Small closure models for large multiscale problems

Wouter Edeling, Daan Crommelin
Scientific Computing Group

CWI Scientific Meeting, June 17th, 2022.
About me

- Aerospace Engineering TU Delft: MSc.
- Arts et Métiers ParisTech & TU Delft: joint-PhD.
- Stanford University: postdoc.
- CWI: postdoc & tenure track.
About me

- Aerospace Engineering TU Delft: MSc.
- Arts et Métiers ParisTech & TU Delft: joint-PhD.
- Stanford University: postdoc.
- CWI: postdoc & tenure track.
Topic: turbulent flow

What is turbulent flow?
What is turbulent flow?

- Not laminar.
What is turbulent flow?
- Not laminar.
- Unsteady.
What is turbulent flow?

- Not laminar.
- Unsteady.
- Mixing.
What is turbulent flow?

- Not laminar.
- Unsteady.
- Mixing.
- Multiscale.
Numerical simulation resolving all spatial & temporal scales:

Credit: turbulence team:
https://www.youtube.com/watch?v=OM0l2YPVMf8
Discretization

- Numerical simulation = discretization: $\omega(x, y) \rightarrow \omega^h(x, y)$
Discretization

- Numerical simulation = discretization: \( \omega(x, y) \rightarrow \omega^h(x, y) \)

Solve equations on each point of a fine mesh.
Problem: multi-scale nature:
→ required mesh resolution (often) much too large.
Filtering

- Problem: multi-scale nature:
  → required mesh resolution (often) much too large.

- Engineering solution:
  \[ \omega = \bar{\omega} + \omega' \]
  → decompose solution 
  → only solve for large scales \( \bar{\omega} \).
Filtering

- Problem: multi-scale nature:
  → required mesh resolution (often) much too large.

- Engineering solution:
  → decompose solution \( \omega = \overline{\omega} + \omega' \).
  → only solve for large scales \( \overline{\omega} \).

- How to get \( \overline{\omega} \)?
  → Use filter \( \overline{\omega} = P\omega \)
Filtering

- Governing equations:

\[
\frac{\partial \omega}{\partial t} + J(\psi, \omega) = \nu \nabla^2 \omega + \mu (F - \omega) \\
\nabla^2 \psi = \omega.
\]
Filtering

- Governing equations:

\[
\frac{\partial \omega}{\partial t} + J(\psi, \omega) = \nu \nabla^2 \omega + \mu (F - \omega)
\]

\[
\nabla^2 \psi = \omega.
\]

- Apply filter:

\[
\frac{\partial \bar{\omega}}{\partial t} + J(\bar{\psi}, \bar{\omega}) = \nu \nabla^2 \bar{\omega} + \mu (F - \bar{\omega}) - \bar{r},
\]

\[
\nabla^2 \bar{\psi} = \bar{\omega}.
\]
Filtering

- **Governing equations:**

\[
\frac{\partial \omega}{\partial t} + J(\psi, \omega) = \nu \nabla^2 \omega + \mu (F - \omega)
\]

\[
\nabla^2 \psi = \omega.
\]

- **Apply filter:**

\[
\frac{\partial \bar{\omega}}{\partial t} + J(\bar{\psi}, \bar{\omega}) = \nu \nabla^2 \bar{\omega} + \mu (F - \bar{\omega}) - \bar{r},
\]

\[
\nabla^2 \bar{\psi} = \bar{\omega}.
\]

- **Sub-Grid Scale (SGS) term** \( \bar{r} \) appears.
Filtering

- Solving both equations side by side:
Closing governing equations

▶ Problem: SGS model is unknown / unclosed $\bar{r} = \bar{r}(\omega, \psi)$.
$\Rightarrow \bar{r}$ must be modelled

$$\frac{\partial \bar{\omega}}{\partial t} + J(\bar{\psi}, \bar{\omega}) = \nu \nabla^2 \bar{\omega} + \mu (F - \bar{\omega}) - \bar{r},$$

$$\nabla^2 \bar{\psi} = \bar{\omega}.$$
Closing governing equations

- Problem: SGS model is unknown / unclosed \( \bar{r} = \bar{r}(\omega, \psi) \).
  \( \rightarrow \) \( \bar{r} \) must be modelled

\[
\frac{\partial \bar{\omega}}{\partial t} + J (\bar{\psi}, \bar{\omega}) = \nu \nabla^2 \bar{\omega} + \mu (F - \bar{\omega}) - \bar{r}, \\
\nabla^2 \bar{\psi} = \bar{\omega}.
\]

- Goal: Learn \( \bar{r} \) from 256 \( \times \) 256 simulation.
Question

What should we learn from data?

<table>
<thead>
<tr>
<th>state size</th>
<th>SGS training data size</th>
<th>QoI size</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 × 64</td>
<td>64 × 64</td>
<td>( Q(\bar{ω}) \in \mathbb{R}^{64 \times 64} )</td>
</tr>
</tbody>
</table>

traditional training

reduced training

\( q(\bar{ω}) \in \mathbb{R}^{d} \)
\( d \ll 64 \times 64 \)
Assumptions

1. There are $d$ global QoI:

$$q_i(t) = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi} \int_0^{2\pi} f_i(\bar{\omega}, \bar{\psi}; x, y, t) \, dx \, dy, \quad i = 1, \ldots, d.$$
Assumptions

1. There are $d$ global QoI:

$$q_i(t) = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi} \int_0^{2\pi} f_i(\bar{\omega}, \bar{\psi}; x, y, t) \, dx \, dy, \quad i = 1, \ldots, d.$$ 

2. Replace $\bar{r} = r(\bar{\psi}, \bar{\omega})$ with reduced SGS term $r$:

$$r := \sum_{i=1}^{d} \tau_i(t) P_i(x, y, t),$$
Assumptions

\[ r := \sum_{i=1}^{d} \tau_i(t)P_i(x, y, t), \]

Justified if:

- \( r \) is ‘just as good’ as \( \bar{r} \) for \( q_i \).
Assumptions

\[ r := \sum_{i=1}^{d} \tau_i(t) P_i(x, y, t), \]

Justified if:
- \( r \) is ‘just as good’ as \( \bar{r} \) for \( q_i \).
- Must tie \( \tau_i \) & \( P_i \) to \( q_i \) physics.
Compute effect of assumptions

- Derive $q_i$ ODEs:

\[
\frac{dq_i}{dt} = \ldots + \left( \frac{\partial f_i}{\partial \bar{\omega}}, r \right) = \ldots + \sum_{j=1}^{d} \tau_j \left( \frac{\partial f_i}{\partial \bar{\omega}}, P_j \right)
\]

- $(A, B) := \left( \frac{1}{2\pi} \right)^2 \int_0^{2\pi} \int_0^{2\pi} AB \, dx \, dy$.

---

Compute effect of assumptions

- Derive $q_i$ ODEs:

$$\frac{dq_i}{dt} = \ldots + \left( \frac{\partial f_i}{\partial \vec{\omega}}, r \right) = \ldots + \sum_{j=1}^{d} \tau_j \left( \frac{\partial f_i}{\partial \vec{\omega}}, P_j \right)$$

- $(A, B) := \left( \frac{1}{2\pi} \right)^2 \int_0^{2\pi} \int_0^{2\pi} AB \, dx \, dy$.

- Every $q_i$ ODE has $d$ SGS terms:

---

Compute effect of assumptions

- Derive $q_i$ ODEs:

\[
\frac{dq_i}{dt} = \ldots + \left( \frac{\partial f_i}{\partial \bar{\omega}}, r \right) = \ldots + \sum_{j=1}^{d} \tau_j \left( \frac{\partial f_i}{\partial \bar{\omega}}, P_j \right)
\]

- $(A, B) := \left( \frac{1}{2\pi} \right)^2 \int_0^{2\pi} \int_0^{2\pi} AB \, dx \, dy$.

- Every $q_i$ ODE has $d$ SGS terms: remove $\sum_{j=1}^{d}$

---

Compute effect of assumptions

- Derive $q_i$ ODEs:

$$\frac{dq_i}{dt} = \ldots + \left( \frac{\partial f_i}{\partial \bar{\omega}}, r \right) = \ldots + \sum_{j=1}^{d} \tau_j \left( \frac{\partial f_i}{\partial \bar{\omega}}, P_j \right)$$

- $(A, B) := \left( \frac{1}{2\pi} \right)^2 \int_{0}^{2\pi} \int_{0}^{2\pi} AB \, dx \, dy$.

- Every $q_i$ ODE has $d$ SGS terms: remove $\sum_{j=1}^{d}$

- Orthogonality condition $\forall t$:

$$\left( \frac{\partial f_i}{\partial \bar{\omega}}, P_j \right) = 0 \text{ if } i \neq j$$

- Separate expansion for $P_j$ and small linear solve $^1$.

---

Extract $\tau_i$ from data

Due to orthogonality, $q_i$ transport equation becomes:

$$\frac{dq_i}{dt} = ... + \left( \frac{\partial f_i}{\partial \omega}, r \right) = ... + \tau_i \left( \frac{\partial f_i}{\partial \omega}, P_i \right)$$

Goal: error $q_{i\text{ref}}(t) - q_i(t) =: \Delta q_i$ is small $\forall t$ in training.
Extract $\tau_i$ from data

- Due to orthogonality, $q_i$ transport equation becomes:

$$\frac{dq_i}{dt} = \ldots + \left( \frac{\partial f_i}{\partial \bar{\omega}}, r \right) = \ldots + \tau_i \left( \frac{\partial f_i}{\partial \bar{\omega}}, P_i \right)$$

- Goal: error $q_i^{\text{ref}}(t) - q_i(t) =: \Delta q_i$ is small $\forall t$ in training.

- Assumption: $\tau_i$ depends upon $\Delta q_i$. 

Assumes $\tau_i \sim \Delta q_i$ imposes linear relaxation towards reference.

$\Delta q_i$ is the only data we need.
Due to orthogonality, \( q_i \) transport equation becomes:

\[
\frac{dq_i}{dt} = \ldots + \left( \frac{\partial f_i}{\partial \bar{\omega}}, \mathbf{r} \right) = \ldots + \tau_i \left( \frac{\partial f_i}{\partial \bar{\omega}}, P_i \right)
\]

Goal: error \( q_i^{\text{ref}}(t) - q_i(t) =: \Delta q_i \) is small \( \forall t \) in training.

Assumption: \( \tau_i \) depends upon \( \Delta q_i \).

Simply equate source term to \( \Delta q_i \):

\[
\tau_i \left( \frac{\partial f_i}{\partial \bar{\omega}}, P_i \right) = \frac{\Delta q_i}{T_i}, \quad T_i = 1, \quad i = 1, \ldots, d
\]

Assumes \( \tau_i \sim \Delta q_i \) + imposes linear relaxation towards reference.

\( \Delta q_i \) is the only data we need.
Example results

$\mathbf{q_1} = \text{energy } E, \mathbf{q_2} = \text{enstrophy } Z.$

$\mathbf{r}$ is ‘just as good’ as $\bar{r}$ for $q_i$.

→ Number of unknowns reduced from $64^2$ to 2.
→ Training data size reduced by factor $64^2/2$. 
Example results

\[ q_1 = \text{energy } E, \quad q_2 = \text{enstrophy } Z. \]

\[ r = -\frac{1}{2} \left[ \frac{\Delta E}{S - E^2/Z} \right] \left( \psi + \frac{E}{Z} \omega \right) + \frac{1}{2} \left[ \frac{\Delta Z}{Z - E^2/S} \right] \left( \omega + \frac{E}{S} \psi \right) \]
### Question

What should we learn from data?

<table>
<thead>
<tr>
<th></th>
<th>State Size</th>
<th>SGS Training Data Size</th>
<th>QoI Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional Training</td>
<td>64 × 64</td>
<td>64 × 64</td>
<td>( Q(\omega) \in \mathbb{R}^{64 \times 64} )</td>
</tr>
<tr>
<td>Reduced Training</td>
<td>64 × 64</td>
<td>?</td>
<td>( q(\omega) \in \mathbb{R}^{d} ) ( d \ll 64 \times 64 )</td>
</tr>
</tbody>
</table>
Conclusion

- $\tau_i$ (or $\Delta q_i$)

<table>
<thead>
<tr>
<th></th>
<th>state size</th>
<th>SGS training data size</th>
<th>QoI size</th>
</tr>
</thead>
<tbody>
<tr>
<td>traditional training</td>
<td>64 × 64</td>
<td>64 × 64</td>
<td>$Q(\omega) \in \mathbb{R}^{64 \times 64}$</td>
</tr>
<tr>
<td>reduced training</td>
<td>64 × 64</td>
<td>$d$</td>
<td>$q(\omega) \in \mathbb{R}^d$</td>
</tr>
</tbody>
</table>

$d \ll 64 \times 64$
Offline training

- Now: train ML model on reduced training data.
- Offline training: train e.g. ANN on $\Delta E, \Delta Z$ database.

\[ \partial \bar{\omega} \partial t + J(\bar{\psi}, \bar{\omega}) = \nu \nabla^2 \bar{\omega} + \mu (F - \bar{\omega}) - \tau_1 (\tilde{\Delta} E) P_1 - \tau_2 (\tilde{\Delta} Z) P_2, \]

\[ \nabla^2 \bar{\psi} = \bar{\omega}. \]
Offline training

- Now: train ML model on reduced training data.
- Offline training: train e.g. ANN on $\Delta E, \Delta Z$ database.
Offline training

- Now: train ML model on reduced training data.
- **Offline** training: train e.g. ANN on $\Delta E$, $\Delta Z$ database.

Now we have a **coupled** PDE - ML system:

\[
\frac{\partial \tilde{\omega}}{\partial t} + J(\tilde{\psi}, \tilde{\omega}) = \nu \nabla^2 \tilde{\omega} + \mu (F - \tilde{\omega}) - \tau_1 (\tilde{\Delta E}) P_1 - \tau_2 (\tilde{\Delta Z}) P_2,
\]

\[
\nabla^2 \tilde{\psi} = \tilde{\omega}.
\]

\[
[\tilde{\Delta E}, \tilde{\Delta Z}] = ANN(X_1, \ldots, X_7)
\]
Prediction with offline surrogate

- Can become unstable:

Why?: ANN was not trained not to operate in a two-way coupled modelling environment.

Other authors reported similar issues.
Prediction with offline surrogate

➤ Can become unstable:

➤ Why?: ANN was not trained not to operate in a two-way coupled modelling environment.

➤ Other authors reported similar issues.
Online training

- online training while ANN is coupled to PDE.

---


Online training

- Online training while ANN is coupled to PDE.
- 1 data point per time step.

---


Online training

▶ **Online training** while ANN is coupled to PDE.
▶ 1 data point per time step.
▶ First step: just do back propagation online:

More sophisticated methods, See Rasp or Sahoo.² ³


Initial results

- $M \Delta t =$ time interval between back propagation steps.
Initial results

- $M\Delta t =$ time interval between back propagation steps.
- $M = 1$, continual online learning:

$\text{coupled}$ LR - ML model conserves HR energy and enstrophy.
Initial results

$M = 20$: 

- coupled LR - ML model conserves HR energy and enstrophy.
Initial results

▶ Thus far, initial conditions were perfect: $q^{\text{ref}} = q$.
▶ Can it recover when $q^{\text{ref}} \neq q$?
Initial results

- Thus far, initial conditions were perfect: $q^{ref} = q$.
- Can it recover when $q^{ref} \neq q$?
- No SGS term before 10 days:
- coupled LR - ML model can quickly recover HR QoI.
Next steps

One of the next steps:

- When do we turn online training on / off? Is a UQ question.
Questions?

Online training: Rasp (2020)\(^4\)

Fig from Rasp 2020: HR state is nudged towards LR state.

- Keeps HR - LR small, helps with online convergence.
- Allows LR to learn ‘what HR would do under similar states’.
- **Applied to Lorenz96.**

Online training: Rasp (2020)$^4$

Fig from Rasp 2020: HR state is nudged towards LR state.

- **However**: nudging & ML correction is applied to entire state.
- **Reduced ML**: ML correction is only applied via $\tau$.

---

Reduced online training

- Modified online (reduced) learning:
  - Extract $\Delta q$ data.
  - Only 1 LR step.
Example results

- Are the results spectrally accurate?
- Map 2D wave numbers $\mathbf{k} = (k_1, k_2)$ to 1D $k$:

$$k - \frac{1}{2} \leq \sqrt{k_1^2 + k_2^2} < k + \frac{1}{2}, \quad k = 0, 1, \ldots, \text{ceil} \left( \sqrt{2}K \right)$$
Example results

▶ Are the results spectrally accurate?
▶ Map 2D wave numbers $k = (k_1, k_2)$ to 1D $k$:

$$k - \frac{1}{2} \leq \sqrt{k_1^2 + k_2^2} < k + \frac{1}{2}, \quad k = 0, 1, \cdots, \text{ceil} \left( \sqrt{2}K \right)$$

▶ **No** accurate spectra, only explicitly track overall $E$ and $Z$. 
Example results

- However, scale-aware QoI are also possible
- Focus on a specific wave-number range via (spectral) filter $T$:
  $\to$ zeros out all wave numbers $k < K$.

$$
\tau_i \left( \frac{\partial f_i}{\partial \omega}, P_i \right) = \Delta q_i \to \tau_i \left( T \frac{\partial f_i}{\partial \omega}, P_i \right) = T(\Delta q_i)
$$
Example results

- However, scale-aware QoI are also possible
- Focus on a specific wave-number range via (spectral) filter $\mathcal{T}$:
  → zeros out all wave numbers $k < K$.

$$
\tau_i \left( \frac{\partial f_i}{\partial \omega}, P_i \right) = \Delta q_i \rightarrow \tau_i \left( \mathcal{T} \frac{\partial f_i}{\partial \omega}, P_i \right) = \mathcal{T}(\Delta q_i)
$$

Focus on wave numbers $k \in [21, 30]$. 