 5. The amount of data used for sets S1 and S2 is such that the total amount of training data remains constant for all sets. For each training configuration, 5 random initializations had been conducted, the small number of initializations was due to time limitations of this study. Such a sample size may only provide some preliminary basis on which to discuss the effect of the training configurations.

In selected optimization runs for the TCF cases, an additional sampling bias based on a cosine distribution along the y-axis is applied, in attempt to generate models with improved wall behaviour. The sampling bias distribution is shown in Figure 3, with $p_{relative}$ being the relative probability of points to be sampled at the wall versus channel centre. The probability distribution here is normalized to ensure the total probability sums to unity across the entire channel.

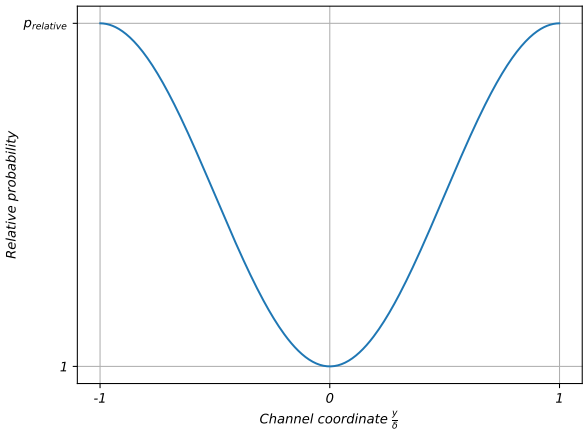


Figure 3. Sampling bias for TCF cases

Training data configurations for the tensor model discovery is shown in Table 4, relatively few data configurations are used to time constraints. Besides, in view of some unsatisfactory results observed in the initial tests, efforts had been diverted to modify the training configuration instead of data configuration.

Table 4. Training data configuration for tensor dissipation case.(% applies to all cases for all filter sizes indicated)

Data configuration	Cases	filter size $\tilde{\Delta}$	Amount of data (%)	$p_{relative}$
T1	HIT, TCF700	9,17,25	1	1
T2	HIT, TCF700, TCF1500	9,17,25	0.5	1

C. Objective function

A variety of objective functions have been employed in different studies. The mean squared error (MSE) with L_2 regularization and a model complexity parameter was used in Waschkowski et al. (2022), the multi-objective optimization required a pareto-front selection. Other examples include mean absolute error (MAE), relative mean absolute error (RMAE) and L_2 norm used by Lav et al. (2023), Schoepplein et al. (2018), and Weatheritt and Sandberg (2017) respectively.

For the scalar dissipation case, the RMAE is employed and defined as follows:

$$Loss(\sigma) = \frac{w_e \epsilon_\sigma}{q} \sum_q \frac{|\sigma_q - \varepsilon_q|}{|\varepsilon_q|} - \frac{w_\rho}{m} \sum_m \rho_m + \epsilon_{neg} \tag{24}$$

The objective is denoted as a loss function since the GEP algorithm is set to minimize the error. The subscripts q indicate the number of data points sample from all training cases. ϵ_σ is a wall penalty term defined in Equation 33,this is treated in subsection IV.E. Moreover, to avoid the output model producing negative scalar dissipation rates, yet another penalty $\epsilon_{neg} = 99 \cdot n_{negativepoints}$ is added to Equation 24. During some preliminary testing of this study, the correlation coefficient, defined as

$$\rho = \frac{\sum_q (\sigma_q - \bar{\sigma})(\varepsilon_q - \bar{\varepsilon})}{\sqrt{\sum_q (\sigma_q - \bar{\sigma})^2} \sqrt{\sum_q (\varepsilon_q - \bar{\varepsilon})^2}}, \quad \bar{\sigma} = \frac{1}{q} \sum_q \sigma_q, \quad \bar{\varepsilon} = \frac{1}{q} \sum_q \varepsilon_q \tag{25}$$

was included into the fitness function, which seemed to produce improved results occasionally. Hence, this was included in the training configuration S4 to investigate it's effect. Note that $\bar{(\cdot)}$ in the above equation denotes the mean.

Implementation of ρ is controlled via the weights w_e and w_ρ, in this study, they take value of either 1 or 0 as tabulated in Table 5.

During some testing of the tensor model discovery, a change in selection scheme from tournament to roulette selection was made, in order to confirm that the selection scheme was not a cause of some unsatisfactory results. Roulette selection constrains the objective function to be strictly positive and formulated as a maximization problem. As such, both a fitness and a loss function can been defined as

$$Loss(\varepsilon^a_{ij_{model}}) = \frac{\epsilon_\gamma}{6q}\sum_{i=1}^{3}\sum_{j=i}^{3}\left(\sum_q \frac{w_\varepsilon\left|\sum_n \gamma_q^{(n)}\mathbf{T}^{(n)}_{ij,q} - \varepsilon^a_{ij,q}\right|}{|\varepsilon^a_{ij,q}|} + \frac{w_a\left|\sum_n \beta_q^{(n)}\mathbf{T}^{(n)}_{ij,q} - a_{ij,q}\right|}{|a_{ij,q}|}\right), \tag{26a}$$

$$fit(\varepsilon^a_{ij_{model}}) = \frac{1}{1 + Loss(\varepsilon^a_{ij_{model}})}\,. \tag{26b}$$

The term ϵ_γ is, similar to the scalar case, a wall penalty term defined in Equation 34. As opposed to a Frobenius norm formulation, which sums all tensor components, here only 6 components are summed since the tensor is symmetric. w_a, w_ε are weightings given to the respective RMAE in ε^a_{ij} and a_{ij}, their values in each training configuration is given in Table 4. The effect of including ρ was not included in the tensor case due to time limitations.

D. Numerical constants

Physical models often contain numerical constants, the original GEP method handles this by generating an array of random numerical constants (RNC) that can be part of the individuals. These constants can also be combined to generate additional constants. It is a well known problem that evolutionary algorithms struggle in numerical constant convergence, Waschkowski et al. (2022) implemented adaptive symbols into genetic expressions which are then optimized with a gradient based optimizer. Regularization to the optimization and a model complexity objective was also added in the study of Waschkowski et al. (2022).

Here, considering the ease of implementation under the pre-existing geppy package, we take inspiration from Waschkowski et al. (2022) to apply a BFGS optimizer to a fixed number of placeholder symbols C_m. The routine for numerical constant optimization implemented in this study is illustrated in Figure 4. Given a population, the algorithm only activates constant optimization every 15 generations and, if a placeholder symbol is within an individual. The 15 generation interval was implemented due to the extensive computational cost of optimizing at every generation. The initial guess input to the BFGS optimizer is taken from constant values corresponding to the fittest individual in previous evaluations. Depending on whether the output fitness is lower than the best fitness so far, the saved constant and fitness values will be updated or not. Regardless, the output fitness will be attached to the individual to be used in subsequent genetic operations.

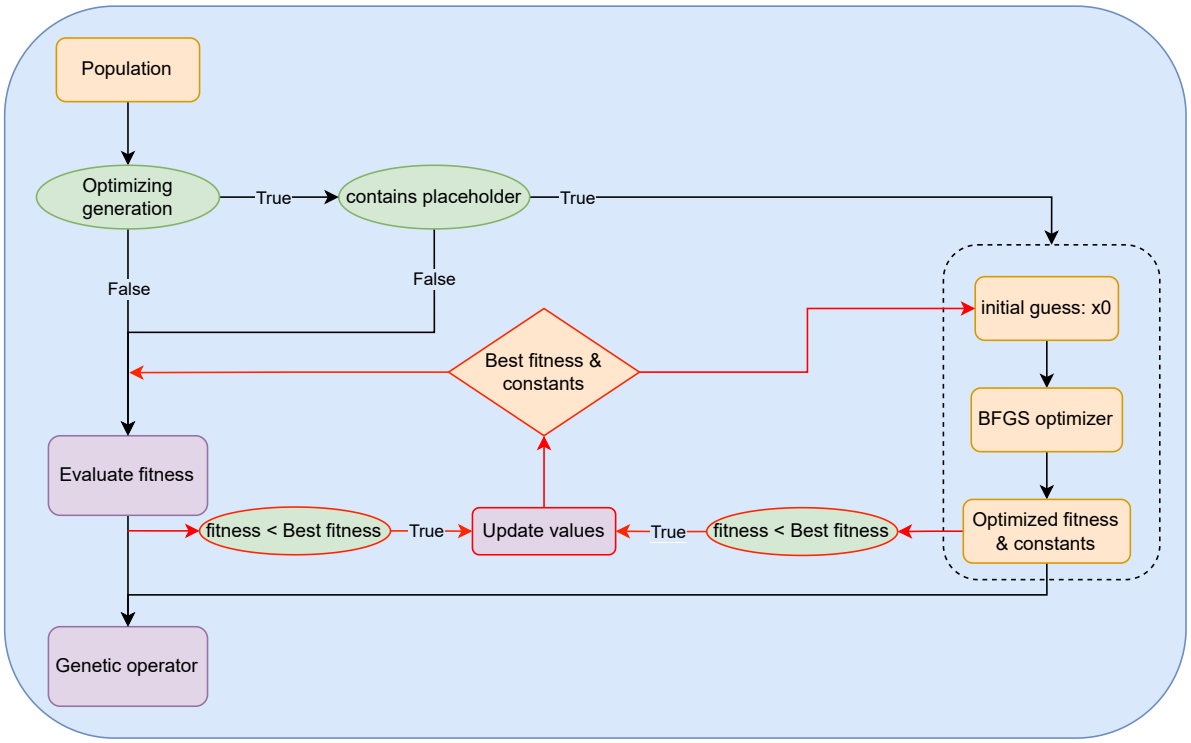


Figure 4. Flowchart of placeholder symbol numerical constant optimization

The difference between the adaptive and placeholder symbols implementation is twofold. First, a fixed maximum number of placeholder symbols, specified by m, can exist in each individual, whereas an arbitrary number of adaptive symbols can be present in an individual (within the limits of the gene length). Secondly, the values of placeholder symbols are not attached to each individual, during non-optimizing generations and for the initial guess of each BFGS optimization, the same set of constants from the best individual is used. The is further illustrated in Appendix C, which shows that during non-optimizing generations, all individuals containing a symbolic constant in their chromosome takes their respective constant values from the list "best_constants". In the case of a proper implementation of adaptive symbols, the lines and blocks highlighted in red in Figure 4 need not be present.

At a later stage of this study, a simple method to implement adaptive symbols within geppy was worked out and also implemented. This training configuration is tabulated in Table 5 as S6. All tensor model discover with constant optimization applied also uses this updated implementation.

E. Wall-scaling penalty

Figure 5 shows the anisotropies of the SGS stress and dissipation rate to be highest in vicinity of solid no-slip walls. This highlights the importance of a model's wall scaling behaviour. To encourage the discovery of models with the appropriate asymptotic wall behaviours, an additional penalty is applied to model expressions that does not possess the proper wall scaling. The required wall scaling is derived as follows. First writing the Taylor expansion for the velocities in wall distance y, requiring velocities to be 0 at the wall and applying the incompressibility condition leads to

$$\bar{u} = \bar{a}y + \dots, \quad \bar{v} = \bar{b}y^2 + \dots, \quad \bar{w} = \bar{c}y + \dots . \tag{27}$$

Substitution into the definition of τ_{ij} yield the scalings

$$\tau_{ij} \propto \begin{bmatrix} y^2 & y^3 & y^2 \\ y^3 & y^4 & y^3 \\ y^2 & y^3 & y^2 \end{bmatrix}, \qquad k_{sgs} \propto y^2, \qquad a_{ij} \propto \begin{bmatrix} y^0 & y^1 & y^0 \\ y^1 & y^0 & y^1 \\ y^0 & y^1 & y^0 \end{bmatrix} . \tag{28}$$

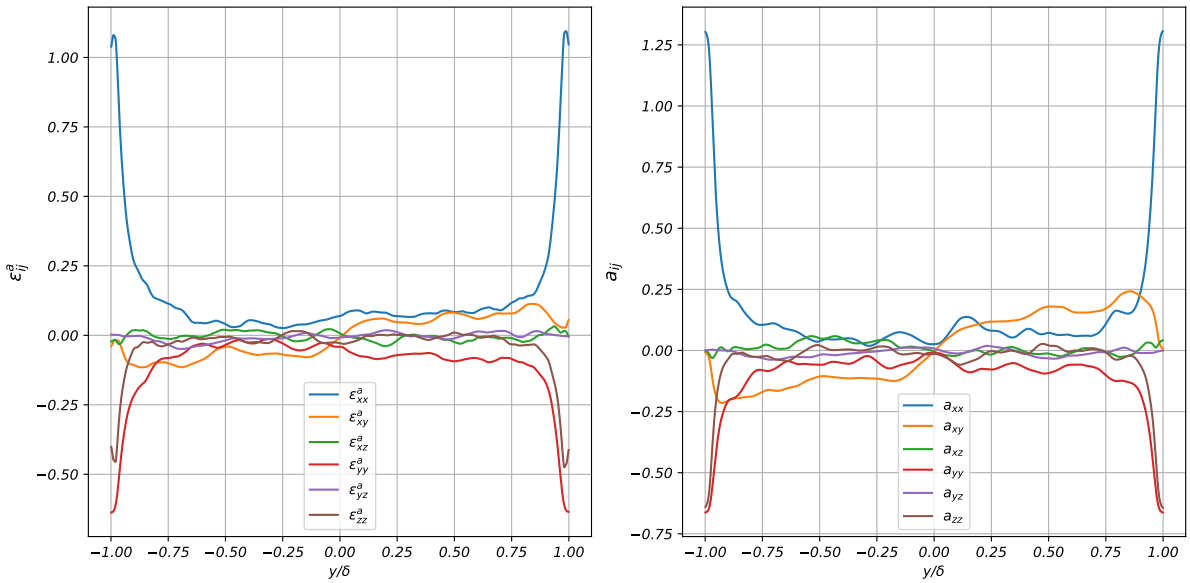


Figure 5. Anisotropies of SGS stress and dissipation rate in TCF with $Re_\tau = 700$, $\tilde{\Delta} = 17$

Applying gradient operator to Equation 27, scaling of the filtered velocity gradient tensor can be found:

$$\frac{\partial \bar{u}_i}{\partial x_j} \propto \begin{bmatrix} y^1 & y^0 & y^1 \\ y^2 & y^1 & y^2 \\ y^1 & y^0 & y^1 \end{bmatrix}. \tag{29}$$

Using Equation 29, scaling of the dissipation rate tensor, it's half trace and anisotropy tensor can also be found as

$$\varepsilon_{ij} \propto \begin{bmatrix} y^0 & y^1 & y^0 \\ y^1 & y^2 & y^1 \\ y^0 & y^1 & y^0 \end{bmatrix}, \qquad \varepsilon \propto y^0, \qquad \varepsilon^a_{ij} \propto \begin{bmatrix} y^0 & y^1 & y^0 \\ y^1 & y^0 & y^1 \\ y^0 & y^1 & y^0 \end{bmatrix}. \tag{30}$$

Also using Equation 29, scaling of the basis tensors $\mathbf{T}^{(n)}_{ij}$ and invariants can be found by simple substitution. The scalings are shown as follows:

$$\begin{aligned} \mathbf{T}^{(1)} &\propto y^2 \begin{bmatrix} y^1 & y^0 & y^1 \\ y^0 & y^1 & y^0 \\ y^1 & y^0 & y^1 \end{bmatrix} & \mathbf{T}^{(2)}, \mathbf{T}^{(3)}, \mathbf{T}^{(4)} &\propto y^4 \begin{bmatrix} y^0 & y^1 & y^0 \\ y^1 & y^0 & y^1 \\ y^0 & y^1 & y^0 \end{bmatrix} \\ \mathbf{T}^{(5)} &\propto y^6 \begin{bmatrix} y^1 & y^2 & y^1 \\ y^2 & y^1 & y^2 \\ y^1 & y^2 & y^1 \end{bmatrix} & \mathbf{T}^{(6)} &\propto y^6 \begin{bmatrix} y^1 & y^0 & y^1 \\ y^0 & y^1 & y^0 \\ y^1 & y^0 & y^1 \end{bmatrix} \\ \mathbf{T}^{(7)}, \mathbf{T}^{(8)}, \mathbf{T}^{(9)} &\propto y^8 \begin{bmatrix} y^0 & y^1 & y^0 \\ y^1 & y^0 & y^1 \\ y^0 & y^1 & y^0 \end{bmatrix} & \mathbf{T}^{(10)} &\propto y^{10} \begin{bmatrix} y^1 & y^2 & y^1 \\ y^2 & y^1 & y^2 \\ y^1 & y^2 & y^1 \end{bmatrix}, \end{aligned} \tag{31}$$

$$II_S \propto y^{0+4}, \quad II_\Omega \propto y^{0+4}, \quad III \propto y^{1+6}, \quad IV \propto y^{1+6}, \quad V \propto y^{0+8}. \tag{32}$$

Note that in Equation 31 and Equation 32, the scaling contribution of the non-dimensionalization term $\frac{k_{sgs}}{\varepsilon}$ is written separately for clarity. The aforementioned wall penalty terms are defined in Equation 33 and Equation 34. They are

applied "softly" to the relative error as shown in Equation 24 and Equation 26.

$$\epsilon_\sigma = \begin{cases} 1, & \sigma \propto y^0, \\ 1.5, & \sigma \not\propto y^0. \end{cases} \tag{33}$$

$$\epsilon_\gamma = \begin{cases} 1, & \varepsilon^a_{xy_{model}} \propto y^1, \\ 1.5, & \varepsilon^a_{xy_{model}} \not\propto y^1. \end{cases} \tag{34}$$

The penalty value of 1.5 was chosen rather arbitrarily, under the consideration to preserve genetic diversity within the population.

This concludes the methodology of the present work, Table 5 and Table 6 summarizes the model discovery configuration that will be applied. In Table 6, the suffixes M and C denotes a modification to the loss function and a correction that is applied, they will be made clear in subsection VI.A. The columns "Additional terminal" and "$\beta = 0$" will also be explained in subsection VI.A. Lastly, "Constant" refers to whether constant optimization is applied.

t

Table 5. Training configuration for scalar dissipation case.(% applies to all cases for all filter sizes indicated) Note that S6 uses an updated version of constant optimization.

Training configuration	Cases	filter size $\tilde{\Delta}$	Amount of data (%)	w_e, w_ρ	$p_{relative}$
S1	HIT	9,17,25	45.39	1,0	1
S2	TCF700	9,17,25	16.67	1,0	1
S3	HIT, TCF700, TCF1500	9,17,25	1	1,0	1
S4	HIT, TCF700, TCF1500	9,17,25	1	1,1	1
S5	HIT, TCF700, TCF1500	9,17,25	1	1,0	5
S6 (adaptive)	HIT, TCF700, TCF1500	9,17,25	1	1,0	1

Table 6. Training configuration for tensor case.

Training configuration	w_a	ϵ_γ	Constant	Additional terminal	Remarks
1	0	1	no	no	-
2	1	1	no	no	-
3	0	1	no	ν^*	-
4	1	1	no	ν^*	-
5	0	1	no	$\left(\frac{\mathcal{P}}{\varepsilon}\right), \nu^*$	-
6	1	ϵ_γ	yes	ν^*	-
6M	1	ϵ_γ	yes	ν^*	-
7	1	ϵ_γ	yes	$\left(\frac{\mathcal{P}}{\varepsilon}\right), \nu^*$	-
7M	1	ϵ_γ	yes	$\left(\frac{\mathcal{P}}{\varepsilon}\right), \nu^*$	-
7MC	1	ϵ_γ	yes	$\left(\frac{\mathcal{P}}{\varepsilon}\right), \nu^*$	-
8MC	0	ϵ_γ	yes	$\left(\frac{\mathcal{P}}{\varepsilon}\right), \nu^*$	Population size=1000, Generations=700
9MC	1	ϵ_γ	yes	$\left(\frac{\mathcal{P}}{\varepsilon}\right), \nu^*$	Population size=1000, Generations=700

V. Scalar dissipation results & discussion

All scalar models mentioned in this section are denoted with the identifier $S\langle n\rangle\_s\langle m\rangle$, where n indicates the training configuration defined in Table 5, and m denotes the initialization seed. For each training configuration, five initializations were conducted unless mentioned otherwise. RMAE and correlation coefficient (ρ) values mentioned are computed after GEP optimization on all data points of all cases. A reference model is chosen as the one proposed by Yoshizawa and Horiuti (1985), $\varepsilon = C_c \frac{k_{sgs}^{3/2}}{\Delta}$, with $C_c = 1.048$. The model constant is a default value used in the open-source CFD solver OpenFoam. This model will be used throughout the section as a reference for comparison.

A. Scalar model results

The expression for each best model (best, according to the metric RMAE-ρ) within a training configuration is shown in Table 7, their corresponding RMAE and correlation coefficients are tabulated in Table 8.

Table 7. Scalar model expressions of best individuals within each training configuration

Identifier	Model expression	Model constants
S1_s3	$C_2\nu\|\|S\|\|^2+\frac{\|\|S\|\|C_3(C_1+III)k_{sgs}^{3/2}}{(C_1+2C_2III)\sqrt{k_{sgs}+\Delta\|\|S\|\|}}$	$C_1 = 0.8331$, $C_2 = -0.1684$, $C_3 = 0.8132$
S2_s3	$C_3\nu\|\|S\|\|^2+\frac{C_1k_{sgs}^2}{\Delta\left((C_2+V)\sqrt{k_{sgs}}-C_4\Delta\|\|S\|\|\right)}$	$C_1 = -1.7516$, $C_2 = -0.9798$, $C_3 = 0.1257$, $C_4 = 0.5779$
S3_s2	$C_2\nu\|\|S\|\|^2+\frac{C_1C_3k_{sgs}^2}{\Delta(\Delta\|\|S\|\|-V\sqrt{k_{sgs}})}$	$C_1 = 1.0409$, $C_2 = 0.3183$, $C_3 = 1.2548$
S4_s3	$C_2\nu\|\|S\|\|^2+\frac{k_{sgs}^{3/2}}{(C_1+C_2(1-2II_\Omega+IV)-III)\Delta}$	$C_1 = 1.5466$, $C_2 = 0.0779$
S5_s4	$\frac{\nu\|\|S\|\|(C_1+C_2+III)\sqrt{k_{sgs}}}{\Delta}+\frac{C_3k_{sgs}^{3/2}}{2\Delta}$	$C_1 = -0.42$, $C_2 = 1.617$, $C_3 = 0.9515$
S6_s5	$C_2\nu\|\|S\|\|^2+\frac{C_1\sqrt{k_{sgs}}\|\|S\|\|\nu+C_3k_{sgs}^{3/2}}{\Delta}$	$C_1 = 1.028$, $C_2 = -0.0328$, $C_3 = 0.5153$

Figure 6 shows the model predictions of the dissipation spectrum of HIT_f17, predictions at other filter sizes are not shown as they exhibit similar trends. As can be seen, all trained models perform significantly better than the default model, although all models consistently over-predict at all wavenumbers. It is no surprise that models S1_s3 and S2_s3 produces the best and worse predictions respectively, as one is trained entirely on HIT cases and the other on TCF cases. Although model S1_s3 predicts the most accurate spectrum, it has larger RMAE than models S3_s2 and S4_s3, whilst it's correlation is the lowest amongst all other models, this is shown in Table 8. This indicates that model S1_s3 better predicts the energy contained within all scales in the spectrum but more inaccurate in terms of the spatial distribution of dissipation.

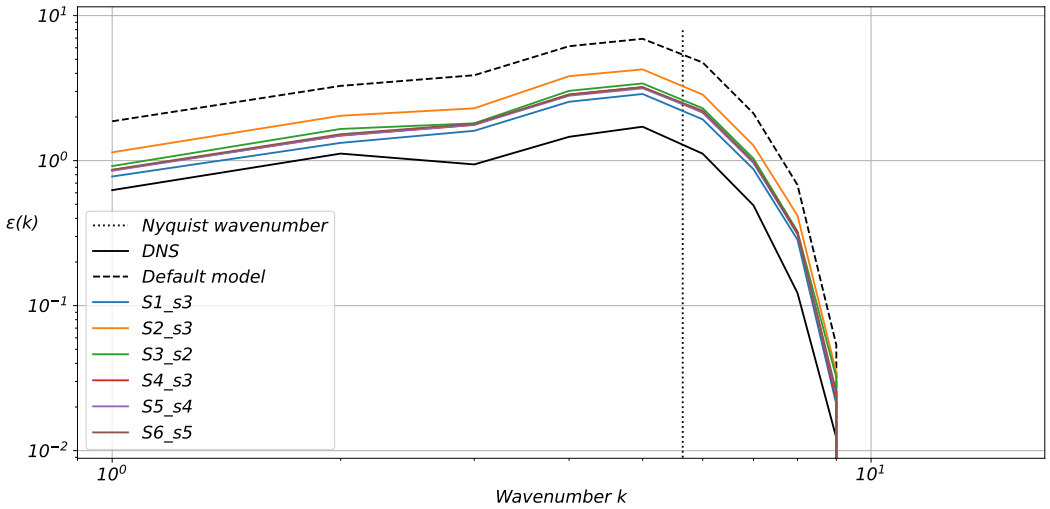
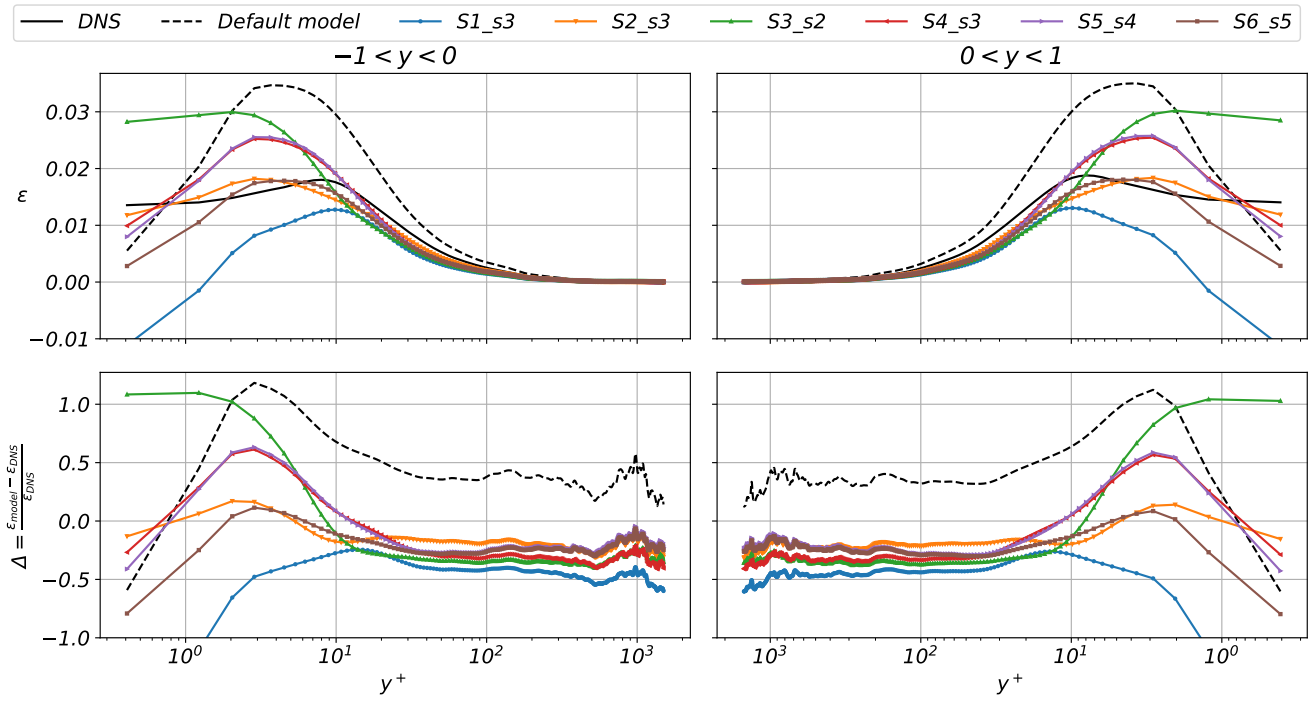


Figure 6. Dissipation spectrum of HIT_f17

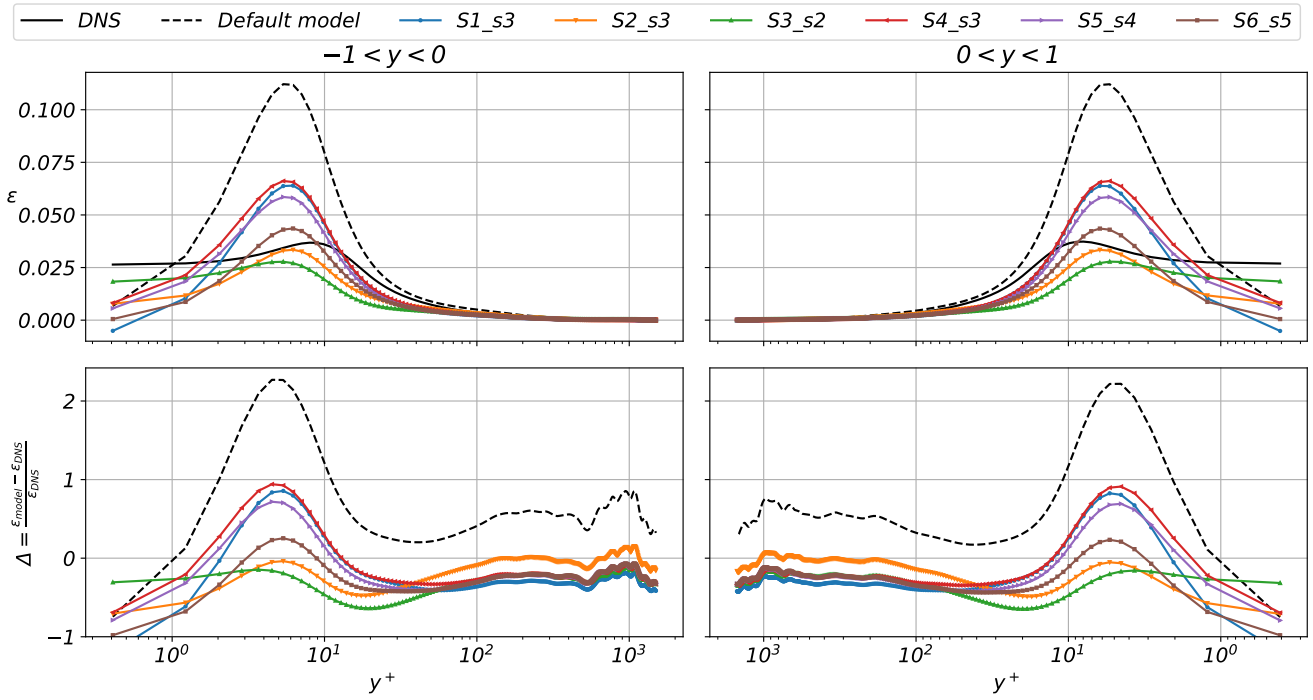
Table 8. RMAE of best model of each training configuration averaged across filters. The column "Score" is the average correlation subtracted from average error. Colour of the entries are implemented per column, with a linear scaling between the minimum and maximum of the corresponding column. Blue and red indicates the best and worse performance respectively.

Model identifier	RMAE			Correlation coefficient			Average error	Average correlation	Score
	TCF1500	TCF700	HIT	TCF1500	TCF700	HIT			
S1_s3	0.5530	0.4945	0.3339	0.9053	0.8064	0.6964	0.4604	0.8027	-0.3423
S2_s3	0.4489	0.4223	0.4751	0.9276	0.8380	0.7576	0.4488	0.8411	-0.3923
S3_s2	0.4909	0.4566	0.3245	0.9144	0.8319	0.7771	0.4240	0.8411	-0.4171
S4_s3	0.4856	0.4535	0.3284	0.9227	0.8180	0.7278	0.4225	0.8228	-0.4003
S5_s4	0.4854	0.4594	0.3458	0.9230	0.8183	0.6974	0.4302	0.8129	-0.3827
S6_s5	0.4771	0.4491	0.3468	0.9230	0.8183	0.6974	0.4244	0.8129	-0.3885

Model predictions of the TCF1500_f9 and TCF1500_f25 are shown in Figure 7. First, it can be observed that model S1_s3 predicts negative values of ε in close vicinity of the channel walls, which are non-physical. Secondly, all models except S1_s3 are able to produce predictions with better agreement w.r.t DNS reference data than the standard isotropic model. Models S2_s3 and S3_s2 particularly, most closely capture the gradient of ε near channel walls. The locations of maximum dissipation that the models predict also seem to be consistently underestimated by $y^+ \approx 3$, except for model S1_s3 with $\tilde{\Delta} = 9$. A general trend of decreasing dissipation rate, with increasing filter size can also be observed. Here, models S2_s3 performs the best which is a sensible result considering that is is trained only on TCF data.



(a) $\tilde{\Delta} = 9$



(b) $\tilde{\Delta} = 25$

Figure 7. Dissipation rate prediction of TCF1500_f9 and TCF1500_f25 by best models from each configuration, including relative error

Given these results, determining which is the best performing model is no simple task. From Table 17, using only averaged error as a criteria, it would be model S4_s3 whilst, if considering only correlation, it would be S2_s3 and S3_s3. If error and correlation are consider with same weighting, then the best model would be S3_s2, followed

by S4_s3 and S2_s3, although one might also argue that S3_s2 performs better for two reasons. First that it does not produce a significant overestimation in vicinity of channel walls at $\tilde{\Delta} = 9$, which S3_s2 does. Secondly, that it captures the gradient of ε close to channel walls significantly better than S4_s3.

B. Scalar model discussion

Table 7 shows that all models except S5_s4, contains a term with $\nu||S||^2$, $||S||^2$ being equivalent to the un-normalized first invariant, albeit with some significant difference in it's corresponding constant. From models S2_s3 and S3_s2, the remaining term takes form of $\frac{C_1 k_{sgs}^2}{C_2\Delta^2||S||+(C_3-V)\Delta\sqrt{k_{sgs}}}$. Figure 8 shows that the $\nu||S||^2$ term contributes to the correct wall scaling of y^0 with the remaining terms dictating the shape of the profile up to $y^+ \approx 50$.

Wall scaling of the S5_s4 model is of order y^1, which is not the desired scaling for ε. Model expressions of S5_s2 and S5_s3 shown in Appendix C does contain the $\nu||S||^2$ term and thus have the correct scaling, they are however not the best performing models amongst the training set. This suggests that whilst the application of wall penalty in the loss function does encourage the models with correct scaling, the penalty value implemented in Equation 33 should be increased. This applies for all scalar model training configurations.

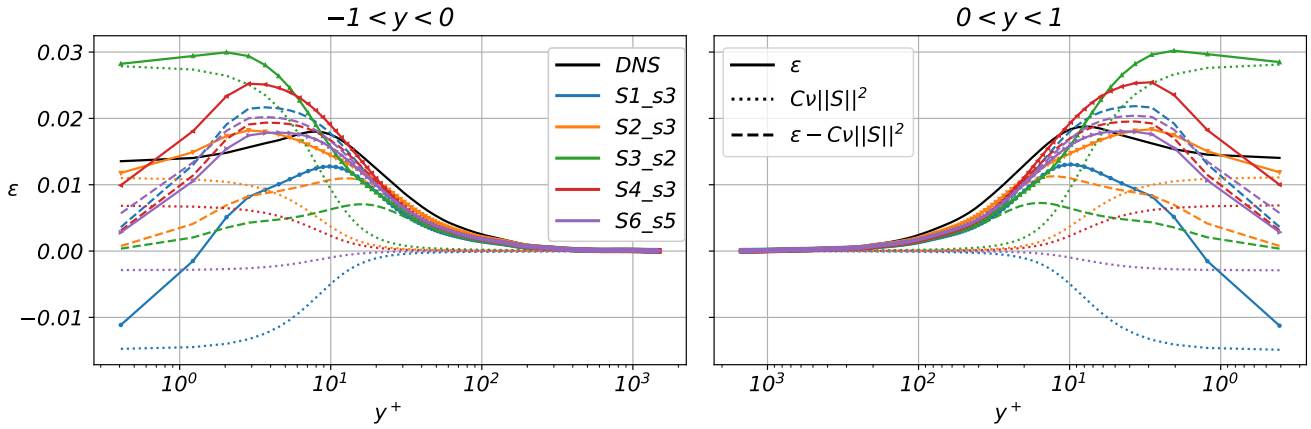


Figure 8. Contribution of $C\nu||S||^2$ and remaining terms of best models from configuration S1-S4 for TCF1500_f9

Regarding the different training configurations, Table 9 shows the RMAE and correlation coefficient of each training configuration averaged. It reflects that with multiple training cases, the configurations S3, S4 and S6 perform more consistently across all cases. The single case training configuration S2 performs particularly well in TCF whilst still producing reasonably good results in the HIT cases as well. Configuration S4 with the correlation coefficient included in the fitness function does not seem to show any improvement in either RMAE or correlation. Moreover, configuration S5 which uses training data with a higher sampling density close to channel walls performs worse both in TCF and HIT cases, for both metrics of RMAE and correlation. Deficiencies in the implementation of numerical constant optimization that has been applied throughout S1-S5 had been explained in subsection IV.D, it is therefore unexpected to see that configuration S6 did not produce a model that outperforms model S3_s2. Given the small number of initializations, between the configurations S2,S3,S4 and S6, the best training configuration cannot be ascertained.

Figure 9 shows the convergence of minimum loss values. Note that S4 is offset from the rest of the training cases since it's fitness function is defined with negative correlation coefficient ρ included whilst the other cases does not. Unfortunately, only the loss values were logged instead of RMAE and ρ separately during training leading, resulting in S4 being unable to be shown on similar scales. For all trainings, they seem to have converged to the final model within 60 generations, except for some minor improvements that are likely due further optimization of constants. This shows that it is unlikely that a further increase in number of generations would result in any significant improvement of the models.

Table 9. RMAE of all models for each training configuration averaged across filters and 5 random initializations. The column "Score" is Average correlation subtracted from average error. Colour of the entries are implemented per column, with a linear scaling between minimum and maximum of the corresponding column. Blue and red indicates the best and worse performance respectively.

Training configuration	RMAE			Correlation coefficient			Average error	Average correlation	Score
	TCF_1500	TCF_700	HIT	TCF_1500	TCF_700	HIT			
S1	0.8445	0.5849	0.3327	0.8473	0.7447	0.7582	0.5874	0.7834	-0.1960
S2	0.4750	0.4417	0.4084	0.9243	0.8258	0.7763	0.4417	0.8421	-0.4004
S3	0.4941	0.4594	0.3400	0.9195	0.8235	0.7921	0.4312	0.8450	-0.4138
S4	0.4946	0.4617	0.3486	0.9227	0.8173	0.7718	0.4350	0.8372	-0.4023
S5	0.5487	0.5246	0.4297	0.9155	0.8083	0.7087	0.5010	0.8108	-0.3098
S6	0.4892	0.4595	0.3503	0.9235	0.8192	0.7680	0.4330	0.8369	-0.4039

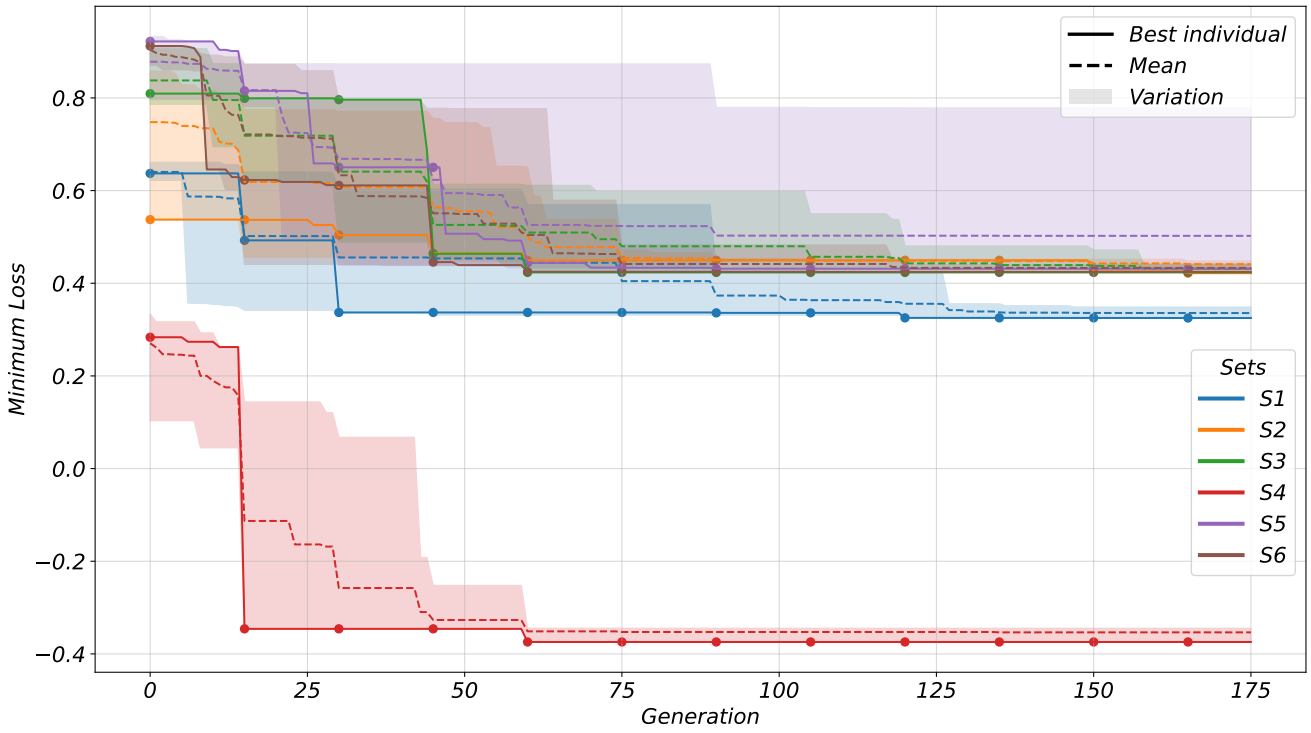


Figure 9. Minimum loss values of scalar dissipation optimization, scatters indicate where constant optimization is activated. "Best": Best performing initialization, "Mean": Mean across all initializations, "Variation": Minimum to maximum range across all initializations

1. Scalar model a-posteriori validation

Since the scalar dissipation model was to be used in the tensor model discovery, it was decided that the models should be validated in a CFD solver for two reasons. First, to confirm it's numerical stability in an actual LES, which determines entirely whether it may be a practically applicable model at all. Secondly, it will aid in the performance assessment of the models based on the simulation output.

The LES solver chosen for this purpose is the open-source CFD software OpenFoam. The setup of the simulation is TCF, similar to the generation of DNS data, with a kinematic viscosity of $\nu = 7.6923e - 5$ and a bulk velocity

forcing of unity. Periodic boundary conditions were applied in the streamwise and spanwise directions, no-slip and zero-gradient conditions at channel walls. A number of simulation configurations was tested due to some unsatisfactory results, details of the setup are tabulated in Table 10. The grading factor G is defined as $\Delta y_i = \Delta y_{i+1} G^{\frac{1}{N_y/2}}$ according to the implementation in OpenFoam and N_y being the total number of cells in the wall-normal direction. Mesh refinement here is implemented rather aggressively in order to obtain a wall cell height of $\Delta y^+ \leq 1$. CN in Table 10 refers to the Crank-Nicolson time stepping scheme, this was implemented in attempt to obtain better agreement of LES simulation with DNS reference data. All simulations were conducted with a time step size of $\Delta t = 0.005s$ which gives a mean Courant number of ≈ 0.12. A total simulation time of 1000s was chosen in order to obtain sufficient samples for the computation of statistical quantities, the statistics presented hereafter are averaged from t=500s-1000s. Case 3 is an exception where the simulation time is 500s; results averaging from 250s, it was terminated due to no improvements being observed. For all cases, the turbulence model implemented is a one equation eddy viscosity model proposed by Yoshizawa and Horiuti (1985), where the SGS eddy viscosity is modelled as

$$\nu_{sgs} = C_k \sqrt{k_{sgs}} \Delta, \tag{35}$$

k_{sgs} is obtained by solving Equation 22. The reference dissipation term was modelled as $\varepsilon = C_c \frac{k_{sgs}^{3/2}}{\Delta}$, as in Yoshizawa and Horiuti (1985). Implementation of the optimized scalar dissipation model is by simple substitution of model expressions in Table 7 in place of ε in Equation 22.

Table 10. Scalar dissipation model validation cases

	Nx,Ny,Nz	Grading factor (G)	Δx^+, Δy^+, Δz^+	Time-stepping scheme	Model	C_k, C_c
Case 1	64,64,64	100	32, 0.94-94, 16	Backward	kEqn	0.094, 1.048
Case 2	64,128,64	70	32,0.6-45, 16	Backward	kEqn	0.094, 1.048
Case 3	64,64,64	100	32, 0.94-94, 16	CN	kEqn	0.094, 1.048
Case 4	64,64,64	100	32, 0.94-94, 16	Backward	S2_s3	0.094, -
Case 5	64,64,64	100	32, 0.94-94, 16	Backward	kEqn	0.0125, 1.048
Case 6	64,64,64	100	32, 0.94-94, 16	Backward	S2_s3	0.0125, -

Note that OpenFoam documentation (OpenCFD Ltd., 2025) references Yoshizawa and Horiuti (1985) and uses $C_k = 0.094$ and $C_c = 1.048$. However, the theoretical prediction of the constants by Yoshizawa and Horiuti (1985) are actually $C_k = 0.043$ and $C_c = 1.84$. The value $C_k = 0.094$ was quoted by the same authors from Deardorff (1970), this value subsequently gives $C_c = 8.3$. The final recommendation of Yoshizawa and Horiuti (1985) for the model constants are $C_k \approx 0.05$ and $C_c \approx 1$.

The mean velocity and stress profiles of the simulations are shown in Figure 10. The friction velocity used for non-dimensionalization is defined as $u_\tau = \sqrt{\frac{dp}{dx}}$

From Figure 10, it can be seen that for Cases 1-4, where the OpenFoam default model constants were used, all quantities show extremely poor agreement with DNS reference data. The relation $U^+ = y^+$ which is supposed to hold within the viscous sublayer is not observed at all. As a result of the deviation in mean velocity profile, the viscous stress in the region $y^+ < 10$ is also severely under-predicted, leading to the kink in total stress profile. The use of the Crank-Nicolson time stepping scheme seems to have negligible effect. The increased grid resolution of Case 2 does seem to show slightly better agreement in terms of it's viscous stress prediction but is insignificant in view the large deviation in the first place.

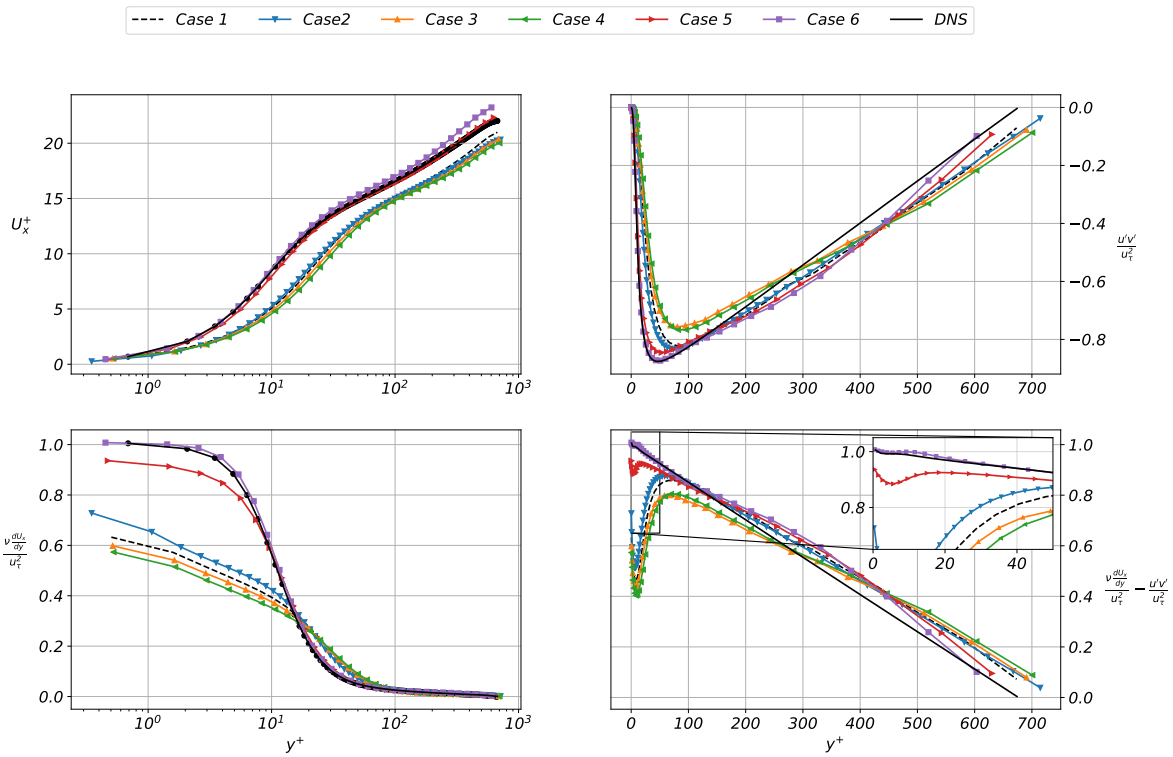


Figure 10. Comparison of LES results. Upper left: Mean velocity U_x^+, Upper right: Turbulent stress $u'v'/u_\tau^2$, Lower left: Viscous stress $\nu \frac{dU_x}{dy}/u_\tau^2$, Lower right: Total stress $(\nu \frac{dU_x}{dy} - u'v')/u_\tau^2$.

Eventually, the situation was remedied by a change of model constant C_k, the final value of 0.0125 was found by trial and error. The proposed value of $C_k = 0.05$ by Yoshizawa and Horiuti (1985) as well as $C_k = 0.025$ were the other tested values, both show improvements w.r.t the cases with default C_k value, but the deviation in mean velocity remains large and therefore are not shown here.

For Case 5, the mean velocity and viscous stress profiles shows good agreement with DNS reference especially in the region of $y^+ > 30$. However, the under-prediction of wall stress value still leads to a small kink in the total stress profile. The S2_s3 model of Case 6, on the other hand, captures the stresses in vicinity of the channel wall more faithfully although turbulent stresses tend to deviate more significantly at $y^+ > 100$.

For all cases, there seems to be some deviation in the linearity of the turbulent stress profile, this is suspected to be the aggressive mesh refinement applied towards the channel walls, but is not further investigated due to the time constraints of this thesis project.

Non-dimensionalized turbulent kinetic energy and dissipation rate is shown in Figure 11. Similarly, Cases 5-6 produces a kinetic energy profile with much better agreement than Cases 1-4, contrary to the mean velocity and stresses shown above, the standard model of Case 5 seems to outperform the optimized model S2_s3 with better matched peak kinetic energy. No time averaged statistical DNS data was available for the dissipation rate. First, we note that contrary to Figure 7, where the a priori evaluation of the standard model is significantly larger than that of the S2_s3 model, they are in the a-posteriori sense similar in magnitude. A possible explanation of this is in the implementation of the dissipation model in OpenFoam which treats the model as:

$$\varepsilon = \frac{C_c k_{sgs}^{t=i} \sqrt{k_{sgs}^{t=i-1}}}{\Delta}, \tag{36}$$

where the latter k_{sgs} is lagged in time, in order to ensure stability, see De Villiers (2006). This treatment would have a larger effect on the standard model, since the influence of such treatment is more direct on the standard model. Due to the presence of terms without k_{sgs} in the S2_s3 model, thee time lag effect might not be as pronounced. Secondly,

although no statistical DNS data in ε is available, one would expect the similar profiles to those shown in Figure 7, where a peak in dissipation occurs at $y^+ \approx 10$. All models seems to reproduce dissipation at the roughly similar locations, except Case 6 which estimates peak dissipation at the channel wall.

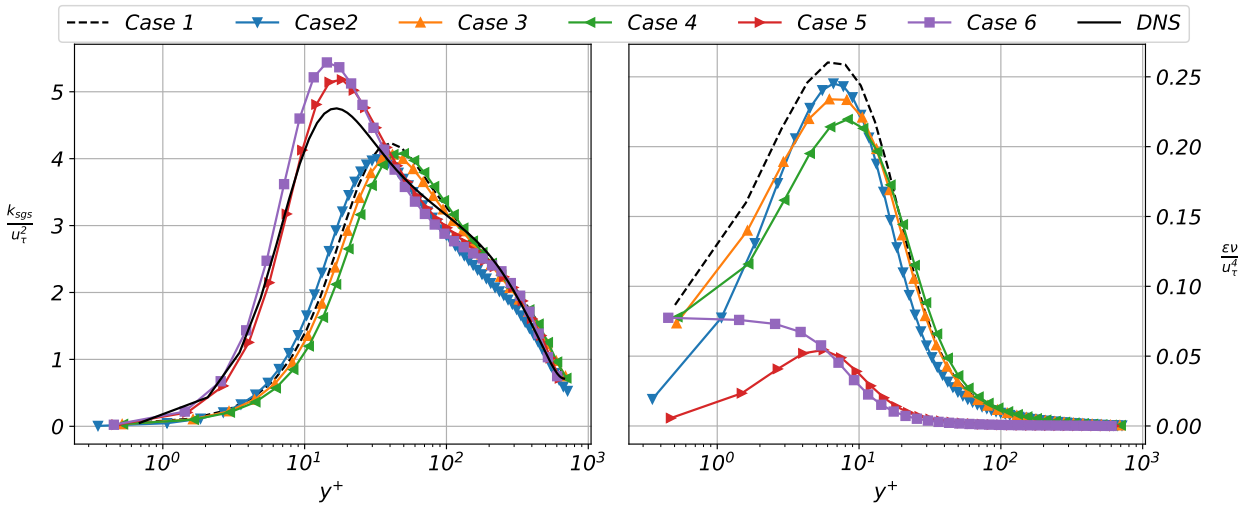


Figure 11. Comparison of LES results. Left: Turbulent kinetic energy $k^+ = k/u_\tau^2$, Right: Dissipation rate $\varepsilon^+ = \varepsilon\nu/u_\tau^4$.

To summarize, such a large reduction in the suggested model constant C_k in order to obtain comparable results w.r.t DNS reference data was unexpected. Most importantly, the LES simulations show that the optimized model is numerically stable and can be applied in actual LES simulations. As for it's performance, the preliminary results show that for the current test case and choice of model constants, the optimized S2_s3 model performs marginally better, particularly in the estimation of wall shear stress and turbulent stress in the near wall regions.

VI. Tensor model results & discussion

The training data configurations that has been considered are tabulated in Table 4 while the other parameters such as the implementation of wall penalties, numerical constant optimization etc are presented in Table 6. This section is divided into two parts. Initial results are presented in subsection VI.A, results which were generated subsequent to some adjustments are shown after in the same section. subsection VI.B provides further review of the results and framework.

A. Initial results

During some initial tests , significant numerical instabilities were found in the discovered ε_{ij}^a models and subsequently in a_{ij}, with RMAE in the order of magnitude of 1e9. To investigate the cause of this, the SGS anisotropy model with no dissipation anisotropy model was calculated, this corresponds to solving for the $\beta^{(n)}$ expressions (Equation 16) with $\gamma^{(n)} = 0$. We refer to this as the G0 model. The result of this for TCF700_17 is shown in Figure 12.

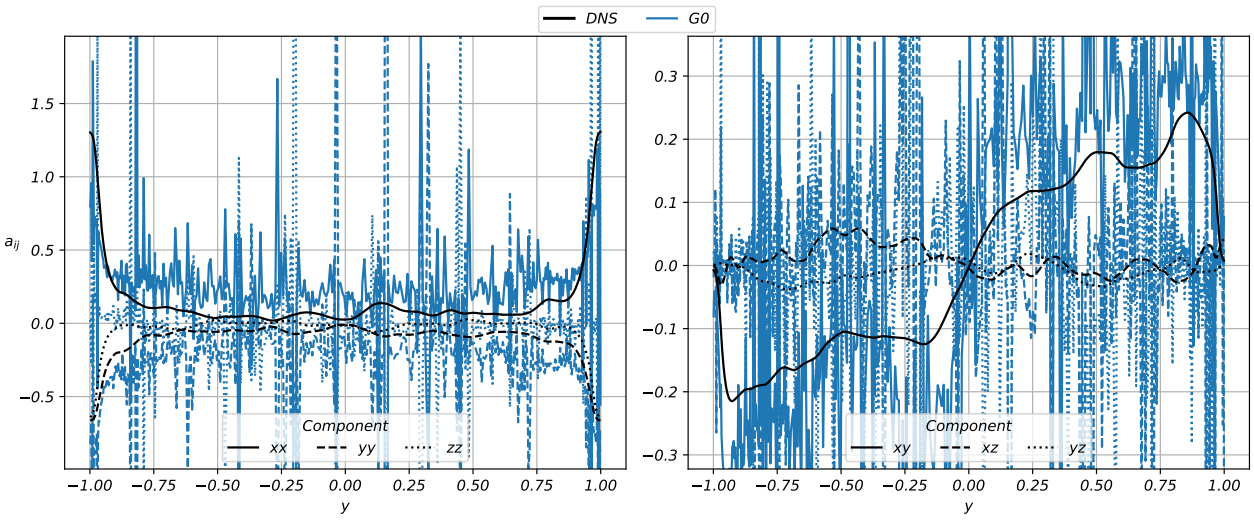


Figure 12. Tensor components of a_{ij} predicted by the G0 model. Case TCF700_f17

An inspection of the $\beta^{(n)}$ coefficients shows that such instabilities occur where $\mathcal{P}/\varepsilon \approx -0.5$. Indeed, when $\mathcal{P}/\varepsilon = -0.5$, the variable θ_3 approaches zero and the expressions for $\beta^{(2,4,5,7,8,10)}$ in Appendix B are mathematically undefined due to division by zero. Numerically, multiple approaches can be employed, first, to set all $\beta^{(n)} = 0$ if $\mathcal{P}/\varepsilon \approx -0.5$. Second, to set $\beta^{(2,4,5,7,8,10)} = 0$, given the condition. Lastly, to solve the values of $\beta^{(2,4,5,7,8,10)}$ numerically using a least square method. The last method would introduce thousands of least square solves per loss function evaluation, which is deemed too costly to be implemented within the training process and thus excluded. Figure 13 shows the resulting G0 model, the three methods mentioned are applied under the condition of

$$\left|\frac{\mathcal{P}}{\varepsilon} + 0.5\right| \leq 0.1 \tag{37}$$

The value on the RHS of Equation 37 is obtained by trial and error until the majority of spikes that are observed in Figure 12 are no longer present. As can be seen, the difference between the setting $\beta^{(n)} = 0$ and $\beta^{(2,4,5,7,8,10)} = 0$ makes no observable difference in the resulting profile of the tensor components. The least square method produces only fractionally different results, this then justifies it's exclusion from the training process.

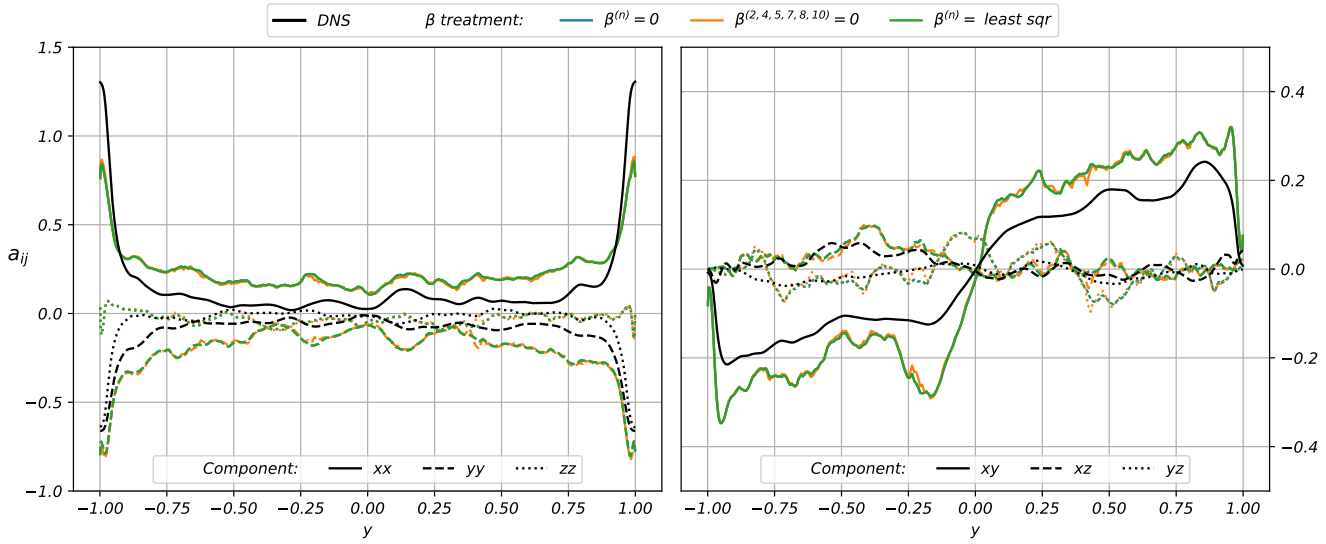


Figure 13. Comparison of G0 model with three different numerical handling of the condition in Equation 37

With a stable conversion from $\gamma^{(n)}$ to $\beta^{(n)}$, subsequent trainings still yielded models with profiles similar to Figure 12. Two main problems were identified. Firstly, mathematically undefined terms such as $\frac{1}{II_\Omega - II_\Omega}$ were present in model expressions. Secondly, the terminal set that has been defined was insufficient to obtain models that has good agreement w.r.t DNS reference. The first problem lead to an additional penalty term ϵ_∞ being added to the loss function in Equation 26, the resulting loss is defined as

$$Loss(\varepsilon^a_{ij_{model}}) = \left(\frac{\epsilon_\gamma}{q} \sum_q \frac{\left| \sum_n \gamma^{(n)}_q \mathbf{T}^{(n)}_{ij,q} - \varepsilon^a_{ij,q} \right|}{|\varepsilon^a_{ij,q}|} + \frac{\left| \sum_n \beta^{(n)}_q \mathbf{T}^{(n)}_{ij,q} - a_{ij,q} \right|}{|a_{ij,q}|} \right) \epsilon_\infty, \tag{38}$$

$$\epsilon_\infty = 1 + 99 \cdot n_{zoo}, \tag{39}$$

where n_{zoo} is the number of undefined terms within the chromosome.

To tackle the second problem, the terminal set was expanded with ν^* and $\mathcal{P}/\varepsilon$. Inclusion of the first term was natural from its frequent occurrence in the scalar model case. The rationale of including the latter term is as follows: first observe in Figure 5 that profiles of a_{ij} and ε^a_{ij} are similar in shape. Secondly, upon substitution of $\gamma^{(n)} = 0$ into the $\beta^{(n)}$ expressions in Appendix B, the non-zero terms $\beta^{(1,2,4,6,7)}$ are functions of II_Ω and $\theta_3(\mathcal{P}/\varepsilon)$. It can also be observed from Figure 13 that the G0 model is reasonably well aligned w.r.t DNS reference. Thus is it deduced that $\mathcal{P}/\varepsilon$ should be included in the terminal set of GEP.

Models T1_1_s1-T1_5_s1 and T1_5_s1_79 were trained without constants optimization and wall penalties implemented, since their purpose is mainly to determine whether or not to implement the aforementioned extra terminals. Their model expressions are shown in Appendix F, Table 11 and Table 12 tabulates the model errors of ε^a_{ij} and a_{ij} respectively. Note that the model T1_5_s1_79 denotes a modified version of the original T1_5_s1 model that GEP output. Due to some numerical instability that was observed in the original T1_5_s1 model, albeit not as severe as those in Figure 12, we further inspect the model expression. The expression of $\gamma^{(7,9)}$ reads as follows:

$$\gamma^{(7)} = \frac{-\nu^* \frac{\mathcal{P}}{\varepsilon}(III - V)(V - \nu^*)(\nu^* - 10.4)}{\left[II_\Omega III\, V - 33.3\frac{\mathcal{P}}{\varepsilon}(II_\Omega - V)\right]\left[80.9 II_\Omega (V - \nu^*)(\nu^* - 10.4) - \nu^* \frac{\mathcal{P}}{\varepsilon}(II_S - \frac{\mathcal{P}}{\varepsilon})\right]}$$

$$\gamma^{(9)} = \frac{\nu^* \left(\frac{\mathcal{P}}{\varepsilon}\right)^3}{II_S(II_S + \frac{\mathcal{P}}{\varepsilon})[II_\Omega + \nu^*(II_S + II_\Omega - IV)]\left(V\left(\frac{\mathcal{P}}{\varepsilon}\right)^2 + \nu^* + 87.3\right)}$$

It was suspected that the complex rational function expression of $\gamma^{(7,9)}$ causes the observed instabilities as the expressions are undefined if the denominator of the function is 0. The terms were then set to 0 to obtain the model as shown Figure 14. We use the notation $\_\langle ab \rangle$ to indicate a modification of $\gamma^{(a,b)} = 0$. A comparison of the original and modified model is given in Appendix E for reference. Returning to the error of the models, both errors in ε^a_{aij} and a_{ij} show that the inclusion of ν^* in T1_3_s1 and T1_4_s1 clearly leads to significantly improved results relative to T1_1_s1 and T1_2_s1. The addition of $\mathcal{P}/\varepsilon$ in the training of model T1_5_s1 lead to larger errors relative to T1_3_s1 except in ε^a_{ij} for the case of TCF700_f9. It is also noted that the error for the case of TCF700 with the smallest filter size is significantly higher than those with larger filters or the HIT cases.

The tensor components of selected models are plotted in Figure 14 for TCF700_f9 and TCF700_f25. Note that the lines of T1_3_s1 is mostly overlapped by T1_5_s1 in the figure. First the case with $\tilde{\Delta} = 25$ is discussed. On the diagonal tensor components of ε^a_{ij}, T1_4_s1 reproduces the increasing and decreasing trend of ε^a_{xx} and ε^a_{yy} respectively towards channel walls, albeit with significantly different gradient and wall values. There is little variation captured for all models in the ε^a_{zz} component. Models T1_3_s1 and T1_5_s1 capture little variation towards channel walls in all diagonal components except for a small peak close to $y = 1$. On the cross components of ε^a_{ij}, models T1_3_s1 and T1_5_s1 notably reproduces ε^a_{xy} with entirely opposite magnitude while T1_4_s1 correctly predicts the sign of ε^a_{xy} throughout the channel centre but produces a sharp spike towards the channel walls.

Regardless of the large variation in the trends of the tensor components predicted by the different models, the resulting a_{ij} profiles are largely similar in the channel centre region for all models. Over large parts of channel centre,

Table 11. RMAE in ε^a_{ij} of models shown per training case and averaged across training cases. All values have been averaged amongst the 6 independent tensor components.

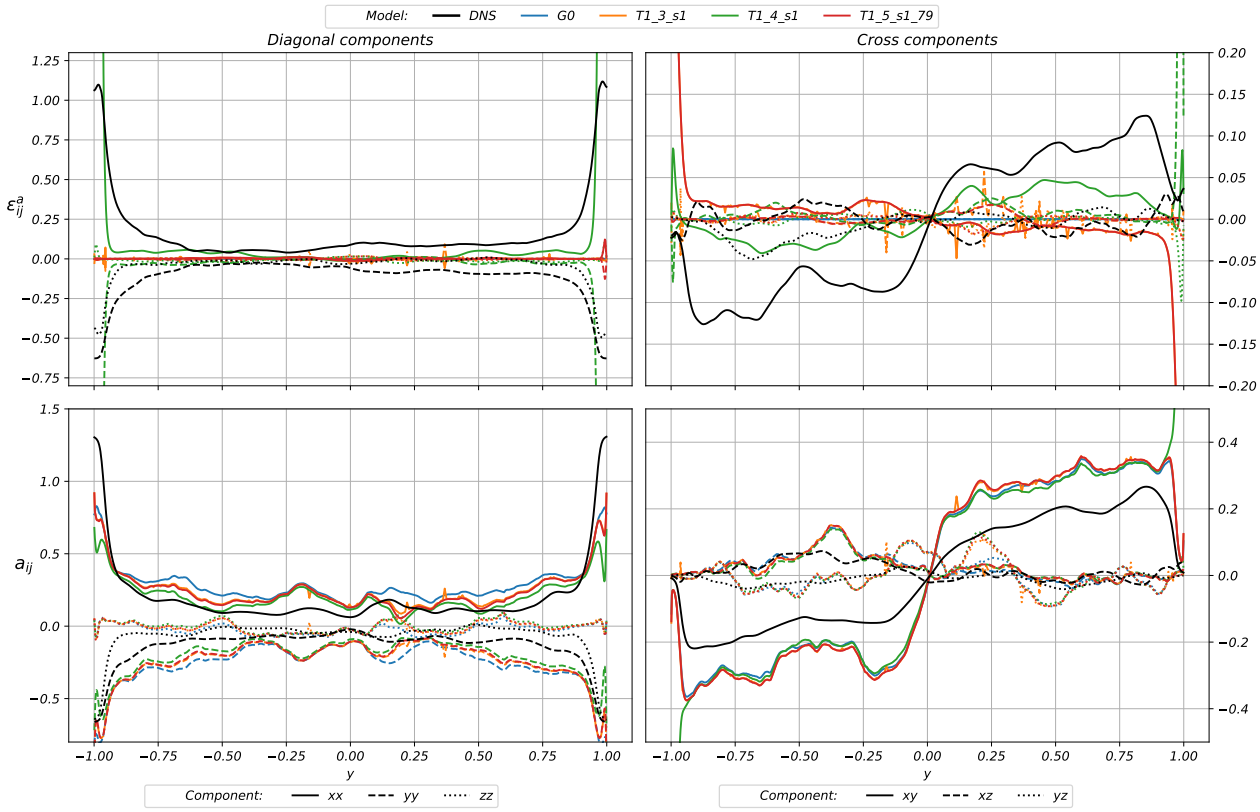
Model identifier	HIT			TCF700			Average RMAE
	$\tilde{\Delta}$ = 9	$\tilde{\Delta}$ = 17	$\tilde{\Delta}$ = 25	$\tilde{\Delta}$ = 9	$\tilde{\Delta}$ = 17	$\tilde{\Delta}$ = 25	
T1_1_s1	996	1588	2332	523	594	498	1088
T1_2_s1	1034	1679	2360	7877	1161	1155	2544
T1_3_s1	4	3	3	23	9	6	8
T1_4_s1	9	16	39	33	18	17	22
T1_5_s1	4	3	3	1280	9	5	217
T1_5_s1_79	4	3	3	1280	9	5	217
T1_6_s1_679	13	19	33	24	12	10	19
T1_7__s1_9	29	54	81	63	27	21	46
G0	1	1	1	1	1	1	1

Table 12. RMAE in a_{ij} of models shown per training case and averaged across training cases. All values have been averaged amongst the 6 independent tensor components.

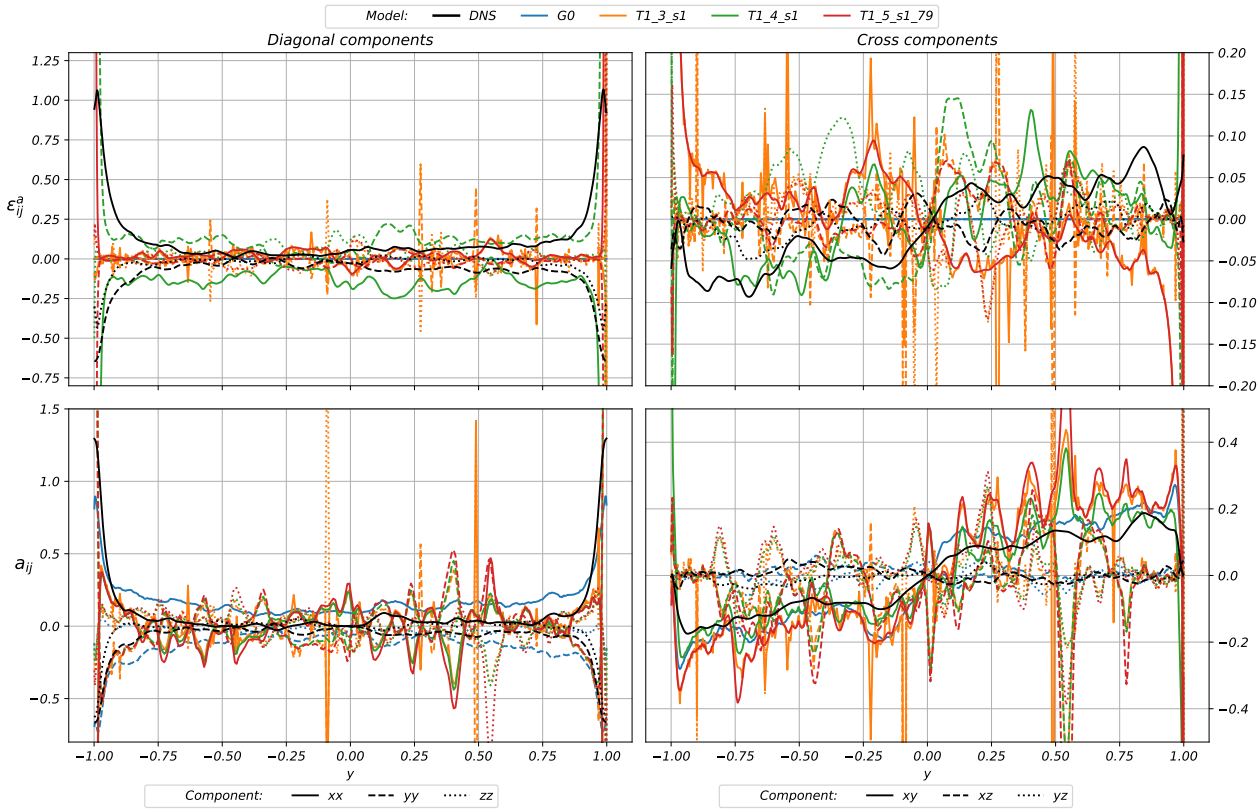
Model identifier	HIT			TCF700			Average RMAE
	$\tilde{\Delta}$ = 9	$\tilde{\Delta}$ = 17	$\tilde{\Delta}$ = 25	$\tilde{\Delta}$ = 9	$\tilde{\Delta}$ = 17	$\tilde{\Delta}$ = 25	
T1_1_s1	718	458	2114	2e8	507	1173	3e7
T1_2_s1	801	646	897	3e10	816	2130	5e9
T1_3_s1	37	32	109	8e6	22	32	1e6
T1_4_s1	38	33	108	9e6	27	41	1e6
T1_5_s1	53	46	121	5e9	47	46	9e8
T1_5_s1_79	37	32	109	5e9	22	32	9e8
T1_6_s1_679	38	32	109	8e6	21	32	1e6
T1_7__s1_9	42	34	112	1e7	29	34	1e6
G0	37	31	108	16	19	28	40

the trained models seem to predict more accurate values than the G0 model, however, referring to Table 12, their corresponding error values are larger than the G0 model. This can be attributed to the larger deviation from DNS reference that is observed in vicinity of channel walls.

For the case with $\tilde{\Delta}$ = 9, large wavelike oscillations and some numerical instabilities are observed for all trained models, particularly for model T1_3_s1. These are consistent with the large magnitude in error shown in Table 12. One can observe, albeit with some difficulty that, should the wavelike oscillations be removed, the general trend of the tensor components remain largely similar to those with $\tilde{\Delta}$ = 25. The exception in this are the ε^a_{xx} and ε^a_{yy} components where their trends have flipped in sign. Lastly, no such numerical instabilities and wavelike oscillations that can be observed in the trained models are exhibited by the G0 reference model.



(a) $\tilde{\Delta} = 25$



(b) $\tilde{\Delta} = 9$

Figure 14. Comparison of a_{ij} and ε_{ij} profiles of models T1_3_s1-T1_5_s1_79. G0 model is provided as reference.

Errors of the models that has been shown supports the addition of ν^* but not $\mathcal{P}/\varepsilon$ into the terminal set. In the T1_5_s1 model, the $\mathcal{P}/\varepsilon$ term only occurs in $\gamma^{(7,9)}$ which was set to 0 in order to remove numerical instabilities. This means that no $\mathcal{P}/\varepsilon$ term is in the T1_5_s1_79 model, and therefore seems to show that the inclusion of the term in the terminal set is unnecessary. But considering that the addition of the latter term is motivated by logical reasoning, it is kept in the terminal set. We proceed to models trained with the implementation of wall penalties and constants optimization.

RMAE of models T1_6_s1_679 and T1_7_s1_9 are shown in Table 11 and Table 12. Note that they are also modified in the manner as described before. Unexpectedly, T1_6_s1_679 show no significant decrease in error relative to model T1_3_s1 while model T1_7_s1_9 even shows an increase in error particularly in the HIT cases. Figure 15 shows the model's tensor components for the TCF700_f25. Firstly, the agreement in ε^a_{xy} of T1_7_s1_9 is significantly improved for the majority of regions close to channel centre, relative to all other trained models, the spike that had been observed in model T1_4_s1 in Figure 14a, however, persists. Secondly there is a large offset in the magnitude of ε^a_{xx} in the channel centre region and a sudden inflection towards the channel walls. The most notable feature of model T1_6_s1_679 is in the relatively flat profile in the diagonal components of ε^a_{ij}, as well as the opposite sign that ε^a_{xx} and ε^a_{yy} w.r.t DNS reference.

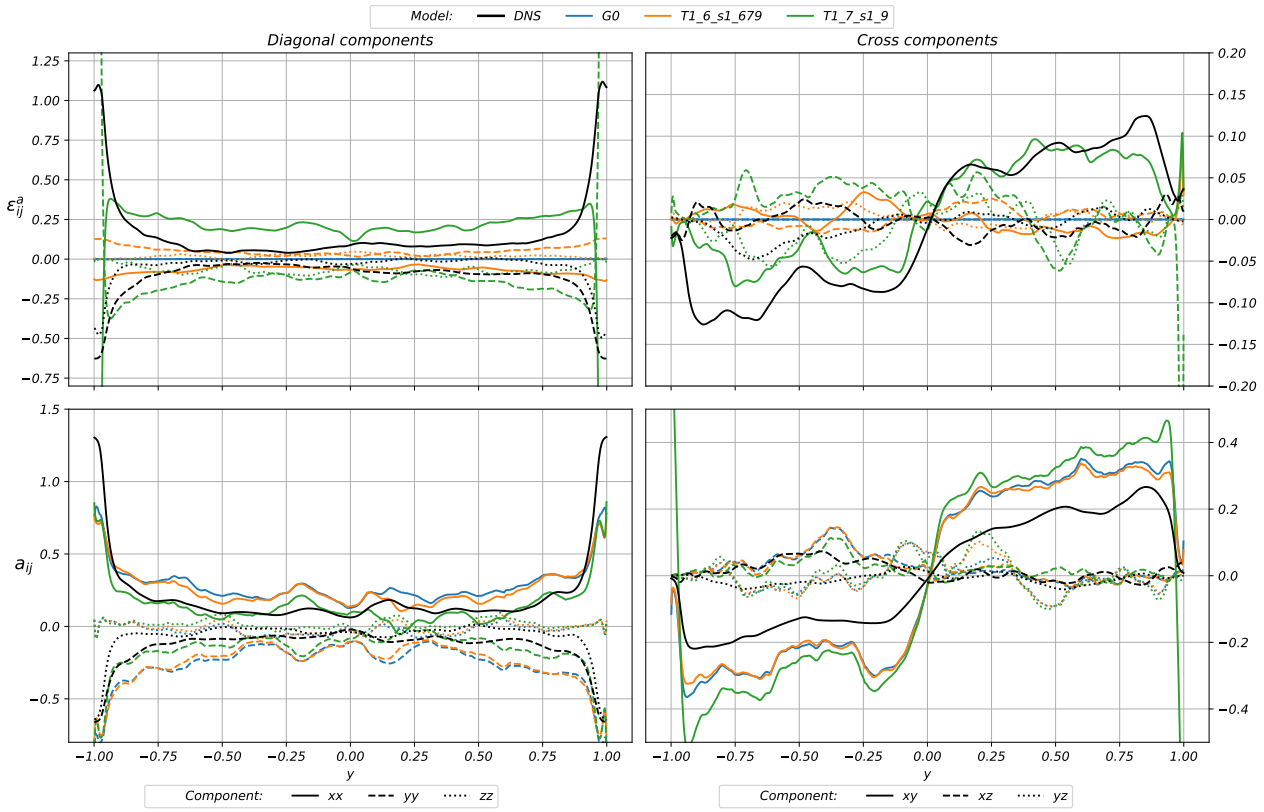


Figure 15. Comparison of a_{ij} and ε_{ij} profiles of models T1_6_s1_679 and T1_7_s1_9.

In view of the unusually large error in all flow cases as well as the particularly great error for TCF700 at the smallest filter size, it was suspected that the use of RMAE in the loss function is inappropriate due to 0 being a valid value that ε^a_{ij} and a_{ij} can take. It was thought that this would over-amplify the RMAE at locations where ε^a_{ij} and a_{ij} are close to 0. Therefore, a modification to use MAE in the loss function was made. The results of this showed no consistent improvement to what has been presented thus far, error for the TCF700_f9 case remain a few orders of magnitude larger that all other cases. Due to these results offering no particularly useful insight, they are not presented here, the interested readers are referred to Appendix G. All models that are train with the modified loss function are denoted with the suffix "M". Subsequent to the above mentioned modification, an error in the processing of the term ε was eventually uncovered. The error lead to negative and extremely small values of ε, which are further amplified

when substituted into k_{sgs}/ε for the non-dimesionalization of the basis tensors and invariants. The error proved to be the cause of the extremely large magnitude in error that is shown in Table 11 and Table 12, whilst having limited and no effect on the other TCF cases and HIT cases respectively. Details on the error and the extent of it's influence are documented in Appendix H. Models that are produced after said correction are denoted with the suffix "C".

Model discovery on configuration T1_7MC, T2_7MC, T1_8MC and T1_9MC were conducted after rectifying the error, each for 3 random initializations. The best model expressions are shown in Appendix F, model error of all initializations are tabulated in Table 13 and Table 14

Table 13. MAE in ε_{ij}^{a} of models shown for each training case and averaged. All values shown averaged across the 6 independent tensor components.

Model identifier	HIT			TCF_700			Average MAE
	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	
T1_7MC_s1	0.580	0.445	0.391	3.035	1.876	1.668	1.332
T1_7MC_s2	0.088	0.047	0.032	0.359	0.192	0.156	0.146
T1_7MC_s3	0.103	0.057	0.042	0.484	0.226	0.175	0.181
T2_7MC_s1	3.010	2.695	2.524	11.951	7.524	6.724	5.738
T2_7MC_s2	0.093	0.055	0.044	0.384	0.187	0.145	0.152
T2_7MC_s2_34	0.082	0.053	0.044	0.208	0.160	0.137	0.114
T2_7MC_s3	1.365	1.290	1.237	3.886	2.681	3.178	2.273
T1_8MC_s1	0.081	0.046	0.033	0.226	0.163	0.138	0.115
T1_8MC_s2	0.093	0.051	0.038	0.383	0.189	0.150	0.151
T1_8MC_s3	0.098	0.061	0.048	0.662	0.261	0.183	0.219
T1_9MC_s1	0.131	0.116	0.112	0.267	0.188	0.162	0.163
T1_9MC_s2	0.097	0.059	0.051	0.398	0.206	0.168	0.163
T1_9MC_s3	3.289	2.729	2.561	13.574	5.278	4.827	5.376
G0	0.081	0.046	0.032	0.215	0.171	0.151	0.116

Table 14. MAE in a_{ij} of models shown per training case and averaged across training cases and the 6 independent tensor components.

Model identifier	HIT			TCF_700			Average MAE
	$\tilde{\Delta}$ = 9	$\tilde{\Delta}$ = 17	$\tilde{\Delta}$ = 25	$\tilde{\Delta}$ = 9	$\tilde{\Delta}$ = 17	$\tilde{\Delta}$ = 25	
T1_7MC_s1	0.608	0.403	0.433	0.741	0.419	0.373	0.496
T1_7MC_s2	0.349	0.347	0.373	0.526	0.326	0.273	0.366
T1_7MC_s3	0.349	0.345	0.373	0.734	0.343	0.274	0.403
T2_7MC_s1	11.714	7.677	5.184	17.997	5.363	3.522	8.576
T2_7MC_s2	0.357	0.347	0.372	0.668	0.346	0.280	0.395
T2_7MC_s2_34	0.343	0.346	0.373	0.359	0.309	0.269	0.333
T2_7MC_s3	1.996	1.763	1.643	2.685	1.809	2.118	2.002
T1_8MC_s1	0.355	0.347	0.367	0.390	0.312	0.269	0.340
T1_8MC_s2	0.358	0.350	0.375	0.662	0.350	0.285	0.397
T1_8MC_s3	0.355	0.352	0.378	0.559	0.336	0.280	0.377
T1_9MC_s1	0.353	0.355	0.378	0.362	0.315	0.274	0.340
T1_9MC_s2	0.337	0.327	0.349	0.687	0.336	0.266	0.384
T1_9MC_s3	2.064	1.697	1.698	8.395	2.484	2.018	3.059
G0	0.344	0.347	0.374	0.350	0.308	0.269	0.332

Firstly, models that are trained on the T1_7MC and T2_7MC configuration exhibits large variability in the error of ε^a_{ij}, between different initializations. Figure 16 shows the convergence of loss function values of said models. The large variation in final loss function value can also be seen here. From the relatively flat profile line towards the end of the evolution, it can be said that apart from some minor improvements that are likely due to a mutation in numerical constants, the models have converged to their final form. A possible explanation to the large variation between initializations is that in the tensor case, a chromosome with 10 genes is used, each with a head length of 25. In the scalar case, where the chromosome contains a single gene, the search space is significantly smaller. In the tensor case, due to the larger search space, the population size of 350 might not provide a sufficiently diverse population, leading to the large variation between initializations. This could also be potentially improved by an increase in the probabilities of genetic operators being applied on a chromosome. The inclusion of the T1_8MC and T1_9MC configurations stem from the above consideration. As can be seen in Figure 16, there is a decrease in the variation of loss function values for those configurations. We also observe in Table 13 that especially for the T1_8 configuration, variation in the error of ε^a_{ij} is particularly small relative to others.

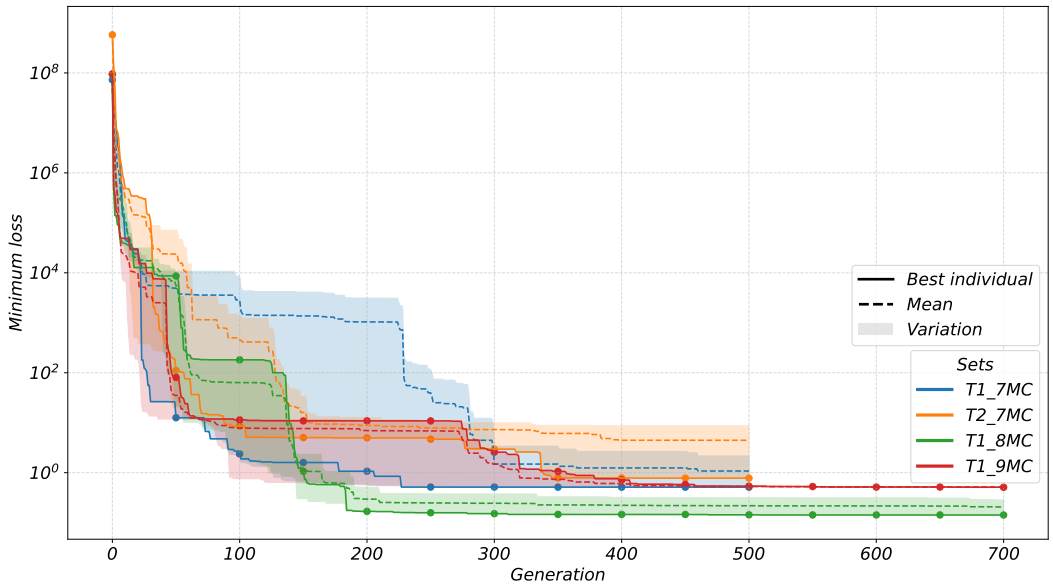


Figure 16. Minimum loss values for 4 training configurations, scatters indicate where constant optimization is activated. "Best": Best performing initialization, "Mean": Mean across all initializations, "Variation": Minimum to maximum range across all initializations

Secondly, we observe that model T2_7MC_s2_34 is the only model that shows an improvement relative to G0 in ε^a_{ij} whilst only being marginally outperformed in a_{ij}. The improvement is mainly attributed to decreased error in the TCF cases, the G0 reference still performs better on the HIT cases.

Three models; T2_7MC_s2_34, T1_8MC_s1 and T1_9MC_s1, are shown in Figure 17. First, we note some of the undesirable traits such as the spike of T1_8MC_s1 in $\varepsilon^a_{xx,yy}$ and T1_9MC_s1 in ε^a_{xy}, which approaches large magnitudes close to the channel walls. In general, it shows that T2_7MC_s2_34 does not reproduce any particular tensor component of ε^a_{xy} very close, but provides, in general better alginment, as opposed to T1_8MC_s1 and T1_9MC_s1 which has particularly god agreement in ε^a_{xy} and $\varepsilon^a_{xx,yy}$ respectively.

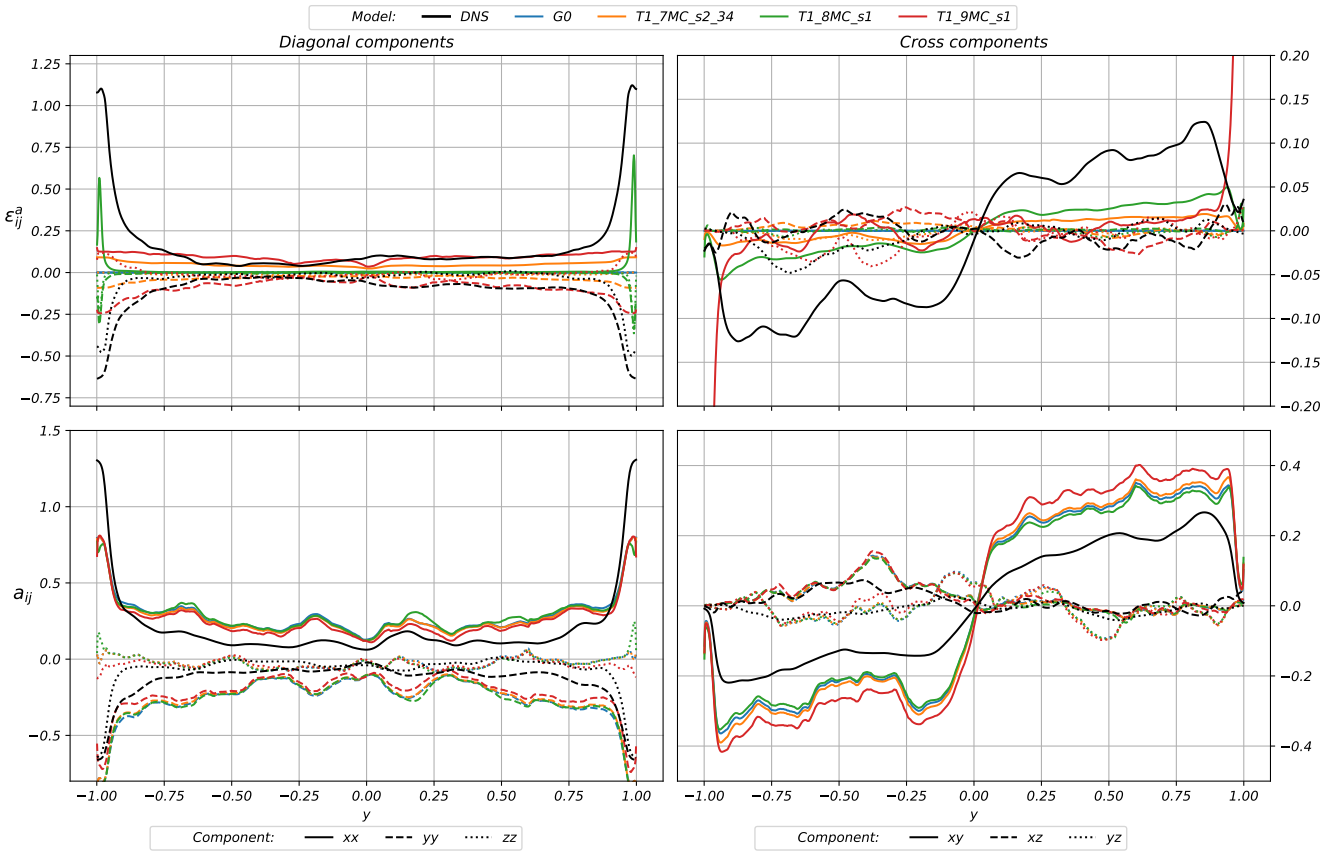


Figure 17. Comparison of a_{ij} and ε_{ij} profiles of models T2_7MC_s2_34, T1_8MC_s1 and T1_9MC_s1. G0 model is provided as reference.

Lastly, for the same selected models, their model expressions are shown as follows:

$$\text{T2\_7MC\_s2\_34} = \mathbf{T}^{(2)}\frac{0.046}{II_\Omega} + (\mathbf{T}^{(3)} + \mathbf{T}^{(4)})\nu^* + \mathbf{T}^{(5)}\frac{\nu^*}{II_S} + \mathbf{T}^{(6)}\frac{0.046}{II_S - V} + \mathbf{T}^{(7)}\frac{0.046}{V} + \mathbf{T}^{(10)}\frac{\nu^*}{\Delta^* - 2V},$$

$$\text{T1\_8MC\_s1} = 0.00066\left((\mathbf{T}^{(3)} + \frac{\mathbf{T}^{(4)}}{II_\Omega} + \mathbf{T}^{(5)} + \frac{\mathbf{T}^{(7)}}{II_\Omega} + \frac{\mathbf{T}^{(9)}}{\Delta^* II_S} + \frac{\mathbf{T}^{(10)}}{II_S}\right) + \frac{0.143\mathbf{T}^{(6)}}{\Delta^* - V} + \mathbf{T}^{(8)}\frac{\nu^*}{-II_S + V + \mathcal{P}/\varepsilon}$$

$$\text{T1\_9MC\_s1} = -0.014\mathbf{T}^{(1)} + \frac{0.362\mathbf{T}^{(2)}}{II_\Omega} - \frac{\nu^*\mathbf{T}^{(4)}}{II_S + 0.784} + \frac{\nu^*\mathbf{T}^{(6)}}{II_S - \mathcal{P}/\varepsilon} - \frac{0.179\mathbf{T}^{(8)}}{V} - \frac{0.188\mathbf{T}^{(9)}}{V}$$
$$\mathbf{T}^{(10)}\frac{0.566}{-II_S + II_\Omega - III + IV + \frac{IV}{\mathcal{P}/\varepsilon} - 8.64\frac{II_\Omega\nu^* - V(0.393II_\Omega - \mathcal{P}/\varepsilon)}{\nu^*(0.393II_\Omega - \mathcal{P}/\varepsilon)} + \frac{\mathcal{P}/\varepsilon}{III}}$$

We note the frequent occurrence of the ν^*, II_S and II_Ω in the model expressions. Further analysis on the expressions themselves might provide additional insight that could lead to the even better models. This however, must be put off as the subject of future work due to the limited scope of this work.

B. Tensor model discussion

To investigate some potential causes of the unsatisfactory results that have been presented, we first review the well known Rotta's model, where the dissipation anisotropy is modelled as $\varepsilon^a_{ij} = a_{ij}$. We replace the ansatz $\varepsilon^a_{ij} = \sum_n^{10}\gamma^{(n)}\mathbf{T}^{\mathbf{n}}$ with the ansatz for a_{ij} in Equation 15. This results in a slightly modified version of Equation 16, which reads:

$$\boldsymbol{\beta} = \frac{1}{\theta_3}(\boldsymbol{Q}\boldsymbol{\beta} - \boldsymbol{\beta} - \boldsymbol{\phi}) \tag{42}$$

This equation is solved in a similar fashion to that had been explained in section III, to obtain explicit expressions for $\beta^{(n)}$.

Table 15. MAE in ε^a_{ij} of models shown per training case and averaged across training cases and the 6 independent tensor components.

Model identifier	HIT			TCF_700			Average MAE
	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	
Rotta	0.309	0.268	0.275	0.462	0.277	0.219	0.302
G0	0.081	0.046	0.032	0.215	0.171	0.151	0.116

Table 16. MAE in a_{ij} of models shown per training case and averaged across training cases and the 6 independent tensor components.

Model identifier	HIT			TCF_700			Average MAE
	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	
Rotta	0.274	0.224	0.230	0.451	0.271	0.213	0.277
G0	0.344	0.347	0.374	0.350	0.308	0.269	0.332

Errors of Rotta's model are shown in Table 15 and Table 16. Relative to all other models including G0, it shows the lowest error in a_{ij} whilst for ε^a_{ij}, all of the best models within each training configuration tabulated in Table 13 performs better. One could argue that since a_{ij} has a more direct influence on flow quantities that are of interest in practical applications, that Rotta's model outperforms all current models. We now turn to examine the tensor components shown in Figure 18.

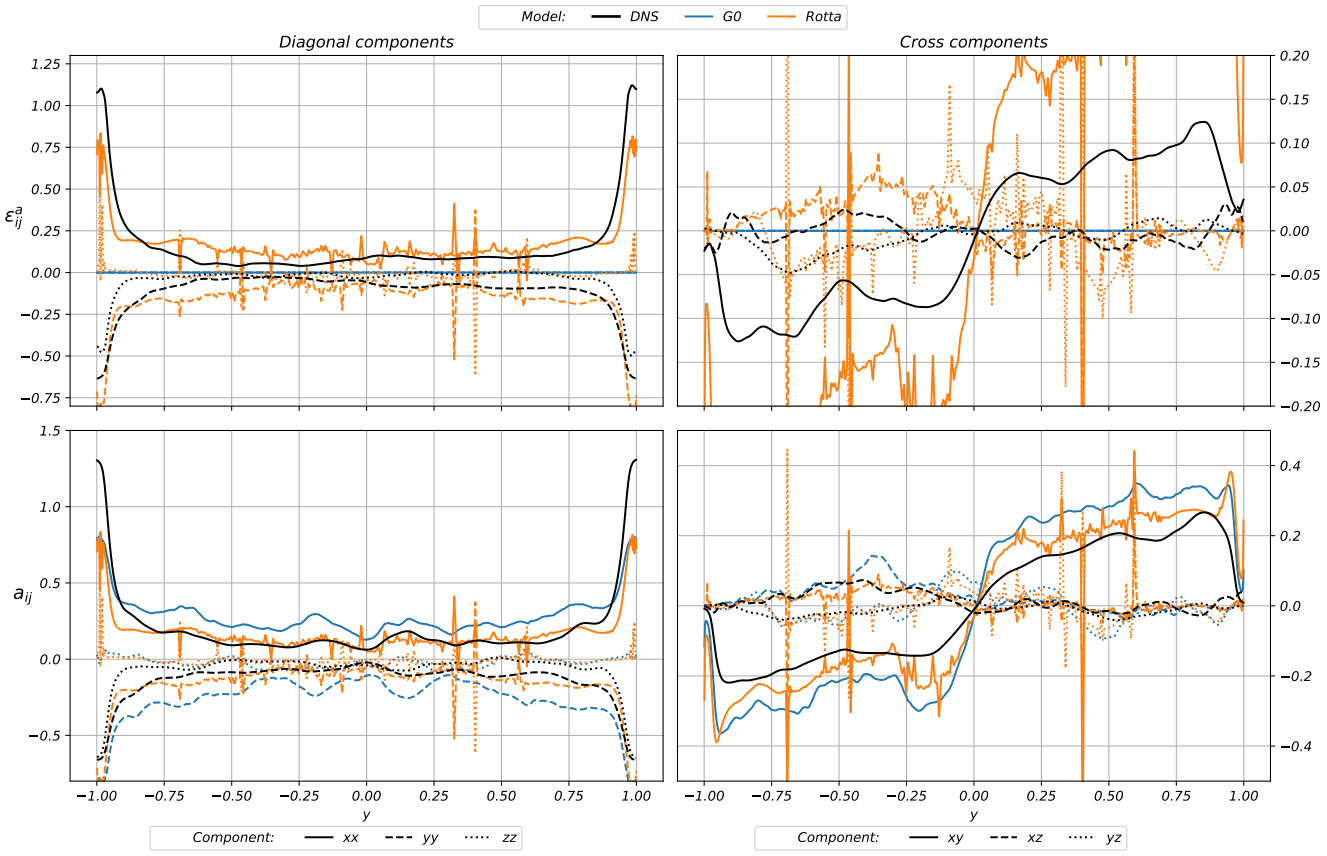


Figure 18. Tensor components of a_{ij} and ε_{ij} produced by Rotta's model. G0 model is provided as reference.

Note no $\beta^{(n)} = 0$ treatment has been applied to Rotta's model here. The figure clearly shows numerical instabilities in the model. This again is likely attributed to the $\beta^{(n)}$ expressions being undefined when the denominator of the expression approaches 0. An attempt was made to set the $\beta^{(n)}$ expressions to 0 as their respective denominators approaches 0, but it had little effect. Apart from the numerical instabilities, there is good agreement in both $\varepsilon^a_{xx,yy}$ and $a_{xx,yy}$. The model fails to capture the decreasing trend of ε^a_{zz} and a_{zz} towards channel walls. All cross components of a_{ij} are reasonably well reproduced. There is however, a large deviation in the agreement of ε^a_{xy}. Based on model error alone, no model proposed in the present work outperforms Rotta's model, although how the observed numerical instabilities affects an actual LES simulation remains to be investigated in future work.

Here we provide four areas of interest that could potentially lead to better models produced under the framework of this project or to better grasp the cause of no satisfactory results being obtained thus far. First, the author recognizes that the amount of data (1,0.5%) used in the training process is a low percentage. Efficiency of the code that is used, relative to other well established GEP symbolic regression algorithms is unknown. This could have contributed to the lengthy model training process, as a results, it was chosen to prioritize the investigation of different training configurations over a single one with a large percentage of data. This is partially supported by the results of the scalar model discovery.

Secondly, upon further review on the background of EASSM and the model for the pressure-strain term, we find that Hickel et al. (2019) mentioned that to a certain extent, the LRR-QI model contains the effect of the dissipation tensor, the suggestion is that the LRR-QI model should read as

$$\begin{aligned}\Pi_{ij} - \varepsilon^d_{ij} = &- C_R \varepsilon a_{ij} + C_1 k_{sgs} \bar{S}_{ij} \\ &+ \frac{3}{11}(2 + 3C_2) k_{sgs} \left(\bar{S}_{ik} a_{kj} + a_{ik} \bar{S}_{kj} - \frac{2}{3} \bar{S}_{kl} a_{lk} \delta_{ij} \right) \\ &+ \frac{1}{11}(10 - 7C_2) k_{sgs} \left(\bar{\Omega}_{ik} a_{kj} - a_{ik} \bar{\Omega}_{kj} \right),\end{aligned} \tag{43}$$

instead of that shown in Equation 6. The difference is in the deviatoric part of the dissipation tensor appearing on the LHS of the equation. There has been some confusion around this since Launder et al. (1975) does not explicitly mention this and neither does Marstorp et al. (2009) and Wallin and Johansson (2000). Whereas Girimaji (1996) applies a model in a similar fashion to Equation 43. The Π_{ij} term and LHS of Equation 43, along with the LRR-QI model as shown in Equation 6, are plotted in Figure 19. We observe that the LRR-QI model is in fact in better agreement with $\Pi_{ij} - \varepsilon^d_{ij}$ than Π_{ij} alone, indicating that the LRR-QI model does to some extent also models the effect of ε^d_{ij}. Therefore, with a type of pressure-strain term model which also partially models the effect of dissipation, the transport equation for a_{ij} should reads as,

$$\left(a_{ij} + \frac{2}{3}\delta_{ij}\right)(\mathcal{P} - \varepsilon) = \mathcal{P}_{ij} + \Pi^*_{ij} + (\Pi^* - \Pi^{*,model}_{ij}) \,, \quad \Pi^*_{ij} = \Pi_{ij} - \varepsilon^d_{ij} \tag{44}$$

and to account for the effect of dissipation that is unaccounted for in the Π_{ij} model, an ansatz should be proposed for the last term in the equation instead.

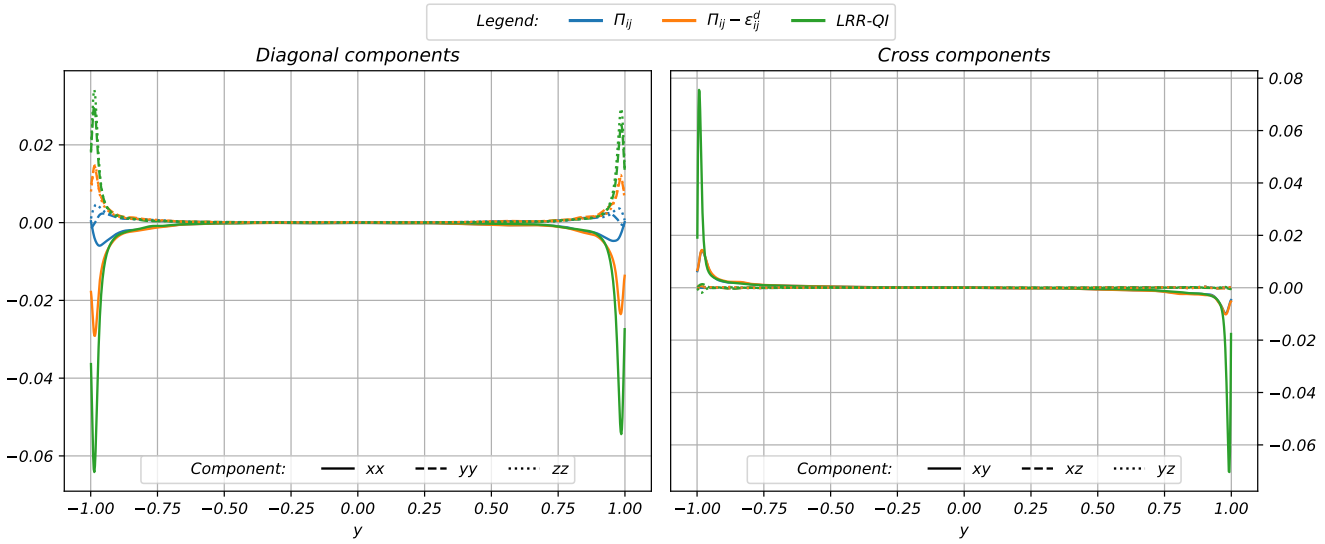


Figure 19. Comparison of tensor components of Π_{ij} and $\Pi_{ij} - \varepsilon^d_{ij}$ from DNS, and the LRR-QI model.

Alternatively, a model for Π_{ij} that dose not contain the effects of dissipation could be applied and thee framework remains largely unchanged. If there can be no clear demarcation between the pressure-strain and dissipation terms, future work may consider the possibility of model discovery on both terms in parallel, or both terms lumped. The modelling of Π_{ij} has effect only on the present work for models that are trained with a_{ij} included in the loss function and not otherwise.

Thirdly, we note that different timescales have been used in the non-dimensionalization of the basis tensors. The present work follows Gnanasundaram (2018) and Marstorp et al. (2009) in their use of k_{sgs}/ε while Hasslberger et al. (2025) and Reissmann et al. (2024) uses $||S||^{-1}$. Figure 20 show the different timescales with different exponents. It can be seen that k_{sgs}/ε produces the sharp spikes in vicinity of channel walls that have also be observed in the trained models. The timescale $||S||^{-1}$ produces smoother variation near channel walls, however it also produces some sharp peaks in channel centre regions when raised to larger exponents. We question whether the choice of $||S||^{-1}$ as a timescale could lead to improved results that produces better tensorial alignment of the model close to channel walls. That said, the use of $||S||^{-1}$ for non-dimensionalization does not eliminate the presence of k_{sgs}/ε in the a_{ij} transport equation. Instead of it being absorbed into the strain rate tensor as $\bar{S}^*_{ij} = S_{ij}\frac{k_{sgs}}{\varepsilon}$, it will appear within the θ_3 term in Equation 16.

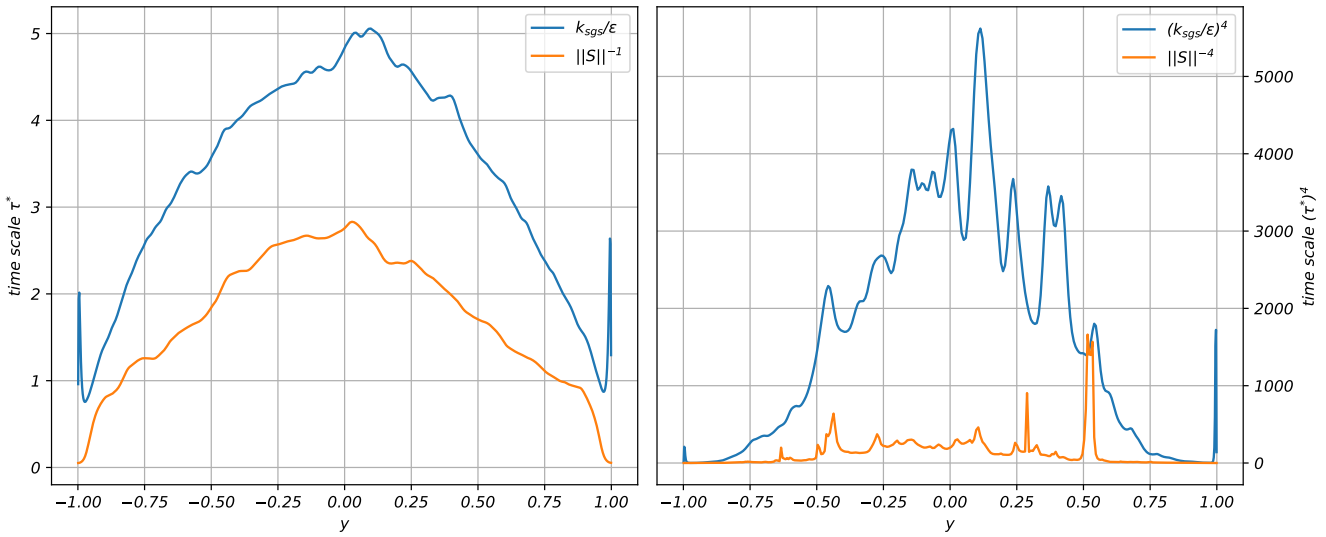


Figure 20. Comparison of tensor components of Π_{ij} and $\Pi_{ij} - \varepsilon^d_{ij}$ from DNS, and the LRR-QI model.

Fourthly, the present work employs an ad-hoc numerical handling of $\beta^{(n)}$ under the condition stated in Equation 37. This is in a mathematical perspective inappropriate, one should investigate whether GEP can regress a model such that the θ_3 terms in the denominator of the $\beta^{(n)}$ expression cancels out, this would yield a model that is consistent throughout a flowfield.

VII. Conclusion

The overarching aim of the present work is to discover a closure model for the SGS stress dissipation tensor that is applicable under the EASSM framework, using GEP. Various features both on the EASSM framework and GEP algorithm, proposed by other authors, have been incorporated into this study. On the framework of EASSM, the assumptions of two-dimensional turbulence and equilibrium between SGS stress production and dissipation were discarded. The former lead to a reduced number of basis tensors in the EASSM framework while the latter further simplifies the transport equation of a_{ij}. The assumptions had been neglected in other works independently but not in conjunction. Regarding the GEP algorithm, the addition of wall scaling penalty terms and numerical constant optimization were inspired by other works and incorporated in the present work. Furthermore, a mix of training data cases had been applied in this work in attempt to discover a model that remains robust in different flow cases. The results are separated into the discovery of the scalar dissipation model and dissipation anisotropy tensor model.

On the discovered scalar dissipation models, all models produced a dissipation spectrum with improved agreement with DNS data, relative to the default model, for the HIT cases. For the TCF cases, all models but S1_s3 outperformed the default model. Amongst these models, S2_s3 and S3_s2 were shown to best match the gradient of dissipation near channel walls. Model S3_s2 which is trained on both HIT and TCF data also proved to perform better as opposed to other models trained on only HIT or TCF cases (amount of data remains the same). A-posteriori validation was conducted for the S2_s3 model on turbulent channel flow at Re_τ. Initial results with default constants provided by the OpenFoam solver lead to unrealistic stress profiles, a drastic reduction in the constant values eventually yield sensible results which firstly, demonstrated the stability of the model. Secondly, showed improvements relative to the default model. For any practical application of the model, it remains to be tested on a far wider range of flow cases to ascertain it's robustness. Future work could also be conducted by substituting the HIT and TCF data for more more complex flow cases such as the Taylor-Green-Vortex or backward facing step, which contains features like laminar-to-turbulent transition and flow separation.

Proceeding to the tensor model discovery, initial results showed large numerical instabilities which was discovered to be a mathematical singularity in the expressions of $\beta^{(n)}$. This was remedied with an ad-hoc numerical handling by setting $\beta^{(n)} = 0$ under the singular condition. Subsequent model training showed that the addition of the terminal

ν^* yielded models with improved results. From the large variation in error of models that are trained with the same configuration but initialized with different initial populations, it was deduced that a significant increase in the population size was necessary to obtain more consistent models within each training configuration. Most models that have been discovered possess a number of undesirable traits such as sharp spikes or a drastic increase to large magnitudes, both in vicinity of channel walls. The best performing model. T2_7MC_s2_34 is marginally outperformed by the G0 reference model in a_{ij}, but provides some tensorial alignment in ε^a_{ij}, leading to a lower error in ε^a_{ij}. Whether the alignment in ε^a_{ij} can effect improved better alignment in a_{ij} in an actual LES simulation remains to be tested.

The tensor model results are rather unsatisfactory, with no model able to show consistent improvements in both ε^a_{ij} and a_{ij} relative to the G0 model. Regarding this, some retrospective review had been presented in this work. We would like to reiterate the most important point is related to the pressure strain model. Literature suggest that the LRR-QI model contains, partially, the effect of dissipation. In this scenario, the use of error in both ε^a_{ij} and a_{ij} becomes problematic. It depends on whether the effect of dissipation can be separated from the pressure-strain model in order to determine whether it is sensible to proposed a model for the dissipation tensor under the EASSM framework. We specify the framework since the ultimate aim is to produce a set of expressions for a_{ij} and not ε^a_{ij}, but one can also conduct a model search purely to gain insight into the physical mechanism of SGS stress dissipation. Under the EASSM framework, two directions seem available, first, retain the pressure-strain model and discover a closure model for the dissipation effect that is unaccounted for in the pressure strain model. The other is to proposed a combined model for the pressure-strain and dissipation term in conjunction.

Acknowledgments

First and foremost, the author would like to extend his gratitude to professor S.Hickel for his suggestion of this thesis topic and subsequently his patience and guidance throughout the process. Thanks also go out to; A.Chakrabory who provided the channel flow data used in this work; staff members in the aerodynamic department who gave their insight in the validation of the scalar model. Last, but by no means any lower in priority, thank you to family and friends who offered their support in challenging times.

References

Beck, A., & Kurz, M. (2021). A perspective on machine learning methods in turbulence modeling. *GAMM-Mitteilungen*, *44*(1). https://doi.org/10.1002/gamm.202100002

Brunton, S. L., & Kutz, J. N. (2019). Data-driven dynamical systems. In *Data-driven science and engineering: Machine learning, dynamical systems, and control* (pp. 229–275). Cambridge University Press.

Chumakov, S. G. (2005). *Subgrid models for large eddy simulation: Scalar flux, scalar dissipation and energy dissipation* [PhD dissertation]. University of Wisconsin-Madison. https://chumakov.info/data/papers/Chumakov_PhD_thesis.pdf

De Villiers, E. (2006). *The potential of large eddy simulation for the modeling of wall bounded flows* [PhD thesis]. Imperial College of Science, Technology and Medicine, London.

Deardorff, J. W. (1970). A numerical study of three-dimensional turbulent channel flow at large reynolds numbers. *Journal of Fluid Mechanics*, *41*(2), 453–480. https://doi.org/10.1017/S0022112070000691

Duraisamy, K., Iaccarino, G., & Xiao, H. (2019). Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, *51*(1), 357–377. https://doi.org/10.1146/annurev-fluid-010518-040547

Ferreira, C. (2006). *Gene expression programming: Mathematical modeling by an artificial intelligence* (Vol. 21). Springer. https://doi.org/10.1007/3-540-32849-1

Fortin, F.-A., Rainville, F.-M.-D., Gardner, M.-A., Parizeau, M., & Gagné, C. (2012). DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, *13*, 2171–2175. http://jmlr.org/papers/v13/fortin12a.html

Gamahara, M., & Hattori, Y. (2017). Searching for turbulence models by artificial neural network. *Physical Review Fluids*, *2*(5). https://doi.org/10.1103/physrevfluids.2.054604

Gao, S., Sun, C., Xiang, C., Qin, K., & Lee, T. H. (2022). Learning Asynchronous Boolean Networks From Single-Cell Data Using Multiobjective Cooperative Genetic Programming. *IEEE Transactions on Cybernetics*, *52*(5), 2916–2930. https://doi.org/10.1109/TCYB.2020.3022430

Gatski, T. B., & Speziale, C. G. (1993). On explicit algebraic stress models for complex turbulent flows. *Journal of Fluid Mechanics*, *254*, 59–78. https://doi.org/10.1017/S0022112093002034

Gerolymos, G. A., & Vallet, I. (2016). The dissipation tensor in wall turbulence. *Journal of Fluid Mechanics*, *807*, 386–418. https://doi.org/10.1017/jfm.2016.610

Girimaji, S. S. (1996). Fully explicit and self-consistent algebraic reynolds stress model. *Theoretical and Computational Fluid Dynamics*, *8*(6), 387–402. https://doi.org/10.1007/BF00455991

Gnanasundaram, A. K. (2018, May). *Explicit algebraic subgrid-scale stress model for homogeneous turbulence* [MSc Thesis]. Delft University of Technology.

Hallbäck, M., Groth, J., & Johansson, A. V. (1990). An algebraic model for nonisotropic turbulent dissipation rate in reynolds stress closures. *Physics of Fluids A: Fluid Dynamics*, *2*(10), 1859–1866. https://doi.org/10.1063/1.857660

Hanjalić, K., & Launder, B. E. (1976). Contribution towards a reynolds-stress closure for low-reynolds-number turbulence. *Journal of Fluid Mechanics*, *74*(4), 593–610. https://doi.org/10.1017/S0022112076001961

Hasslberger, J., Reissmann, M., Sandberg, R. D., & Klein, M. (2025). Exploiting semantic backpropagation to impose near-wall scaling constraints in machine-learned symbolic subgrid-scale closures. *Physics of Fluids*, *37*(3), 035203. https://doi.org/10.1063/5.0260447

Hickel, S., Gnanasundaram, A. K., & Pestana, T. C. S. (2019). A one-equation explicit algebraic subgrid-scale stress model. *Proceedings of the 11th International Symposium on Turbulence and Shear Flow Phenomena (TSFP11)*. https://api.semanticscholar.org/CorpusID:207985653

Inagaki, K., & Kobayashi, H. (2023). Analysis of anisotropic subgrid-scale stress for coarse large-eddy simulation. *Physical Review Fluids*, *8*(10). https://doi.org/10.1103/physrevfluids.8.104603

Kasten, C., Shin, J., Sandberg, R. D., Pfitzner, M., Chakraborty, N., & Klein, M. (2022). Modelling subgrid-scale scalar dissipation rate in turbulent premixed flames using gene expression programming and deep artificial neural networks. *Physics of Fluids*. https://api.semanticscholar.org/CorpusID:250509912

Kim, S., Lu, P. Y., Mukherjee, S., Gilbert, M., Jing, L., Ceperic, V., & Soljacic, M. (2021). Integration of neural network-based symbolic regression in deep learning for scientific discovery. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(9), 4166–4177. https://doi.org/10.1109/tnnls.2020.3017010

Launder, B. E., Reece, G. J., & Rodi, W. (1975). Progress in the development of a reynolds-stress turbulence closure. *Journal of Fluid Mechanics*, *68*(3), 537–566. https://doi.org/10.1017/S0022112075001814

Lav, C., Banko, A. J., Waschkowski, F., Zhao, Y., Elkins, C. J., Eaton, J. K., & Sandberg, R. D. (2023). A coupled framework for symbolic turbulence models from deep-learning. *International Journal of Heat and Fluid Flow*, *101*, 109140. https://doi.org/https://doi.org/10.1016/j.ijheatfluidflow.2023.109140

Lee, M. J., & Reynolds, W. C. (1985). On the structure of homogeneous turbulence. In F. Durst, B. E. Launder, J. L. Lumley, F. W. Schmidt, & J. H. Whitelaw (Eds.), *Turbulent shear flows 5* (pp. 54–66). Springer. https://doi.org/10.1007/978-3-642-71435-1_6

Li, H., Waschkowski, F., Zhao, Y., & Sandberg, R. D. (2023). Turbulence model development based on a novel method combining gene expression programming with an artificial neural network. https://arxiv.org/abs/2301.07293

Maplesoft. (2025). *Maple* [Version 2025.1, Waterloo Maple Inc.]. https://www.maplesoft.com/products/Maple/

Marstorp, L., Brethouwer, G., Grundestam, O., & Johansson, A. V. (2009). Explicit algebraic subgrid stress models with application to rotating channel flow. *Journal of Fluid Mechanics*, *639*, 403–432. https://doi.org/10.1017/S0022112009991054

McConaghy, T. (2011). Ffx: Fast, scalable, deterministic symbolic regression technology. In R. Riolo, E. Vladislavleva, & J. H. Moore (Eds.), *Genetic programming theory and practice ix* (pp. 235–260). Springer New York. https://doi.org/10.1007/978-1-4614-1770-5_13

Montecchia, M., Brethouwer, G., Johansson, A. V., & Wallin, S. (2017). Taking large-eddy simulation of wall-bounded flows to higher reynolds numbers by use of anisotropy-resolving subgrid models. *Phys. Rev. Fluids*, *2*, 034601. https://doi.org/10.1103/PhysRevFluids.2.034601

Mundhenk, T. N., Landajuela, M., Glatt, R., Santiago, C. P., Faissol, D. M., & Petersen, B. K. (2021). Symbolic regression via neural-guided genetic programming population seeding. *ArXiv*, *abs/2111.00053*. https://api.semanticscholar.org/CorpusID:240354126

OpenCFD Ltd. (2025). K equation (les) — openfoam user guide [Accessed: 2025-11-24].

Perot, B., & Natu, S. (2004). A model for the dissipation rate tensor in inhomogeneous and anisotropic turbulence. *Physics of Fluids*, *16*(11), 4053–4065. https://doi.org/10.1063/1.1801392

Pestana, T., & Hickel, S. (2019). Regime transition in the energy cascade of rotating turbulence. *Phys. Rev. E*, *99*, 053103. https://doi.org/10.1103/PhysRevE.99.053103

Pope, S. B. (1975). A more general effective-viscosity hypothesis. *Journal of Fluid Mechanics*, *72*(2), 331–340. https://doi.org/10.1017/S0022112075003382

Pouransari, Z., & Rasam, A. (2023). Non-equilibrium extension of the explicit algebraic subgrid-scale stress model with application to turbulent channel flow at low reynolds numbers. *Physics of Fluids*, *35*(3), 035131. https://doi.org/10.1063/5.0141650

Rasam, A., Brethouwer, G., Schlatter, P., Li, Q., & Johansson, A. V. (2011). Effects of modelling, resolution and anisotropy of subgrid-scales on large eddy simulations of channel flow. *Journal of Turbulence*, *12*, N10. https://doi.org/10.1080/14685248.2010.541920

Reissmann, M., Fang, Y., Ooi, A., & Sandberg, R. (2024). Constraining genetic symbolic regression via semantic backpropagation. https://arxiv.org/abs/2409.07369

Reissmann, M., Hasslberger, J., Sandberg, R. D., & Klein, M. (2021). Application of gene expression programming to a-posteriori les modeling of a taylor green vortex. *Journal of Computational Physics*, *424*, 109859. https://doi.org/https://doi.org/10.1016/j.jcp.2020.109859

Rotta, J. C. (1951). Statistische theorie nichthomogener turbulenz. *Zeitschrift für Physik*, *131*, 51–77. https://api.semanticscholar.org/CorpusID:123243529

Schmelzer, M., Dwight, R. P., & Cinnella, P. (2019). Discovery of algebraic reynolds-stress models using sparse symbolic regression. *Flow, Turbulence and Combustion*, *104*(2–3), 579–603. https://doi.org/10.1007/s10494-019-00089-x

Schoepplein, M., Weatheritt, J., Sandberg, R., Talei, M., & Klein, M. (2018). Application of an evolutionary algorithm to les modelling of turbulent transport in premixed flames. *Journal of Computational Physics*, *374*, 1166–1179. https://doi.org/https://doi.org/10.1016/j.jcp.2018.08.016

Speziale, C. G., & Gatski, T. B. (1997). Analysis and modelling of anisotropies in the dissipation rate of turbulence. *Journal of Fluid Mechanics*, *344*, 155–180. https://doi.org/10.1017/S002211209700596X

Wallin, S., & Johansson, A. V. (2000). An explicit algebraic reynolds stress model for incompressible and compressible turbulent flows. *Journal of Fluid Mechanics*, *403*, 89–132. https://doi.org/10.1017/S0022112099007004

Waschkowski, F., Li, H., Deshmukh, A., Grenga, T., Zhao, Y., Pitsch, H., Klewicki, J., & Sandberg, R. D. (2022). Gradient information and regularization for gene expression programming to develop data-driven physics closure models. https://arxiv.org/abs/2211.12341

Weatheritt, J., & Sandberg, R. D. (2017). The development of algebraic stress models using a novel evolutionary algorithm. *International Journal of Heat and Fluid Flow*, *68*, 298–318. https://doi.org/10.1016/j.ijheatfluidflow.2017.09.017

Yoshizawa, A., & Horiuti, K. (1985). A statistically-derived subgrid-scale kinetic energy model for the large-eddy simulation of turbulent flows. *Journal of the Physical Society of Japan*, *54*(8), 2834–2839. https://doi.org/10.1143/JPSJ.54.2834

A. Appendix Cayley-Hamilton

The following matrices are used in Equation 11 to express $\bar{\mathbf{S}}^*\mathbf{T}^{(n)} + \mathbf{T}^{(n)}\bar{\mathbf{S}}^*$ and $\mathbf{T}^{(n)}\bar{\mathbf{\Omega}}^* - \bar{\mathbf{\Omega}}^*\mathbf{T}^{(n)}$ in terms of the basis tensors. They are directly quoted from Speziale and Gatski (1997).

$$H_{n,\xi} = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{3}II_S & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{2}{3}II_\Omega & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{2}II_S & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{2}{3}IV & 0 & 2II_\Omega & II_S & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & -IV & 0 & 0 & II_\Omega & 0 & 0 & 0 & 0 & -2 \\ 0 & \frac{1}{3}III & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{3}V & 0 & IV & \frac{1}{3}III & 0 & \frac{1}{2}II_S & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{3}V - \frac{1}{6}II_SII_\Omega & 0 & 0 & \frac{2}{3}IV & 0 & -\frac{1}{3}II_S & \frac{1}{3}II_\Omega & 0 & 0 \end{bmatrix} \tag{45}$$

$$J_{n,\xi} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -II_\Omega & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & II_\Omega & 2II_S & 0 & 0 & 0 & 0 & -3 & 0 \\ 0 & \frac{1}{2}II_\Omega & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ II_\Omega^2 & 0 & 0 & -2IV & 0 & -2II_\Omega & 0 & 0 & 0 & 0 \\ 2V - II_SII_\Omega & 0 & -2IV & 0 & 0 & II_S & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-1}{2}II_\Omega & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & II_\Omega^2 & 2II_SII_\Omega - 2V & 0 & 0 & 0 & 0 & -2II_\Omega & 0 \end{bmatrix} \tag{46}$$

Fully expanded implicit expressions of $\beta^{(n)}$ from Equation 11

$$
\begin{aligned}
\beta^{(1)} &= \left[\left(C_1 - \frac{4}{3}\right) + \left[\frac{1}{3}\beta^{(3)} II_S - \frac{2}{3}\beta^{(4)} II_\Omega + \frac{2}{3}\beta^{(6)} IV - \frac{1}{3}\beta^{(9)} V\right] \left(\frac{9C_2 - 5}{11}\right) \right. \\
&\quad \left. + \left[-\beta^{(2)} II_\Omega + \beta^{(7)} II_\Omega^2 + \beta^{(8)}(2V - II_S II_\Omega)\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(1)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(2)} &= \left[\left[-\frac{1}{2}\beta^{(5)} II_S - \beta^{(7)} IV + \frac{1}{3}\beta^{(8)} III + \beta^{(10)} \left(-\frac{1}{3}V - \frac{1}{6} II_S II_\Omega\right)\right] \left(\frac{9C_2 - 5}{11}\right) \right. \\
&\quad \left. + \left[\beta^{(1)} + \frac{1}{2}\beta^{(6)} II_\Omega\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(2)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(3)} &= \left[\left[2\beta^{(1)} + 2\beta^{(6)} II_\Omega + \beta^{(9)} IV\right] \left(\frac{9C_2 - 5}{11}\right) \right. \\
&\quad \left. + \left[\beta^{(5)} II_\Omega - 2\beta^{(8)} IV + \beta^{(10)} II_\Omega^2\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(3)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(4)} &= \left[\left[\beta^{(6)} II_S + \frac{1}{3}\beta^{(9)} III\right] \left(\frac{9C_2 - 5}{11}\right) \right. \\
&\quad \left. + \left[2\beta^{(5)} II_S - 2\beta^{(7)} IV + \beta^{(10)}(2 II_S II_\Omega - 2V)\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(4)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(5)} &= \left[\left[-\beta^{(2)} + \beta^{(7)} II_\Omega + \frac{2}{3}\beta^{(10)} IV\right] \left(\frac{9C_2 - 5}{11}\right) \right. \\
&\quad \left. + \left[-\beta^{(3)} - \frac{1}{2}\beta^{(9)} II_\Omega\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(5)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(6)} &= \left[\left[\beta^{(4)} + \frac{1}{2}\beta^{(9)} II_S\right] \left(\frac{9C_2 - 5}{11}\right) \right. \\
&\quad \left. + \left[3\beta^{(2)} - 2\beta^{(7)} II_\Omega + \beta^{(8)} II_S\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(6)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(7)} &= \left[\left[-\frac{1}{3}\beta^{(10)} II_S\right] \left(\frac{9C_2 - 5}{11}\right) + \left[-\beta^{(6)}\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(7)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(8)} &= \left[\left[\beta^{(5)} + \frac{1}{3}\beta^{(10)} II_\Omega\right] \left(\frac{9C_2 - 5}{11}\right) + [0] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(8)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(9)} &= \left[\left[-\beta^{(6)}\right] \left(\frac{9C_2 - 5}{11}\right) \quad + \left[-3\beta^{(5)} - 2\beta^{(10)} II_\Omega\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(9)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right) \\
\beta^{(10)} &= \left[\left[-2\beta^{(7)}\right] \left(\frac{9C_2 - 5}{11}\right) + \left[-\beta^{(9)}\right] \left(\frac{1 + 7C_2}{11}\right) - \gamma^{(10)} \right] / \left(\frac{\mathcal{P}}{\varepsilon} - 1 + C_R\right)
\end{aligned}
\tag{47}
$$

B. Appendix Explicit Beta Expressions

$$
\begin{aligned}
\beta^{(1)} = & \Bigg[2187\left(-4+3C_1-3\gamma^{(1)}\right)\theta_3^3+2916\left(-II_\Omega^2\gamma^{(7)}+\left(II_S\gamma^{(8)}+\gamma^{(2)}\right)II_\Omega-2V\gamma^{(8)}\right)\theta_3^2 \\
& -4536\left(-\frac{3II_\Omega\gamma^{(6)}}{7}+C_1-\gamma^{(1)}-\frac{4}{3}\right)II_\Omega\theta_3+288II_\Omega^3\gamma^{(7)} \\
& +576\left(-2II_S\gamma^{(8)}+\gamma^{(2)}\right)II_\Omega^2+4032II_\Omega V\gamma^{(8)}\Bigg]/ \\
& \left(6561\theta_3^4-3240II_\Omega\,\theta_3^2+256II_\Omega^2\right) \\
\beta^{(2)} = & \frac{81\theta_3^2\gamma^{(2)}+\left(18II_\Omega\gamma^{(6)}-36C_1+36\gamma^{(1)}+48\right)\theta_3+32\gamma^{(8)}\left(-\frac{II_SII_\Omega}{4}+V\right)}{-81\theta_3^3+8II_\Omega\theta_3} \\
\beta^{(3)} = & \Bigg[288II_\Omega^3\gamma^{(10)}+\left(-2916\theta_3^2\gamma^{(10)}+1944\theta_3\gamma^{(9)}-576\gamma^{(5)}\right)II_\Omega^2+ \\
& \left(-2916\theta_3^2\gamma^{(5)}-4032IV\gamma^{(8)}+4536\theta_3\gamma^{(3)}\right)II_\Omega+5832IV\,\theta_3^2\gamma^{(8)}-6561\theta_3^3\gamma^{(3)}\Bigg] \\
& /\left(6561\theta_3^4-3240II_\Omega\,\theta_3^2+256II_\Omega^2\right) \\
\beta^{(4)} = & \Bigg[-6561\gamma^{(4)}\theta_3^4+\left(-5832II_\Omega II_S\gamma^{(10)}-5832II_S\gamma^{(5)}+5832IV\gamma^{(7)}+5832V\gamma^{(10)}\right)\theta_3^3+ \\
& \left(\left(3888II_S\gamma^{(9)}+3240\gamma^{(4)}\right)II_\Omega-2592V\gamma^{(9)}-2592\gamma^{(6)}IV+2592II_S\gamma^{(3)}\right)\theta_3^2 \\
& +\Bigg(576\gamma^{(10)}II_\Omega^2II_S+\left(-1152II_S\gamma^{(5)}-576IV\gamma^{(7)}-576V\gamma^{(10)}\right)II_\Omega \\
& +\left(-3456II_S\gamma^{(8)}-3456\gamma^{(2)}\right)IV+3456V\gamma^{(5)}\Bigg)\theta_3-256II_\Omega^2\gamma^{(4)} \\
& +\left(512II_S\gamma^{(3)}-512\gamma^{(6)}IV-512V\gamma^{(9)}\right)II_\Omega+\left(1536C_1-1536\gamma^{(1)}-2048\right)IV-1536V\gamma^{(3)}\Bigg] \\
& /\left(6561\theta_3^5-3240II_\Omega\theta_3^3+256II_\Omega^2\theta_3\right) \\
\beta^{(5)} = & \frac{-18II_\Omega\theta_3\gamma^{(9)}+81\theta_3^2\gamma^{(5)}+32IV\gamma^{(8)}-36\theta_3\gamma^{(3)}}{\left(-81\theta_3^2+8II_\Omega\right)\theta_3} \\
\beta^{(6)} = & \Bigg[-6561\theta_3^3\gamma^{(6)}+\left(5832II_\Omega\gamma^{(7)}-2916II_S\gamma^{(8)}-8748\gamma^{(2)}\right)\theta_3^2 \\
& +\left(-1296II_\Omega\gamma^{(6)}+3888C_1-3888\gamma^{(1)}-5184\right)\theta_3+1152II_\Omega II_S\gamma^{(8)}-576II_\Omega^2\gamma^{(7)}-3456V\gamma^{(8)}\Bigg] \\
& /\left(6561\theta_3^4-3240II_\Omega\theta_3^2+256II_\Omega^2\right)
\end{aligned}
\tag{48}
$$

$$
\begin{aligned}
\beta^{(7)} &= \left[-6561\theta_3^4\gamma^{(7)} + 2916\theta_3^3\gamma^{(6)} + 648\left(II_\Omega\gamma^{(7)} + 2II_S\gamma^{(8)} + 6\gamma^{(2)}\right)\theta_3^2 + 576\left(II_\Omega\gamma^{(6)} - 3C_1 + 3\gamma^{(1)} + 4\right)\theta_3 \right. \\
&\quad \left. + 1536\gamma^{(8)}\left(-\frac{II_S II_\Omega}{3} + V\right)\right] / \left(6561\theta_3^5 - 3240II_\Omega\,\theta_3^3 + 256II_\Omega^2\theta_3\right) \\
\beta^{(8)} &= -\frac{\gamma^{(8)}}{\theta_3} \\
\beta^{(9)} &= \left[-6561\theta_3^3\gamma^{(9)} + \left(5832II_\Omega\gamma^{(10)} + 8748\gamma^{(5)}\right)\theta_3^2 + \left(-1296II_\Omega\gamma^{(9)} - 3888\gamma^{(3)}\right)\theta_3 - 576II_\Omega^2\gamma^{(10)} + 3456IV\gamma^{(8)}\right] \\
&\quad / \left(6561\theta_3^4 - 3240II_\Omega\,\theta_3^2 + 256II_\Omega^2\right) \\
\beta^{(10)} &= \left[-6561\theta_3^4\gamma^{(10)} + 2916\theta_3^3\gamma^{(9)} + 648\left(II_\Omega\gamma^{(10)} - 6\gamma^{(5)}\right)\theta_3^2 + 576\left(II_\Omega\gamma^{(9)} + 3\gamma^{(3)}\right)\theta_3 - 1536IV\gamma^{(8)}\right] \\
&\quad / \left(6561\theta_3^5 - 3240II_\Omega\theta_3^3 + 256II_\Omega^2\theta_3\right)
\end{aligned}
\tag{49}
$$

C. Appendix Numerical Constant Optimization Routine

Listing 1. Pseudocode for BFGS optimization within evolutionary loop

```
#global
best_constants=[1,2,3,4,5] #
best_fitness = 100
symbolic_constants=[C1,C2,C3,C4,C5]

def evaluate_fitness(individual, constants):
  ... #fitness calculation
  return fitness

def BFGS_optimizer(objective_function, initial_conditions):
  ... #Gradient based BFGS optimizer
  return optimized_fitness, optimized_constants

#beginning of each generation
if generation % 15 == 0: #optimize every 15 generations
  for individual in generation:
    #check if individual contains symbolic constants
    constants_in_individual = constants_in_individual
    if constants_in_individual == True:
      #use best_constants as initial guess to optimizer
      x0 = best_constants
      fitness, optimized_constants = BFSG_optimizer(evaluate_fitness, x0)
      if fitness<= best_fitness:
        #update value of constants and fitness
        best_fitness= fitness
        best_constants = optimized_constants
    else:
      #individual does not contain symbolic constants
      fitness = evaluate_fitness(individual, best_constants)
      if fitness<= best_fitness:
        best_fitness = fitness
else:
  for individual in generation:
    fitness = evaluate_fitness(individual, best_constants)
    if fitness<= best_fitness:
      best_fitness = fitness

new_generation = apply_genetic_operations(generation)
```

Table 17. RMAE of best model for each training configuration. Colour scale is applied per column with blue being the best and red being the worse.

Identifier	T15 f9	T15 f17	T15 f25	T7 f9	T7 f17	T7 f25	H f9	H f17	H f25
S1	0.6739	0.5180	0.4671	0.5628	0.4732	0.4474	0.3665	0.3293	0.3058
S2	0.5438	0.4260	0.3769	0.5191	0.3962	0.3516	0.4575	0.4810	0.4867
S3	0.5848	0.4679	0.4200	0.5334	0.4297	0.4066	0.3325	0.3234	0.3177
S4	0.5670	0.4683	0.4214	0.5329	0.4302	0.3974	0.3540	0.3255	0.3058
S5	0.5564	0.4671	0.4327	0.5250	0.4382	0.4150	0.3792	0.3425	0.3156

Table 18. Correlation coefficient of best model for each training configuration. Colour scale is applied per column with blue being the best and red being the worse.

Identifier	T15 f9	T15 f17	T15 f25	T7 f9	T7 f17	T7 f25	H f9	H f17	H f25
S1	0.8706	0.9185	0.9269	0.7573	0.8152	0.8468	0.7528	0.6985	0.6380
S2	0.8996	0.9363	0.9469	0.7890	0.8542	0.8707	0.8031	0.7549	0.7147
S3	0.8777	0.9215	0.9438	0.7485	0.8557	0.8915	0.8189	0.7744	0.7381
S4	0.9062	0.9269	0.9349	0.7633	0.8304	0.8602	0.7789	0.7261	0.6783
S5	0.9084	0.9264	0.9342	0.7659	0.8298	0.8592	0.7479	0.6985	0.6459

D. Appendix Scalar dissipation Model Expressions & Performance

Table 19. Model expressions and constants of all scalar dissipation models

Identifier	Model expressions	Constants
S1_s1	$\frac{(C_3-1)(C_3C_5+C_2)k_{sgs}^{\frac{3}{2}}-\Delta\|\|S\|\|^2\nu(C_2+C_4+IV)}{\Delta}$	[-2.1233, 1.0124, 0.0873, 36.3387]
S1_s2	$C_1\left(1+\frac{C_2k_{sgs}}{\nu\|\|S\|\|^2\left(\frac{C_1\Delta}{\sqrt{k_{sgs}}}-III^2(2C_3+1)\left(\frac{\Delta(III+V)}{\sqrt{k_{sgs}}}-\frac{\nu}{k_{sgs}}\right)\right)}\right)\nu\|\|S\|\|^2$	[0.7762, 0.5028, 0.7593]
S1_s3	$\frac{\|\|S\|\|\left(C_5(C_3+III)k_{sgs}^{\frac{3}{2}}+C_4\nu\|\|S\|\|(C_3+C_4+2III)\sqrt{k_{sgs}}+\Delta\|\|S\|\|^2\nu C_4\right)}{(C_3+C_4+2III)\sqrt{k_{sgs}}+\Delta\|\|S\|\|}$	[0.8331, -0.1684, 0.8132]
S1_s4	$\frac{C_2\left(C_1k_{sgs}^{\frac{3}{2}}+\Delta\|\|S\|\|^2\nu\right)}{\Delta(C_3-III)}$	[0.7775, 1.256, 2.0908]
S1_s5	$\frac{C_4^2k_{sgs}^{\frac{3}{2}}+\nu\left(C_1\sqrt{k_{sgs}}-4\Delta\left(C_4-\frac{II\Omega}{2}\right)\|\|S\|\|C_4\right)\|\|S\|\|}{2C_4\Delta}$	[33.6632, 0.777]
S2_s1	$\frac{-C_1C_3\Delta\sqrt{k_{sgs}}\|\|S\|\|^2\nu+C_3\Delta^2\|\|S\|\|^3\nu-k_{sgs}^2(C_2+1)}{\Delta\left(C_1\sqrt{k_{sgs}}-\Delta\|\|S\|\|\right)}$	[-3.3778, 2.3382, -0.1398, 0.9403, 1.1304]
S2_s2	$\frac{C_4k_{sgs}^{\frac{3}{2}}+\nu\|\|S\|\|\left((C_2-III)\sqrt{k_{sgs}}+C_1\Delta\|\|S\|\|\right)}{\Delta}$	[-0.141, 1.9149, -0.0037, 0.5081, 8.3372]

Continued on next page

S2_s3	$\frac{-\Delta\|\|S\|\|^2\nu C_4(C_3+V)\sqrt{k_{sgs}}+C_4C_5\Delta^2\|\|S\|\|^3\nu+C_1k_{sgs}^2}{\Delta\left((C_3+V)\sqrt{k_{sgs}}-C_5\Delta\|\|S\|\|\right)}$	[-1.7516, 0.7923, -0.9798, -0.1257, 0.5779]
S2_s4	$\frac{k_{sgs}^{\frac{3}{2}}+\left(C_2^2+C_2III-C_5^2+C_5III\right)\nu\|\|S\|\|\sqrt{k_{sgs}}+C_3\Delta(-C_2-C_5)\nu\|\|S\|\|^2}{\Delta(C_2+C_5)}$	[1.1767, 1.8845, 0.1369, -4.7542, 0.0457]
S2_s5	$\frac{-C_3C_4k_{sgs}^{\frac{3}{2}}+2\left(\frac{C_5\sqrt{k_{sgs}}}{2}+\|\|S\|\|\Delta(C_3-1)\right)\nu\|\|S\|\|}{2\Delta}$	[0.6816, -1.744, 0.8905, -1.1887, 2.5828]
S3_s1	$\frac{C_2k_{sgs}^{\frac{5}{2}}-2\nu\left(\Delta k_{sgs}-\frac{\nu\sqrt{k_{sgs}}}{2}\right)\|\|S\|\|^2(C_3-C_4)}{2\Delta k_{sgs}-\nu\sqrt{k_{sgs}}}$	[0.9984, 0.9771, 1.0229]
S3_s2	$\frac{C_4\Delta^2\|\|S\|\|^3\nu-C_4\Delta\sqrt{k_{sgs}}\|\|S\|\|^2V\nu+C_2C_5k_{sgs}^2}{\Delta\left(\Delta\|\|S\|\|-V\sqrt{k_{sgs}}\right)}$	[1.0409, 0.3183, 1.2548]
S3_s3	$\frac{C_4k_{sgs}^{\frac{3}{2}}-C_2\Delta\|\|S\|\|^2\nu(C_3-2)}{\Delta C_2}$	[1.2637, 1.9445, 0.6321]
S3_s4	$\frac{C_1C_5k_{sgs}^{\frac{3}{2}}+\left(C_2C_5\sqrt{k_{sgs}}-\|\|S\|\|\Delta(C_5-1)\right)\nu\|\|S\|\|}{\Delta}$	[0.4147, 2.1903, 1.1043]
S3_s5	$\frac{C_2C_4\Delta^2\|\|S\|\|^3\nu-C_2C_4\Delta\sqrt{k_{sgs}}\|\|S\|\|^2V\nu+C_5k_{sgs}^2}{\Delta\left(\Delta\|\|S\|\|-V\sqrt{k_{sgs}}\right)}$	[1.8082, 0.1764, 1.306]
S4_s1	$\frac{C_4C_1\Delta\|\|S\|\|^2\nu+C_5k_{sgs}^{\frac{3}{2}}}{C_1\Delta}$	[10.8341, 3.799, -2.0012, 0.0505, 5.424]
S4_s2	$\frac{C_1k_{sgs}^{\frac{3}{2}}-\Delta\|\|S\|\|^2\nu(C_4-C_5)}{C_3\Delta}$	[0.4832, 0.7977, 0.9651, 0.3449, 0.3937]
S4_s3	$\frac{k_{sgs}^{\frac{3}{2}}+(1-2II_\Omega+IV)\nu\|\|S\|\|^2\Delta C_4^2+(C_1-III)\nu\|\|S\|\|^2\Delta C_4}{(C_1+C_4(1-2II_\Omega+IV)-III)\Delta}$	[1.5466, 0.0505, 0.501, 0.0779, 0.5006]
S4_s4	$\frac{-C_1k_{sgs}^{\frac{3}{2}}+\nu\|\|S\|\|\left((-III+1)\sqrt{k_{sgs}}+C_4\Delta\|\|S\|\|\right)}{\Delta}$	[-0.4808, 0.5793, 0.2826, -0.0557, 0.0495]
S4_s5	$-\frac{C_4C_5k_{sgs}^{\frac{3}{2}}}{\Delta}$	[1.0281, 1.1957, 4.4605, 0.2007, -2.5202]
S5_s1	$\|\|S\|\|\left(II_\Omega k_{sgs}-\nu\|\|S\|\|(C_1+C_5+V)\right)C_2$	[1.0438, -0.0737, 2.0389, -0.0775, 0.5915]
S5_s2	$\frac{\left(C_1C_2\sqrt{k_{sgs}}\|\|S\|\|^2\nu^2-\nu\|\|S\|\|^2C_5\Delta(C_1-V)k_{sgs}\right)\|\|S\|\|}{\sqrt{k_{sgs}}\left(C_5\Delta\|\|S\|\|-C_3\sqrt{k_{sgs}}\right)\left(\Delta\|\|S\|\|(C_1-V)-III\sqrt{k_{sgs}}\right)}+$ $\frac{\left(\left(-C_1k_{sgs}^2(C_1-V)\Delta+k_{sgs}^{\frac{3}{2}}C_5III\nu\right)\|\|S\|\|+C_1k_{sgs}^{\frac{5}{2}}III\right)\|\|S\|\|}{\sqrt{k_{sgs}}\left(C_5\Delta\|\|S\|\|-C_3\sqrt{k_{sgs}}\right)\left(\Delta\|\|S\|\|(C_1-V)-III\sqrt{k_{sgs}}\right)}$	[0.6667, -0.6039, 0.1599, -8.6762, 1.4618]
S5_s3	$\frac{-C_3C_4\nu\Delta\|\|S\|\|^2(C_3+1)(C_3+C_4)\sqrt{k_{sgs}}}{\left(C_3(C_3+1)(C_3+C_4)\sqrt{k_{sgs}}+C_1^2C_2C_4\Delta\|\|S\|\|\right)\Delta}+$ $\frac{\left(k_{sgs}^2C_3^3+k_{sgs}^2(C_4+1)C_3^2+k_{sgs}^2C_4C_3-C_1^2C_4^2\nu\Delta^2\|\|S\|\|^3\right)C_2}{\left(C_3(C_3+1)(C_3+C_4)\sqrt{k_{sgs}}+C_1^2C_2C_4\Delta\|\|S\|\|\right)\Delta}$	[0.2813, 0.5045, -1.0947, -0.0774, -0.4983]
S5_s4	$\frac{\left(2\nu\|\|S\|\|(C_2+C_4+III)+C_5k_{sgs}\right)\sqrt{k_{sgs}}}{2\Delta}$	[-2.2252, -0.42, -4.5458, 1.617, 0.9515]
S5_s5	$\frac{\left(C_4k_{sgs}+\nu\|\|S\|\|\right)\sqrt{k_{sgs}}}{\Delta}$	[0.5028, -0.4017, 5.9059, 0.4766, 0.0528]

Continued on next page

S6_s1	$\frac{(\Delta\|\|S\|\|+\sqrt{k_{sgs}})\|\|S\|\|^2\Delta\sqrt{k_{sgs}}\left[0.507k_{sgs}^{\frac{3}{2}}+0.052\nu\Delta\|\|S\|\|^2)\right]+0.005\nu k_{sgs}^2\|\|S\|\|^2}{1}$
S6_s2	$\frac{\left(1.196+\frac{\|\|S\|\|\nu}{\sqrt{k_{sgs}}\left(\Delta\|\|S\|\|-III\sqrt{k_{sgs}}-V\sqrt{k_{sgs}}-0.7836\sqrt{k_{sgs}}\right)}+\frac{0.472k_{sgs}}{\nu\|\|S\|\|}\right)\|\|S\|\|\sqrt{k_{sgs}}\,\nu}{\Delta}$
S6_s3	$\frac{-0.0564\Delta\,\|\|S\|\|^2\nu+\sqrt{k_{sgs}}\,\|\|S\|\|\nu+0.4892k_{sgs}^{\frac{3}{2}}}{\Delta}$
S6_s4	$\frac{0.0555\Delta\,\|\|S\|\|^2\nu+0.5001k_{sgs}^{\frac{3}{2}}}{\Delta}$
S6_s5	$\frac{\left(-0.0328\Delta\|\|S\|\|+1.028\sqrt{k_{sgs}}\right)\left(\|\|S\|\|\nu+0.5013k_{sgs}\right)}{\Delta}$

E. Appendix Numerical Handling of Singularities

We point out that for all such modifications, they are made when some kind of underlying trend in the original model is observed, whereafter the model expressions are inspected and certain terms set to 0. If the resulting model contains reduced numerical instabilities and maintains the original trend that was observed, the modifications are adopted.

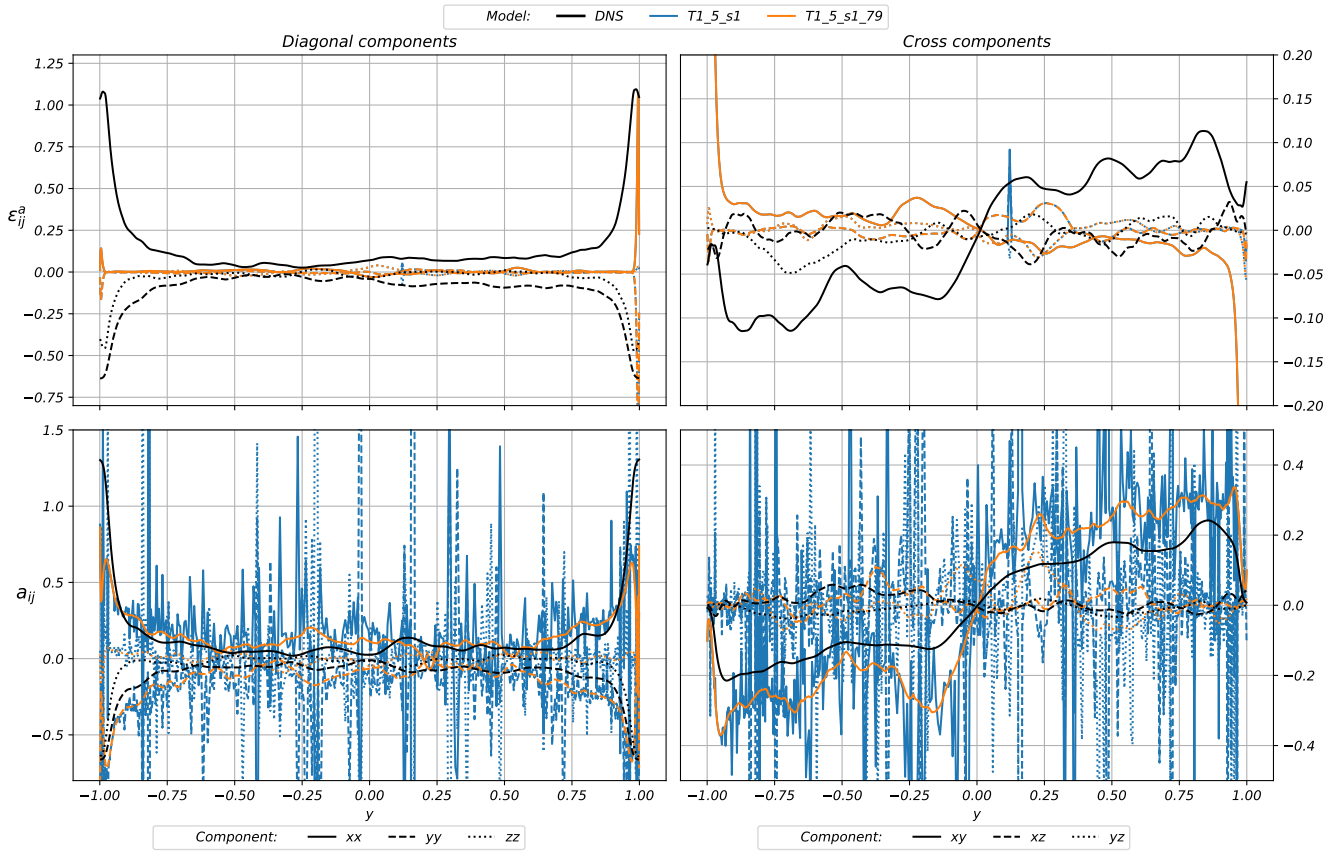


Figure 21. Comparison of models T1_5_s1 and T1_5_79 for TCF700_f17

F. Appendix Tensor Model Expressions

Model T1_1_s1 expression:

$$\varepsilon_{ij}^{a} = \Delta^{*}\frac{IV}{V}\mathbf{T}^{(1)} + \Delta^{*}\mathbf{T}^{(3)} + \frac{\Delta^{*}}{V}\mathbf{T}^{(6)} + \frac{\Delta^{*}}{II_S}\mathbf{T}^{(9)} - \frac{\Delta^{*}II_{\Omega}\mathbf{T}^{(2)}}{(2771II_{\Omega} + 76.4IV(II_{\Omega}(IIS + 17.8II_{\Omega}(II_S + 64) - II_{\Omega} + III + 2IVV + 40.5) - 24.7))} + \frac{\Delta^{*}IV^{2}\mathbf{T}^{(4)}}{(\Delta^{*}II_S^{2}IV - \Delta^{*}IV(II_{\Omega} - III + 2.27) + 51.7\Delta^{*} + II_SIII + 2II_{\Omega} - III - 64.4)} + \frac{37.25\mathbf{T}^{(7)}}{(V^{2}(II_{\Omega} + 37.7)(II_SII_{\Omega} - II_S - 412II_{\Omega} + III + IV(II_S + 49.7)))} \frac{\Delta^{*}IIIV\mathbf{T}^{(8)}}{(II_{\Omega}IV - V(-10.5II_SII_{\Omega}IVV^{2}(II_S - II_{\Omega}) + III + 37.9))} - \frac{5.43\mathbf{T}^{(10)}}{(8.37\Delta^{*}IV + IIIV - IV(II_{\Omega} - 1))}$$

Model T1_2_s1 expression:

$$\varepsilon_{ij}^{a} = -\left(\frac{IV}{II_S - II_{\Omega}}\right)\mathbf{T}^{(1))} + \Delta^{*}(\mathbf{T}^{(3)} + \mathbf{T}^{(4)}) - 0.013(\frac{\Delta^{*}}{III} + 1)\mathbf{T}^{(5)} + \frac{\Delta^{*}}{II_{\Omega}}\mathbf{T}^{6} + \frac{IV}{\Delta^{*} - V(II_{\Omega} + V - 153)}\mathbf{T}^{8}$$

Model T3_1_s1 expression:

$$\varepsilon_{ij}^{a} = \nu^{*}(\mathbf{T}^{1} + \mathbf{T}^{3} + \mathbf{T}^{4}) + \mathbf{T}^{2}\frac{III\nu^{*}}{\Delta^{*}(II_{\Omega} - III\nu^{*})(V - 80.3)} + \mathbf{T}^{5}\frac{IIIV\nu^{*}(II_S - II_{\Omega} + III)}{2II_{\Omega}III\nu^{*}(II_S - II_{\Omega} + III) - III\nu^{*}(IV + 2\nu^{*}) + \left(III^{2}V(II_{\Omega} + 41.4) - IV\nu^{*}(II_S + III(II_{\Omega} + III))\right)(II_S - II_{\Omega} + III)} + \mathbf{T}^{6}\frac{\nu^{*2}(1 - II_S)}{II_S^{2}II_{\Omega} + \nu^{*}(II_S - 1)(0.017II_{\Omega} - IV + \nu^{*} + 43.9)}\frac{1}{IV - 45.6\nu^{*} + 87.7} - \frac{\nu^{*2}\mathbf{T}^{9}}{IV + \nu^{*}(\Delta^{*}IIIV(\Delta^{*} + II_S) - II_{\Omega} + V)} - \frac{\nu^{*}\mathbf{T}^{10}}{1.63\Delta^{*}IV(V - 77.6)(IV(II_S - 1) + \nu^{*}) - 8.27}$$

Model T4_1_s1 expression:

$$\varepsilon_{ij}^{a} = \nu^{*}(\mathbf{T}^{(2)} + \mathbf{T}^{(3)} + \mathbf{T}^{(4)}) + \mathbf{T}^{(7)}\frac{\nu^{*}}{II_{\Omega}} - 0.011\frac{\Delta^{*}}{II_S}\mathbf{T}^{(8)} - 0.19\frac{IV}{II_{\Omega}V}\mathbf{T}^{(9)} - \frac{\nu^{*}}{\Delta^{*}\left(67.9\,II_S\,V - III\right)}\mathbf{T}^{(10)}$$

Model T5_1_s1 expression:

$$\varepsilon_{ij}^{a} = \nu^{*}(\mathbf{T}^{(1)} + \mathbf{T}^{(3)} + \mathbf{T}^{(4)}) - \frac{0.011}{\nu^{*}}\mathbf{T}^{(5)} + \frac{-\nu^{*}\frac{\mathcal{P}}{\varepsilon}(III - V)(V - \nu^{*})(\nu^{*} - 10.4)}{\left[II_{\Omega}III\,V - 33.3\frac{\mathcal{P}}{\varepsilon}(II_{\Omega} - V)\right]\left[80.9II_{\Omega}(V - \nu^{*})(\nu^{*} - 10.4) - \nu^{*}\frac{\mathcal{P}}{\varepsilon}(II_S - \frac{\mathcal{P}}{\varepsilon})\right]}\mathbf{T}^{(7)} + \frac{\nu^{*}\left(\frac{\mathcal{P}}{\varepsilon}\right)^{3}}{II_S(II_S + \frac{\mathcal{P}}{\varepsilon})[II_{\Omega} + \nu^{*}\,(II_S + II_{\Omega} - IV)]\left(V\left(\frac{\mathcal{P}}{\varepsilon}\right)^{2} + \nu^{*} + 87.3\right)}\mathbf{T}^{(9)}$$

Model T6_1_s1 expression:

$$\varepsilon_{ij}^{a} = \mathbf{T}^{(2)}\frac{0.25}{II_S} + \nu^{*}(\mathbf{T}^{(3)} + \mathbf{T}^{(4)}) + \mathbf{T}^{(5)}\frac{\nu^{*}}{II_S} + \mathbf{T}^{(8)}\frac{0.123}{V} + \mathbf{T}^{(10)}\frac{\nu^{*}}{V} - \mathbf{T}^{(6)}\frac{0.21V}{11.4V^2 + 0.664V + \nu^{*}(\Delta^{*} - III + 0.275\nu^{*} - 0.867)} + \mathbf{T}^{(7)}\frac{IV\nu^{*}}{II_S\nu^{*2}(II_S II_\Omega V - IV)(II_S + II_\Omega - IV(II_\Omega + III)) + II_\Omega IV} + \mathbf{T}^{(9)}\frac{\nu^{*}}{\Delta^{*}II_\Omega V + \Delta^{*} + II_S V - II_\Omega + III - V}$$

Model T7_1_s1 expression:

$$\varepsilon_{ij}^{a} = \mathbf{T}^{(1)}\frac{IV}{II_S II_\Omega} + \nu^{*}(\mathbf{T}^{(2)} + \mathbf{T}^{(3)} + \mathbf{T}^{(4)}) + \mathbf{T}^{(5)}\frac{\nu^{*}}{II_\Omega} + \mathbf{T}^{(7)}\frac{0.524}{V} + \mathbf{T}^{(8)}\frac{\nu^{*}}{II_\Omega} + \mathbf{T}^{(10)}\frac{\nu^{*}}{V} + \mathbf{T}^{(9)}\frac{0.524IV}{II_S^2 V + IV(\Delta^{*}II_S^2 - \Delta^{*}II_S + V - \mathcal{P}/\varepsilon)}$$

Model T1_7MC_s1 expression:

$$\varepsilon_{ij}^{a} = \mathbf{T}^{(1)}\frac{IV\nu^{*}}{\Delta^{*} - II_\Omega + IV(-2II_S + II_\Omega + IIIV^2\mathcal{P}/\varepsilon - IV\mathcal{P}/\varepsilon + \mathcal{P}/\varepsilon)} + \mathbf{T}^{(2)}0.06 + \mathbf{T}^{(3)}\frac{II_\Omega III(IV + \nu^{*})}{V(-\Delta^{*}II_\Omega + II_S II_\Omega III + III^2)} + \mathbf{T}^{(4)}\nu^{*2} + \mathbf{T}^{(5)}\frac{\nu^{*}}{II_S} + \mathbf{T}^{(7)}\frac{0.562\,V(II_S + III)}{(V(II_S + III)(II_S + \nu^{*}) + \mathcal{P}/\varepsilon)(II_S(II_\Omega - III) + II_\Omega + IV + \nu^{*})}$$

Model T1_7MC_s2 expression:

$$\varepsilon_{ij}^{a} = \mathbf{T}^{(4)}\nu^{*} + \mathbf{T}^{(5)}\frac{III\nu^{*}}{II_S^3} + \mathbf{T}^{(9)}\frac{\nu^{*}}{II_\Omega}$$

Model T1_7MC_s3 expression:

$$\varepsilon_{ij}^{a} = -\mathbf{T}^{(1)}0.003 + (\mathbf{T}^{(2)} + \mathbf{T}^{(3)} + \mathbf{T}^{(4)})\nu^{*} + \mathbf{T}^{(5)}\frac{0.054}{II_S} + \mathbf{T}^{(6)} - \frac{\nu^{*}}{V - \nu^{*}} + \mathbf{T}^{(7)}\frac{\nu^{*}}{II_\Omega} - \mathbf{T}^{(8)}\frac{\nu^{*}}{\Delta^{*} - II_\Omega + \mathcal{P}/\varepsilon - 0.6} - \mathbf{T}^{(9)}\frac{0.003}{V}\mathbf{T}^{(10)}\frac{0.003}{V}$$

Model T2_7MC_s1 expression:

$$\varepsilon_{ij}^{a} = \mathbf{T}^{(2)}0.726 + (\mathbf{T}^{(3)} + \mathbf{T}^{(4)})\nu^{*} + \mathbf{T}^{(5)}0.2 + \mathbf{T}^{(6)}\frac{1.79\,IV\nu^{*}}{-\Delta^{*} + II_S + V} + \mathbf{T}^{(7)}\frac{II_S}{II_\Omega + V - \mathcal{P}/\varepsilon} + \mathbf{T}^{(8)}\frac{0.218}{V(\mathcal{P}/\varepsilon\,(\nu^{*} + \mathcal{P}/\varepsilon) + 0.976)} + \mathbf{T}^{(9)}\frac{\nu^{*}}{II_\Omega(II_S - 0.915)} + \mathbf{T}^{(10)}\frac{-II_\Omega III(0.45\,II_\Omega + 0.446) + 0.45\,IV}{II_\Omega^2 III(2II_\Omega + \nu^{*})}$$

Model T2_7MC_s2 expression:

$$\varepsilon_{ij}^{a} = \mathbf{T}^{(2)}\frac{0.046}{II_\Omega} + (\mathbf{T}^{(3)} + \mathbf{T}^{(4)})\nu^{*} + \mathbf{T}^{(5)}\frac{\nu^{*}}{II_S} + \mathbf{T}^{(6)}\frac{0.046}{II_S - V} + \mathbf{T}^{(7)}\frac{0.046}{V} + \mathbf{T}^{(10)}\frac{\nu^{*}}{\Delta^{*} - 2V}$$

Model T2_7MC_s3 expression:

$$\varepsilon_{ij}^{a} = \mathbf{T}^{(1)}\frac{II_{\Omega}III}{IIIIV - 0.964(II_{\Omega}III - \nu^{*}\mathcal{P}/\varepsilon)(1.58II_{S}(\Delta^{*} - V\mathcal{P}/\varepsilon) - \mathcal{P}/\varepsilon + 0.709)} - \mathbf{T}^{(2)}0.061 + \mathbf{T}^{(4)}\nu^{*} + \mathbf{T}^{(5)}\frac{0.683}{II_{\Omega}} + \mathbf{T}^{(6)}\frac{0.066}{\Delta^{*}(II_{\Omega} - IV)} + \mathbf{T}^{(8)}\frac{0.282\Delta^{*}}{III + V} + \mathbf{T}^{(9)}\frac{\nu^{*}}{II_{\Omega}} + \mathbf{T}^{(10)}\frac{II_{S}III\nu^{*2}}{III\nu^{*2}(V - 8.86(II_{S} - II_{\Omega})(II_{\Omega} - V + 0.937)) - 1.42IV\mathcal{P}/\varepsilon(\Delta^{*} + III + IV)}$$

Model T1_8MC_s1 expression:

$$\varepsilon_{ij}^{a} = 0.001\mathbf{T}^{(3)} + \frac{0.001}{II_{\Omega}}\mathbf{T}^{(4)} + 0.001\mathbf{T}^{(5)} + \frac{0.143}{\Delta^{*} - V}\mathbf{T}^{(6)} + \frac{0.001}{II_{\Omega}}\mathbf{T}^{(7)} + \frac{\nu^{*}}{-II_{S} + V + \mathcal{P}/\varepsilon}\mathbf{T}^{(8)} + \frac{0.001}{\Delta^{*}II_{S}}\mathbf{T}^{(9)} + \frac{0.001}{II_{S}}\mathbf{T}^{(10)}$$

Model T1_8MC_s2 expression:

$$\varepsilon_{ij}^{a} = \frac{V\nu^{*}\mathcal{P}/\varepsilon}{\Delta^{*}V - 0.005\mathcal{P}/\varepsilon}\mathbf{T}^{(1)} + \nu^{*}\mathbf{T}^{(3)} + \nu^{*}\mathbf{T}^{(4)} + \frac{\nu^{*}}{II_{\Omega}}\mathbf{T}^{(7)} + \frac{\nu^{*}}{2\Delta^{*} + II_{S} + \mathcal{P}/\varepsilon}\mathbf{T}^{(8)} + \frac{0.059}{V}\mathbf{T}^{(9)}$$

Model T1_8MC_s3 expression:

$$\varepsilon_{ij}^{a} = -0.018\frac{IV}{2.687\Delta^{*}\nu^{*} + II_{S}IV}\mathbf{T}^{(1)} + (\nu^{*})^{2}\mathbf{T}^{(2)} + \nu^{*}\mathbf{T}^{(4)} - 0.018\frac{V(\Delta^{*}IV\nu^{*} + II_{S})}{II_{S}V(\Delta^{*}IV\nu^{*} + II_{S}) + 2.687(\nu^{*})^{2}}\mathbf{T}^{(6)} - 0.293\frac{\nu^{*}}{II_{S}}\mathbf{T}^{(8)} + \frac{\nu^{*}}{II_{\Omega}}\mathbf{T}^{(9)} - \frac{0.00906}{V}\mathbf{T}^{(10)}$$

Model T1_9MC_s1 expression:

$$\varepsilon_{ij}^{a} = -0.014\mathbf{T}^{(1)} + \frac{0.362}{II_{\Omega}}\mathbf{T}^{(2)} - \frac{\nu^{*}}{II_{S} + 0.784}\mathbf{T}^{(4)} + \frac{\nu^{*}}{II_{S} - \mathcal{P}/\varepsilon}\mathbf{T}^{(6)} - \frac{0.179}{V}\mathbf{T}^{(8)} - \frac{0.188}{V}\mathbf{T}^{(9)} + \frac{0.566}{-II_{S} + II_{\Omega} - III + IV + IV/(\mathcal{P}/\varepsilon) - 8.636(II_{\Omega}\nu^{*} - V(0.393II_{\Omega} - \mathcal{P}/\varepsilon))/(\nu^{*}(0.393II_{\Omega} - \mathcal{P}/\varepsilon)) + (\mathcal{P}/\varepsilon)/III}\mathbf{T}^{(10)}$$

Model T1_9MC_s2 expression:

$$\varepsilon_{ij}^{a} = -0.039\mathbf{T}^{(1)} + \nu^{*}\mathbf{T}^{(3)} + \nu^{*}\mathbf{T}^{(4)} - \frac{0.139}{V}\mathbf{T}^{(9)} + \frac{0.099\nu^{*}}{II_{S} - 0.156V}\mathbf{T}^{(10)}$$

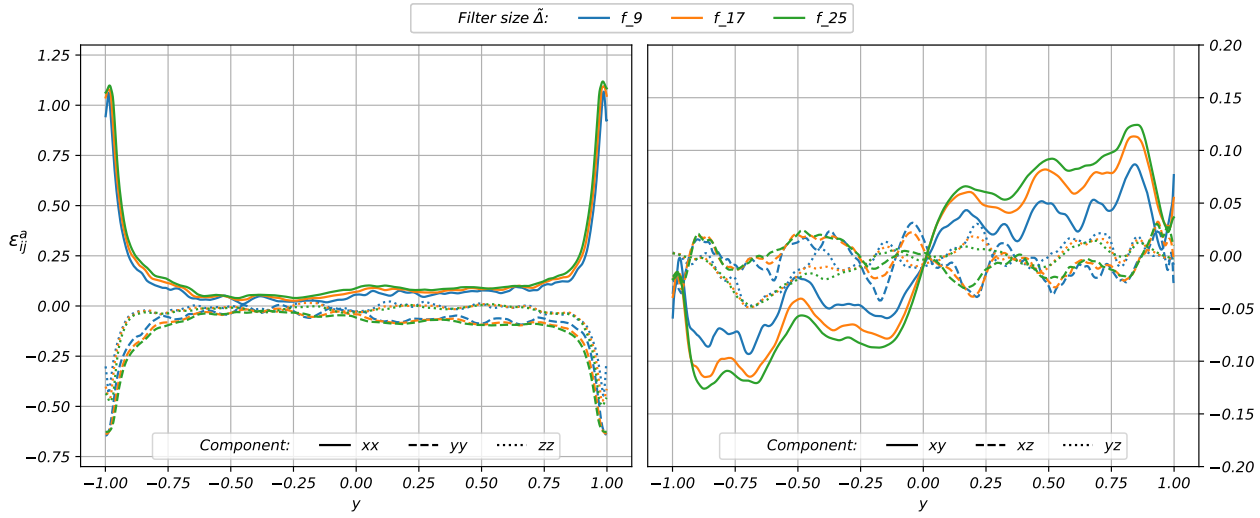
Model T1_9MC_s3 expression:

$$\varepsilon_{ij}^{a} = \Delta^{*}\mathbf{T}^{(1)} - \frac{II_{S}V\nu^{*}}{II_{S}V(\Delta^{*} + V - 1) + II_{S}\mathcal{P}/\varepsilon - IIIV}\mathbf{T}^{(2)} + \nu^{*}\mathbf{T}^{(3)} + \nu^{*}\mathbf{T}^{(4)} + \nu^{*}\mathbf{T}^{(5)} + \frac{\mathcal{P}/\varepsilon}{V}\mathbf{T}^{(6)} + \frac{\mathcal{P}/\varepsilon}{V}\mathbf{T}^{(7)} + \frac{-\Delta^{*} - II_{\Omega} - III - 1.409IV + 0.554}{II_{S}(-2.544\Delta^{*} + V + \mathcal{P}/\varepsilon)(II_{S} + II_{\Omega} - V)}\mathbf{T}^{(8)} + \frac{0.261}{V}\mathbf{T}^{(9)} + \frac{1.783\nu^{*}}{\Delta^{*}(II_{S} + IV)}\mathbf{T}^{(10)}$$

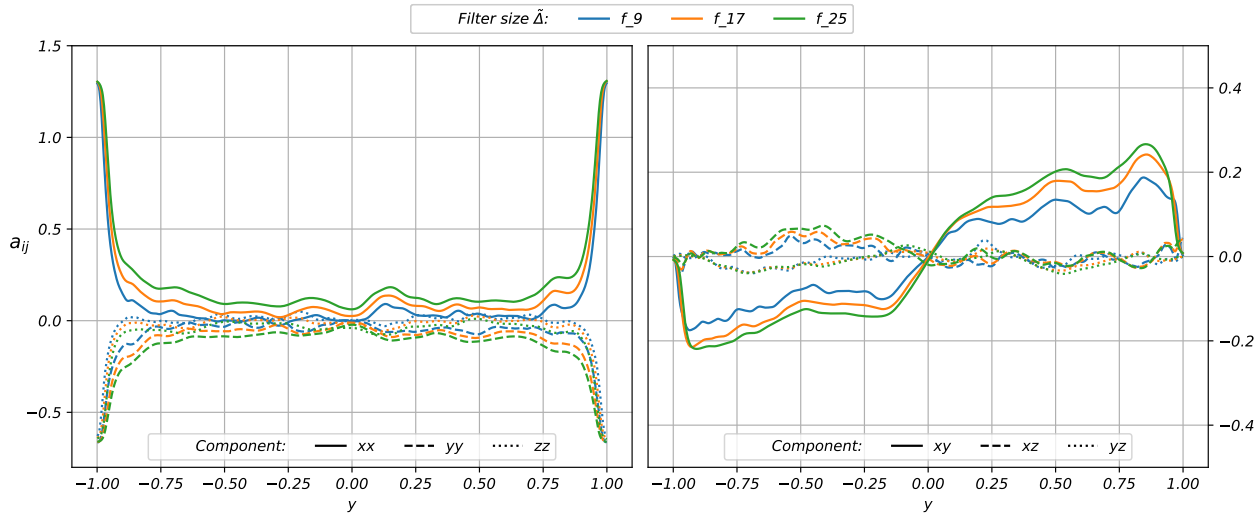
G. Modification to Loss Function & Results

Figure 22 shows that the tensor components of a_{ij} and ε^a_{ij} at three filter sizes. There is a significant decrease in magnitude of the diagonal components of a_{ij} with decreasing filter size. This correlation is less pronounced for ε^a_{ij}. Moreover, it is also noted that towards the centre of the channel,values of the diagonal components of a_{ij} is very close to zero for $\tilde{\Delta} = 9$. In view of this, it is suspected that the use of RMAE in the loss function may be inappropriate as it will lead to a bias in the optimization where errors located at points of low a_{ij} values are over-amplified. The loss function is thus modified as

$$Loss(\varepsilon^a_{ij_{model}}) = \left[\frac{\epsilon_\gamma}{q}\sum_q\left(|\sum_n \gamma^{(n)}_q \mathbf{T}^{(n)}_{ij,q} - \varepsilon^a_{ij,q}|+|\sum_n \beta^{(n)}_q \mathbf{T}^{(n)}_{ij,q} - a_{ij,q}|\right)\right]\epsilon_\infty, \tag{54}$$



(a) ε^a_{ij} tensor components.



(b) a_{ij} tensor components.

Figure 22. Comparison of DNS reference of a_{ij} and ε_{ij} profiles on data case TCF700.

With the modified loss function, model training was done on configurations T6M and T7M, each for three initializations. The errors of the best models from each training configuration is shown in Table 20 and Table 21. The model expression are presented in Appendix E. As can be seen in the tables, the order of magnitude of error in the case of TCF700_f9 is still significantly higher than those with larger filter sizes and HIT cases. This thus disproves

that the large error in the trained models, is due to the implementation of RMAE in the loss function but rather caused by some other factors. We do note that for a_{ij}, model T1_7M_s1 outperforms the G0 model in all HIT cases and TCF700 with the largest filter width.

Table 20. MAE in ε^a_{ij} of models shown per training case and averaged across training cases and the 6 independent tensor components.

Model identifier	HIT			TCF700			Average MAE
	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	
T1_6M_s3	0.536	0.465	0.416	670.983	1.838	1.595	112.639
T1_7M_s1	0.130	0.100	0.095	15.092	0.232	0.192	2.640
G0	0.081	0.046	0.032	0.215	0.171	0.151	0.116

Table 21. MAE in a_{ij} of models shown per training case and averaged across training cases and the 6 independent tensor components.

Model identifier	HIT			TCF700			Average MAE
	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	
T1_6M_s3	0.426	0.390	0.391	13119.288	0.359	0.290	2186.857
T1_7M_s1	0.315	0.299	0.319	127622.656	0.332	0.257	21270.696
G0	0.344	0.347	0.374	0.350	0.308	0.270	0.332

Figure 23 shows the tensor component profiles of TCF700_f25. It can be seen that model improvement for all a_{ij} components, relative to G0, mainly occurs away from channel walls. The shape of the diagonal components remain significantly different from G0 reference but the a_{xy} component of T1_7M_s1 is largely similar to G0. One can also observe that the large deviation of a_{xy} at channel walls contributes to it's larger averaged error w.r.t G0. For the tensor components of ε^a_{ij}, the only notable observation is some visual correlation in ε^a_{xy} in the centre of the channel, albeit with large error in magnitude.

Figure 23. Comparison of a_{ij} and ε_{ij} profiles of models T1_6M_s3 and T1_7M_s1. G0 model is provided as reference.

H. Correction to Scalar Dissipation

Results in Appendix G show that the implementation of RMAE in the loss function does not lead to the exceedingly large magnitude of error in the case of TCF700 with the smallest filter size. Further investigation eventually uncovered an error within a data processing step in the algorithm. An error in the scalar dissipation rate ε caused the term $\frac{k_{sgs}}{\varepsilon}$, used to non-dimensionalize the basis tensors and invariants, to take extreme values. Table 22 shows the range and mean values of k_{sgs}/ε. It can be seen that for all filter sizes, the error's effect on the mean of k_{sgs}/ε is minimal. The error mainly affects the case with the smallest filter where negative and extremely small values of ε occur, leading to negative and exceedingly large values in k_{sgs}/ε. A comparison of ε is shown in Figure 24, where it indicates that the error is only occurs within $y^+ < 20$. Lastly, it should also be noted that the error only affects TCF cases and not HIT cases.

Table 22. Ranges and means before and after correction for all filter sizes.

$\tilde{\Delta}$	Range Before (min, max)	Range After (min, max)	Mean Before	Mean After
9	-111.833, 44672.711	0.098, 47.028	2.926	2.921
17	0.132, 34.864	0.129, 34.864	3.956	3.955
25	0.152, 28.871	0.153, 28.871	4.826	4.826

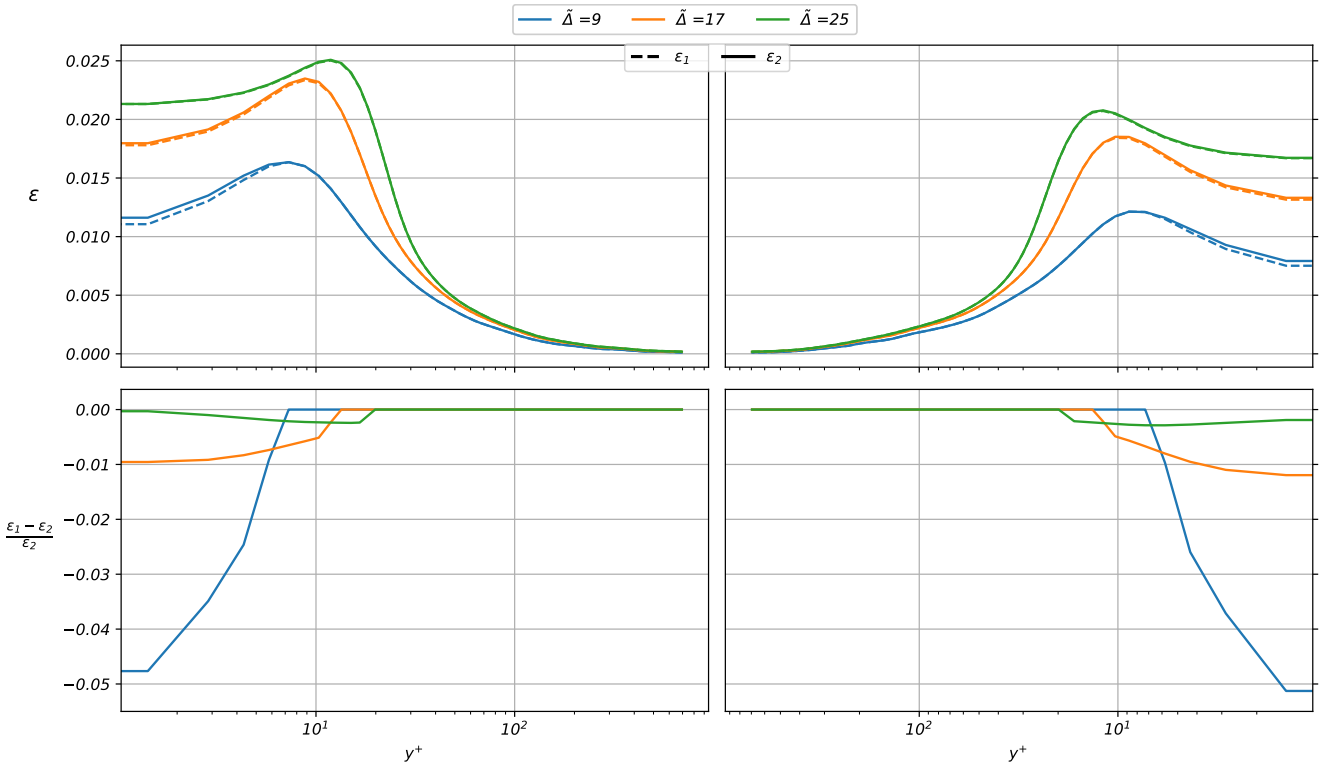


Figure 24. Scalar dissipation ε before and after correction for TCF_700. Subscript 1 and 2 indicates before and after correction respectively.

The reference model G0 and models T1_6M_s3 and T1_7M_s1 were re-evaluated using the corrected ε, the errors are shown in Table 23 and Table 24. We note that error for the HIT cases are identical to those in Table 20 and Table 21, confirming that the cases are not affected. Mainly, it can be seen that the model error in the TCF case with $\tilde{\Delta} = 9$ is in the same order of magnitude compared to other filter sizes and HIT cases. Furthermore, the difference in error for the TCF cases with larger filter sizes remain remain small.

Table 23. MAE in ε_{ij}^a of models shown per training case and averaged across training cases and the 6 independent tensor components.

Model identifier	HIT			TCF700			Average MAE
	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	
T1_6M_s3	0.536	0.465	0.416	2.882	1.806	1.580	1.281
T1_7M_s1	0.130	0.100	0.095	0.424	0.231	0.191	0.195
G0	0.081	0.046	0.032	0.215	0.171	0.151	0.116

Table 24. MAE in a_{ij} of models shown per training case and averaged across training cases and the 6 independent tensor components.

Model identifier	HIT			TCF700			Average MAE
	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	$\tilde{\Delta} = 9$	$\tilde{\Delta} = 17$	$\tilde{\Delta} = 25$	
T1_6M_s3	0.426	0.390	0.391	0.573	0.358	0.290	0.405
T1_7M_s1	0.315	0.299	0.319	0.708	0.328	0.255	0.371
G0	0.344	0.347	0.374	0.350	0.308	0.269	0.332