Turbulence Modeling Final Project
Nonequilibrium Closure Model

Ryan Darragh

December 17, 2015

1 Introduction

Turbulence is a complicated physical phenomenon found in many engineering and natural applications that is notoriously difficult to understand and simulate. The resolutions required to properly simulate realistic flows are well beyond the current computational capabilities due to resolution requirements and a large separation of scales. In order to address these issues engineers have developed several ways to simplify the Navier-Stokes (NS) equations. Algebraic models seek to develop an algebraic relationship between eddy viscosity and a length scale within the flow. They work somewhat well for simple flows, but for more complex flows Reynolds-Averaged Navier-Stokes (RANS) models are required. These models average the NS equations and model the Reynolds stresses in some way. RANS models are more complicated than algebraic models, but they still only capture the largest scales of turbulence. To capture smaller scales large eddy simulations (LES) are used. These capture down to scales somewhere in the inertial range rather than only the largest scales. If even more resolution is required one must turn to direct numerical simulation (DNS) in order to fully simulate the flow and capture all scales. Unfortunately as more scales are captured in a simulation, the cost of the simulation also increases. The high cost of LES and DNS has led to the popularity of RANS equations primarily in industry. This project will apply a nonequilibrium closure developed in [2] to a RANS model, namely the standard $k-\varepsilon$ model developed by Launder and Sharma.

RANS models begin by averaging the NS equations and splitting the components into an averaged $\overline{\ }$ and perturbation $'$ component. Doing so produces a simpler system with the exception of the open $\overline{u_i'u_j'}$ term. To get around this a number of models have been developed each taking a different, yet usually fairly similar, approach to closing the RANS equation. Generally these models consist of some set of transport equations. They can either consist of a single equation, like the Spalart-Allmaras model, or two equations like the standard $k-\varepsilon$ or $k-\omega$ models. More complicated models also exist with even more equations, however we will focus on two equation models. Two equation models generally consist of a transport equation for the turbulent kinetic energy k and a transport equation for some secondary variable. Most of the differences in two equation models arise in the choice of the second variable. Two of the most popular two equation models are the $k-\varepsilon$ and $k-\omega$ models. These two models are very similar and in fact one can derive one from the other with a simple change of variables. However, of particular interest to us is their use of the Boussinesq hypothesis which assumes $(\overline{u_i'u_j'})_{\text{dev}} \sim \overline{S}_{ij}$. This assumptions leads to

$$a_{ij} = -2\frac{\nu_T}{k}\overline{S}_{ij}, \tag{1}$$

a statement that is not necessarily true for all types of flow as the anisotropy does not always depend directly on the instantaneous strain rate. In more complex flows with curvature or some sort of pulsing action driving the flow, the anisotropy tensor will become dependent upon the straining history instead.

In [2], Hamlington and Dahm present a new type of closure that does not rely on the Boussinesq hypothesis. Instead they develop a relationship between the anisotropy tensor and an effective strain rate $\tilde{S}_{ij}$ that depends upon the straining history. This type of closure is known as a nonequilibrium closure and this closure can be applied to any model by replacing the Boussinesq hypothesis assumption with their nonequilibrium result. This project will also apply their closure to the standard $k-\varepsilon$ model using the Python code package FEniCS ([5]) and simulations will be performed on a steady channel flow.

2 Model Development Theory

2.1 The Standard $k-\varepsilon$ Model

The Standard $k-\varepsilon$ model is among the most popular of the RANS models. It was originally developed in 1945 by Chou and was worked on by a number of people up until 1974 when Launder and Sharma presented what is now known as the Standard $k-\varepsilon$ (SKE) model. Since then further work has been done, particularly on improving the models performance near walls. In this section we will formulate the SKE starting from the Reynolds-Averaged Navier-Stokes equations and follow along the derivation given both in class in Wilcox's book [7].

As has been previously stated the SKE model begins with the Reynolds-Averaged Navier-Stokes equations and the continuity equation. To simplify matters we'll assume the fluid to be incompressible. Thus our starting point is

$$\frac{\partial u_j}{\partial x_j} = 0 \tag{2}$$

$$\rho\frac{\partial \overline{u}_i}{\partial t} + \rho\frac{\partial}{\partial x_j}\left(\overline{u}_j\overline{u}_i + \overline{u_j'u_i'}\right) = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\mu\overline{S}_{ji}\right) \tag{3}$$

where $u_i = \overline{u}_i + u_i'$ is the velocity, $p = \overline{p} + p'$ is the pressure, ρ is the density, and μ the viscosity. S_{ji} is the usual definition $S_{ij} = \overline{S}_{ij} + S_{ij}' = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$ which we note is symmetric. We can define the specific Reynolds stress tensor as $\tau_{ij} = -\overline{u_i'u_j'}$. Simple inspection reveals that we have one equation from continuity and three from the NS equations for a total of four equations. However τ_{ij} has introduced six unknown quantities to go along with the three velocity components and pressure, and thus we see that a model is required since our system is not closed.

The next step is to develop a transport equation for τ_{ij}. This can be done by multiplying Equation 3 by u_j' and then doing a lot of math to simplify the result. Doing so yields the following transport equation

$$\frac{D\tau_{ij}}{Dt} = -\tau_{ik}\frac{\partial \overline{u}_j}{\partial x_k} - \tau_{jk}\frac{\partial \overline{u}_i}{\partial x_k} + \varepsilon_{ij} - \Pi_{ij} + \frac{\partial}{\partial x_k}\left[\nu\frac{\partial \tau_{ij}}{\partial x_k} + C_{ijk}\right] \tag{4}$$

where we have defined the following

$$\Pi_{ij} = \overline{\frac{p'}{\rho}\left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i}\right)} \tag{5}$$

$$\varepsilon_{ij} = 2\nu\overline{\frac{\partial u_i'}{\partial x_k}\frac{\partial u_j'}{\partial x_k}} \tag{6}$$

$$\rho C_{ijk} = \rho\overline{u_i'u_j'u_k'} + \overline{p'u_i'}\delta_{jk} + \overline{p'u_j'}\delta_{ik}. \tag{7}$$

We again see a higher order correlation, triple in this case, than what we started with (second order). In fact if we were to continue on in this manner we would continuously get higher order terms each time and never be able to close the system without resorting to other methods. However we can get around this issue by taking the trace of Equation 4 and deriving another transport equation.

As motivation recall our definition for τ_{ij} and that the turbulent kinetic energy (TKE) of a flow is given by

$$k = \frac{1}{2}\overline{u_i'u_i'}. \tag{8}$$

Taking the trace of τ_{ij} provides us with the relation $\tau_{jj} = -2k$. So we would expect to obtain a transport equation for k by taking the trace of Equation 4. Doing so we see that this is indeed the case and end up with

$$\frac{Dk}{Dt} = \tau_{ij}\frac{\partial \overline{u}_i}{\partial x_j} - \varepsilon + \frac{\partial}{\partial x_j}\left[\nu\frac{\partial k}{\partial x_j} - \frac{1}{2}\overline{u_i'u_i'u_j'} - \frac{1}{\rho}\overline{p'u_j'}\right] \tag{9}$$

where we have slightly changed the definition of ε to be

$$\varepsilon = \nu\overline{\frac{\partial u_i'}{\partial x_k}\frac{\partial u_i'}{\partial x_k}}. \tag{10}$$

We now need to close Equation 9 by specifying τ_{ij}, the dissipation ε, the turbulent transport $\frac{1}{2}\overline{u_i'u_i'u_j'}$, and the pressure diffusion $\frac{1}{\rho}\overline{p'u_j'}$. This will be done by making an assumption for three of the terms and doing what Wilcox refers to as "drastic surgery". The dissipation term is left for the model.

The Boussinesq approximation is used to close the first term by assuming

$$\tau_{ij} = 2\nu_T \overline{S}_{ij} - \frac{2}{3}k\delta_{ij} \tag{11}$$

where ν_T is the eddy viscosity. This assumption will be of primary concern in our later development of the nonequilibrium closure.

The next assumption involves the turbulent transport and pressure diffusion. Here we assume gradient-diffusion which assumes the turbulent energy and pressure are transported like a molecular process. There isn't really any theoretical reason to believe this to be the case, but it does tends to work. Additionally these two terms are generally grouped together resulting in

$$\frac{1}{2}\overline{u_i' u_i' u_j'} + \frac{1}{\rho}\overline{p' u_j'} = -\frac{\nu_T}{\sigma_k}\frac{\partial k}{\partial x_j}, \tag{12}$$

where σ_k is a closure coefficient. This equation defines σ_k which will also be assumed to be constant.

At this point, aside from the dissipation, we have a closed equation. Combining everything together produces the k transport equation

$$\frac{Dk}{Dt} = \tau_{ij}\frac{\partial \overline{u}_i}{\partial x_j} - \varepsilon + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_T}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right]. \tag{13}$$

Now that a 'closed' k equation has been developed we must deal with the ε term. To do this we will develop an additional transport equation for ε and solve that in conjunction with the k equation to form our model. We would like the derivation of the ε equation to begin with the definition in Equation 10. However, this equation does not produce the exact dissipation. One can instead obtain an analytically exact equation for ε by solving

$$2\nu\overline{\frac{\partial u_i'}{\partial x_j}\frac{\partial}{\partial x_j}\left[\mathcal{N}(u_i)\right]} = 0 \tag{14}$$

where

$$\mathcal{N}(u_i) = \rho\frac{Du_i}{Dt} + \frac{\partial p}{\partial x_i} - \mu\frac{\partial^2 u_i}{\partial x_k \partial x_k}. \tag{15}$$

Once again a bunch of math is done and we end up with our exact transport equation for ε,

$$\begin{aligned}\frac{D\varepsilon}{Dt} = -2\nu\left[\overline{u_{i,k}' u_{j,k}'} + \overline{u_{k,i}' u_{k,j}'}\right]\frac{\partial \overline{u}_i}{\partial x_j} - 2\nu\overline{u_k' u_{i,j}'}\frac{\partial^2 \overline{u}_i}{\partial x_k \partial x_j} - 2\nu\overline{u_{i,k}' u_{i,m}' u_{k,m}'} - 2\nu^2\overline{u_{i,km}' u_{i,km}'} \\ + \frac{\partial}{\partial x_j}\left[\nu\frac{\partial \varepsilon}{\partial x_j} - \nu\overline{u_j' u_{i,m}' u_{i,m}'} - 2\frac{\nu}{\rho}\overline{p_{,m}' u_{j,m}'}\right].\end{aligned} \tag{16}$$

As can obviously be seen this equation is very complicated. New double and triple correlations have popped up that are extremely difficult to measure experimentally. The difficulty in dealing with this equation is of concern to Wilcox, however other researchers have used DNS and experiment to approximate constants and close the equation. These efforts have resulting in Equation 18.

At this point we are ready to present the full SKE Model. It consists of the following set of equations, relations, and constants.

Equations for the turbulence kinetic energy and dissipation rate:

$$\frac{Dk}{Dt} = \tau_{ij}\frac{\partial \overline{u}_i}{\partial x_j} - \varepsilon + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_T}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] \tag{17}$$

$$\frac{D\varepsilon}{Dt} = C_{\varepsilon 1}\frac{\varepsilon}{k}\tau_{ij}\frac{\partial \overline{u}_i}{\partial x_j} - C_{\varepsilon 2}\frac{\varepsilon^2}{k} + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_T}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right] \tag{18}$$

Kinematic eddy viscosity:

$$\nu_T = C_\mu\frac{k^2}{\varepsilon} \tag{19}$$

Closure coefficients:

$$C_{\varepsilon 1} = 1.44, \qquad C_{\varepsilon 2} = 1.92, \qquad C_\mu = 0.09, \qquad \sigma_k = 1.0, \qquad \sigma_\varepsilon = 1.3 \tag{20}$$

2.2 Nonequillibruim Closure

2.2.1 Motivation

To provide some motivation for the need of a nonequillibruim closure we will first consider the results of Yu and Girimaji in [8]. Yu and Girimaji use DNS to simulate periodically sheared homogeneous flow. Figure 1 shows some of their results with $\frac{\omega}{S_{\max}} = 0.5$ where ω is the shearing frequency. Similar results are also obtained at higher values of $\frac{\omega}{S_{\max}}$. These results make it quite clear that the anisotropy a_{ij} (b_{12} in the figure) lags behind the strain rate S_{ij}. Therefore this is an example the Boussinesq approximation failing.

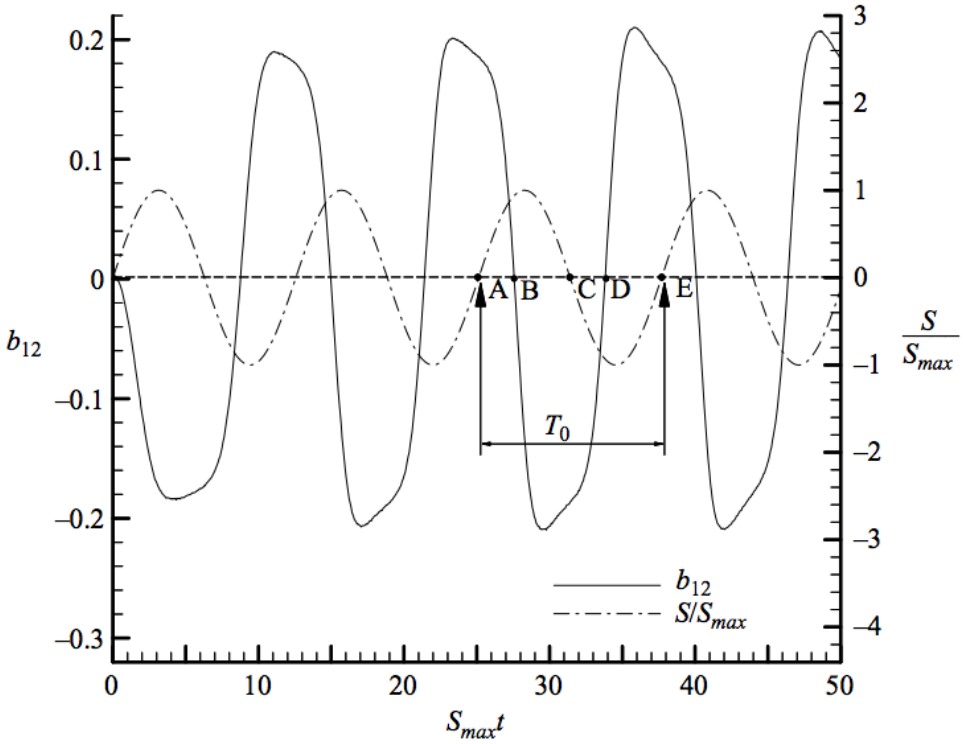

Figure 1: DNS results of periodically sheared homogeneous flow with $\frac{\omega}{S_{\max}} = 0.5$ by Yu and Girimaji from [8]. Here we see the time dependance of the strain rate and anisotropy (b_{12}). It is clear that the anisotropy lags behind the strain rate, and thus $a_{ij} \sim \overline{S}_{ij}$ does not hold in this case.

Even though a_{ij} may not always proportional to S_{ij} the Boussinesq approximation may still be "good enough" to produce valid results. In order to investigate if this inaccuracy even models to the SKE model Hamlington and Dahm compared the SKE model to an LES by Bardina ([1]) for an impulsively sheared homogenous turbulent flow. Results of this comparison are shown in Figure 2. We see the turbulent kinetic energy, k, starts at the same value for both simulations, however as time increases the SKE fails to predict an initial decrease in k that the LES predicts. Since LES is the more accurate of the two this can be seen as a failure of the SKE model. That is, the SKE is not correctly modeling the flow by a not insignificant amount, and so the results of Yu and Girimaji cannot be ignored. One must account for the time lag between strain rate and anisotropy in a nonequilibrium flow.

2.2.2 Intuitive Overview

To provide some insight to the closure we will first present the physical considerations [2] brings up before they formally derive the closure. The starting place for this consideration is

$$a_{ij} = -\frac{\left(\overline{u_i' u_j'}\right)_{\text{dev}}}{k} \tag{21}$$

which tells us that the anisotropy tensor is directly related to the deviatoric part of the Reynolds stress tensor. So if we understand how the Reynolds stresses evolve through time we can also understand how the anisotropy evolves.

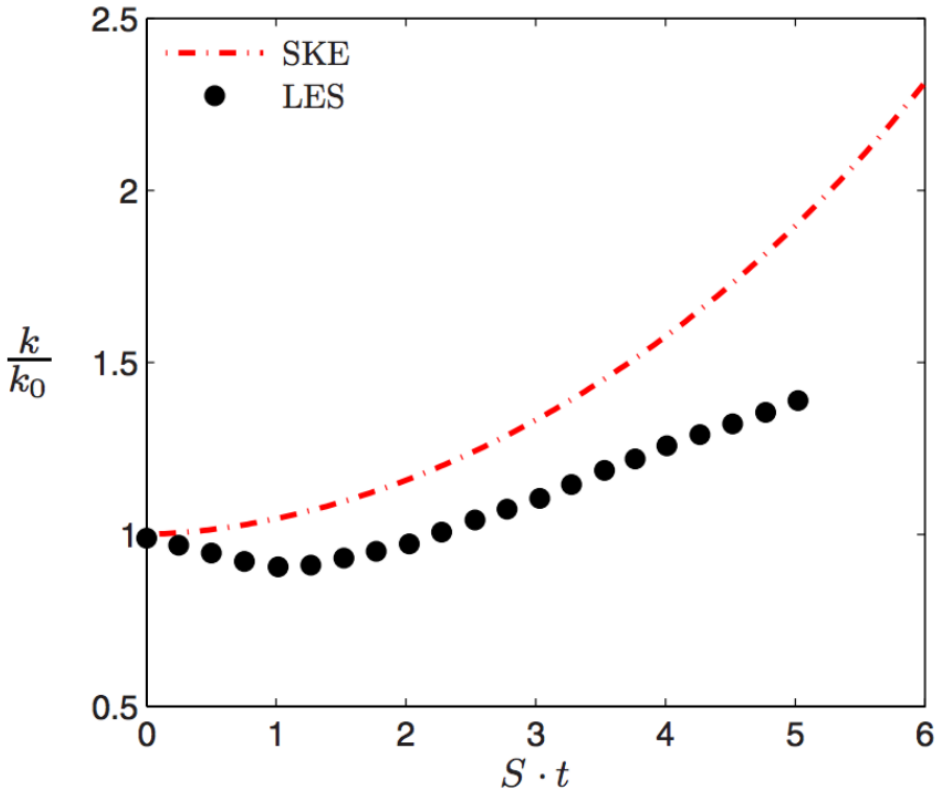

Figure 2: Comparison of $k-\varepsilon$ model and LES under impulsively sheared homogeneous turbulence. The LES results were obtained by Bardina *et al.* in [1], and the SKE by Hamlington and Dahm in [2]. The red line shows the $k-\varepsilon$ model's results and the black the LES solution. We see that the results do not match as both models evolve through time.

We will take this relation a step further by considering the Biot-Savart integral which relates the velocity fluctuations to the vorticity fluctuations as seen in

$$\overline{u_i' u_j'}(\boldsymbol{x}, t) = \frac{1}{4\pi} \int_{\check{\boldsymbol{x}}} \frac{1}{4\pi} \int_{\hat{\boldsymbol{x}}} \epsilon_{ikl} \epsilon_{jmn} \overline{\omega_k'(\check{\boldsymbol{x}}, t) \omega_m'(\hat{\boldsymbol{x}}, t)} \frac{(x_l - \check{x}_l)(x_n - \hat{x}_n)}{|x - \check{x}|^3 |x - \hat{x}|^3} d^3\check{\boldsymbol{x}} d^3\hat{\boldsymbol{x}}. \tag{22}$$

This provides us with a relation between the vorticity fluctuation correlation and the anisotropy tensor. Since ω_i' is a small scale quantity, it depends upon the nearby relative alignment of the overall vorticity. Because of this dependency we can consider the vorticity transport equation

$$\frac{D\omega_i}{Dt} = \omega_j S_{ij} + \nu \frac{\partial^2}{\partial x_j \partial x_j} \omega_i. \tag{23}$$

The first term on the right is nonlinear in general, and provides us with our dependence on S_{ij}. Since we are interested in how ω_i depends on the strain rate we will focus on this term.

Next we split the strain rate into a local, S_{ij}^L, and background, S_{ij}^B, component where $S_{ij} = S_{ij}^L + S_{ij}^B$. The local component is essentially how a fluid element strains itself through its vorticity whereas the background component is how the rest of the fluid strains a given fluid element. It has been shown in [3] that the instantaneous vorticity tends to align itself with the S_{ij}^B field. Thus we should expect the vorticity's evolution to depend on the background strain instead of the local strain, as the local strain has little to no effect. By ignoring the local strain we also get rid of the nonlinearity in Equation 23, leaving us with a quasilinearized equation than can be written as

$$\frac{D\omega_i}{Dt} \approx \omega_j S_{ij}^B. \tag{24}$$

From this equation we can consider what happens to the vorticity as the background strain rate varies. If S_{ij}^B varies slowly the vorticity will remain aligned. However if S_{ij}^B varies rapidly then the vorticity will realign with S_{ij}^B on the time scale of $1/S^B$ where $S^B = \left(2S_{ij}^B S_{ij}^B\right)^2$.

Now we will consider an impulsively sheared flow that has been subdivided into many fluid elements. Before the shearing begins each fluid element will have some (likely unique or near unique) randomly oriented vorticity vector as there is no overall preferred direction in which the vorticity would be expected to point. When the shear is applied the vorticity for each fluid element will then begin to align with S^B_{ij}. Since each fluid element begins in a different orientation, and they all reorient like $1/S^B$ as opposed to instantly reorienting, they will continue to have different strain rates as time progresses and likely all be in a different part of their reorientation process at any given time. That is to say one element may be 75% of the way to fully aligning with the background strain while another is only 30% of the way there.

Additionally the fluid elements are convecting through the flow and interacting with each other. We can consider a fluid element some time t. This element's vorticity alignment will be dependent upon all the fluid elements that met at this location at this time. Each of the fluid elements leading up to this point will have had some different starting vorticity orientation, and they will each have been subjected to differing strain histories meaning they will all be at different points in their reorientation to the background strain. Figure 3 shows pictorially what this may look like. Thus in order to determine the vorticity of our fluid element at t we need to take an ensemble average over all the fluid elements. The a_{ij} that results from this average will not be proportional to the instantaneous strain rate due to the straining histories. Instead we can define an effective strain rate $\tilde{S}_{ij}$ such that $a_{ij} \sim \tilde{S}_{ij}$. Putting this all together we can set up a sort of linear system with input $\left\langle S^B_{ij}(\tau)\right\rangle$, impulse input $h(t-\tau)$, and output $\tilde{S}_{ij}$. This system then provides a relation between the three quantities that brings all of this together,

$$\tilde{S}_{ij}(\boldsymbol{x},t) = \int_{-\infty}^{t} \left\langle S^B_{ij}(\tau)\right\rangle_{R(\tau)} h(t-\tau) D\tau \tag{25}$$

where $\langle\ \rangle_{R(\tau)}$ represents an ensemble average over $R(\tau)$.

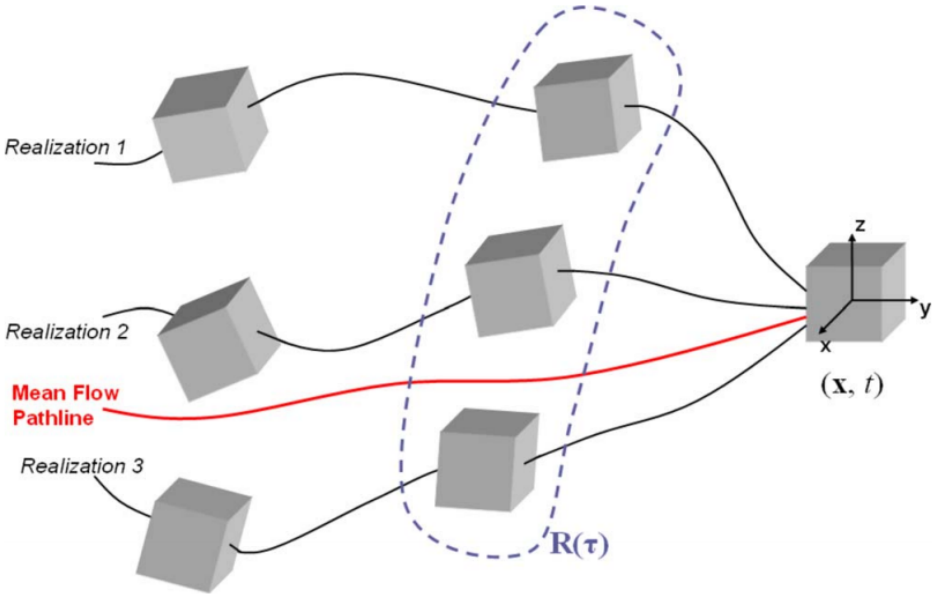

Figure 3: Image from [2] showing fluid elements traveling along different pathlines and undergoing different strain rates before meeting at time t. $R(\tau)$ represents an ensemble of these fluid elements.

A more formal derivation of the effective strain rate can be found in [2]. To just go over the highlights though, they begin with a transport equation for a_{ij},

$$\frac{Da_{ij}}{Dt} = \frac{1}{k}\left(\frac{D\overline{u_i'u_j'}}{Dt} - \frac{\overline{u_i'u_j'}}{k}\frac{Dk}{Dt}\right). \tag{26}$$

As you can see, within this transport equation are two more transport equations, one for the Reynolds stresses and one for the turbulent kinetic energy. By subbing in both of these transport equations and doing a lot of algebra/calculus,

they eventually end up with the linearized equation

$$\frac{Da_{ij}}{Dt} = -\frac{1}{\Lambda_m} a_{ij} + \alpha_2 \overline{S}_{ij} \tag{27}$$

where $\Lambda_m = C_\Lambda \frac{k}{\varepsilon}$ is the turbulence memory timescale and α_2 and C_Λ are some constants.

Further work can be done to show

$$a_{ij}(t) = \int_{-\infty}^{t} \alpha_2 \overline{S}_{ij}(\tau) h(t-\tau) D\tau \tag{28}$$

which is similar to Equation 25. By specifying $h(t-\tau) = e^{-(t-\tau)/\Lambda_m(t)}$ they can then write

$$a_{ij}(t) = -2C_\mu \frac{k}{\varepsilon} \frac{1}{\Lambda_m(t)} \int_{-\infty}^{t} \overline{S}_{ij}(\tau) e^{-(t-\tau)/\Lambda_m(t)} D\tau. \tag{29}$$

Equation 29 is sort of similar to $a_{ij} \sim \tilde{S}_{ij}$, except we still need to specify $\tilde{S}_{ij}$. We can do that by brining the $\Lambda_m(t)$ into the integral and defining

$$\tilde{S}_{ij}(t) = \int_{-\infty}^{t} \overline{S}_{ij}(\tau) \frac{e^{-(t-\tau)/\Lambda_m(t)}}{\Lambda_m(t)} D\tau. \tag{30}$$

Putting this definition into Equation 29 yields the relationship

$$a_{ij}(t) = -2C_\mu \frac{k}{\varepsilon} \tilde{S}_{ij}(t). \tag{31}$$

If we take this a step further by using the SKE's definition of $\nu_T = C_\mu \frac{k^2}{\varepsilon}$ we can write

$$a_{ij} = -2\frac{\nu_T}{k} \tilde{S}_{ij} \tag{32}$$

which is identical to Equation 1 except we have replaced $\overline{S}_{ij}$ with $\tilde{S}_{ij}$.

2.2.3 Calculating $\tilde{S}_{ij}$

As seen in Equation 32 the entire implementation of this closure is dependent upon our ability to calculate $\tilde{S}_{ij}$, and as seen in Equation 25 or 30 it appears to require calculating a complicated time integral over pathlines, ensemble averaging, and dealing with material derivatives that rotate in time. Instead of dealing with all of that, Hamlington and Dahm derive a form of the effective strain rate in [2] that is more conducive to actually calculating values. They begin by doing a sort of Taylor expansion of $\overline{S}_{ij}$ with material derivatives instead of normal derivatives, and inserting the effective strain to that result.

By doing a bit of algebra and accounting for rotation and strain rate effects on a_{ij} they end up with a much simpler expression

$$\tilde{S}_{ij} = \overline{S}_{ij} + \sum_{n=1}^{\infty} (-\Lambda_m)^n \left[\frac{\mathring{D}^n \overline{S}_{ij}}{Dt^n} - \frac{\delta_{ij}}{3} \frac{\mathring{D}^n \overline{S}_{ll}}{Dt^n} \right]_t \tag{33}$$

where $\frac{\mathring{D}}{Dt}$ is an Oldroyd derivative which is defined as

$$\frac{\mathring{D}\overline{S}_{ij}}{Dt} \equiv \frac{D\overline{S}_{ij}}{Dt} - \frac{\partial \overline{u}_i}{\partial x_l} \overline{S}_{lj} - \frac{\partial \overline{u}_j}{\partial x_l} \overline{S}_{li}. \tag{34}$$

We note that Equation 33 has an infinite sum in it. While this may seem problematic, we actually only need the first few terms of the series as the higher order terms tend to have negligible effects. In fact for weak enough nonequilibrium effects we only need to consider the first term. Including additional terms would provide greater accuracy, however it also has the potential to lead to numerical instability. Luckily many flows only require $n = 1$ and so the instability can be avoided.

3 Implementation

3.1 Incorporating into the SKE

With all the theory developed in Section 2 we are now ready to insert the nonequilibrium closure into the SKE. Starting with Equation 17 we can consider the production (first term on the right) and rewrite it in terms of the anisotropy tensor

$$\tau_{ij}\frac{\partial \overline{u}_i}{\partial x_j} = -ka_{ij}\overline{S}_{ij}. \tag{35}$$

Applying Equation 32 then gives us $2\nu_T\overline{S}_{ij}\overline{S}_{ij}$, making the total equation

$$\frac{Dk}{Dt} = 2\nu_T\tilde{S}_{ij}\overline{S}_{ij} - \varepsilon + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_T}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right]. \tag{36}$$

We can do the same with the ε equation from Equation 18 to produce

$$\frac{D\varepsilon}{Dt} = 2C_{\varepsilon 1}C_\mu k\tilde{S}_{ij}\overline{S}_{ij} - C_{\varepsilon 2}\frac{\varepsilon^2}{k} + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_T}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right] \tag{37}$$

where again only the first term is affected by the new closure.

Finally, we need the Navier-Stokes equations if we actually want to get velocities out of our implementation. To do that we will need to modify Equation 3 a bit. Normally we would say $\overline{u_i'u_j'} = 2\nu_T\overline{S}_{ij} - \frac{2}{3}k\delta_{ij}$ and reduce the equation down to

$$\frac{D\overline{u}}{Dt} = -\frac{\partial \tilde{p}}{\partial x_i} + 2\left[\nu + \nu_T\right]\frac{\partial \overline{S}_{ij}}{\partial x_j} \tag{38}$$

where $\tilde{p} = \frac{\overline{p}}{\rho} + \frac{2}{3}k$. However since we need to apply our new closure we have to change $\overline{u_i'u_j'}$ to $\overline{u_i'u_j'} = 2\nu_T\tilde{S}_{ij} - \frac{2}{3}k\delta_{ij}$. This change gives us

$$\frac{D\overline{u}_i}{Dt} = -\frac{\partial \tilde{p}}{\partial x_i} + 2\frac{\partial}{\partial x_j}\left[\nu\overline{S}_{ij} + \nu_T\tilde{S}_{ij}\right] \tag{39}$$

which is again very similar to the equilibrium case. We just are no longer able to pull the strain rate out of the parentheses which might result in a bit of extra coding depending on how the original equilibrium model was implemented.

3.2 Wall Functions

We'll discuss the exact problem we are solving in Section 4.1, however before we do that we need to discuss corrections to the SKE in the presence of walls. Wilcox describes the corrections to the ε equation in [7]. Motivation for adding corrections arises from discrepancies in the predictions of most RANS models near walls, such as their failure to predict the correct peak in k.

Corrections are determined by considering the asymptotic behavior of the velocity perturbations as you approach the wall. In doing so a number of damping functions can be defined to modify the ε equation to be of the form

$$\overline{u}\frac{\partial \tilde{\varepsilon}}{\partial x} + \overline{v}\frac{\partial \tilde{\varepsilon}}{\partial y} = C_{\varepsilon 1}f_1\frac{\tilde{\varepsilon}}{k}\nu_T\left(\frac{\partial \overline{u}}{\partial y}\right)^2 - C_{\varepsilon 2}f_2\frac{\tilde{\varepsilon}^2}{k} + E + \frac{\partial}{\partial y}\left[\left(\nu + \frac{\nu_T}{\sigma_\varepsilon}\right)\frac{\partial \tilde{\varepsilon}}{\partial y}\right] \tag{40}$$

where $\varepsilon = \varepsilon_0 + \tilde{\varepsilon}$ and $\nu_T = C_\mu f_\mu\frac{k^2}{\tilde{\varepsilon}}$. There have been numerous efforts to determine good damping functions to use, however the one we will consider is the Launder-Sharma model developed in 1974. They specify

$$f_\mu = e^{-3.4/(1+Re_T/50)^2} \tag{41}$$

$$f_1 = 1 \tag{42}$$

$$f_2 = 1 - 0.3e^{-Re_T^2} \tag{43}$$

$$\varepsilon_0 = 2\nu\left(\frac{\partial\sqrt{k}}{\partial y}\right)^2 \tag{44}$$

$$E = 2\nu\nu_T\left(\frac{\partial^2\overline{u}}{\partial y^2}\right)^2 \tag{45}$$

$$C_{\varepsilon 1} = 1.44, \qquad C_{\varepsilon 2} = 1.92, \qquad C_\mu = 0.09, \qquad \sigma_k = 1.0, \qquad \sigma_\varepsilon = 1.3. \tag{46}$$

We incorporate these damping functions by adding them into Equation 37. Failure to add in these terms results in very poor results that do not match what one would expect from theory as was empirically discovered when attempting to implement the nonequilibrium SKE model (see Appendix A).

In addition to these damping functions we also included one for the k equation. This consisted of adding

$$D = \frac{1}{2k}\frac{\partial k}{\partial x_j}\frac{\partial k}{\partial x_j} \tag{47}$$

to the right hand side of Equation 36. The inclusion of this term was also as important as the damping functions in the ε equation.

3.3 Weak Forms

We will be implementing the nonequilibrium SKE model in a finite element code. In order to do that we must write our model equations in their weak forms. Doing this involves multiplying our governing equations by some test function (q, v_i, v^k, v^ε in our case) and then integrating (possibly by parts if we need to change what term has a derivative on it). Additionally we need to specify what terms are explicit and which are implicit. Mimicking FEniCS's (see Section 4.2) notation we will denote explicit terms with an _ after the variable, and leave implicit variables alone. So $u_{i\_}$ would be the explicit version of the velocities, and u_i the implicit version. Skipping all the intermediate algebra our nondimensionalized governing equations become:

$$F = \int \overline{u}_{j\_}\frac{\partial \overline{u}_i}{\partial x_j}v_i dV + \int \frac{2}{Re}\frac{\partial \overline{S}_{ij}}{\partial x_j}v_i dV + \int \frac{2}{Re}\nu_{T\_}\frac{\partial \tilde{S}_{ij}}{\partial x_j}v_i dV - \int \frac{\partial \tilde{p}}{\partial x_j}v_j dV - \int \frac{\partial \overline{u}_j}{\partial x_j}q dV - \int f_j v_j dV \tag{48}$$

$$F_k = \int \overline{u}_{j\_}\frac{\partial k}{\partial x_j}v^k dV + \int \frac{1}{Re}\left(1+\frac{\nu_T}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\frac{\partial v^k}{\partial x_j}dV - \int \frac{2\nu_{T\_}}{Re}\tilde{S}_{ij\_}\overline{S}_{ij}v^k dV + \int \frac{\varepsilon}{Re}\frac{k}{k_\_}v^k dV + \int \frac{1}{Re}D_\_ v^k dV \tag{49}$$

$$F_\varepsilon = \int \overline{u}_{j\_}\frac{\partial \varepsilon}{\partial x_j}v^\varepsilon dV + \int \frac{1}{Re}\left(1+\frac{\nu_k}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\frac{\partial v^\varepsilon}{\partial x_j}dV - \int \frac{2\nu_{T\_}}{Re}C_{\varepsilon 1}\frac{\varepsilon_\_}{k_\_}\tilde{S}_{ij\_}\overline{S}_{ij}v^\varepsilon dV + \int \frac{f_{2\_}}{Re}C_{\varepsilon 2}\varepsilon\frac{\varepsilon_\_}{k_\_}v^\varepsilon dV - \int \frac{1}{Re}Ev^\varepsilon dV \tag{50}$$

where

$$\tilde{S}_{ij} = \overline{S}_{ij} - \Lambda_m\left[\overline{u}_l\frac{\partial \overline{S}_{ij}}{\partial x_l} - \frac{\partial \overline{u}_i}{\partial x_l}\overline{S}_{lj} - \frac{\partial \overline{u}_j}{\partial x_l}\overline{S}_{li} - \frac{\delta_{ij}}{3}\overline{u}_k\frac{\partial \overline{S}_{ll}}{\partial x_k} + \frac{2}{3}\delta_{ij}\frac{\partial \overline{u}_l}{\partial x_k}\overline{S}_{kl}\right]. \tag{51}$$

In the case where we have $\tilde{S}_{ij\_}$ we take each of the $\overline{S}_{ij}$ terms in Equation 51 to be explicit. Additionally $\Lambda_M = 0.01$ as specified in [2].

Equation 48 is our Navier-Stokes equations along with our continuity equation (fifth term) and a forcing term (sixth term) to drive the flow. Our k equation, Equation 49 is fairly similar to what we would expect from the strong form, except for the fourth term where we have a $\frac{k}{k_\_}$ that would normally cancel out to become one. However we need an explicit term to solve for and so we can include this since it is effectively like multiplying by one. The ε equation, Equation 50, is also unsurprising, except for perhaps the fact that it is so similar to the k equation as is hopefully evident from the equations being written on top of each other. The damping functions are those from the Launder-Sharma model presented in Section 3.2.

4 Code

4.1 Problem Setup

In order to test our closure we will apply it to a steady channel flow. The benefit of this is that we should not expect to see any nonequilibrium effects. This will allow us to more easily debug any issues that may arise and compare to a working equilibrium SKE model. The downside to this is that we will be unable to test how well the nonequilibrium closure part of the model performs. In order to test that we would need a flow with some curvature such as a pulsed channel flow, a diffusor, or a bent channel.

Our problem is set up on an 8×50 rectangular mesh with a channel half-height of one. The grid is stretched near the walls according to a tangent function to provide more points near the wall and resulting in higher resolution. No slip boundary conditions are applied at the walls ($k = \varepsilon = \overline{u} = 0$), and periodic boundary conditions are used on the in and outflows. We have chosen a Reynolds number of $Re = 7890$ as well which should be high enough to see some turbulence.

4.2 FEniCS

FEniCS is a collection of Python packages such as DOLFIN, FFC, UFL, UFC, and FIAT among others. It provides a simple-to-use and extensive library to solve partial differential equations using the finite element method. Unlike more conventional languages like C, FORTRAN, or even normal Python, FEniCS is just-in-time (JIT) compiled, making it a bit faster to run and making up for some of the slowness (relative to something like FORTRAN) that is inherent to the language. FEniCS has been presented in [5] among other papers and can be downloaded for free at `http://fenicsproject.org/`.

Actually writing code in FEniCS is very intuitive, aside from a few peculiarities (like the difference between `dot` and `inner`), since gradients are taken with the `grad` function, divergences with `div`, etc. Everything is stored symbolically until it needs to be evaluated, making the finite elements part very simple as you just use the `Function` command to declare your test functions and the `solve` command to solve the system.

4.3 CBC.PDESys Solver

My first attempt at implementing the nonequilibrium model was in the CBC.PDESys found at `https://launchpad.net/cbcpdesys` ([6]). It is an extension to the FEniCS project that has implemented the SKE, Spalart-Allmaras, and V2F models. Additionally it has implemented several example problems such as a turbulent channel and diffusor. These would be great test cases for the nonequilibrium model. Unfortunately it has not been updated recently, so it doesn't work completely out of the box. In particular it does not run on OSX 10.11.1 which was not particularly obvious from the error messages. Aside from that it is not coded very intuitively (at least to me) and suffers from a lack of comments and documentation. All of this together with additional errors of unknown origin caused me to abandon the CBC.PDESys solver and use Ryan King's Krylov solver from [4] instead.

4.4 Ryan King's Solver

In order to start with a working SKE solver I used Ryan King's Krylov solver as presented in [4]. This code is also implemented using FEniCS, just in a simpler way than CBC.PDESys. It works by first iterating ten times before turning on the turbulence. At that point it then iterates using the equilibrium SKE model as coded by Ryan. Once that solution is obtained the nonequilibrium section kicks in and starts solving the problem from where the SKE left off. Additionally it uses successive over relaxation to speed up convergence and applies limiters to k and ε to ensure they remain positive.

5 Results

Using the problem setup described in Section 4.1 we obtained the following results after 235 equilibrium iterations 84 nonequilibrium iterations. Convergence was obtained in both the equilibrium and nonequilibrium cases.

Figure 4 shows a comparison of the turbulent kinetic energy k. We see very good agreement between the equilibrium SKE and nonequilibrium models and we see what is expected based on theory. Figure 5 shows similar agreement in ε. To show that there are slight differences between the models Figure 6 has zoomed in on the peak in dissipation. We see that the nonequilibrium model has a slightly steeper slope coming off of the peak, but the differences are very minimal.

Additionally we can compare the peak of the dissipation to the peak of the TKE. Figure 7 shows this comparison for the equilibrium case. We see that the dissipation peaks closer to the wall as is expected. Figure 8 shows the same plot with the nonequilibrium model where we again see the dissipation peaking nearer the wall than the TKE.

Figures 9 and 10 show the log layer on a normal and log axis. Additionally Figure 10 shows the theoretical law of the wall: $\frac{1}{\kappa}\ln y^+ + C$ with the accepted values of $\kappa = 0.41$ and $C = 5.5$. We don't see particularly good agreement with the law of the wall. Our slope is a bit too high and our C a bit low, but this is more likely a problem with the solver rather than the nonequilibrium closure.

6 Conclusion

We have presented a nonequilibrium closure that replaces the Boussinesq approximation commonly used in RANS models. Instead we have derived a relationship between the anisotropy tensor and an effective strain rate that depends on the straining history rather than the instantaneous strain rate. This model was then applied to a finite element solver on a SKE model implemented by Ryan King on a turbulent channel flow in FEniCS. Results between the

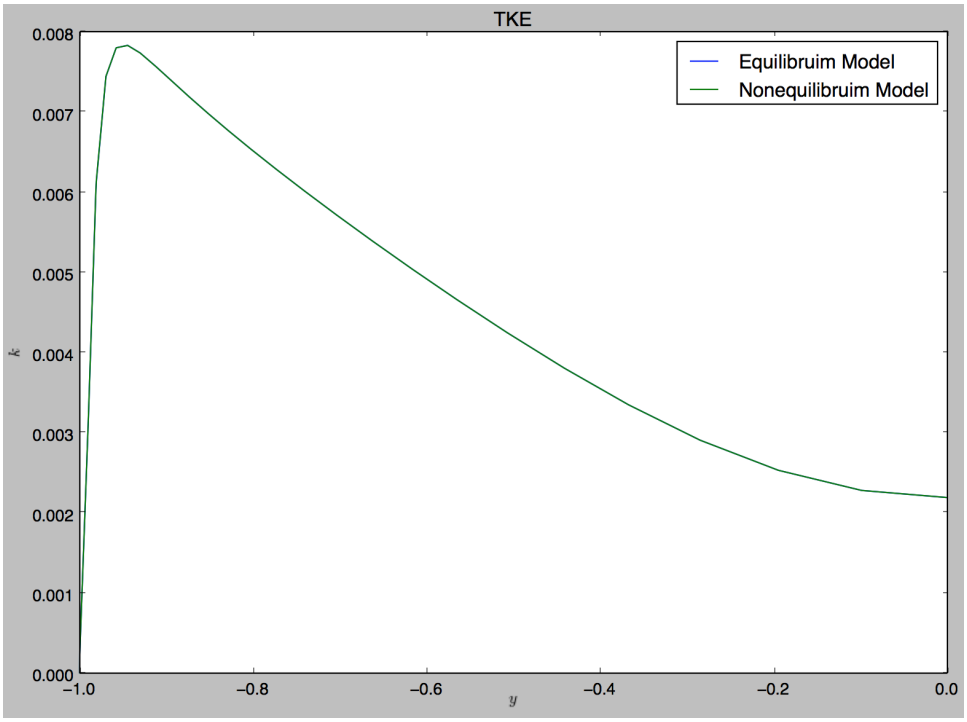

Figure 4: Comparison of turbulent kinetic energy k obtained with equilibrium SKE and nonequilibrium SKE models in a turbulent channel.

equilibrium and noneqilibrium models showed very good agreement as would be expected in this problem, indicating that the nonequilibrium solver was indeed correctly implemented.

In order to provide a bit better closure on the value of the nonequilibrium closure we have also included the results from [2] which shows the same comparison in Figure [1] with the addition of the nonequilibrium results. It shows much better agreement with the LES results indicating that the closure is indeed an improvement upon the original SKE model.

Once you understand what you are doing, this closure would likely be very easy to implement into any code that currently uses the Boussinesq approximation, particularly in FEniCS where calculating the $\tilde{S}_{ij}$ is made easier with its automatic differentiation functions. Because of this ease of implementation the nonequilibrium closure seems to be something that should always be used in the common models to account for nonequilibrium effects, unless they are being accounted for in another way.

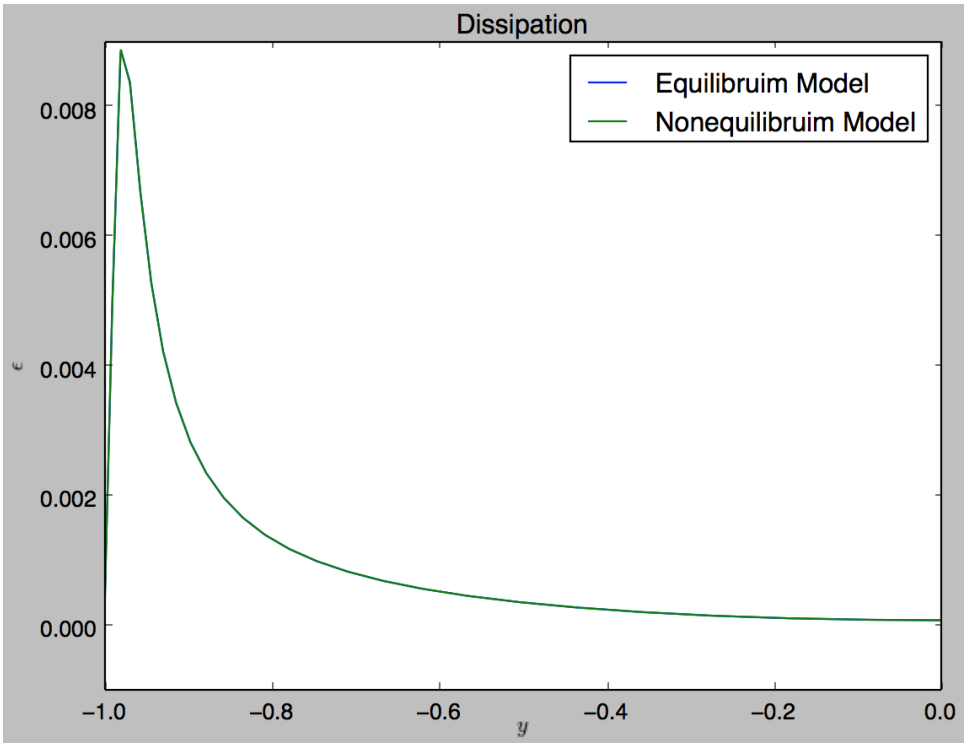

Figure 5: Comparison of dissipation ε obtained with equilibrium SKE and nonequilibrium SKE models in a turbulent channel.

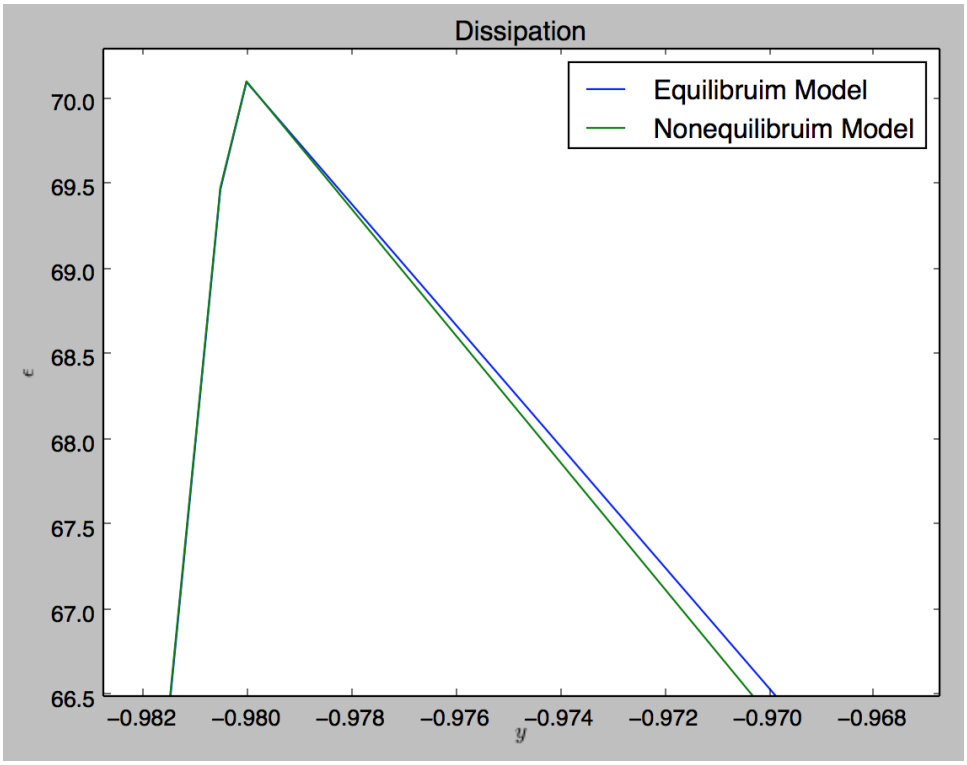

Figure 6: Comparison of dissipation ε obtained with equilibrium SKE and nonequilibrium SKE models in a turbulent channel zoomed in around the dissipation peak.

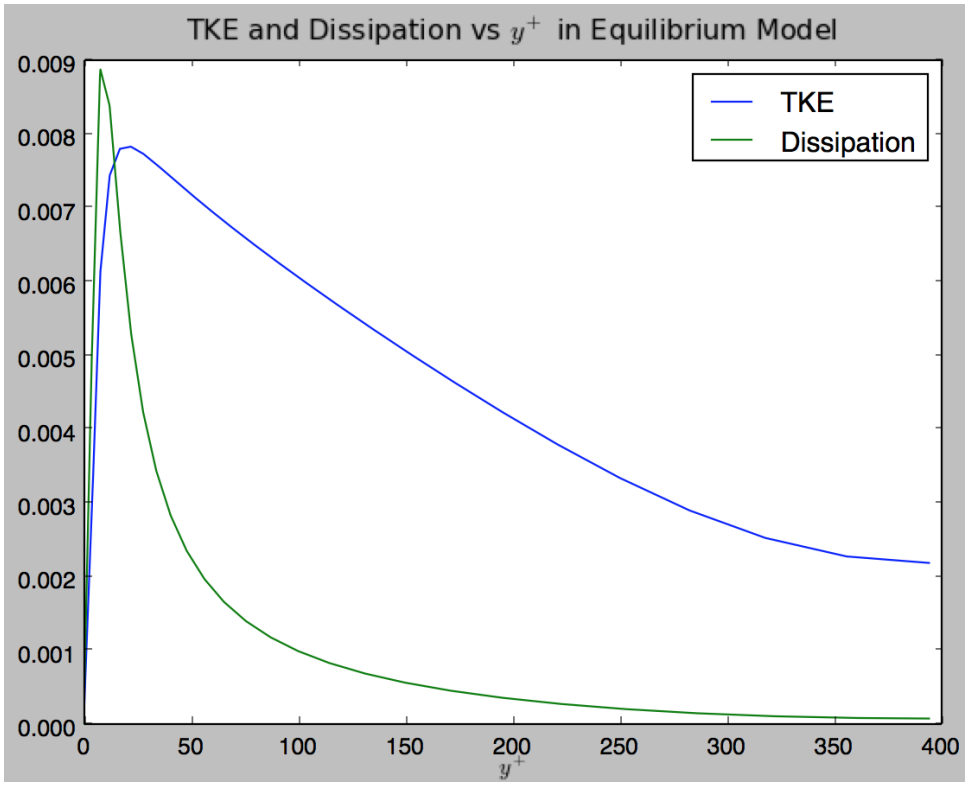

Figure 7: Comparison of turbulent kinetic energy and dissipation with the equilibrium model. The dissipation is expected to peak closer to the wall, which we see.

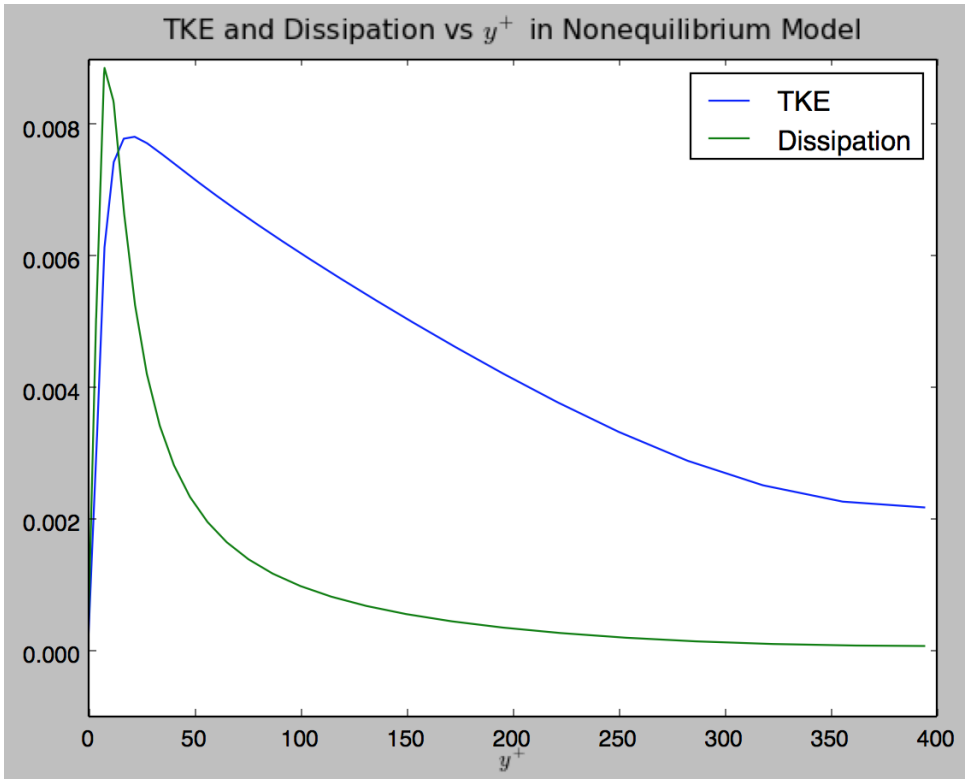

Figure 8: Comparison of turbulent kinetic energy and dissipation with the nonequilibrium model. The dissipation is expected to peak closer to the wall, which we see.

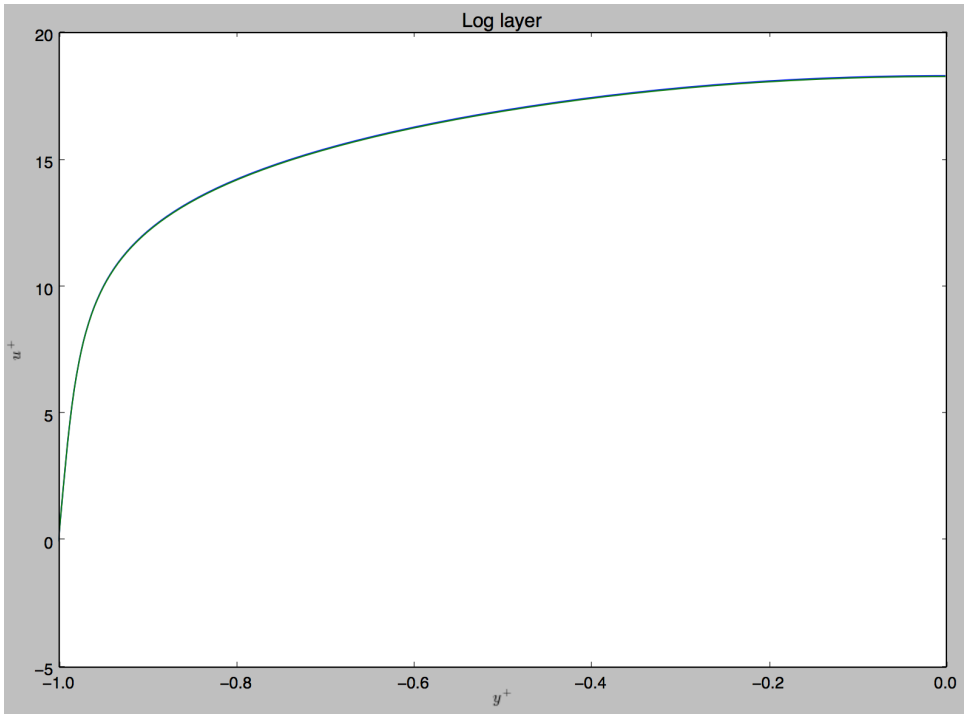

Figure 9: Comparison of log layer obtained with equilibrium SKE and nonequilibrium SKE models in a turbulent channel.

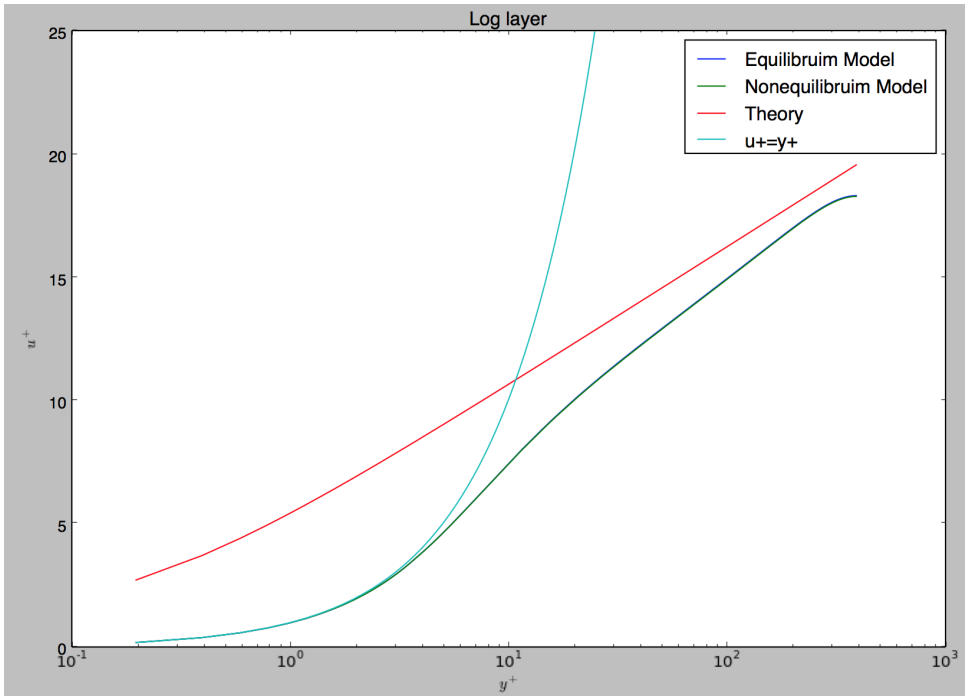

Figure 10: Comparison of log layer obtained with equilibrium SKE and nonequilibrium SKE models in a turbulent channel. Theory curve is given by $\frac{1}{\kappa}\ln y^+ + C$ with $\kappa = 0.41$ and $C = 5.5$.

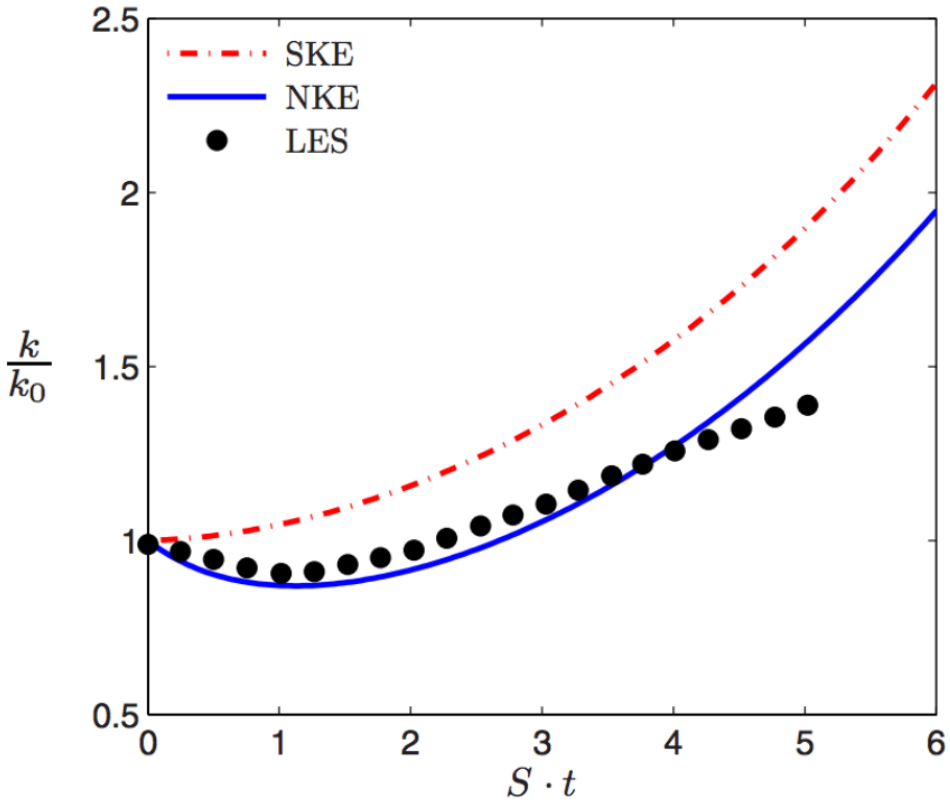

Figure 11: Comparison of equilibrium SKE, nonequilibrium SKE, and LES results. This figure appears in [2] who obtained the SKE results while the LES results are from [1].

References

[1] Bardina, Ferziger, and Reynolds. Improved turbulence models based on large eddy simulation of homogeneous, incompressible, turbulent flow. 1983.

[2] Peter E. Hamlington and Werner J. A. Dahm. Reynolds stress closure for nonequilibrium effects in turbulent flows. *Physics of Fluids*, 20(11):–, 2008.

[3] Peter E. Hamlington, Jörg Schumacher, and Werner J. A. Dahm. Local and nonlocal strain rate fields and vorticity alignment in turbulent flows. *Phys. Rev. E*, 77:026303, Feb 2008.

[4] Ryan King. Advanced turbulence modeling final project. 2015.

[5] Anders Logg, Kent-Andre Mardal, Garth N. Wells, et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012.

[6] Mikael Mortensen. Cbc.pdesys. 2011.

[7] D.C. Wilcox. *Turbulence Modeling for CFD*. DCW Industries, Incorporated, 1994.

[8] Dazhi Yu and Sharath Girimaji. Direct numerical simulations of homogeneous turbulence subject to periodic shear. *Journal of Fluid Mechanics*, 566, 2006.

A The Importance of Wall Functions

In order to illustrate the importance of wall functions we have included the original results obtained when the nonequilibrium closure was implemented without the functions described in Section 3.2. Figure 12 we see a comparison between the TKE for the equilibrium SKE model with wall corrections and the nonequilibrium SKE model without wall corrections. The peak for the equilibrium version is higher and closer to the wall than the nonequilibrium case. Figure 13 shows similar results for the dissipation. We know these results cannot be correct because this is a steady channel flow with no nonequilibrium effects. Figures 14 and 15 also show the log layer comparison for the same two models where again we see a discrepancy between the models when they should line up.

Hopefully this can be taken as a warning in case you see results similar to this to check that you are using wall functions before you waste time looking for other errors. Your time will be better spent implementing the whole, wall-corrected model the first time rather than trying to ignore the wall corrections and thinking that will help you debug your code more easily. By ignoring the wall corrections you are just adding in errors that are best fixed with the wall corrections.

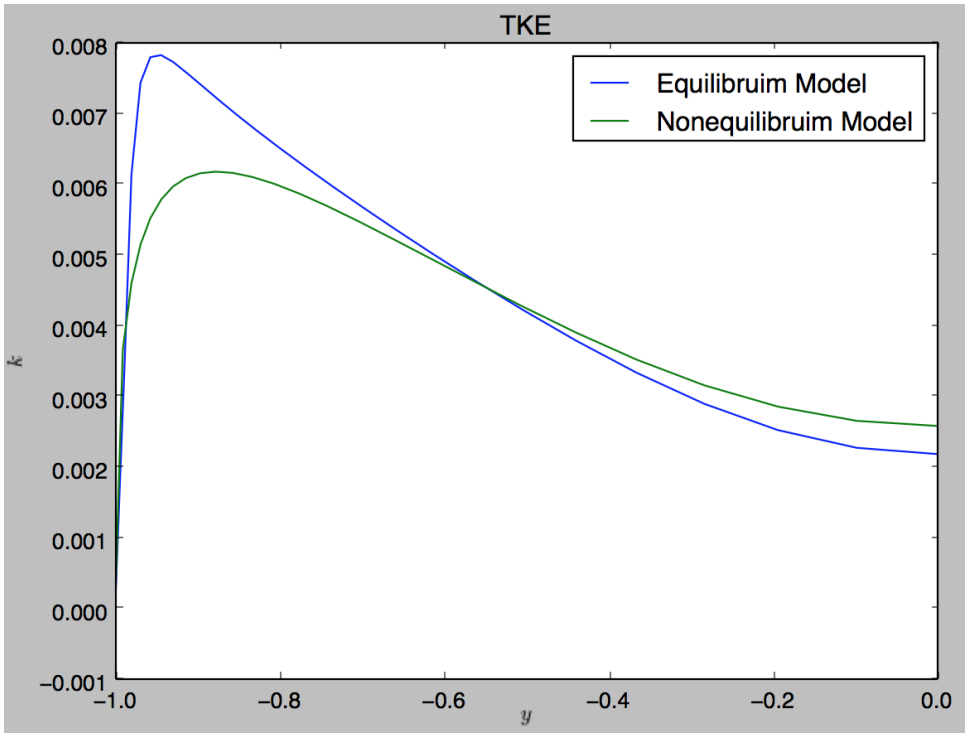

Figure 12: Comparison of k between the equilibrium SKE with wall corrections and the nonequilibrium SKE without wall corrections.

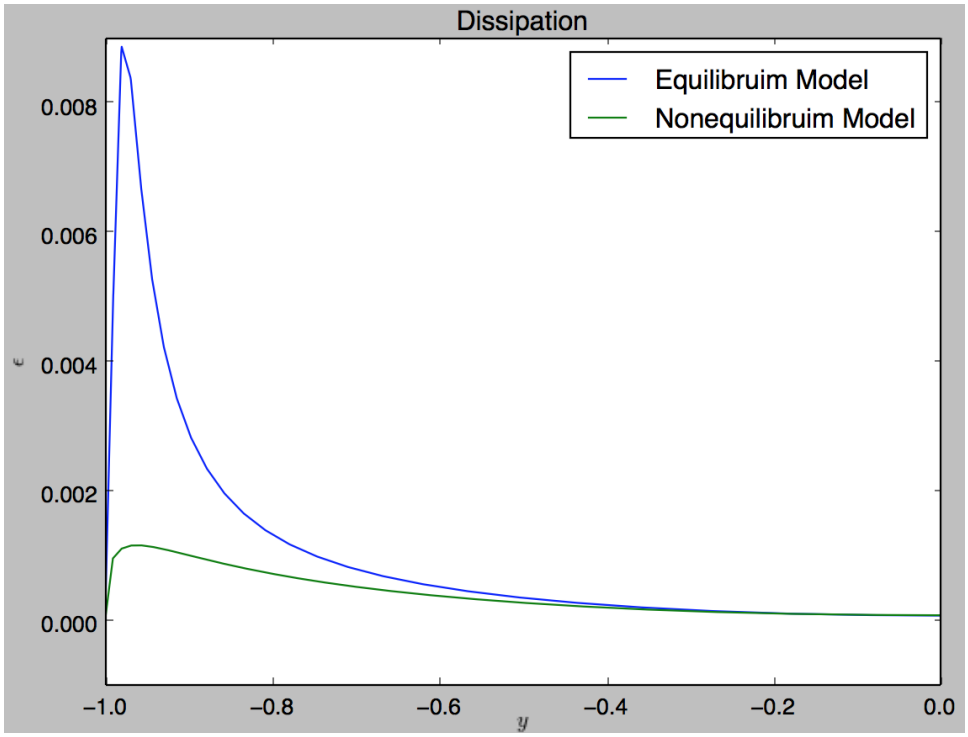

Figure 13: Comparison of ε between the equilibrium SKE with wall corrections and the nonequilibrium SKE without wall corrections.

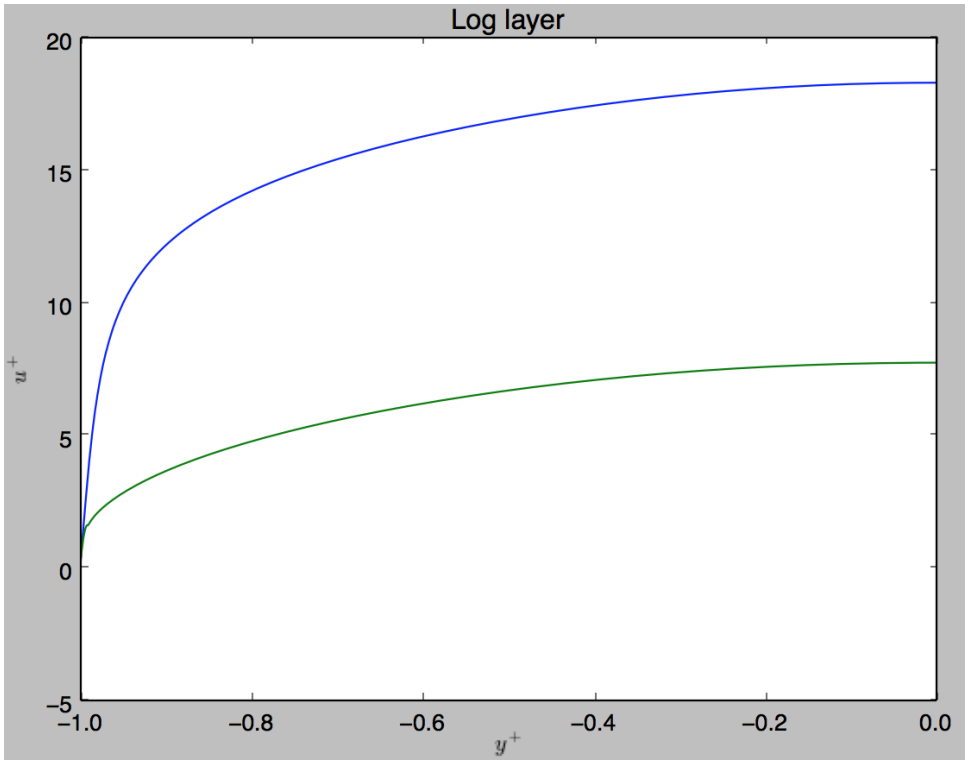

Figure 14: Comparison of the log layer between the equilibrium SKE with wall corrections and the nonequilibrium SKE without wall corrections.

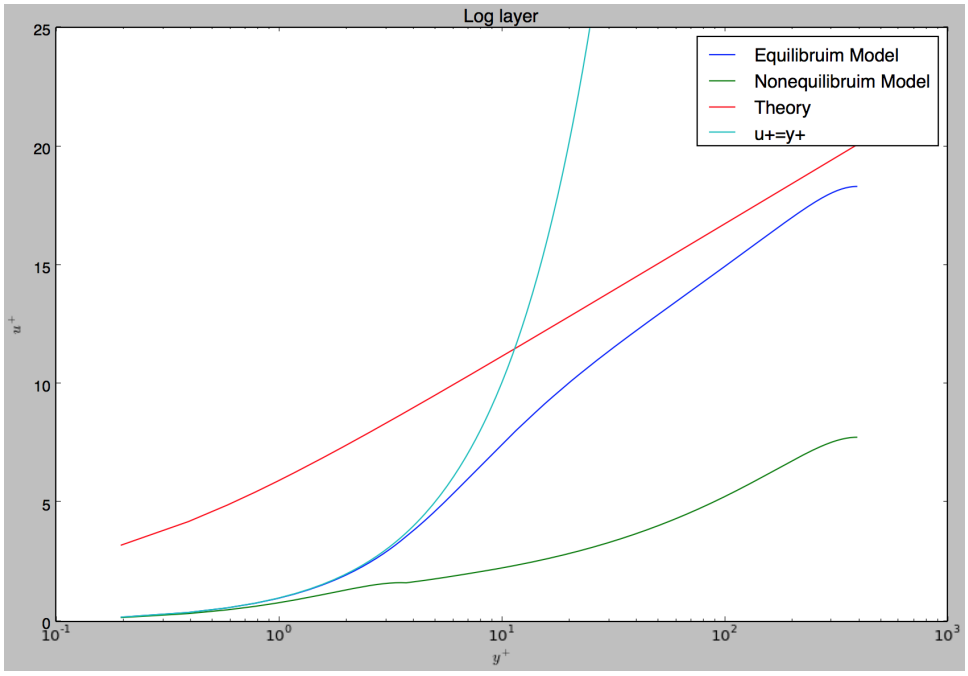

Figure 15: Comparison of the log layer between the equilibrium SKE with wall corrections and the nonequilibrium SKE without wall corrections.