CHALMERS
UNIVERSITY OF TECHNOLOGY

# Effect of Numerical Dissipation of the Predictive Accuracy of Wall-Modelled LES

Timofey Mukha

6 December 2019

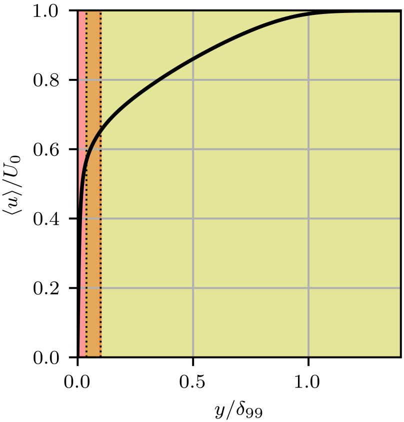
# Wall-Modelled LES

**Turbulence modelling approach**

- Resolved LES in the outer layer, scales $\sim \delta$.
- Turbulence below the overlap layer unresolved.
- Special modelling to compensate for that.
- Grid independent of $\delta_\nu$, i.e. "+"-units!

**Grid size scaling for a flat-plate TBL**

- Wall-resolved LES: $N \sim \mathrm{Re}^{1.85}$
- Wall-modelled LES: $N \sim \mathrm{Re}$



# Wall-Stress Modelling

## Compensating for the unresolved scales

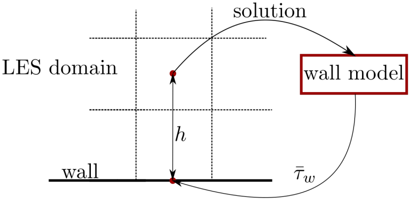
**For each time-step, for each wall-face**

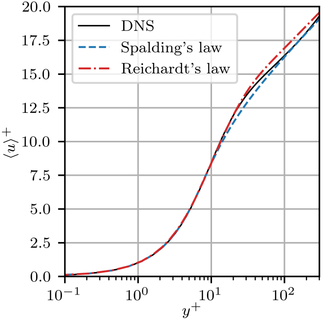
- Sample LES solution from distance $h$.
- A wall model predicts $\bar{\tau}_w$.
- $\bar{\tau}_w$ enforced at the face.
- By adding additional viscosity.

**What's inside the wall model?**

- An equation relating $\bar{\tau}_w$ to the solution.
- Example: Spalding's law

$$y^+ = \langle u \rangle^+ + e^{-\kappa B} \left[ e^{\kappa \langle u \rangle^+} - \sum_{m=0}^{3} \frac{(\kappa \langle u \rangle^+)^m}{m!} \right]$$



# Current Status in OpenFOAM

## libWallModelledLES

### Open-source library

- Several algebraic and ODE-based wall models.
- Assign $h$ on a per-face basis.
- Control all parameters.
- Convenient framework for adding new models.
- Sill in active development.
- Supports multiple version of OpenFOAM.

libwallmodelledles / wallModels / LOTWWallModelFvPatchScalarField.H

```
class LOTWWallModelFvPatchScalarField
:
    public wallModelFvPatchScalarField
{
protected:

    // Protected Data

        //- Pointer to the root finder
        autoPtr<RootFinder> rootFinder_;

        //- Pointer to the LOTW to be used
        autoPtr<LawOfTheWall> law_;

    // Protected Member Functions
        //- Write root finder and LOTW properties to stream
        virtual void writeLocalEntries(Ostream &) const;

        //- Calculate the turbulence viscosity
        virtual tmp<scalarField> calcNut() const;

        //- Calculate the friction velocity
        virtual tmp<scalarField> calcUTau(const scalarField & magGradU) const;


public:

    //- Runtime type information
    TypeName("LOTWWallModel");
```

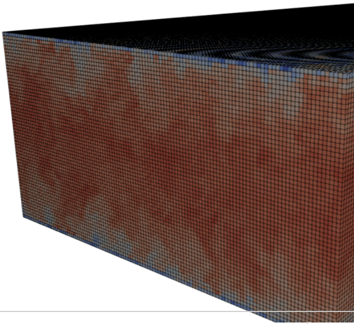
# Making WMLES as Accurate as Possible

Two main tracks

- Improving the accuracy of the wall model.
  - For conditions where the 'laws of the wall' don't hold.
- Determining other optimal modelling parameters
  - Mesh resolution and topology — controls truncation error size, but also min resolved eddy size.
  - SGS modelling — controls $\nu_{sgs}$.
  - Numerical schemes — can be more or less dissipative.
  - **All three control numerical dissipation and interact in a non-trivial way.**

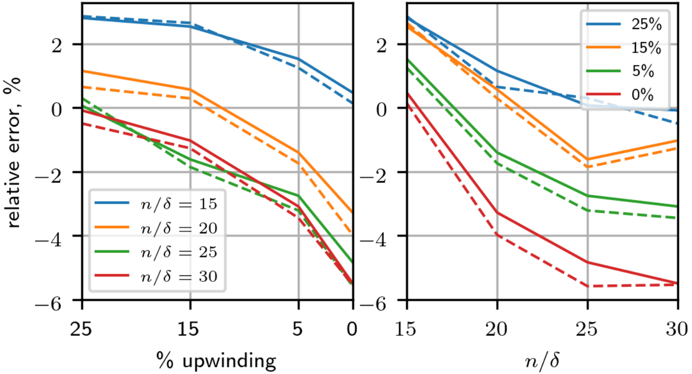
# Current Study

- Channel flow at $Re_b = U_b\delta/\nu = 125\,000$ used as the test case.
  - SGS model fixed to WALE, scheme dissipation and mesh size are varied.
  - Domain meshed with cubic cells, $n/\delta$ defines the resolution.
  - Considered $n/\delta$: 15, 20, 25, 30.
  - For convective fluxes, a linear blend of *linear* and *linearUpwind* schemes is used.
  - The weight of the *linearUpwind* scheme controls dissipation.
  - Considered weights: 25%, 15%, 5%, and 0%.
  - DNS data by Lee and Moser used as reference.



# Results

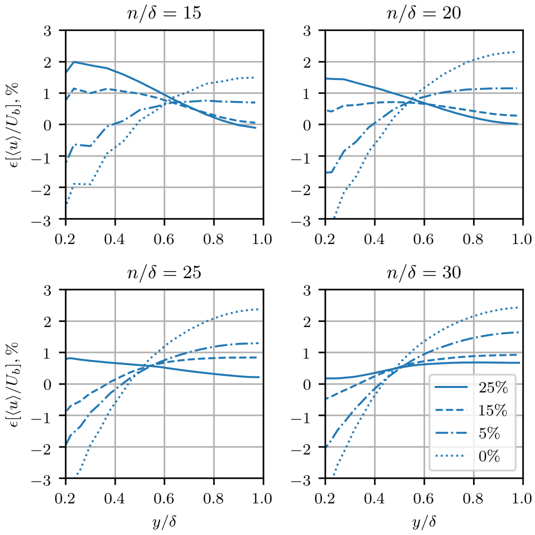
## Relative Errors in $u_\tau$




- Generally, less dissipation -> larger underprediction.
- $u_\tau$ error close to that of $\langle u(h) \rangle$.
- Wall model accuracy chiefly determined by the input velocity.

# Results

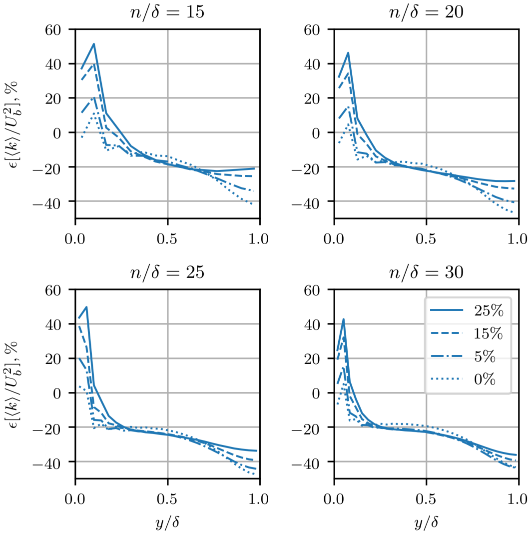
## Relative Errors in $\langle u \rangle$




- Monotounous dependency on % upwinding.
- Little to none improvement with $n/\delta$.
- Best result with $n/\delta = 30$, and 25% upwinding.
- But 15% slightly better considereing all $n/\delta$.

# Results

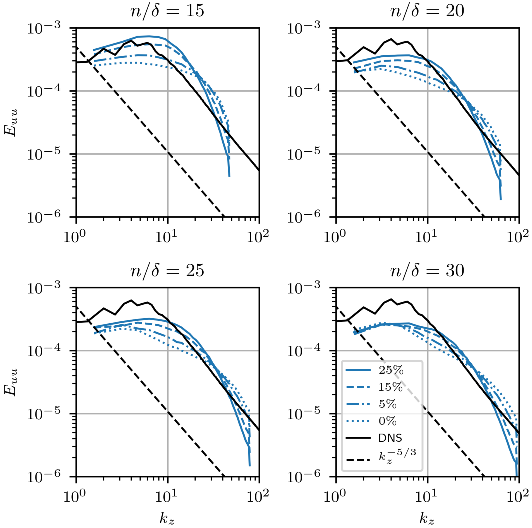
## Relative Errors in $\langle k \rangle$




- $y/\delta < 0.3$: over-prediction.
- $y/\delta > 0.3$: under-prediction.
- Less influence of upwinding for larger $n/\delta$.
- Less over-prediction for lower % upwinding.

# Results

## Energy spectra




- $y = 0.1\delta$
- No interial range at low $n/\delta$.
- Upwinding damps high-$k_z$ modes.
- Less influence of upwinding for larger $n/\delta$. (same as $\langle k \rangle$!)
- Increased energy in low $k_z$ with increased dissipation.

# Conclusions

- Effects of numerical dissipation on the accuracy of WMLES is considered.
- 16 channel flow simulations at $Re_b = 125\,000$ are performed.
- Mesh resolution and scheme dissipativity is altered.

- For $\langle u \rangle$, dissipation leads to better results and $n/\delta$ has almost no effect.
- For $\langle k \rangle$ and $E_{uu}$ large $n/\delta$ and less upwinding improve results.
- Suprisingly, increased dissipation leads to larger kinetic energy of large eddies.