Parameterizing Vertical Turbulent Mixing Coefficients In The Ocean Surface Boundary Layer Using Machine Learning

Neural Networks And Equation Discovery

Aakash Sane (Princeton University)

Brandon Reichl (NOAA – GFDL)
Alistair Adcroft (Princeton University)
Laure Zanna (NYU)

CESM OMWG Feb. 8th, 2024

m2lines.github.io

Key points:

1. Neural networks - within the existing energetics based physics framework to improve vertical diffusivity in the OSBL.
2. Networks implemented in MOM6.
3. JRA forced simulations performed: Bias reduction in shallow mixed layer depth and upper ocean stratification.

JAMES | Journal of Advances in Modeling Earth Systems®

Research Article | Open Access

Parameterizing Vertical Mixing Coefficients in the Ocean Surface Boundary Layer Using Neural Networks

Aakash Sane, Brandon G. Reichl, Alistair Adcroft, Laure Zanna

First published: 23 October 2023 | https://doi.org/10.1029/2023MS003890 |

4. Equation Discovery: Successes, challenges, results.

Vertical mixing (OSBL)

$$\langle w'\phi' \rangle = -\kappa \frac{\partial \langle \phi \rangle}{\partial z}$$

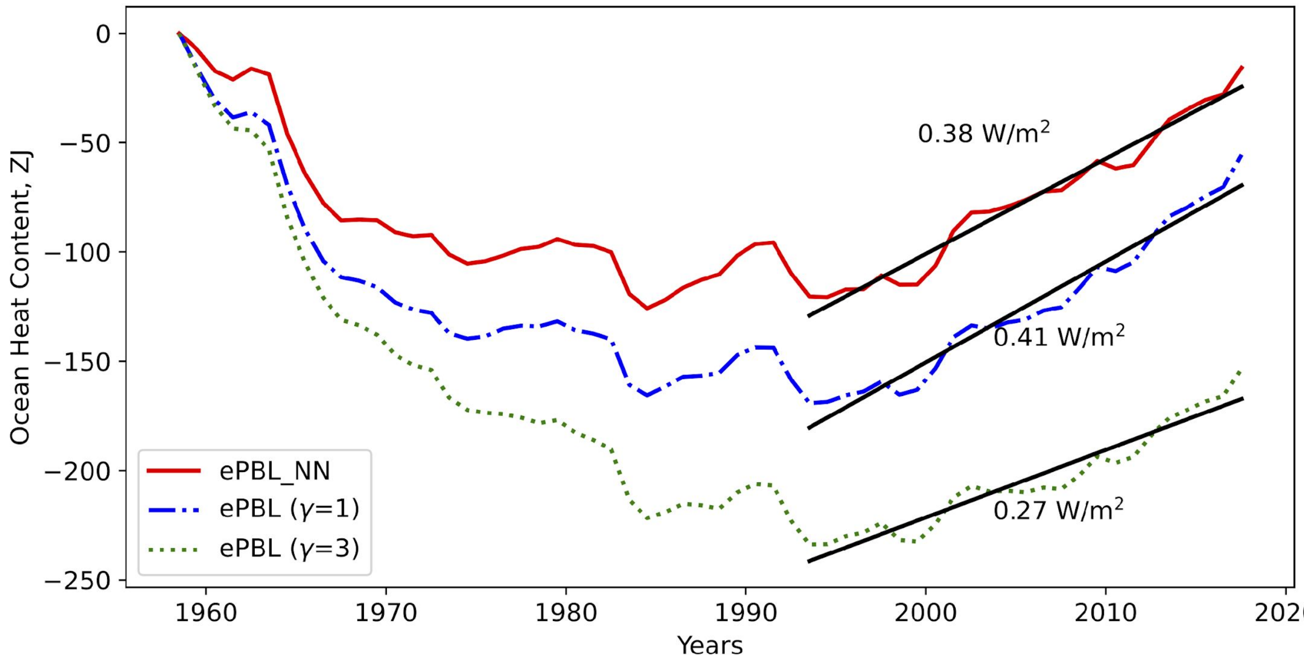
$$\kappa = g(z) \cdot L \cdot v$$

Vertical mixing

$$\langle w'\phi'\rangle = -\kappa \frac{\partial\langle\phi\rangle}{\partial z}$$

$$\kappa = g(z)\cdot L\cdot v$$

Changing ad-hoc parameters changes rate of ocean warming!

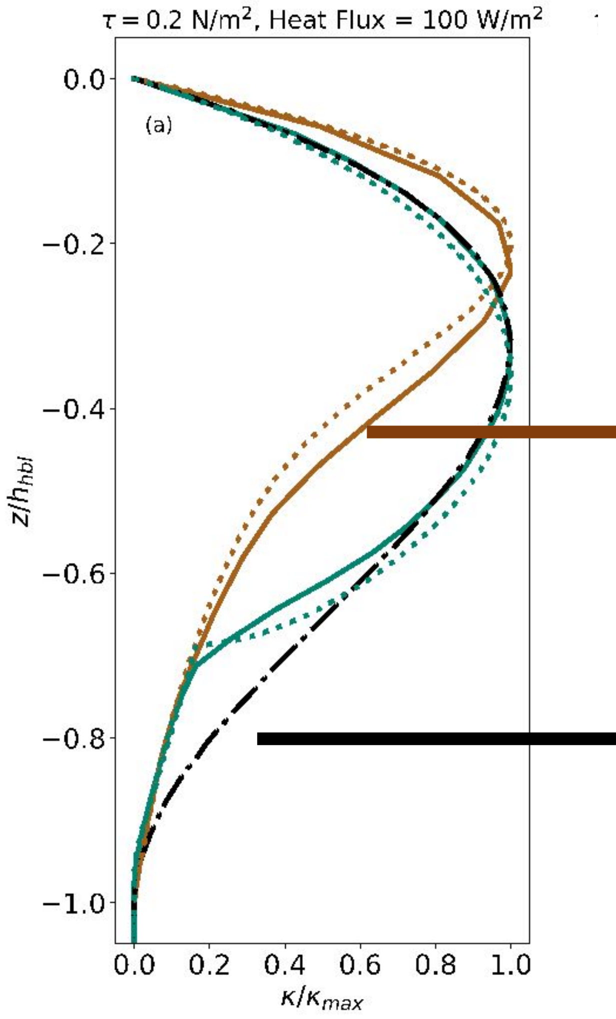


Sane et al. (2023), JAMES

Vertical mixing

$$\langle w'\phi'\rangle = -\kappa \frac{\partial\langle\phi\rangle}{\partial z}$$

$$\kappa = g(z)\cdot L\cdot v$$



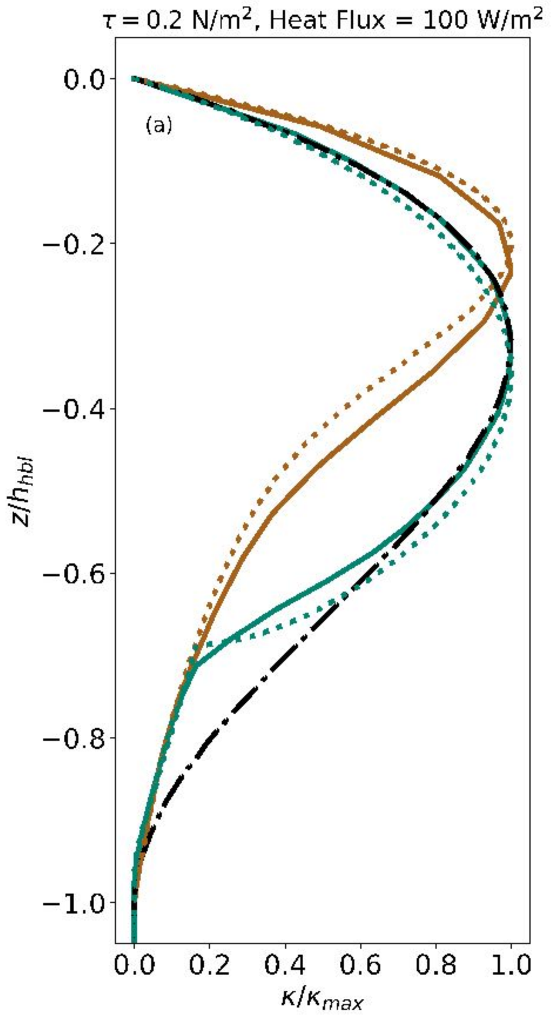
What we need

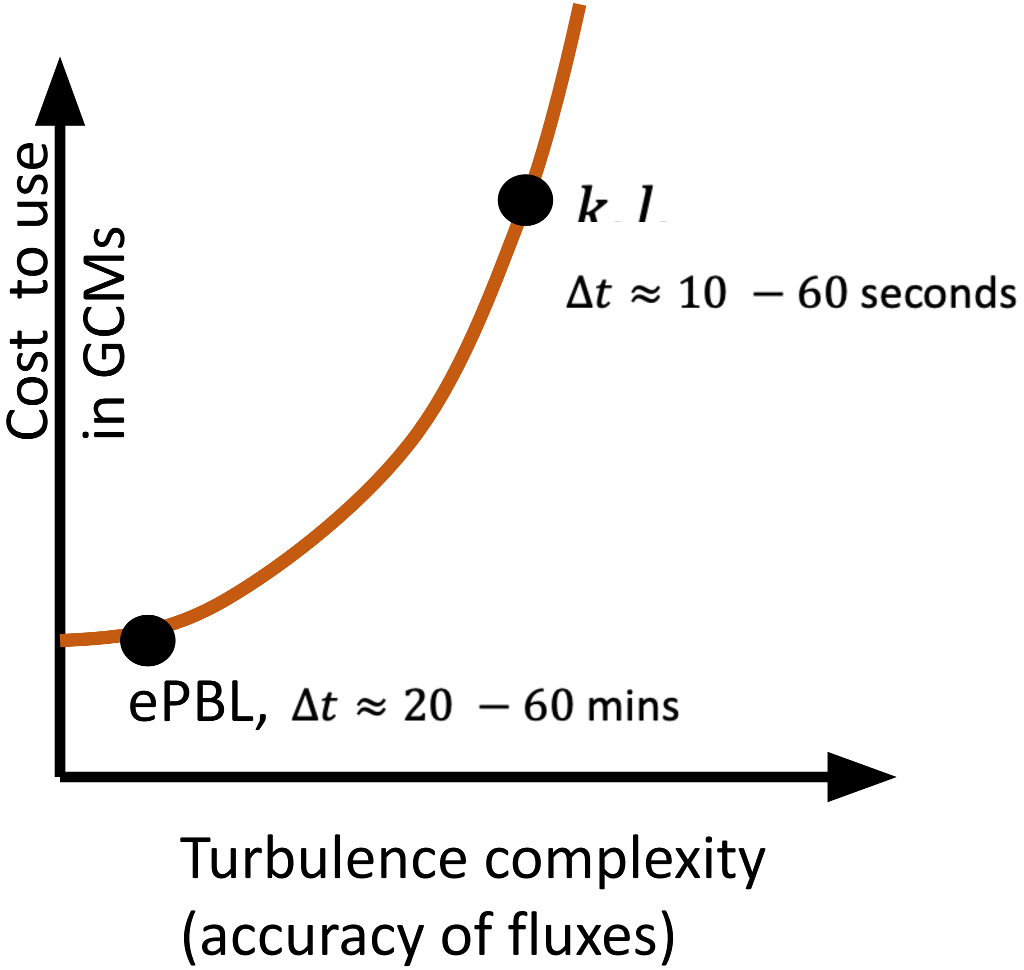
What we currently use in ePBL scheme (Reichl and Hallberg (2018).

Vertical mixing

$$\langle w'\phi' \rangle = -\kappa \frac{\partial \langle \phi \rangle}{\partial z}$$

$$\kappa = g(z) \cdot L \cdot v$$

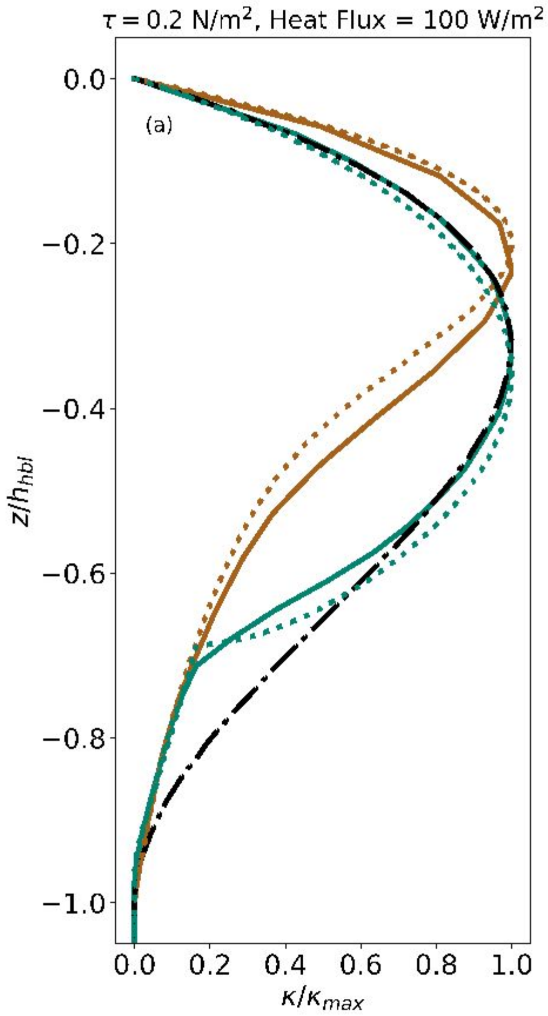


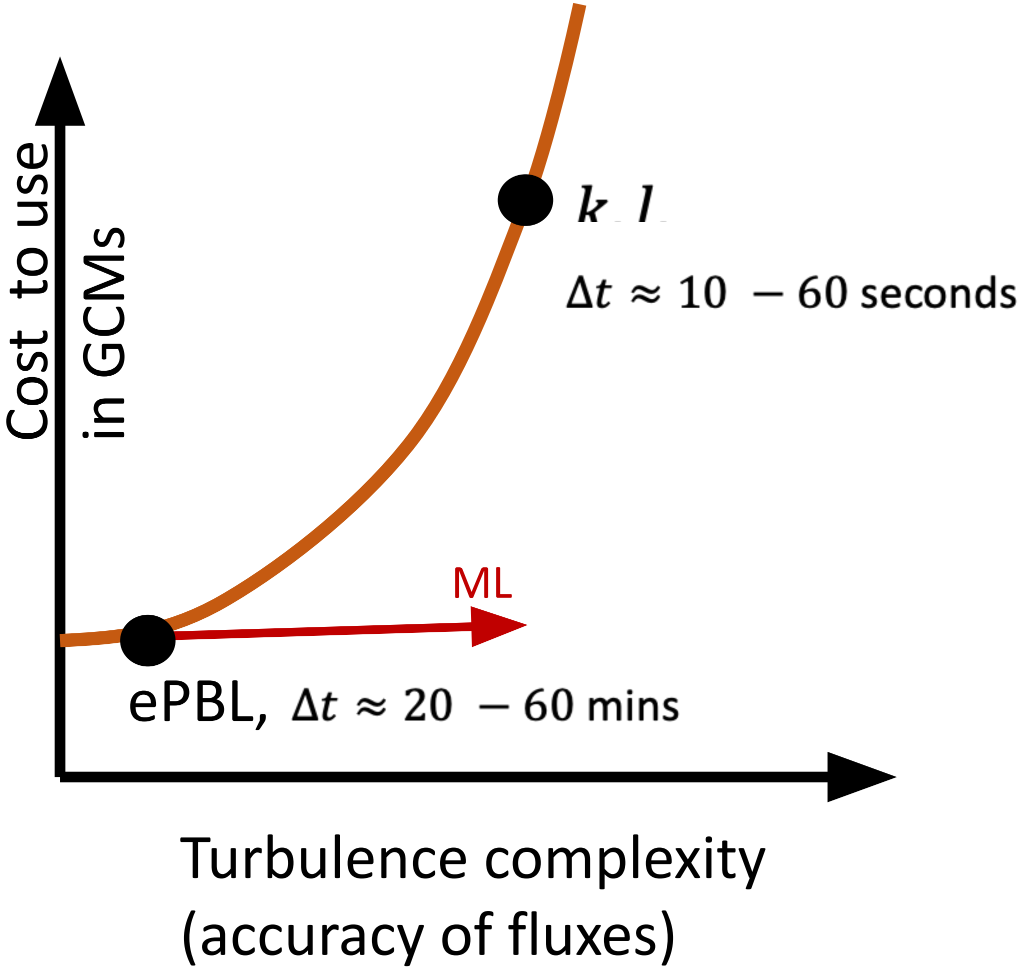


Vertical mixing

$$\langle w'\phi'\rangle = -\kappa \frac{\partial\langle\phi\rangle}{\partial z}$$

$$\kappa = g(z)\cdot L \cdot v$$



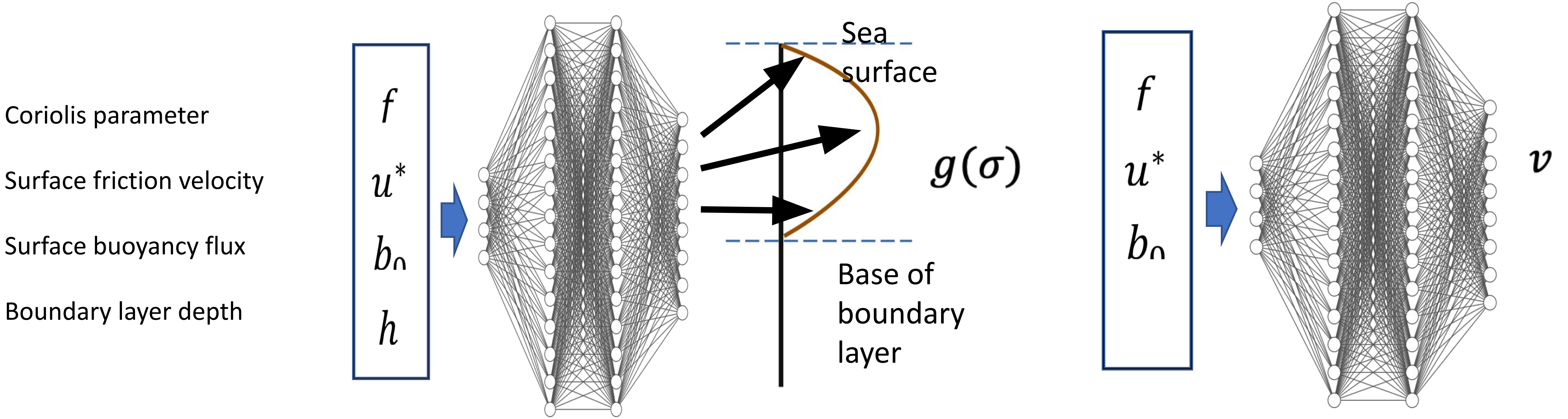


Neural network approach: $\kappa = g(\sigma) \cdot h \cdot v$

Training data: General Ocean Turbulence Model (1-D turbulence model),
Second moment closure schemes, inexpensive, κ is in output.

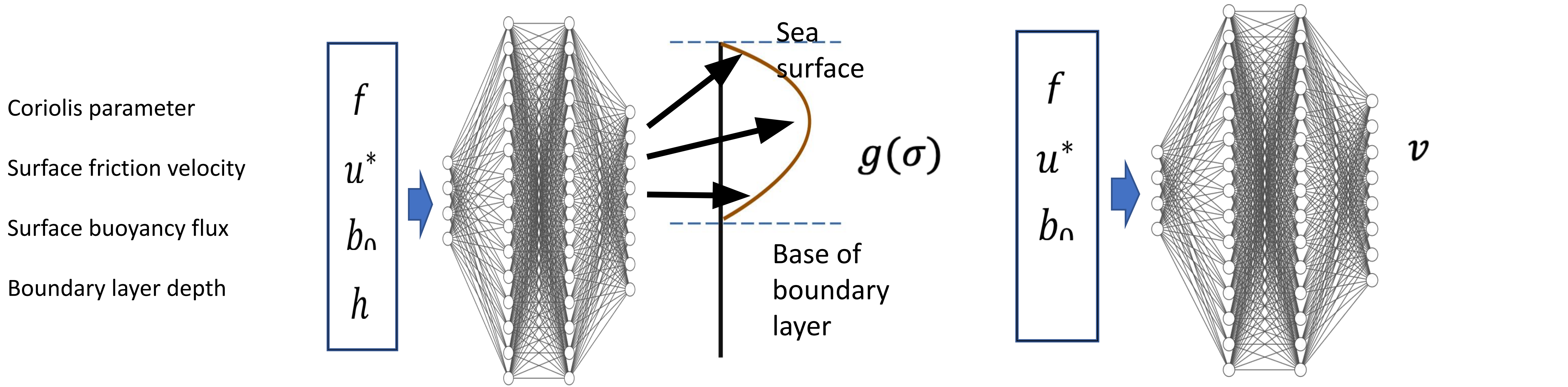
Neural network approach: $\kappa = g(\sigma) \cdot h \cdot v$

Training data: General Ocean Turbulence Model (1-D turbulence model), Second moment closure schemes, inexpensive, κ is in output.



Neural network approach: $\kappa = g(\sigma) \cdot h \cdot v$

Training data: General Ocean Turbulence Model (1-D turbulence model), Second moment closure schemes, inexpensive, κ is in output.

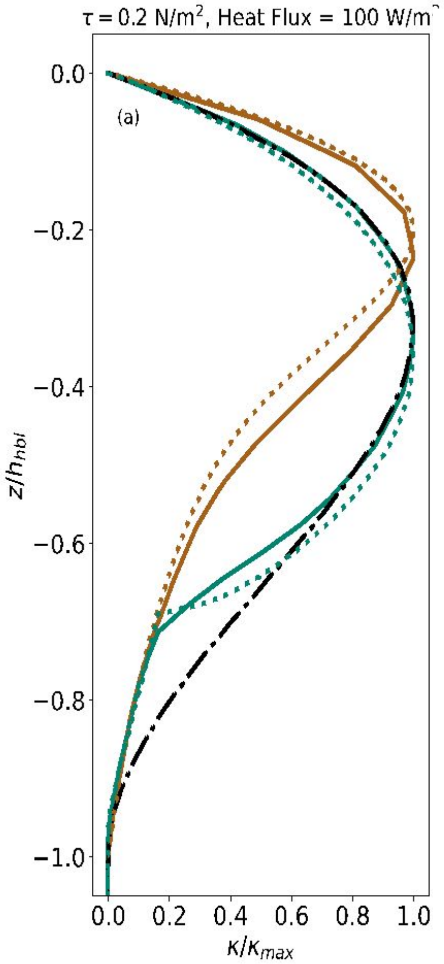


κ obtained from above is substituted in the ePBL mixing scheme (Reichl & Hallberg, 2018)
Showed reduction of some biases in OMIP experiments (Sane et al. 2023, JAMES)

Equation Discovery: (shape function)

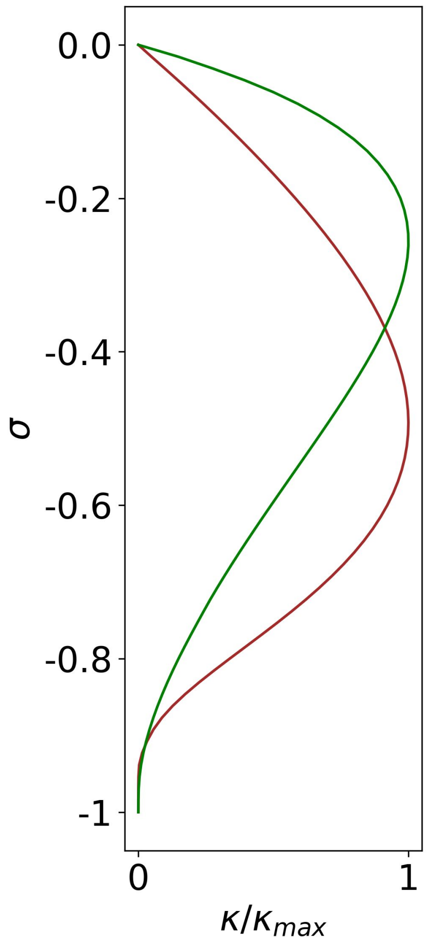
$$\kappa = \textcolor{red}{g(\sigma)} \cdot h \cdot v$$

$k - \epsilon$



GLS

Umlauf and Burchard (2003)



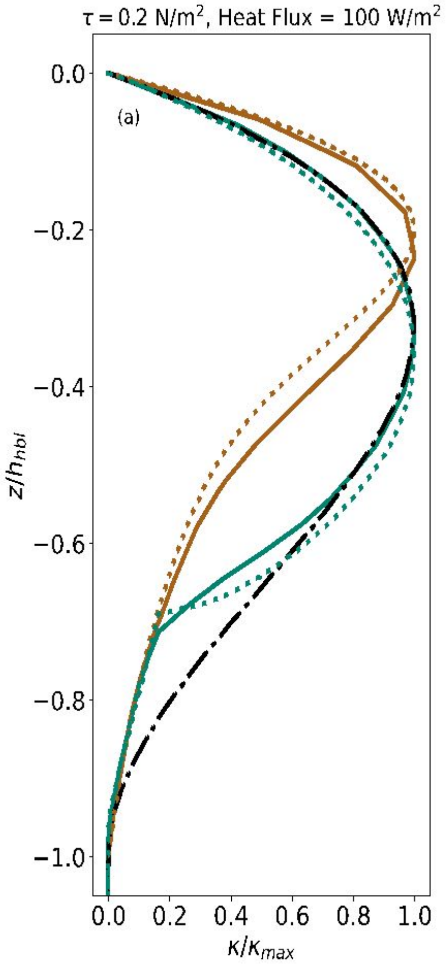
$$g = F\left(\sigma, \frac{h}{L_{Ek}}, \frac{h}{L_{\{MO\}}}\right)$$

Tried Genetic Programming, challenging to constrain to strictly positive values!

Equation Discovery: (shape function)

$$\kappa = g(\sigma) \cdot h \cdot v$$

$k - \epsilon$



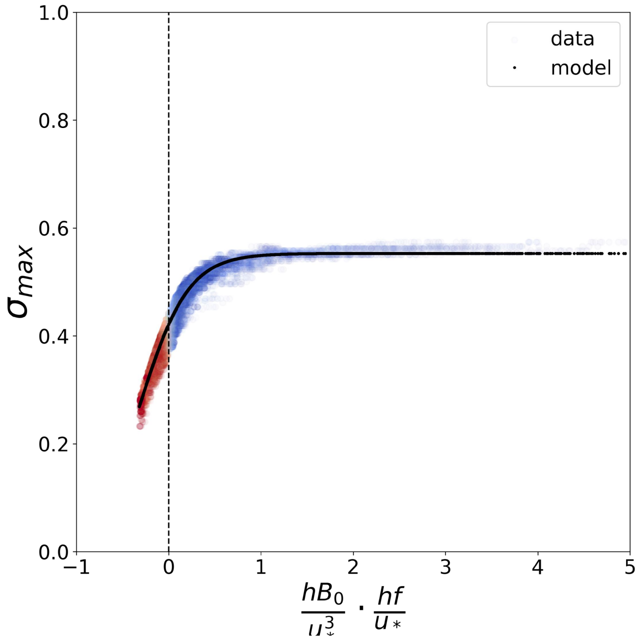
GLS

Umlauf and Burchard (2003)

σ_{max}

$$g = F\left(\sigma, \frac{h}{L_{Ek}}, \frac{h}{L_{\{MO\}}}\right)$$

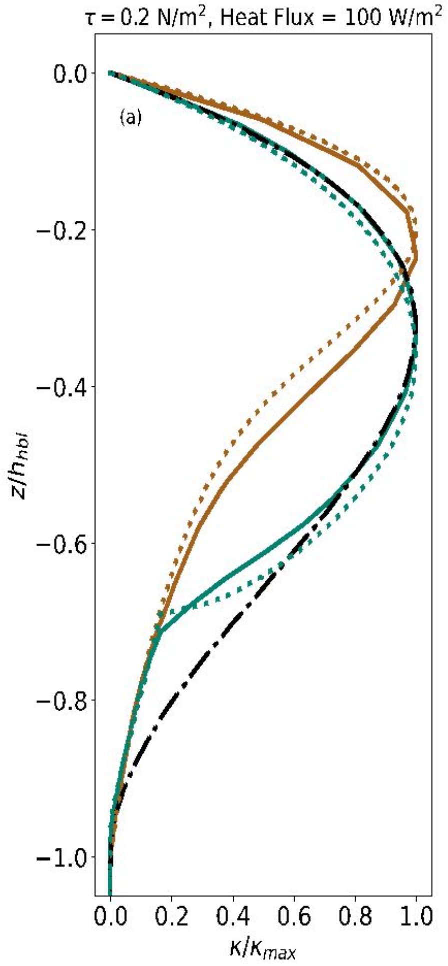
$$g = F\left(\sigma, \frac{h}{L_{Ek}}, \frac{h}{L_{\{MO\}}}\right)$$



Equation Discovery: (shape function)

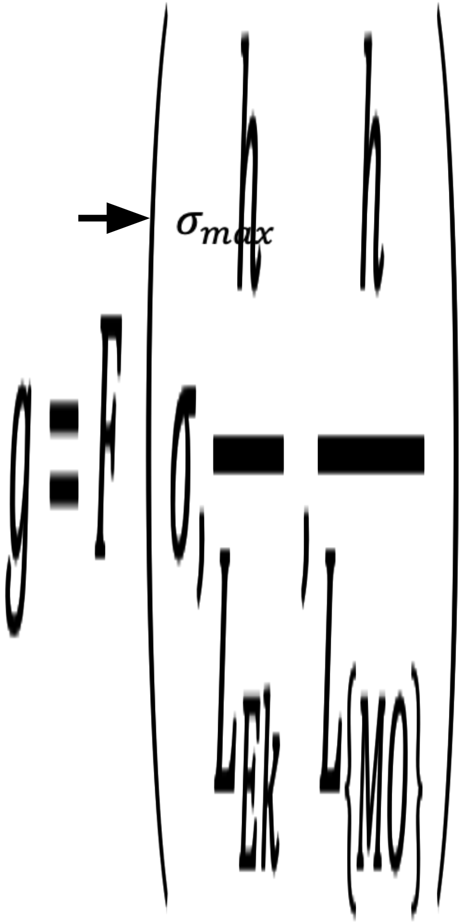
$$\kappa = g(\sigma) \cdot h \cdot v$$

$k - \epsilon$

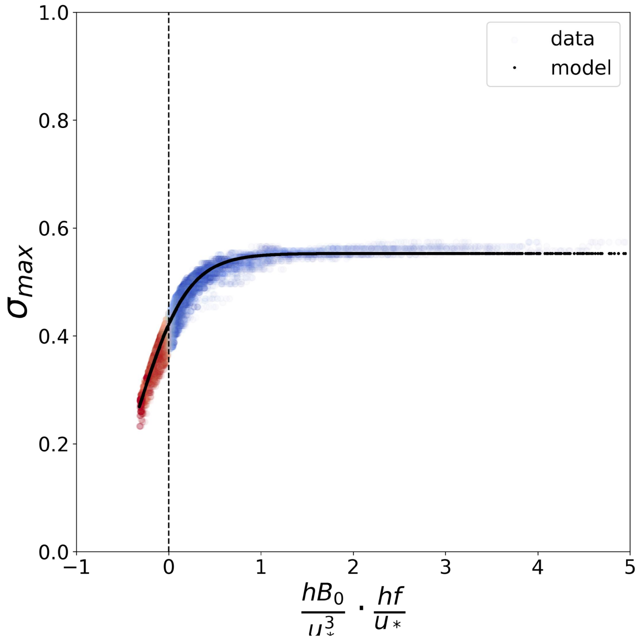


GLS

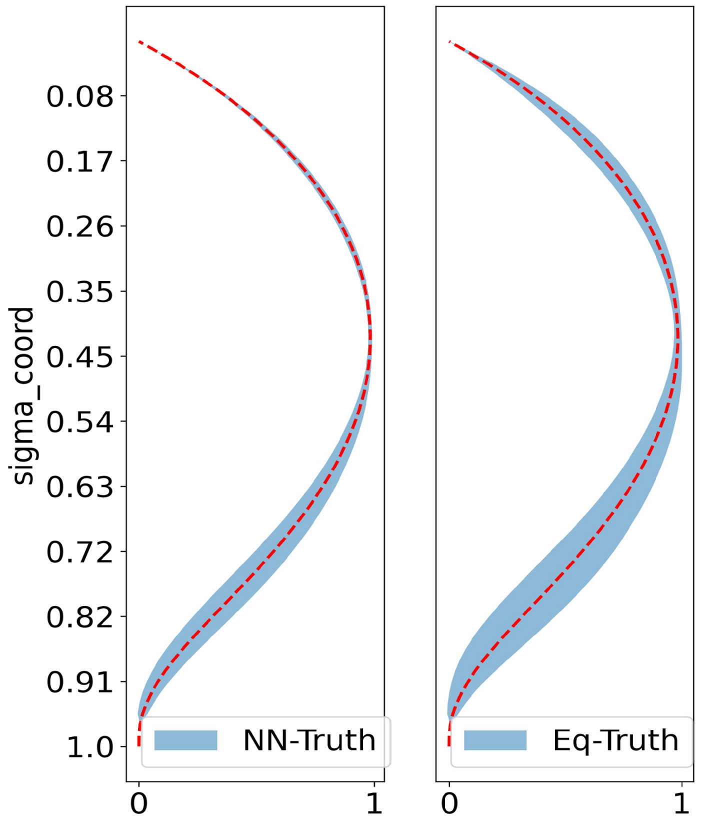
Umlauf and Burchard (2003)



$$g = F\left(\sigma, \frac{h}{L_{Ek}}, \frac{h}{L_{\{MO\}}}\right)$$



Error around the mean profile



Equation Discovery: (velocity)
using Genetic Programming

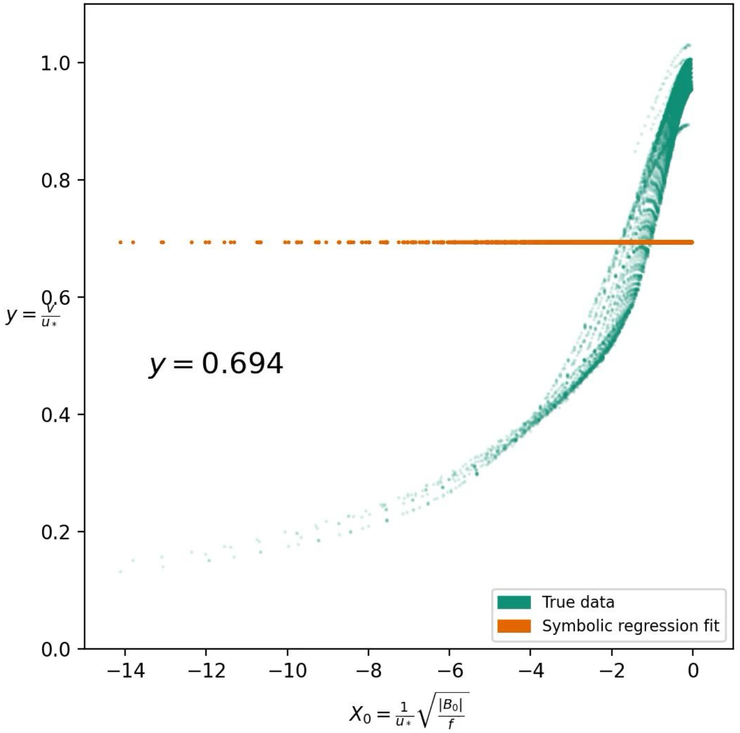
$$x = \left(\frac{1}{u_*}\right)\sqrt{\frac{|B|}{f}}$$

B: Surface buoyancy flux
u_*: Surface friction velocity
f: Coriolis parameter

Stable:

$$\frac{v}{u_*} = -\frac{0.1448}{x - 1.12 + \frac{0.476}{x - 0.6}} + c_0$$

$$\kappa = g(\sigma) \cdot h \cdot v$$

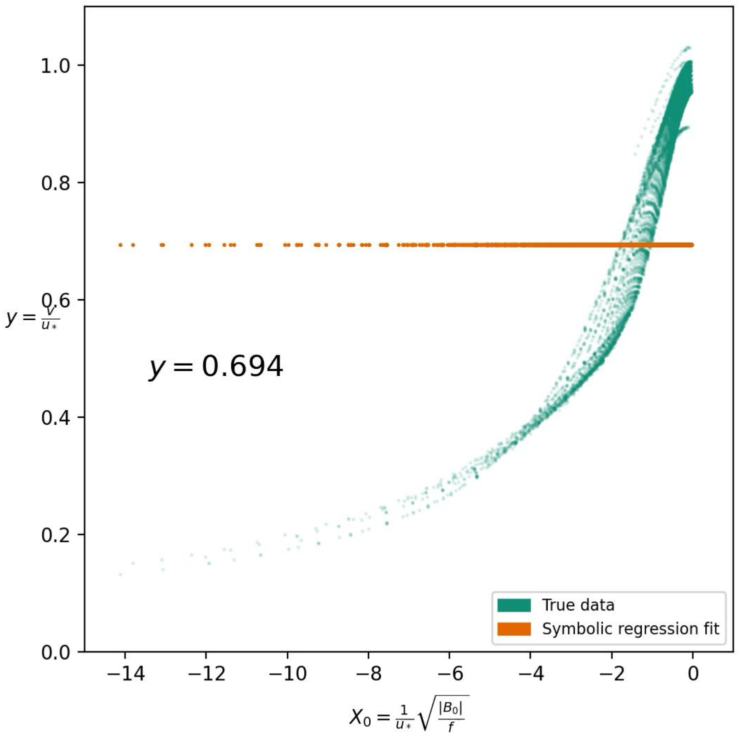


Equation Discovery: (velocity)
using Genetic Programming

$$\kappa = g(\sigma) \cdot h \cdot v$$

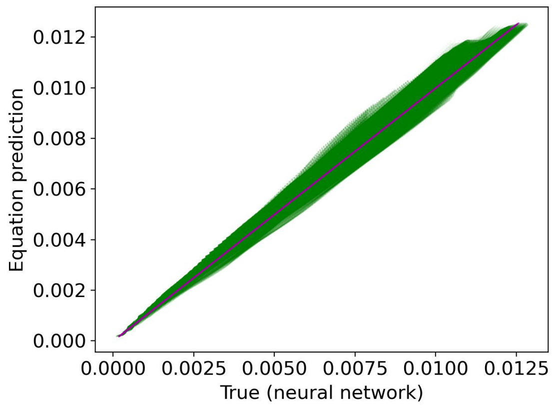
$$x = \left(\frac{1}{u_*}\right)\sqrt{\frac{|B|}{f}}$$

B: Surface buoyancy flux
u_*: Surface friction velocity
f: Coriolis parameter



Stable:

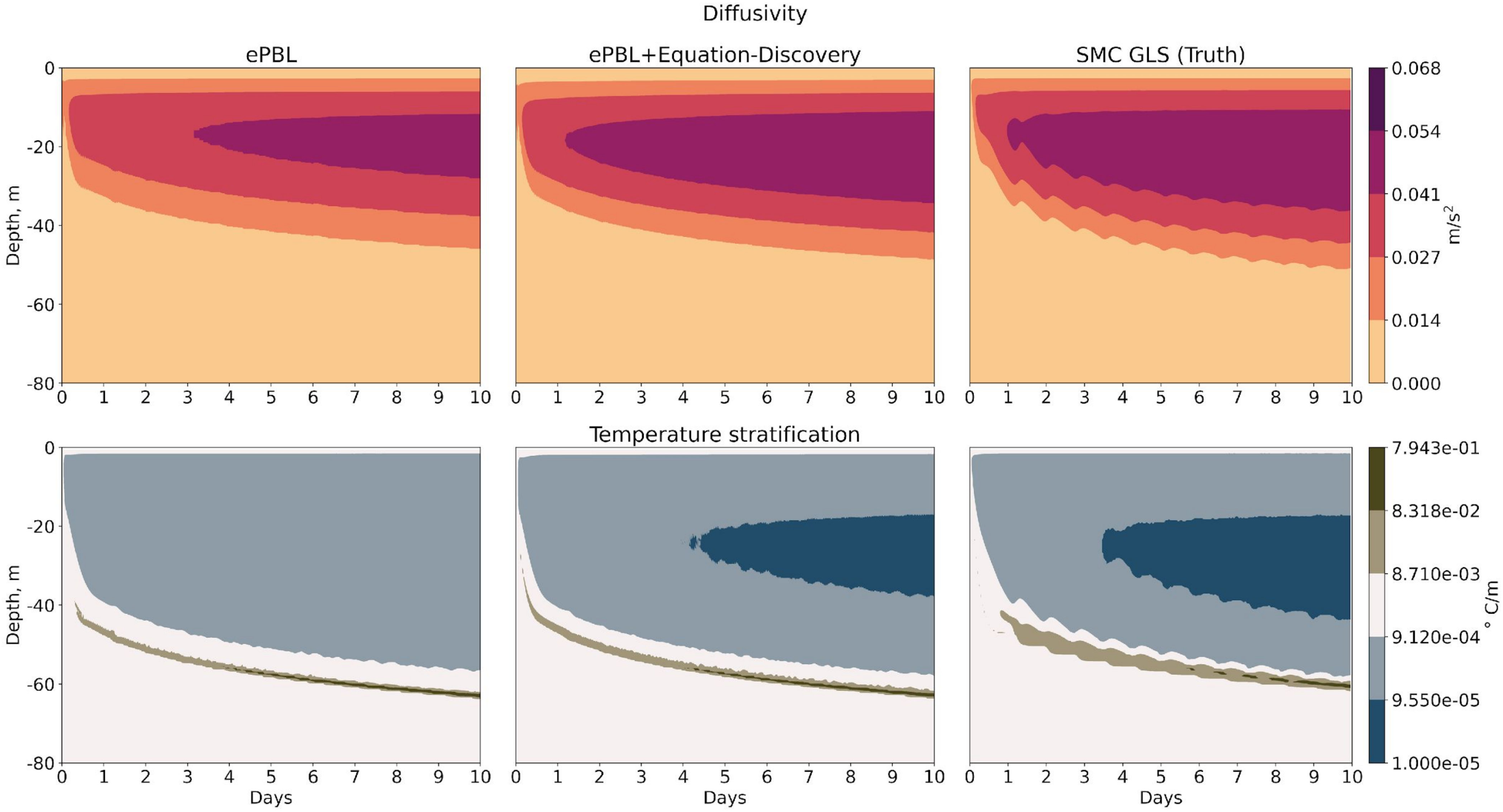
$$\frac{v}{u_*} = -\frac{0.1448}{x - 1.12 + \frac{0.476}{x - 0.6}} + c_0$$



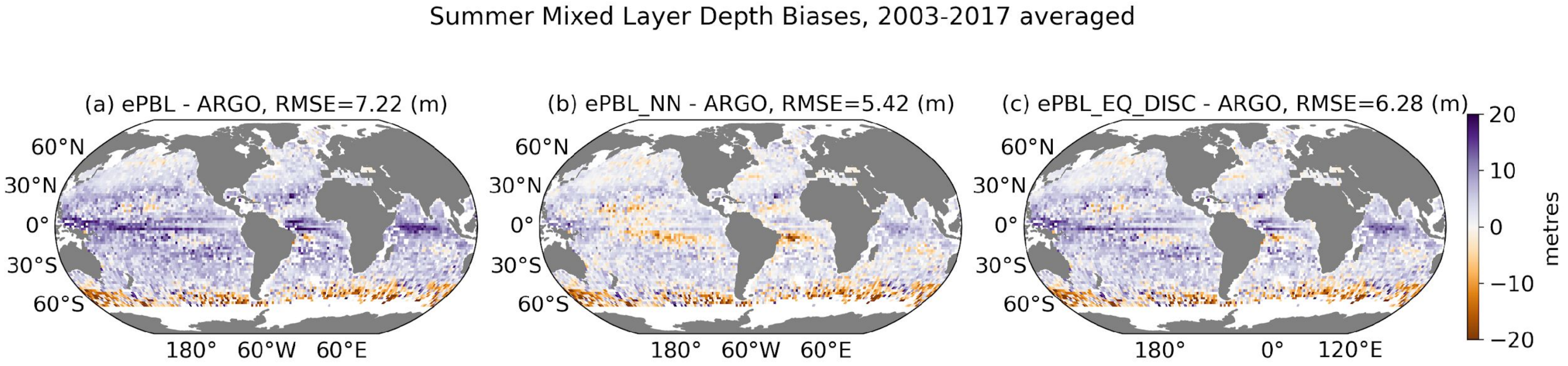
Unstable:

$$\frac{v}{u_*} = -\frac{0.1x\sqrt{\frac{f}{\Omega}}}{1 + \frac{\left(45\, e^{-\frac{f}{\Omega}} + 3.29\right)u^2 f}{B}} + c_1$$

Single Column model results



JRA forced results (MOM6, ¼ ^{o} grid, OM4): summer MLD (work in progress)

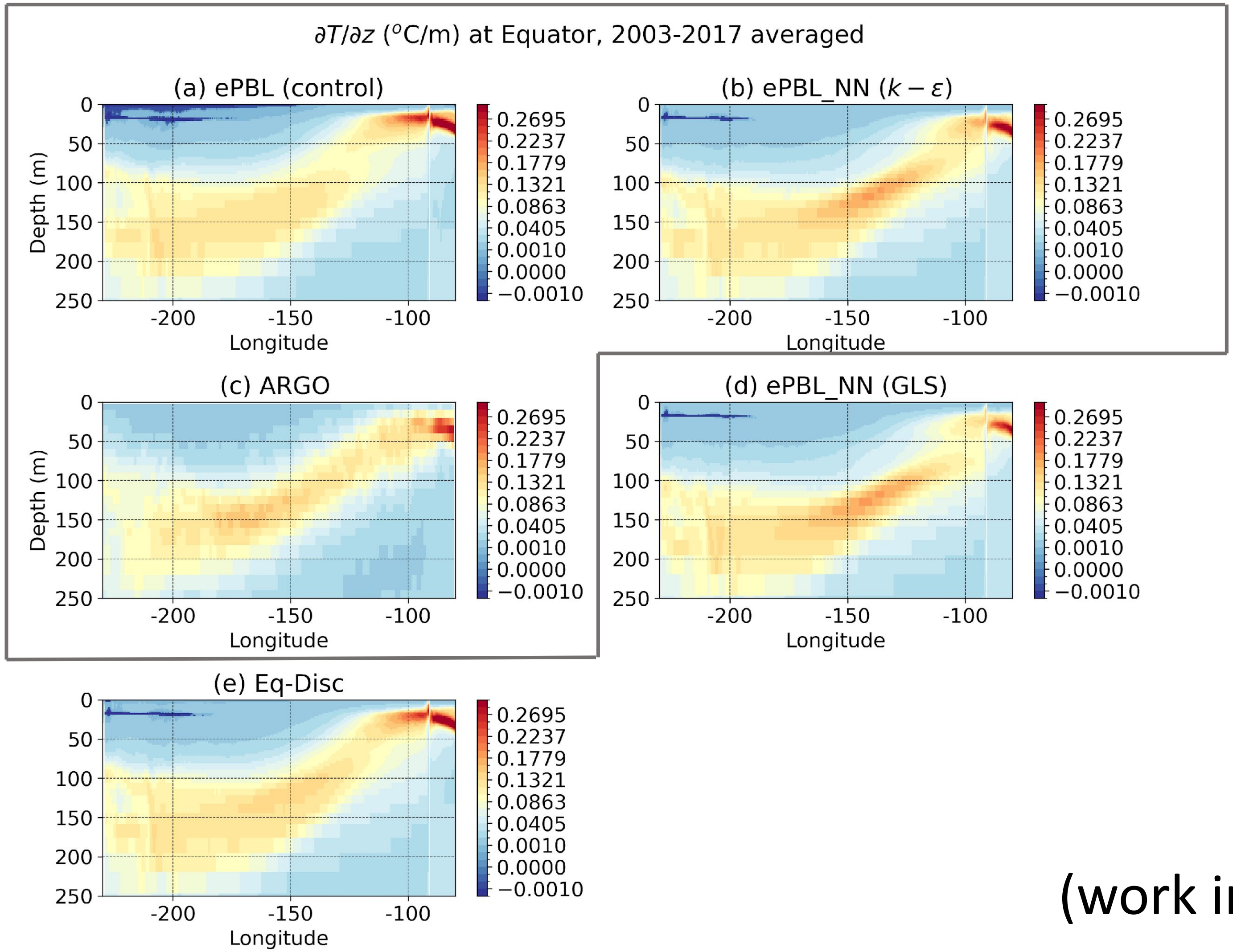


control

ePBL enhanced with neural network predicted diffusivity (Sane et al. 2023)

ePBL diffusivity discovered equations

JRA forced results (MOM6, ¼ ° grid, OM4): Pacific Equator vertical transect



Sane et al.
JAMES
2023

(work in progress)

Concluding remarks:

1. Neural networks - trained and implemented in MOM6
2. NN trained on Second Moment Closure schemes end up reducing some biases in OMIP experiments.
3. Symbolic regression (Genetic Programming) can be challenging.
4. With approximate equations that replace NNs, some bias reduction was observed.

Thank you