Equation Discovery for Sub-Grid Scale Terms

CEVE 543 Spring’22

Karan Jakhar (kj29@rice.edu)

Rice University

Mechanical Engineering Department

Houston, Texas 77005

1. Introduction

The resolution of climate models is limited by computational cost. Therefore, we must rely on parametrizations to represent processes occurring below the resolved scale (subgrid scale) by the models. Here, we focus on parametrization of momentum flux of gravity waves. In the Earth's atmosphere, gravity waves are a mechanism that produces the transfer of momentum from troposphere to the stratosphere and mesosphere. It is a wave moving through a stable layer of atmosphere. Analogy of how gravity wave looks like can be thought of as rock being thrown into the pond, where ripples migrate outwards from the point the rock hits the water [1].

Parametrization (or closure) have been developed for decades using the idealized theories of the bulk effect of the sub-grid scale process on the resolved scale [2]. This approach neglects certain physical effects. Data driven approaches using neural networks has shown promise in parametrization [3], although this black box approach may not necessarily obey laws of physics. A complementary route has been to use a white box approach to discover closed-form equations. Equation discovery has been attempted for ocean [4], and toy models like Navier-Stokes equation [5] have been successfully discovering the sub-grid scale terms or missing terms in PDEs.

I am using a data driven approach to parametrize the momentum flux of gravity waves in this work.

2. Methods

2.1 Filtering Procedure

To adequately resolve wind velocity $\mathbf{u} = (u, v, \omega)$ on a relatively coarser grid, a low-pass filtering operation ($\overline{\ \ }$) is performed. The resulting filtered velocity $\overline{\mathbf{u}}$ can be adequately resolved on a coarser grid. Here, u, v and ω are wind velocities in zonal (along the latitude), meridional (along the longitude) and along the altitude. Leonard [6] defines filtering operation as

$$\bar{u}(x) = G * u = \int_{-\infty}^{\infty} G(r)u(x-r)dr,$$

here, G is the filter function. Gaussian filter is commonly used with mean zero and variance $\sigma = \frac{1}{2}\Delta^2$. Here Δ is the filter size, this dictates the size of smallest flow structures to be resolved. The velocity fields can be decomposed into filtered (resolved) and residual part

$$u = \bar{u} + u'$$

In this work we are interested in modeling Reynolds stress τ (one of the parameters of momentum flux of gravity waves)

$$\tau = \overline{u'\omega'}.$$

The Reynolds's stress is parametrized for coarse grained models.

2.2 Filtered data

We are using the data generated by the Weather Research and Forecasting (WRF) model [7]. The model realistically (as real as possible) simulates the fully compressible and non-hydrostatic atmosphere of the Earth. Due to its high-resolution, we do not need to parametrize momentum flux due to gravity waves in WRF model. Hence, we can use this data to model Reynolds stress.

The WRF data has spatial resolution of $1024 \times 1024 \times 100$ (x, y, z) grid points. Here, (x, y, z) are spatial grid points along latitude, longitude and altitude respectively. Each grid point is 3 km apart along latitude and longitude, and 500 m apart along altitude.

We calculate the Reynolds stress and the filtered wind speed $(\bar{u}, \bar{v}, \bar{\omega})$ using gaussian filter function. Two sets of data are generated using the filter width $\Delta = 250$ km and 1000 km. This data of filtered wind velocities and Reynold's stress is used for the equation discovery.

Figure 1 shows a snapshot of τ and respective $\bar{u}$ and $\bar{\omega}$. Please note that the data is in NetCDF4 format, and we used 'netcdf4' package in python to read it.

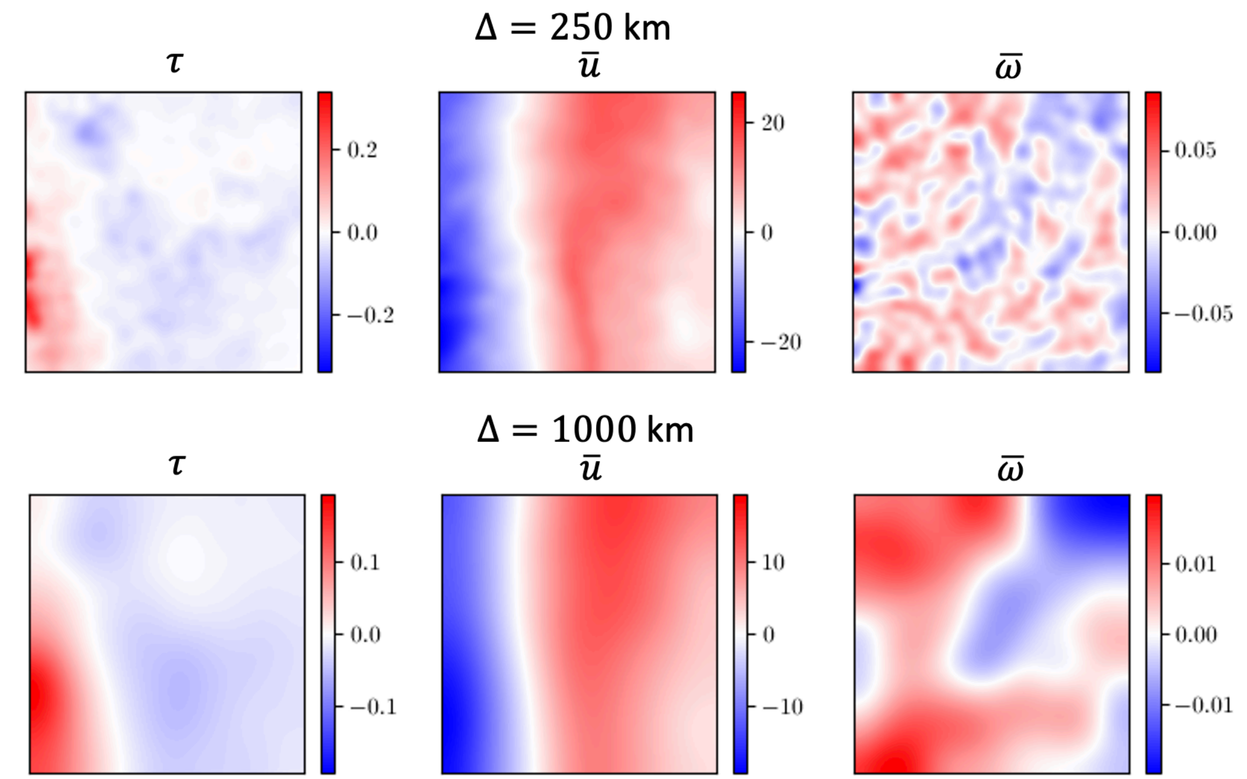

Figure 1. Snapshot of τ and respective $\bar{u}$ and $\bar{\omega}$ at altitude of 30 km for $\Delta = 250$ km and 1000 km. Small scale features are resolved for $\Delta = 250$ km, but not for $\Delta = 1000$ km. The snapshots span an area of 3072×3072 sq. km.

2.3 Equation Discovery Procedure

The data driven approach to discover τ has two steps. First, obtain the τ by using WRF data; second, discover the differential equations satisfied by τ,

$$\tau = f(\bar{u}, \bar{\omega}).$$

First, we pick a set of basis terms containing all the possible terms of $f(\bar{u}, \bar{\omega})$. We tend to pick moderately large set to guarantee that all the terms of $f(\bar{u}, \bar{\omega})$ are contained. The size of the library is limited by the memory of the available computer clusters. An algorithm is applied to search the subset of basis functions that are all the terms of $f(\bar{u}, \bar{\omega})$ and to determine the corresponding weights. Using $\tau, \bar{u}$ and $\bar{\omega}$ obtained from the WRF data, we construct the following system

$$\begin{bmatrix}\tau_1\\ \tau_2\\ .\\ .\\ .\\ \tau_M\end{bmatrix}=\begin{bmatrix}\frac{\partial\bar{u}_1}{\partial x} & \frac{\partial\bar{\omega}_1}{\partial y} & . & . & . & \frac{\partial^k\bar{u}_1}{\partial x^a\partial y^b\partial z^c}\frac{\partial^l\bar{\omega}_1}{\partial x^d\partial y^e\partial z^f}\\ \frac{\partial\bar{u}_2}{\partial x} & \frac{\partial\bar{\omega}_2}{\partial y} & . & . & . & \frac{\partial^k\bar{u}_1}{\partial x^a\partial y^b\partial z^c}\frac{\partial^l\bar{\omega}_1}{\partial x^d\partial y^e\partial z^f}\\ . & . & . & & & .\\ . & . & & . & & .\\ . & . & & & . & .\\ \frac{\partial\bar{u}_M}{\partial x} & \frac{\partial\bar{\omega}_M}{\partial y} & . & . & . & \frac{\partial^k\bar{u}_1}{\partial x^a\partial y^b\partial z^c}\frac{\partial^l\bar{\omega}_1}{\partial x^d\partial y^e\partial z^f}\end{bmatrix}\begin{bmatrix}w_1\\ w_2\\ .\\ .\\ .\\ w_N\end{bmatrix},$$

where M is the number of grid points used, N is the number of basis functions, $\frac{\partial\bar{u}_1}{\partial x},\frac{\partial\bar{\omega}_1}{\partial y}$, … are basis functions and w is weight or coefficient of each basis function. Basis functions comprises of combination of spatial derivatives of $\bar{u}$ and $\bar{\omega}$ upto k^{th} and l^{th} derivative respectively. The derivatives are calculated in all the three spatial dimensions (x,y,z), hence, $a+b+c=k$ and $d+e+f=l$. For this work we used library upto $k=4$ and $l=4$. This was limited by the memory of clusters, computational time and time-limited nature of the project. The spatial derivatives are calculated using 'findiff' package, it calculates numerical derivatives using finite difference scheme. Second order of accuracy is used for derivatives calculation.

We wish to limit the parametrization of τ to limited number of terms. Since N can be large, we need to get most of the weights $w=0$, hence a sparse method needs to be used. Denoting $\eta=[\tau_1,\dots\tau_M]'$, $\Phi=\begin{bmatrix}\frac{\partial\bar{u}_1}{\partial x} & . & . & . & \frac{\partial^k\bar{u}_1}{\partial x^a\partial y^b\partial z^c}\frac{\partial^l\bar{\omega}_1}{\partial x^d\partial y^e\partial z^f}\\ . & . & & & .\\ . & & . & & .\\ . & & & . & .\\ \frac{\partial\bar{u}_M}{\partial x} & . & . & . & \frac{\partial^k\bar{u}_1}{\partial x^a\partial y^b\partial z^c}\frac{\partial^l\bar{\omega}_1}{\partial x^d\partial y^e\partial z^f}\end{bmatrix}$, and $\boldsymbol{w}=[w_1,\dots,w_N]'$. Finding the weight-vector $\boldsymbol{w}$ is equivalent to solving sparse regression problem

$$\eta=\Phi\boldsymbol{w},$$

where η and Φ are known and $\boldsymbol{w}$ is to be determined sparsely. To do this we can use threshold least squares [8] or lasso [5]. In this paper we use threshold sparse Bayesian regression algorithm, which takes advantage of Bayesian inference to provide error bars to quantify uncertainties in the data-driven process.

2.4 Threshold Sparse Bayesian Regression

Here, we employ the sparse Bayesian regression method [9] to reveal τ parametrization. This technique assumes Gaussian priori distributions for each weight. The width of the Gaussian priori of each regression weight provides measure of uncertainty of that regression weight. The sparse regression is iteratively applied to the library of functions, and then pruning of the library of functions is carried out by discarding the functions with an uncertainty higher than prespecified threshold. This uncertainty threshold, δ, is the only parameter that requires setting in this method. The algorithm finishes when the uncertainty measures of each regression weight stop changing from iteration to iteration. This method provides an error associated with the weights discovered. Figure 2 shows the algorithm.

Threshold sparse Bayesian regression: $\eta = \Phi \mathbf{w}$

Input: η, Φ, threshold
Output: $\hat{\mu}$, $\hat{\Sigma}$
Calculate the posterior distribution $p(\mathbf{w}|\eta)$ in $\eta = \Phi\mathbf{w}$, let the mean be $\hat{\mu}$;
For components of $\hat{\mu}$ with absolute value less than the threshold, set them as 0;
while $\hat{\mu} \neq \boldsymbol{0}$ **do**
    Delete the columns of Φ whose corresponding weight is 0, getting Φ';
    Calculate the posterior distribution $p(\mathbf{w}'|\eta)$ in $\eta = \Phi'\mathbf{w}'$, let the mean be $\hat{\mu}'$;
    Update the corresponding components of $\hat{\mu}$ using $\hat{\mu}'$;
    For components of $\hat{\mu}$ with absolute value less than the threshold, set them as 0;
    if $\hat{\mu}$ *is the same as the one on the last loop* **then**
        break;
    end
end
Set the submatrix of $\hat{\Sigma}$ corresponding to non-zero components of $\hat{\mu}$ as the last estimated posterior variance in the preceding procedure, and set other elements of $\hat{\Sigma}$ as 0.

Figure 2. Threshold sparse Bayesian regression [9].

3. Results

We used data for two different filter sizes $\Delta = 250$ km and 1000 km for discovery process. We formed the library of basis functions $\left(\frac{\partial^k \bar{u}_1}{\partial x^a \partial y^b \partial z^c} \frac{\partial^l \bar{\omega}_1}{\partial x^d \partial y^e \partial z^f}\right)$ upto $k = l = 2$ and $k = l = 4$. The equations for τ were discovered by increasing the threshold from 50 to 500. 100 grid points were randomly selected out of $1024 \times 1024 \times 120$ grid points for the discovery process. The data was normalized before inputting in the regression algorithm.

Increasing number of terms were discovered with increasing threshold. With increasing number of discovered terms with increasing value of threshold, R^2 for both filter sizes increased as well (Figure 3). For $k = l = 4$ and $\Delta = 1000$ km, R^2 stayed lowed for threshold value of 500, while R^2 value approached 1, for $\Delta = 250$ km. This indicates filter size of $\Delta = 1000$ km isn't adequate for equation discovery process. Discovered equation for τ is not included in the text because of large number of discovered terms (50 terms for threshold value of 500). We hoped to discover similar terms for $\Delta = 250$ km and $\Delta = 1000$ km, but that was not the case here, hence we cannot comment on the robustness of the discovered terms. Here, the robust terms would be terms discovered for both filter sizes. We will need to repeat the discovery process for a smaller filter size. We have not analyzed the error bar for each discovered terms because of absence of any robust terms.

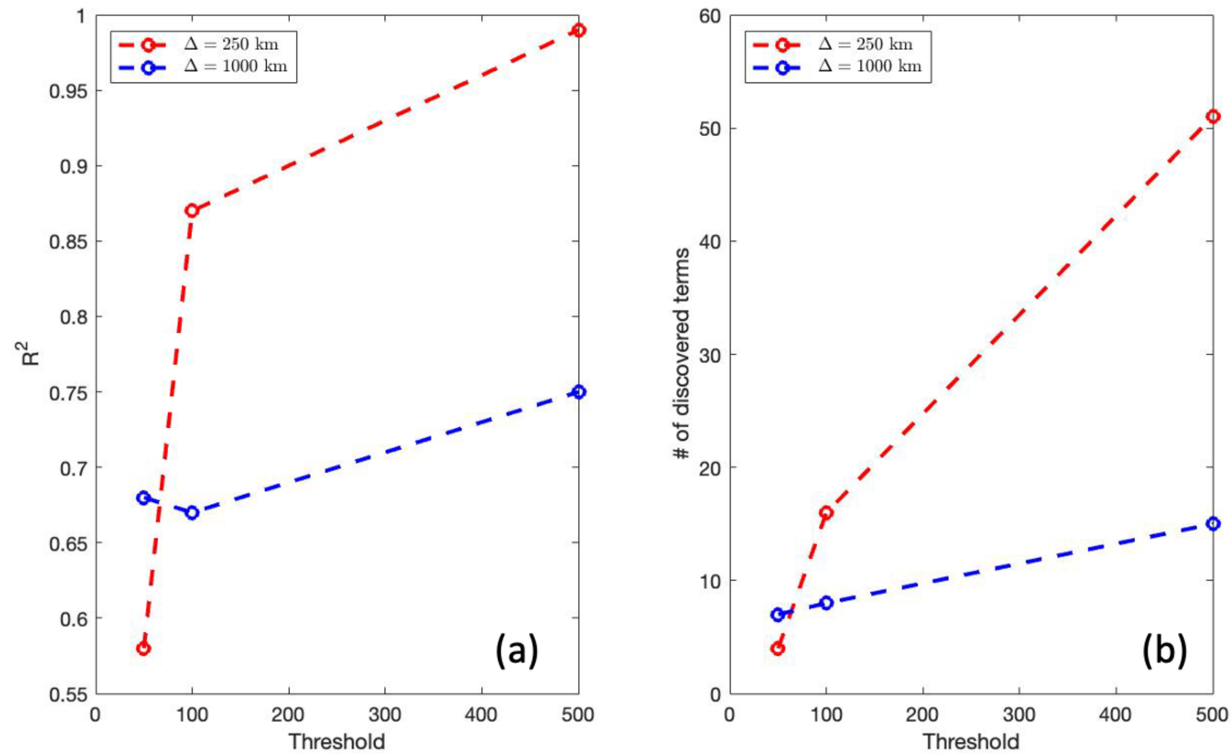

Figure 3. (a) R^2 value and (b) Number of discovered terms with increasing value of threshold for $\Delta = 250$ km and 1000 km. The library of basis functions with $k = l = 4$ is used here.

4. Discussion and Conclusion

We used a data driven approach to parametrize the momentum flux of gravity waves (represented by Reynold's stress, τ). High resolution data generated by WRF model was used. We used threshold sparse Bayesian regression algorithm to sparsely discover weights or coefficients of the terms in the library of basis functions. The discovery process was attempted for two filter sizes and three threshold values.

The equations discovered for $\Delta = 1000$ km showed poor correlations, while equations discovered for $\Delta = 250$ km showed promising results. Since the same terms were not discovered for both filter sizes, we cannot comment on the robustness of the discovered terms.

We need to repeat the discovery process for smaller filter sizes to get robust parametrization for τ. We can also include a greater number of grid points or include more terms in the library of basis functions to get more meaningful results.

Appendix

The code for this work is available on GitHub: https://github.com/jakharkaran/CEVE-543-project.git

Bibliography

[1] D. C. Fritts and M. J. Alexander, "Gravity wave dynamics and effects in the middle atmosphere: MIDDLE ATMOSPHERE GRAVITY WAVE DYNAMICS," *Rev. Geophys.*, vol. 41, no. 1, Mar. 2003, doi: 10.1029/2001RG000106.

[2] M. Okraschevski, S. Hoffmann, K. Stichling, R. Koch, and H.-J. Bauer, "Fluid dynamics beyond the continuum: A physical perspective on large-eddy simulation," *Phys. Rev. Fluids*, vol. 6, no. 10, p. L102601, Oct. 2021, doi: 10.1103/PhysRevFluids.6.L102601.

[3] L. Zanna and T. Bolton, "Deep Learning of Unresolved Turbulent Ocean Processes in Climate Models," in *Deep Learning for the Earth Sciences*, 1st ed., G. Camps-Valls, D. Tuia, X. X. Zhu, and M. Reichstein, Eds. Wiley, 2021, pp. 298–306. doi: 10.1002/9781119646181.ch20.

[4] L. Zanna and T. Bolton, "Data-Driven Equation Discovery of Ocean Mesoscale Closures," *Geophysical Research Letters*, vol. 47, no. 17, p. e2020GL088376, 2020, doi: 10.1029/2020GL088376.

[5] H. Schaeffer, "Learning partial differential equations via data discovery and sparse optimization," *Proc. R. Soc. A.*, vol. 473, no. 2197, p. 20160446, Jan. 2017, doi: 10.1098/rspa.2016.0446.

[6] A. Leonard, "Energy Cascade in Large-Eddy Simulations of Turbulent Fluid Flows," in *Advances in Geophysics*, vol. 18, Elsevier, 1975, pp. 237–248. doi: 10.1016/S0065-2687(08)60464-1.

[7] W. C. Skamarock *et al.*, "A Description of the Advanced Research WRF Model Version 4," p. 162.

[8] "Discovering governing equations from data by sparse identification of nonlinear dynamical systems." https://www.pnas.org/doi/10.1073/pnas.1517384113 (accessed May 09, 2022).

[9] S. Zhang and G. Lin, "Robust data-driven discovery of governing physical laws with error bars," *Proc. R. Soc. A.*, vol. 474, no. 2217, p. 20180305, Sep. 2018, doi: 10.1098/rspa.2018.0305.