Exceptional service in the national interest





Proper Orthogonal Decomposition (POD) Closure Models for Turbulent Flows

Irina K. Tezaur

Extreme Scale Data Science & Analytics Department, Sandia National Laboratories, Livermore CA

Data Science Reading Group Seminar

Thursday, May 18, 2017





Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND2017-5377 PE





^b Department of Mechanical Engineering, NUST College of Electrical & Mechanical Engineering, National University of Sciences & Technology, Islamabad, Pakistan





• My interest in this work:





• My interest in this work:

• I am interested in *stable POD-Galerkin* ROMs for fluid problems (next slides).





• My interest in this work:

- I am interested in *stable POD-Galerkin* ROMs for fluid problems (next slides).
- Authors of this paper have *similar requirements* for ROMs as me: use ROMs for longtime integration, "extreme model reduction", QoI = statistics of flow, etc.





• My interest in this work:

- I am interested in *stable POD-Galerkin* ROMs for fluid problems (next slides).
- Authors of this paper have *similar requirements* for ROMs as me: use ROMs for longtime integration, "extreme model reduction", QoI = statistics of flow, etc.
- Methods in this paper are alternatives to my work on **basis rotation**¹.

¹ M. Balajewicz, **I. Tezaur**, E. Dowell. "Minimal subspace rotation on the Stiefel manifold for stabilization and enhancement of projectionbased reduced order models for the compressible Navier-Stokes equations", *JCP* 321 (2016) 224-241.





• My interest in this work:

- I am interested in *stable POD-Galerkin* ROMs for fluid problems (next slides).
- Authors of this paper have *similar requirements* for ROMs as me: use ROMs for longtime integration, "extreme model reduction", QoI = statistics of flow, etc.
- Methods in this paper are alternatives to my work on **basis rotation**¹.
- I am working with T. Iliescu to try to understand how to extend methods such as those in the paper to *compressible flow* problems and to make them more rigorous.

¹ M. Balajewicz, **I. Tezaur**, E. Dowell. "Minimal subspace rotation on the Stiefel manifold for stabilization and enhancement of projectionbased reduced order models for the compressible Navier-Stokes equations", *JCP* 321 (2016) 224-241.

Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]

Outline



1. Motivation/Background

- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]



• We are interested in the *compressible captive-carry problem*.









• We are interested in the *compressible captive-carry problem*.





 Of <u>primary interest</u> are *long-time predictive simulations*: ROM run at same parameters as FOM but much longer in time.



• We are interested in the *compressible captive-carry problem*.



- Of <u>primary interest</u> are *long-time predictive simulations*: ROM run at same parameters as FOM but much longer in time.
 - → <u>Qols:</u> statistics of flow, e.g., pressure Power Spectral Densities (*PSDs*) [right].





• We are interested in the *compressible captive-carry problem*.



- Of <u>primary interest</u> are *long-time predictive simulations*: ROM run at same parameters as FOM but much longer in time.
 - → <u>Qols:</u> statistics of flow, e.g., pressure Power Spectral Densities (*PSDs*) [right].



 <u>Secondary interest</u>: ROMs robust w.r.t. *parameter changes* (e.g., Reynolds, Mach number) for enabling *uncertainty quantification*.

Proper Orthogonal Decomposition (POD)/ Galerkin method to model reduction





 $\dot{a}_{Mk} = f(a_{M1}, \dots, a_{MM})$

Extreme Model Reduction



- Most realistic applications (e.g., high *Re* compressible cavity): basis that captures
 >99% snapshot energy is required to accurately reproduce snapshots.
 - \rightarrow leads to M > O(1000) except for **toy problems** and/or **low-fidelity** models.
- Higher order modes are in general *unreliable for prediction*, so including them in the basis is unlikely to improve the *predictive* capabilities of a ROM.

Figure (right) shows projection error for POD basis constructed using 800 snapshots for cavity problem. Dashed line = end of snapshot collection period.



We are looking for an approach that enables <u>extreme model reduction</u>: ROM basis size is O(10) or O(100).





Projection-based MOR necessitates *truncation*.

• POD is, by definition and design, biased towards the *large, energy producing* scales of the flow (i.e., modes with large POD eigenvalues).



- POD is, by definition and design, biased towards the *large, energy producing* scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a <u>data compression</u> point of view (i.e., small POD eigenvalues) but are crucial for the <u>dynamical</u> <u>equations</u>.



- POD is, by definition and design, biased towards the *large, energy producing* scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a <u>data compression</u> point of view (i.e., small POD eigenvalues) but are crucial for the <u>dynamical</u> <u>equations</u>.
- For fluid flow applications, higher-order modes are associated with energy <u>dissipation</u>



- POD is, by definition and design, biased towards the *large, energy producing* scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a <u>data compression</u> point of view (i.e., small POD eigenvalues) but are crucial for the <u>dynamical</u> <u>equations</u>.
- For fluid flow applications, higher-order modes are associated with energy <u>dissipation</u>
 - \Rightarrow low-dimensional ROMs can be <u>inaccurate</u> and <u>unstable</u>.



Projection-based MOR necessitates *truncation*.

- POD is, by definition and design, biased towards the *large, energy producing* scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a <u>data compression</u> point of view (i.e., small POD eigenvalues) but are crucial for the <u>dynamical</u> <u>equations</u>.
- For fluid flow applications, higher-order modes are associated with energy <u>dissipation</u>
 - \Rightarrow low-dimensional ROMs can be <u>inaccurate</u> and <u>unstable</u>.

For a low-dimensional ROM to be stable and accurate, the <u>truncated/unresolved subspace</u> must be accounted for.



- POD is, by definition and design, biased towards the *large, energy producing* scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a <u>data compression</u> point of view (i.e., small POD eigenvalues) but are crucial for the <u>dynamical</u> <u>equations</u>.
- For fluid flow applications, higher-order modes are associated with energy <u>dissipation</u>
 - \Rightarrow low-dimensional ROMs can be <u>inaccurate</u> and <u>unstable</u>.



Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]

Section 2: POD-Galerkin-ROM (POD-G-ROM

• Governing equations of *incompressible flow*:

$$\begin{aligned} \mathbf{u}_t - Re^{-1}\Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p &= 0\\ \nabla \cdot \mathbf{u} &= 0 \end{aligned} \right\}, \tag{1}$$

Section 2: POD-Galerkin-ROM (POD-G-ROM

• Governing equations of *incompressible flow*:

$$\begin{aligned} \mathbf{u}_t - Re^{-1}\Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p &= 0\\ \nabla \cdot \mathbf{u} &= 0 \end{aligned} \right\}, \tag{1}$$

• POD approximation of velocity solution² *u*:

$$\mathbf{u}(\mathbf{x},t) \approx \mathbf{u}_r(\mathbf{x},t) \equiv \mathbf{U}(\mathbf{x}) + \sum_{j=1}^r a_j(t) \boldsymbol{\varphi}_j(\mathbf{x}), \tag{6}$$

where U(x) = base flow, $\varphi_j(x)$ = POD modes.

² Pressure ROM can be obtained by solving pressure-Poisson equation. Pressure term drops out from (1) following projection due to BCs. See [36,56].

Section 2: POD-Galerkin-ROM (POD-G-ROM

• Governing equations of *incompressible flow*:

$$\begin{aligned} \mathbf{u}_t - Re^{-1}\Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p &= 0\\ \nabla \cdot \mathbf{u} &= 0 \end{aligned} \right\}, \tag{1}$$

• POD approximation of velocity solution² *u*:

$$\mathbf{u}(\mathbf{x},t) \approx \mathbf{u}_r(\mathbf{x},t) \equiv \mathbf{U}(\mathbf{x}) + \sum_{j=1}^r a_j(t) \boldsymbol{\varphi}_j(\mathbf{x}), \tag{6}$$

where $\boldsymbol{U}(\boldsymbol{x})$ = base flow, $\boldsymbol{\varphi}_j(\boldsymbol{x})$ = POD modes.

• Projecting (1) onto reduced basis $\varphi_j(x)$, the following **<u>POD-G(alerkin)-ROM</u>** is obtained:

$$\left(\frac{\partial \mathbf{u}_r}{\partial t}, \boldsymbol{\phi}\right) + \left((\mathbf{u}_r \cdot \nabla)\mathbf{u}_r, \boldsymbol{\phi}\right) + \left(\frac{2}{Re}\mathbb{D}(\mathbf{u}_r), \nabla \boldsymbol{\phi}\right) = 0 \quad \forall \boldsymbol{\phi} \in \mathbf{X}^r, \qquad (7)$$

where X^r = reduced subspace, $\mathbb{D}(u_r) = \frac{1}{2}\nabla u_r + \frac{1}{2}(\nabla u_r)^T$ = deformation tensor of u_r .

² Pressure ROM can be obtained by solving pressure-Poisson equation. Pressure term drops out from (1) following projection due to BCs. See [36,56].

• **<u>POD-G-ROM</u>** algebraic system:

$$\dot{a}_k(t) = b_k + \sum_{m=1}^r A_{km} a_m(t) + \sum_{m=1}^r \sum_{n=1}^r B_{kmn} a_n(t) a_m(t),$$
(10)

where:

$$\boldsymbol{b}_{k} = -\left(\boldsymbol{\varphi}_{k}, \boldsymbol{\mathsf{U}} \cdot \nabla \boldsymbol{\mathsf{U}}\right) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_{k}, \frac{\nabla \boldsymbol{\mathsf{U}} + \nabla \boldsymbol{\mathsf{U}}^{\mathrm{T}}}{2}\right), \tag{11}$$

$$A_{km} = -(\boldsymbol{\varphi}_k, \mathbf{U} \cdot \nabla \boldsymbol{\varphi}_m) - (\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \mathbf{U}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_k, \frac{\nabla \boldsymbol{\varphi}_m + \nabla \boldsymbol{\varphi}_m^T}{2} \right), \quad (12)$$

$$\boldsymbol{B}_{kmn} = -(\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \boldsymbol{\varphi}_n). \tag{13}$$

• **<u>POD-G-ROM</u>** algebraic system:

$$\dot{a}_k(t) = b_k + \sum_{m=1}^r A_{km} a_m(t) + \sum_{m=1}^r \sum_{n=1}^r B_{kmn} a_n(t) a_m(t),$$
(10)

where:

$$\boldsymbol{b}_{k} = -(\boldsymbol{\varphi}_{k}, \boldsymbol{\mathsf{U}} \cdot \nabla \boldsymbol{\mathsf{U}}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_{k}, \frac{\nabla \boldsymbol{\mathsf{U}} + \nabla \boldsymbol{\mathsf{U}}^{\mathrm{T}}}{2} \right), \tag{11}$$

$$A_{km} = -(\boldsymbol{\varphi}_k, \mathbf{U} \cdot \nabla \boldsymbol{\varphi}_m) - (\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \mathbf{U}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_k, \frac{\nabla \boldsymbol{\varphi}_m + \nabla \boldsymbol{\varphi}_m^T}{2} \right), \quad (12)$$

$$B_{kmn} = -(\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \boldsymbol{\varphi}_n). \tag{13}$$

• **<u>POD-G-ROM</u>** in matrix form:

$$\dot{a} = b + Aa + a^T Ba \tag{8}$$

• **<u>POD-G-ROM</u>** algebraic system:

$$\dot{a}_k(t) = b_k + \sum_{m=1}^r A_{km} a_m(t) + \sum_{m=1}^r \sum_{n=1}^r B_{kmn} a_n(t) a_m(t),$$
(10)

where:

$$\boldsymbol{b}_{k} = -(\boldsymbol{\varphi}_{k}, \boldsymbol{\mathsf{U}} \cdot \nabla \boldsymbol{\mathsf{U}}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_{k}, \frac{\nabla \boldsymbol{\mathsf{U}} + \nabla \boldsymbol{\mathsf{U}}^{\mathrm{T}}}{2} \right), \tag{11}$$

$$A_{km} = -(\boldsymbol{\varphi}_k, \mathbf{U} \cdot \nabla \boldsymbol{\varphi}_m) - (\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \mathbf{U}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_k, \frac{\nabla \boldsymbol{\varphi}_m + \nabla \boldsymbol{\varphi}_m^T}{2} \right), \quad (12)$$

$$B_{kmn} = -(\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \boldsymbol{\varphi}_n). \tag{13}$$

• **<u>POD-G-ROM</u>** in matrix form:

$$\dot{a} = b + Aa + a^T Ba \tag{8}$$

• POD-G-ROM (8) has been successfully used for *laminar flows*.

• **<u>POD-G-ROM</u>** algebraic system:

$$\dot{a}_k(t) = b_k + \sum_{m=1}^r A_{km} a_m(t) + \sum_{m=1}^r \sum_{n=1}^r B_{kmn} a_n(t) a_m(t),$$
(10)

where:

$$\boldsymbol{b}_{k} = -(\boldsymbol{\varphi}_{k}, \boldsymbol{\mathsf{U}} \cdot \nabla \boldsymbol{\mathsf{U}}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_{k}, \frac{\nabla \boldsymbol{\mathsf{U}} + \nabla \boldsymbol{\mathsf{U}}^{\mathrm{T}}}{2} \right), \tag{11}$$

$$A_{km} = -(\boldsymbol{\varphi}_k, \mathbf{U} \cdot \nabla \boldsymbol{\varphi}_m) - (\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \mathbf{U}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_k, \frac{\nabla \boldsymbol{\varphi}_m + \nabla \boldsymbol{\varphi}_m^T}{2} \right), \quad (12)$$

$$B_{kmn} = -(\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \boldsymbol{\varphi}_n). \tag{13}$$

• **<u>POD-G-ROM</u>** in matrix form:

$$\dot{a} = b + Aa + a^T Ba \tag{8}$$

- POD-G-ROM (8) has been successfully used for *laminar flows*.
- For *structurally-dominated turbulent flows*, POD-G-ROM *fails*: effect of discarded modes $\varphi_{r+1}, ..., \varphi_N$ need to be included in some way.

• **<u>POD-G-ROM</u>** algebraic system:

$$\dot{a}_k(t) = b_k + \sum_{m=1}^r A_{km} a_m(t) + \sum_{m=1}^r \sum_{n=1}^r B_{kmn} a_n(t) a_m(t),$$
(10)

where:

$$\boldsymbol{b}_{k} = -(\boldsymbol{\varphi}_{k}, \boldsymbol{\mathsf{U}} \cdot \nabla \boldsymbol{\mathsf{U}}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_{k}, \frac{\nabla \boldsymbol{\mathsf{U}} + \nabla \boldsymbol{\mathsf{U}}^{\mathrm{T}}}{2} \right), \tag{11}$$

$$A_{km} = -(\boldsymbol{\varphi}_k, \mathbf{U} \cdot \nabla \boldsymbol{\varphi}_m) - (\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \mathbf{U}) - \frac{2}{Re} \left(\nabla \boldsymbol{\varphi}_k, \frac{\nabla \boldsymbol{\varphi}_m + \nabla \boldsymbol{\varphi}_m^T}{2} \right), \quad (12)$$

$$B_{kmn} = -(\boldsymbol{\varphi}_k, \boldsymbol{\varphi}_m \cdot \nabla \boldsymbol{\varphi}_n). \tag{13}$$

• **<u>POD-G-ROM</u>** in matrix form:

$$\dot{a} = b + Aa + a^T Ba \tag{8}$$

- POD-G-ROM (8) has been successfully used for *laminar flows*.
- For *structurally-dominated turbulent flows*, POD-G-ROM *fails*: effect of discarded modes $\varphi_{r+1}, ..., \varphi_N$ need to be included in some way.

"<u>POD closure problem</u>"

Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows

3. Background on Turbulence Modeling/Large Eddy Simulation

- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]

Turbulence



• For sufficiently high Reynolds number, flow becomes *turbulent*.



Turbulence



• For sufficiently high Reynolds number, flow becomes *turbulent*.



• Turbulence is *nonlinear, chaotic, 3D* phenomenon.

Turbulence



• For sufficiently high Reynolds number, flow becomes *turbulent*.



- Turbulence is *nonlinear, chaotic, 3D* phenomenon.
- Kolmogorov hypothesis / energy cascade:
 - Kinetic energy enters the turbulence through the production mechanism at largest scales of motion.
 - Energy is transferred (by inviscid processes) to smaller and smaller scales.
 - At smallest scales, energy is dissipated by viscous action.

Turbulence Modeling



- Direct Numerical Simulation (DNS): solves full Navier-Stokes (NS) equations
 (1) → requires fine meshes in boundary layer to resolve fine scales.
 - Too computationally expensive to be feasible for realistic complex flows.
- <u>Large Eddy Simulation (LES)</u>: reduces computational cost of DNS by ignoring smallest length scales (most computationally expensive to resolve).
 - LES equations obtained by low-pass-filtering full NS equations.
- <u>**Reynolds Averaged Navier-Stokes (RANS)**</u>: time-averaged versions of NS equations → turbulence is modeled, not resolved.



Uriel Frisch
Large Eddy Simulation (LES)







Left: energy cascade / "Kolmogorov spectrum" (energy transfer from large to small scales); LES filter filters out small scales.

- Four conceptual steps of LES (Pope, Chapter 13):
 - (i) **Filtering operation** to decompose velocity into filtered (or resolved) component $\overline{u}(x,t)$ and residual (or subgrid-scale) component u'(x,t).
 - (ii) *Equations for evolution* of the filtered velocity are derived from the NS equations.
 - (iii) *Closure* is obtained by modeling the residual-stress tensor (most simply with *eddyviscosity model*).
 - (iv) Model filtered equations are solved *numerically* for $\overline{u}(x,t)$, which provides approximation of large-scale motions in one realization of turbulent flow.

LES Filtering

• General *filtering operation* defined by:

$$\overline{\boldsymbol{u}}(\boldsymbol{x},t) = \int \boldsymbol{G}(\boldsymbol{r},\boldsymbol{x})\boldsymbol{u}(\boldsymbol{x}-\boldsymbol{r},t)d\boldsymbol{r}$$

where **G** is a specified rapidly-decaying "filter function", which has an associated "cut-off" length and time scale. Scales smaller than these cut-offs are eliminated using filter.



 $_{\text{Figure 13.1:}}$ Filters G(r): box filter, dashed line; Gaussian filter, solid line; sharp spectral, dot-dashed line.

LES Filtering

• General *filtering operation* defined by:

$$\overline{\boldsymbol{u}}(\boldsymbol{x},t) = \int \boldsymbol{G}(\boldsymbol{r},\boldsymbol{x})\boldsymbol{u}(\boldsymbol{x}-\boldsymbol{r},t)d\boldsymbol{r}$$

where **G** is a specified rapidly-decaying "filter function", which has an associated "cut-off" length and time scale. Scales smaller than these cut-offs are eliminated using filter.

Given a filter, any field can be split up into *filtered* u
 and *sub-filtered* u
 scale:

 $\boldsymbol{u}(\boldsymbol{x},t) = \overline{\boldsymbol{u}}(\boldsymbol{x},t) + \boldsymbol{u}'(\boldsymbol{x},t)$



 $_{\rm Figure \ 13.1:}$ Filters G(r): box filter, dashed line; Gaussian filter, solid line; sharp spectral, dot-dashed line.



Figure 13.2: Upper curves: sample of velocity field U(x) and the corresponding filtered field $\overline{U}(x)$ (bold line), using the Gaussian filter with $\Delta \approx 0.35$. Lower curves: residual field u'(x) and the filtered residual field $\overline{u'(x)}$ (bold line).

LES Filtered Governing Equations



• Applying filtering operation to (1) gives the following equations for the *filtered variables*:

$$\nabla \cdot \overline{\boldsymbol{u}} = 0$$
$$\overline{\boldsymbol{u}}_t - \frac{1}{Re} \Delta \overline{\boldsymbol{u}} + \nabla \cdot (\overline{\boldsymbol{u}} \overline{\boldsymbol{u}}) + \nabla \cdot \boldsymbol{\tau} + \nabla \overline{p} = 0$$

where au is the *subfilter-scale stress tensor*:

$$\tau = \overline{u}\overline{u} - \overline{u}\overline{u}$$

LES Filtered Governing Equations



• Applying filtering operation to (1) gives the following equations for the *filtered variables*:

$$\nabla \cdot \overline{\boldsymbol{u}} = 0$$
$$\overline{\boldsymbol{u}}_t - \frac{1}{Re} \Delta \overline{\boldsymbol{u}} + \nabla \cdot (\overline{\boldsymbol{u}} \overline{\boldsymbol{u}}) + \nabla \cdot \boldsymbol{\tau} + \nabla \overline{p} = 0$$

where au is the *subfilter-scale stress tensor*:

$$\boldsymbol{\tau} = \overline{\boldsymbol{u}}\overline{\boldsymbol{u}} - \overline{\boldsymbol{u}}\overline{\boldsymbol{u}}$$

• \overline{uu} is most difficult term to model, as it requires knowledge of unfiltered velocity field, which is unknown.

LES Filtered Governing Equations



• Applying filtering operation to (1) gives the following equations for the *filtered variables*:

$$\nabla \cdot \overline{\boldsymbol{u}} = 0$$
$$\overline{\boldsymbol{u}}_t - \frac{1}{Re} \Delta \overline{\boldsymbol{u}} + \nabla \cdot (\overline{\boldsymbol{u}} \overline{\boldsymbol{u}}) + \nabla \cdot \boldsymbol{\tau} + \nabla \overline{p} = 0$$

where au is the *subfilter-scale stress tensor*:

$$au = \overline{u}\overline{u} - \overline{u}\overline{u}$$

- *uu* is most difficult term to model, as it requires knowledge of unfiltered velocity field, which is unknown.
- Common approaches to model \overline{uu} : *eddy-viscosity (EV) models*

$$\tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = \nu_T \left(\frac{\partial \bar{u}_j}{\partial x_i} + \frac{\partial \bar{u}_i}{\partial x_j}\right)$$

where v_T is the eddy-viscosity.

- Expression for v_T : *"eddy-viscosity ansatz*"
- Examples: mixing-length, Smagorinsky, etc. parameters based on Kolmogorov spectrum.

Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]



- POD-G-ROM (8) has been successfully used for *laminar flows*.
- For *structurally-dominated turbulent flows*, POD-G-ROM *fails*: effect of discarded modes $\varphi_{r+1}, ..., \varphi_N$ need to be included in some way.

"POD closure problem"



- POD-G-ROM (8) has been successfully used for *laminar flows*.
- For *structurally-dominated turbulent flows*, POD-G-ROM *fails*: effect of discarded modes $\varphi_{r+1}, ..., \varphi_N$ need to be included in some way.

"POD closure problem"

- Natural way to tackle POD closure problem: "eddy-viscosity" (EV) turbulence modeling.
 - EV model states that role of discarded modes is to extract energy from system.
 - Concept of energy cascade has been confirmed numerically in POD setting.



- POD-G-ROM (8) has been successfully used for *laminar flows*.
- For *structurally-dominated turbulent flows*, POD-G-ROM *fails*: effect of discarded modes $\varphi_{r+1}, ..., \varphi_N$ need to be included in some way.

"POD closure problem"

- Natural way to tackle POD closure problem: "eddy-viscosity" (EV) turbulence modeling.
 - EV model states that role of discarded modes is to extract energy from system.
 - Concept of energy cascade has been confirmed numerically in POD setting.
- <u>EV-POD-ROM</u> formulation:

 $\dot{a} = (b + \tilde{b}(a)) + (A + \tilde{A}(a))a + a^{T}Ba$ (24)

where $\tilde{b}(a)$ and $\tilde{A}(a)$ correspond to numerical discretization of EV closure model.



- POD-G-ROM (8) has been successfully used for *laminar flows*.
- For *structurally-dominated turbulent flows*, POD-G-ROM *fails*: effect of discarded modes $\varphi_{r+1}, ..., \varphi_N$ need to be included in some way.

"POD closure problem"

- Natural way to tackle POD closure problem: "eddy-viscosity" (EV) turbulence modeling.
 - EV model states that role of discarded modes is to extract energy from system.
 - Concept of energy cascade has been confirmed numerically in POD setting.
- <u>EV-POD-ROM</u> formulation:

$$\dot{a} = (b + \tilde{b}(a)) + (A + \tilde{A}(a))a + a^{T}Ba$$
(24)

where $\tilde{b}(a)$ and $\tilde{A}(a)$ correspond to numerical discretization of EV closure model.

(24) is equivalent to adding $\nu_T \left(\frac{\partial \overline{u}_j}{\partial x_i} + \frac{\partial \overline{u}_i}{\partial x_j} \right)$ to equations (1) and projecting.



- POD-G-ROM (8) has been successfully used for *laminar flows*.
- For *structurally-dominated turbulent flows*, POD-G-ROM *fails*: effect of discarded modes $\varphi_{r+1}, ..., \varphi_N$ need to be included in some way.

"POD closure problem"

- Natural way to tackle POD closure problem: "eddy-viscosity" (EV) turbulence modeling.
 - EV model states that role of discarded modes is to extract energy from system.
 - Concept of energy cascade has been confirmed numerically in POD setting.
- <u>EV-POD-ROM</u> formulation:

$$\dot{a} = (b + \tilde{b}(a)) + (A + \tilde{A}(a))a + a^{T}Ba$$
(24)

where $\tilde{b}(a)$ and $\tilde{A}(a)$ correspond to numerical discretization of EV closure model.

(24) is equivalent to adding $v_T \left(\frac{\partial \overline{u}_j}{\partial x_i} + \frac{\partial \overline{u}_i}{\partial x_j} \right)$ to equations (1) and projecting.

- **Four EV-POD-ROMs** of the form (24) proposed/evaluated:
- 1. *ML-POD-ROM* (ML = mixing length). 3. *VMS-POD-ROM* (VMS = variational multi-scale)
- 2. **S-POD-ROM** (S = Smagorinsky). 4. **DS-POD-ROM** (DS = dynamic subgrid)

POD Filter and Lengthscale (Section 3.1-3.2)

- **POD/Galerkin projection filter** (Section 3.1):
 - In POD, there is no explicit spatial filter used ⇒ to develop LES-type POD closure models, a POD filter needs to be introduced.
 - **Natural filter is Galerkin projection**: for all $u \in \mathcal{H}$, the Galerkin projection $\overline{u} \in X^r$ is the solution to the following equation:

$$(\mathbf{u} - \bar{\mathbf{u}}, \boldsymbol{\varphi}) = 0 \quad \forall \boldsymbol{\varphi} \in \mathbf{X}^r.$$
 (14)

By doing POD/Galerkin projection to build the ROM, one is applying a filter. In the context of LES, filtered equations require introduction of closure model to model effect of neglected POD modes. This is where idea of adding EV models to ROM equations comes from.

POD Filter and Lengthscale (Section 3.1-3.2)

- **POD/Galerkin projection filter** (Section 3.1):
 - In POD, there is no explicit spatial filter used ⇒ to develop LES-type POD closure models, a POD filter needs to be introduced.
 - **Natural filter is Galerkin projection**: for all $u \in \mathcal{H}$, the Galerkin projection $\overline{u} \in X^r$ is the solution to the following equation:

$$(\mathbf{u} - \bar{\mathbf{u}}, \boldsymbol{\varphi}) = 0 \quad \forall \boldsymbol{\varphi} \in \mathbf{X}^r.$$
 (14)

By doing POD/Galerkin projection to build the ROM, one is applying a filter. In the context of LES, filtered equations require introduction of closure model to model effect of neglected POD modes. This is where idea of adding EV models to ROM equations comes from.

• **POD lengthscale** (Section 3.2): implicitly defined by neglected modes $\{\varphi_j\}_{j=r+1}^N$

$$\delta := \left(\frac{1}{L_1 L_2} \int_0^{L_1} \int_0^{L_2} \frac{\langle u_{i>} u_{i>} \rangle}{\langle u_{i>} u_{i>} j \rangle} dx_1 dx_2\right)^{1/2}.$$
(23)

where $u_{i>} = \sum_{j=r+1}^{N} a_j^i \varphi_j$, $\langle \cdot \rangle =$ spatial average in homogeneous direction, L_1, L_2 are streamwise and spanwise dimensions of computational domain.

Section 3.3.1: ML*-POD-ROM



- Mixing length model: $v_T = v_{ML} = \alpha U_{ML}L_{ML}$.
 - U_{ML} = characteristic velocity scale (estimated using dimensional analysis; Sec. 3.2).
 - L_{ML} = characteristic length scale (estimated using dimensional analysis; Sec. 3.2).
 - $\alpha = O(1)$ non-dimensional parameter that characterized energy being dissipated.
- *ML-POD-ROM* is of form (24) with:

$$\widetilde{b}_{k}(\mathbf{a}) = -v_{ML} \left(\nabla \boldsymbol{\varphi}_{k}, \frac{\nabla \mathbf{U} + \nabla \mathbf{U}^{T}}{2} \right),$$

$$\widetilde{A}_{km}(\mathbf{a}) = -v_{ML} \left(\nabla \boldsymbol{\varphi}_{k}, \frac{\nabla \boldsymbol{\varphi}_{m} + \nabla \boldsymbol{\varphi}_{m}^{T}}{2} \right).$$
(27)

- Remarks:
 - Different values of α may result in different dynamics (α varies in real turbulent flow).
 - v_{ML} typically computed once at beginning of simulation.
 - Improvements to ML-POD-ROM where v_{ML} are mode dependent have been proposed in [30,32,58].

Section 3.3.2: S*-POD-ROM

- Smagorinsky model: $v_T = v_S = (C_S \delta)^2 ||\mathbb{D}(\boldsymbol{u}_r)||_{F}$.
 - C_S = Smagorinsky constant.
 - $\delta = \text{length scale}$ (estimated using dimensional analysis; Sec. 3.2).
 - $||\mathbb{D}(\boldsymbol{u}_r)||_F$ = Frobenius norm of deformation tensor.
 - $v_S =$ "EV ansatz".
- S-POD-ROM is of form (24) with:

$$\tilde{b}_{k}(\mathbf{a}) = -2(C_{S}\delta)^{2} \left(\nabla \boldsymbol{\varphi}_{k}, \|\mathbb{D}(\mathbf{u}_{r})\| \frac{\nabla \mathbf{U} + \nabla \mathbf{U}^{T}}{2} \right),$$
(30)
$$\tilde{A}_{km}(\mathbf{a}) = -2(C_{S}\delta)^{2} \left(\nabla \boldsymbol{\varphi}_{k}, \|\mathbb{D}(\mathbf{u}_{r})\| \frac{\nabla \boldsymbol{\varphi}_{m} + \nabla \boldsymbol{\varphi}_{m}^{T}}{2} \right).$$
(31)

- Remarks:
 - Main advantage over ML-POD-ROM: EV coefficient recomputed at every timestep.
 - EV terms are nonlinear and need to be handled efficiently discussed in Section 4.

Section 3.3.3: VMS*-POD-ROM



- **VMS LES**: based on principle of locality of energy transfer (energy is transferred mainly between neighboring scales) shown to be valid in POD context [43].
- Decompose space of POD modes into 2 spaces, one of "large" and one of "small" scale modes: $X^r = X_L^r \bigoplus X_S^r$ where:

$$\mathbf{X}_{L}^{r} := \operatorname{span}\left\{\boldsymbol{\varphi}_{1}, \boldsymbol{\varphi}_{2}, \dots, \boldsymbol{\varphi}_{r_{L}}\right\} \qquad \mathbf{X}_{S}^{r} := \operatorname{span}\left\{\boldsymbol{\varphi}_{r_{L}+1}, \boldsymbol{\varphi}_{r_{L}+2}, \dots, \boldsymbol{\varphi}_{r}\right\}.$$

• Decompose ROM solution into "large resolved" and "small resolved" scales: $u_r = u_r^L + u_r^S$ where: r_L r_L r_L

$$\mathbf{u}_r^L = \mathbf{U} + \sum_{j=1}^r a_j \boldsymbol{\varphi}_j, \qquad \mathbf{u}_r^S = \sum_{j=r_L+1}^r a_j \boldsymbol{\varphi}_j.$$

• VMS-POD-ROM has the form:

$$\begin{bmatrix} \dot{\mathbf{a}}^{L} \\ \dot{\mathbf{a}}^{S} \end{bmatrix} = \begin{bmatrix} \mathbf{b}^{L} \\ \mathbf{b}^{S} \end{bmatrix} + \mathbf{A}^{r} \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix} + \begin{bmatrix} \mathbf{A}^{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^{S} + \widetilde{\mathbf{A}}^{S}(\mathbf{a}^{S}) \end{bmatrix} \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix} + \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix}^{T} \mathbf{B} \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix}.$$
(38)

Section 3.3.3: VMS-POD-ROM



• VMS-POD-ROM has the form:

$$\begin{bmatrix} \dot{\mathbf{a}}^{L} \\ \dot{\mathbf{a}}^{S} \end{bmatrix} = \begin{bmatrix} \mathbf{b}^{L} \\ \mathbf{b}^{S} \end{bmatrix} + \mathbf{A}^{r} \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix} + \begin{bmatrix} \mathbf{A}^{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^{S} + \widetilde{\mathbf{A}}^{S}(\mathbf{a}^{S}) \end{bmatrix} \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix} + \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix}^{T} \mathbf{B} \begin{bmatrix} \mathbf{a}^{L} \\ \mathbf{a}^{S} \end{bmatrix}.$$
(38)

- Remarks:
 - EV term applied to small scales only, following principle of energy transfer locality:

$$\widetilde{A}_{kj}^{S}(\mathbf{a}) = -2(C_{S}\delta)^{2}\left(\nabla\boldsymbol{\varphi}_{k}, \|\mathbb{D}(\mathbf{u}_{r}^{S}+\mathbf{U})\|\frac{\nabla\boldsymbol{\varphi}_{j}+\nabla\boldsymbol{\varphi}_{j}^{T}}{2}\right).$$

- Unlike S-POD-ROM, VMS-POD-ROM acts only on small resolved scales, whereas in S-POD-ROM, it acts on all (small and large) resolved scales.
- (38) is coupled through two terms:
 - (i) $\boldsymbol{a}^T \boldsymbol{B} \boldsymbol{a}$: represents nonlinear convective term $(\boldsymbol{u}^r \cdot \nabla) \boldsymbol{u}^r$
 - (ii) $A^T a$: represents nonlinear term $(u^r \cdot \nabla) u^r$ linearized around base flow U.
- EV terms are nonlinear and need to be handled efficiently discussed in Section 4.

Section 3.3.4 : DS*-POD-ROM



- Dynamic subgrid model: $v_T = v_{DS} = (C_S(\mathbf{x}, t)\delta)^2 ||\mathbb{D}(\mathbf{u}_r)||_{F}$.
- **DS-POD-ROM** is of form (24) with:

$$\tilde{b}_{k}(\mathbf{a}) = -2(C_{S}\delta)^{2} \left(\nabla \boldsymbol{\varphi}_{k}, \|\mathbb{D}(\mathbf{u}_{r})\| \frac{\nabla \mathbf{U} + \nabla \mathbf{U}^{T}}{2} \right),$$
(30)
$$\tilde{A}_{km}(\mathbf{a}) = -2(C_{S}\delta)^{2} \left(\nabla \boldsymbol{\varphi}_{k}, \|\mathbb{D}(\mathbf{u}_{r})\| \frac{\nabla \boldsymbol{\varphi}_{m} + \nabla \boldsymbol{\varphi}_{m}^{T}}{2} \right).$$
(31)

• Least-squares problem for $C_S(x, t)$ is obtained by applying filtering twice to ROM equations, assuming $C_S(x, t)$ is constant under double filtering, and equating terms.

$$\Rightarrow \begin{bmatrix} \widetilde{\mathbf{u}_{s}}(\mathbf{x},t) \\ = \frac{\left[\widetilde{\mathbf{u}_{r}}\widetilde{\mathbf{u}_{r}} - \widetilde{\mathbf{u}_{r}}\widetilde{\mathbf{u}_{r}}\right] : \left[2\delta^{2}\|\mathbb{D}(\overline{\mathbf{u}_{r}})\|\mathbb{D}(\overline{\mathbf{u}_{r}}) - 2\widetilde{\delta}^{2}\|\mathbb{D}(\widetilde{\mathbf{u}_{r}})\|\mathbb{D}(\widetilde{\mathbf{u}_{r}})\right]}{\left[2\delta^{2}\|\mathbb{D}(\overline{\mathbf{u}_{r}})\|\mathbb{D}(\overline{\mathbf{u}_{r}}) - 2\widetilde{\delta}^{2}\|\mathbb{D}(\widetilde{\mathbf{u}_{r}})\|\mathbb{D}(\widetilde{\mathbf{u}_{r}})\right] : \left[2\delta^{2}\|\mathbb{D}(\overline{\mathbf{u}_{r}})\|\mathbb{D}(\overline{\mathbf{u}_{r}}) - 2\widetilde{\delta}^{2}\|\mathbb{D}(\widetilde{\mathbf{u}_{r}})\|\mathbb{D}(\widetilde{\mathbf{u}_{r}})\right]} \right]}.$$
rks:

$$(63)$$

- Remarks:
 - v_{DS} can take on negative values can be interpreted as **backscatter** (inverse transfer of energy from high index POD modes to low index modes).
 - Notion of backscatter is well-established in LES.
 - EV terms are nonlinear and need to be handled efficiently discussed in Section 4.

Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]

Ensuring Computational Efficiency (Section 4.1)



- All EV-POD-ROMs have *nonlinear closure model* terms except ML-POD-ROMs.
- Two approaches used to ensure *computational efficiency*:
 - (i) Instead of updating closure terms in ROMs at every time step, re-compute them every 1.5 time units (every 20K time steps for numerical example considered).
 - (ii) Two-level algorithm: create 2 meshes (coarse and fine); discretize/compute closure terms $\tilde{b}(a^l)$, $\tilde{A}(a^l)$ on coarse mesh.

```
 \begin{array}{l} \ell = 0; \text{ compute } \mathbf{b}, \mathbf{A}, \mathbf{B} \text{ on the } \underline{\text{fine mesh}}; \\ \text{for } \ell = 0 \text{ to } M - 1 \\ \text{compute } \widetilde{\mathbf{b}}(\mathbf{a}^{\ell}), \widetilde{\mathbf{A}}(\mathbf{a}^{\ell}) \text{ on the } \underline{\text{coarse mesh}} \\ \mathbf{a}^{\ell+1} := \widetilde{F}(\mathbf{a}^{\ell}); \\ \text{end for} \end{array}   two-level algorithm  \begin{array}{l} M = \text{total number} \\ \text{of time steps.} \end{array}
```

- Both (i) and (ii) were applied to all ROMs for fair comparison (even ML-POD-ROM).
- In [50], it was shown that two-level algorithm achieves same level of accuracy as one-level algorithm while decreasing computational cost by order of magnitude.

Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]



Numerical Example: 3D Flow Past Re = 1000 Cylinder (Section 4.1)



- FOM = finite difference DNS solver + 2nd order Crank-Nicolson & Adams-Bashforth timeintegration scheme (dt = 7.5e-4).
- 1000 *snapshots* collected of velocity field over time [0,75]; ROM run for time [0, 300].
- **POD basis** of size r = 6 created using snapshots; modes capture 84% of snapshot energy.
- POD ROMs created using *continuous projection* and P2 finite elements + forward Euler timeintegration scheme (dt = 2e-3).
- Projection of *quadratic nonlinearities* in incompressible NS equations pre-computed.
- **Closure model terms** computed using (i) and (ii) on previous slide; coarse mesh for (ii) had 37x49x17 grid points (coursing factor $R_c = 4$ in both radial and azimuthal directions).

Numerical Example: Estimation of Parameters (Section 4.1)



- Length scales in EV-POD-ROMs estimated using *dimensional analysis* (Section 3.2).
- "Correct" values for *EV constants* α in ML-POD-ROM and C_S in S-POD-ROM and VMS-POD-ROM are not known in POD context. These parameters are estimated as follows:
 - Run POD-ROM on short time interval [0,15] with several different values of EV constants.
 - Choose value that gives closest result to DNS.
 - <u>Note</u>: these EV constants are optimal only on short time interval tested might be non-optimal for (longer) time-interval where ROM is run.
- For *VMS-POD-ROM*, only first mode considered large resolved scale, so $r_L = 1$.
- For **DS-POD-ROM**, since v_{DS} can be negative, a standard "clipping" procedure is used to ensure numerical stability of discretization: let $C_S(x, t) = \max\{C_S(x, t), -0.2\}$, where 0.2 is determined numerically (see paper for details).

Numerical Example: Estimation of Parameters (Section 4.1)



- Length scales in EV-POD-ROMs estimated using *dimensional analysis* (Section 3.2).
- "Correct" values for *EV constants* α in ML-POD-ROM and C_S in S-POD-ROM and VMS-POD-ROM are not known in POD context. These parameters are estimated as follows:
 - Run POD-ROM on short time interval [0,15] with several different values of EV constants.
 - Choose value that gives closest result to DNS.
 - <u>Note</u>: these EV constants are optimal only on short time interval tested might be non-optimal for (longer) time-interval where ROM is run.
- For *VMS-POD-ROM*, only first mode considered large resolved scale, so $r_L = 1$.
- For **DS-POD-ROM**, since v_{DS} can be negative, a standard "clipping" procedure is used to ensure numerical stability of discretization: let $C_S(x, t) = \max\{C_S(x, t), -0.2\}$, where 0.2 is determined numerically (see paper for details).

<u>Comment</u>: authors consider one basis size r. Approach could be modified so $v_T \rightarrow 0$ as $r \rightarrow K$ for consistency.

Sandia National Laboratories

Criteria for Evaluating ROMs (Section 4.1)

- *Five criteria* are used for evaluating ROMs:
 - (i) Kinetic energy spectrum.
 - (ii) Mean velocity.
 - (iii) Reynolds stresses.
 - (iv) Root mean square (rms) values of velocity and fluctuations.
 - (v) Time-evolution of POD coefficients.

Sandia National Laboratories

Criteria for Evaluating ROMs (Section 4.1)

- *Five criteria* are used for evaluating ROMs:
 - (i) Kinetic energy spectrum.
 - (ii) Mean velocity.
 - (iii) Reynolds stresses.
 - (iv) Root mean square (rms) values of velocity and fluctuations.
 - (v) Time-evolution of POD coefficients.

Comment: These are similar criteria to those we are interested in!

Kinetic Energy Spectrum (Section 4.1)





- All energy spectra calculated from average kinetic energy (KE) at a single point.
- POD-G-ROM: over-estimates energy spectrum.
- ML-POD-ROM: underestimates energy spectrum, especially at higher frequencies.
- S-POD-ROM: more accurate than ML-POD-ROM, but displays high oscillations at higher frequencies.
- VMS-POD-ROM: improvement over S-POD-ROM, with smaller oscillations at higher frequencies.

DS-POD-ROM and VMS-POD-ROM yield most accurate energy spectra, with DS-POD-ROM slightly better than VMS-POD-ROM.

Mean Velocity Components (Section 4.1)





- $\langle u \rangle$ = mean streamwise velocity
- $\langle v \rangle$ = mean normal velocity
- $\langle w \rangle =$ mean spanwise velocity
- $\langle \cdot \rangle$ denotes time-averaging for t=[0,300] and spatial averaging performed in *yz*-direction.
- Mean streamwise velocity computed accurately by all POD-ROMs.
- POD-G-ROM yields inaccurate results for mean normal velocity; all other POD-ROMs performed significantly better.

Mean spanwise velocity results similar for all EV-ROMs; POD-G-ROM performed better than all EV-ROMs over certain regions worse over others.

Reynolds Stresses (Section 4.1)





- $\langle u \langle u \rangle, v \langle v \rangle \rangle$: *xy*-component of Reynolds stress.
- $\langle u \langle u \rangle, w \langle w \rangle \rangle$: *xz*-component of Reynolds stress.
- $\langle v \langle v \rangle, w \langle w \rangle \rangle$: *yz*-component of Reynolds stress.
- POD-G-ROM Reynolds stresses are consistently most inaccurate.

EV-ROMs have similar behaviors; no clear "winner".

Root Mean Square Values (Section 4.1)





- $\langle u \rangle_{rms} = \langle u \langle u \rangle, u \langle u \rangle \rangle$: rms of streamwise velocity fluctuations.
- $\langle v \rangle_{rms} = \langle v \langle v \rangle, v \langle v \rangle \rangle$: rms of normal velocity fluctuations.
- $\langle w \rangle_{rms} = \langle w \langle w \rangle, w \langle w \rangle \rangle$: rms of spanwise velocity fluctuations.
- POD-G-ROM rms values of velocity fluctuations are consistently the most inaccurate.
- DS-POD-ROM and VMS-POD-ROMs consistently outperformed other two EV-ROMs.

S-POD-ROM consistently performs worse than DS-POD-ROM and VMS-POD-ROM, but is clearly more accurate than ML-POD-ROM.

Time Evolution of POD Coefficients (Section 4.1)





- Left: a_1 , right: a_4 .
 - POD-G-ROM's time evolutions of a₁ and a₄ are clearly inaccurate.
 - ML-POD-ROMs time evolutions also inaccurate.
- S-POD-ROM more accurate than ML-POD-ROM.
- VMS-POD-ROM more accurate than S-POD-ROM.
- DS-POD-ROM yields accurate results.
- More variability in coefficients for DS-POD-ROM due to C_S varying with space and time.

VMS-POD-ROM and DS-POD-ROM perform best.

CPU Times (Section 4.1)



• To measure computational efficiency of the four POD-ROMs, define *speed-up factor*:

 $S_f = \frac{\text{CPU time of DNS}}{\text{CPU time of POD-ROM}}$

(68)

• Table 1 gives speed-up factors for POD-ROMs:

Table 1 Speed-up factors of POD-ROMs.

	POD-G-	ML-POD-	S-POD-	VMS-POD-	DS-POD-
	ROM	ROM	ROM	ROM	ROM
Sy	665	659	36	41	23

 POD-G-ROM is most *efficient*; as sophistication of turbulence model increases, model becomes *more expensive*, not surprisingly.

CPU Times (Section 4.1)



• To measure computational efficiency of the four POD-ROMs, define *speed-up factor*:

 $S_f = \frac{\text{CPU time of DNS}}{\text{CPU time of POD-ROM}}$

(68)

• Table 1 gives speed-up factors for POD-ROMs:

Table 1 Speed-up factors of POD-ROMs.										
	POD-G- ROM	ML-POD- ROM	S-POD- ROM	VMS-POD- ROM	DS-POD- ROM					
Sy	665	659	36	41	23					

 POD-G-ROM is most *efficient*; as sophistication of turbulence model increases, model becomes *more expensive*, not surprisingly.

<u>Comment</u>: it may be possible to improve these speed-up factors using hyperreduction to handle non-linear terms (e.g., DEIM, gappy POD).

Summary of Results (Section 4.1)



- VMS and DS approaches yield most accurate POD closure models (i.e., give most accurate average and instantaneous numerical results):
 - Best energy spectra.
 - Best rms values.
 - Best time evolution of POD coefficients a_1 and a_4 .
 - With respect to other criteria (mean velocity components, Reynolds stresses), DS-POD-ROM and VMS-POD-ROM perform at least as well as other POD-ROMs.

Summary of Results (Section 4.1)



- **VMS** and **DS** approaches yield **most accurate** POD closure models (i.e., give most accurate average and instantaneous numerical results):
 - Best energy spectra.
 - Best rms values.
 - Best time evolution of POD coefficients a_1 and a_4 .
 - With respect to other criteria (mean velocity components, Reynolds stresses), DS-POD-ROM and VMS-POD-ROM perform at least as well as other POD-ROMs.

<u>Comment:</u> one cannot make these definitive conclusions from data presented because all the ROMs do not have the same computational cost (see previous slide)... but it is possible to improve computational efficiency of EV-ROMs using hyper-reduction like DEIM or gappy POD.
Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]

Future Work (Section 5)



- Study *more efficient time-discretization approaches* and take advantage of parallel computing in POD-ROMs.
- Investigate *hybrid approach*: using DS-POD-ROM to calculate α only when flow displays high level of variability, use this value in ML-POD-ROM (linear) as long as flow does not experience sudden transitions.
- *Higher Reynolds number* structurally dominated turbulent flows.
- Combining two-level algorithms in conjunction with *EIM* and *DEIM* to handle efficiently nonlinearities in turbulence models.
- Application to problems in *optimal control, optimization, data assimilation.*

Follow Up Work by Iliescu et al.



• Filtered ROMs (F-ROMs)

- ROM terms are explicitly filtered using projection or differential filter.
 - E.g., "*Evolve-then-filter*" approach: do one step of ROM, filter ROM amplitudes $a_i(t)$, repeat.
- Calibrated ROMs (C-ROMs)
 - Turbulence model terms, e.g., \widetilde{A} , are obtained by solving optimization problem.

$$min_{\widetilde{A}}\sum_{j=1}^{M}||a(t_j)-a^{snap}(t_j)||^2$$

Unlike LES-ROMs, F-ROMs and C-ROMs are *consistent* and more *computationally efficient* (no nonlinear turbulence model terms to compute).

<u>References:</u>

- D. Wells, Z. Wang, X. Xie, T. Iliescu, "An Evolve-Then-Filter Regularized Reducer Order Model for Convection-Dominated Flows", <u>https://arxiv.org/abs/1506.07555</u>.
- X. Xie, M. Mohebujjaman, L.G. Rebholz, T. Iliescu, "Calibrated Filtered Reduced Order Modeling", <u>https://arxiv.org/abs/1702.06886</u>.
- Technical seminar by X. Xie in March 2017: "LES ROMs".

Other Follow Up Work



- <u>Closure models with auto-tuned data-driven coefficients:</u>
 - Free parameters in closure models are "learned" online using data-driven multi-parameter extremum seeking (MES) algorithm.
 - Takes into account parametric uncertainties
 - Employs robust Lyapunov control theory.

M. Benosman, J. Borggaard, O. San, B. Kramer. "Learning-based robust stabilization for reduced-order models of 2D and 3D Boussinesq equations", Applied Mathematical Modelling 49 (2017) 162–181.

Outline



- 1. Motivation/Background
- 2. Section 2: POD/Galerkin ROMs for Incompressible Flows
- 3. Background on Turbulence Modeling/Large Eddy Simulation
- 4. Section 3: POD Closure Models
 - Mixing Length (ML)
 - Smagorinsky (S)
 - Variational Multi-Scale (VMS)
 - Dynamic Subgrid (DS)
- 5. Section 4.1: Computational Efficiency
- 6. Section 4.2: Numerical Results for 3D Flow Around Cylinder
- 7. Section 5 and Beyond: Future/Follow-Up Work
- [8. Basis Rotation (My Work an Alternative Approach)]

3D Compressible Navier-Stokes Equations

• We start with the 3D compressible Navier-Stokes equations in *primitive specific volume form:*

$$\zeta_{,t} + \zeta_{,j}u_j - \zeta u_{j,j} = 0$$

$$u_{i,t} + u_{i,j}u_j + \zeta p_{,i} - \frac{1}{Re}\zeta\tau_{ij,j} = 0$$

$$p_{,t} + u_j p_{,j} + \gamma u_{j,j}p - \left(\frac{\gamma}{PrRe}\right)\left(\kappa(p\zeta)_{,j}\right)_{,j} - \left(\frac{\gamma-1}{Re}\right)u_{i,j}\tau_{ij} = 0$$
(1)

[PDEs]

3D Compressible Navier-Stokes Equations

 We start with the 3D compressible Navier-Stokes equations in <u>primitive</u> <u>specific volume form</u>:

$$\zeta_{,t} + \zeta_{,j}u_{j} - \zeta u_{j,j} = 0$$

$$u_{i,t} + u_{i,j}u_{j} + \zeta p_{,i} - \frac{1}{Re}\zeta\tau_{ij,j} = 0$$

$$p_{,t} + u_{j}p_{,j} + \gamma u_{j,j}p - \left(\frac{\gamma}{PrRe}\right)\left(\kappa(p\zeta)_{,j}\right)_{,j} - \left(\frac{\gamma-1}{Re}\right)u_{i,j}\tau_{ij} = 0$$
(1)

• Spectral discretization $(\boldsymbol{q}(\boldsymbol{x},t) \approx \sum_{i=1}^{n} a_i(t) \boldsymbol{U}_i(\boldsymbol{x})) + \text{Galerkin projection}$ applied to (1) yields a system of <u>*n* coupled quadratic ODEs</u>:

[PDEs]

$$\frac{d\boldsymbol{a}}{dt} = \boldsymbol{C} + \boldsymbol{L}\boldsymbol{a} + [\boldsymbol{a}^T \boldsymbol{Q}^{(1)} \boldsymbol{a} + \boldsymbol{a}^T \boldsymbol{Q}^{(2)} \boldsymbol{a} + \dots + \boldsymbol{a}^T \boldsymbol{Q}^{(n)} \boldsymbol{a}]^T$$
(2)

where $\boldsymbol{C} \in \mathbb{R}^{n}$, $\boldsymbol{L} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{Q}^{(i)} \in \mathbb{R}^{n \times n}$ for all i = 1, ..., n.

ROM Instability Problem



Stability can be a real problem for compressible flow ROMs!

- A compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but *unstable* for other choices of basis size (Bui-Tanh *et al.* 2007).
- Some* remedies:

	Continuous Projection	Discrete Projection
Change projection (<i>a priori</i>)	Energy inner products (Rowley <i>et al.,</i> Serre <i>et al.,</i> <u>IKT</u> <i>et al.</i>)	Energy inner products (Rowley, <i>et al</i> .), Petrov- Galerkin Projection (Carlberg <i>et al</i> .)
Change ROM equations (<i>a posteriori</i>)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang,)	Eigenvalue reassignment (<u>IKT</u> et al.)
Change ROM basis (<i>a posteriori</i>)	Basis rotation (Balajewicz, <u>IKT</u> , <i>et al</i> .)	Optimization-based right basis modification (Amsallem <i>et al.</i>)

ROM Instability Problem



Stability can be a real problem for compressible flow ROMs!

- A compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but *unstable* for other choices of basis size (Bui-Tanh *et al.* 2007).
- Some* remedies:

	Continuous Projection	Discrete Projection
Change projection (<i>a priori</i>)	Energy inner products (Rowley <i>et al.,</i> Serre <i>et al.,</i> <u>IKT</u> <i>et al</i> .)	Energy inner products (Rowley, <i>et al</i> .), Petrov- Galerkin Projection (Carlberg <i>et al.</i>)
Change ROM equations (<i>a posteriori</i>)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang,)	Eigenvalue reassignment (<u>IKT</u> et al.)
Change ROM basis (<i>a posteriori</i>)	Basis rotation (Balajewicz, <u>IKT</u> , <i>et al</i> .)	Optimization-based right basis modification (Amsallem <i>et al.</i>)

- Yellow box: Wang, Akhtar, Borggaard, Iliescu.
- Blue box: Balajewicz, <u>Tezaur</u>, Dowell.

ROM Instability Problem



Stability can be a real problem for compressible flow ROMs!

- A compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but *unstable* for other choices of basis size (Bui-Tanh *et al.* 2007).
- Some* remedies:

	Continuous Projection	Discrete Projection
Change projection (<i>a priori</i>)	Energy inner products (Rowley <i>et al.,</i> Serre <i>et al.,</i> <u>IKT</u> <i>et al.</i>)	Energy inner products (Rowley, <i>et al.</i>), Petrov- Galerkin Projection (Carlberg <i>et al.</i>)
Change ROM equations (<i>a posteriori</i>)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang,)	Eigenvalue reassignment (<u>IKT</u> et al.)
Change ROM basis (<i>a posteriori</i>)	Basis rotation (Balajewicz, <u>IKT</u> , <i>et al</i> .)	Optimization-based right basis modification (Amsallem <i>et al.</i>)

- Yellow box: Wang, Akhtar, Borggaard, Iliescu.
- Blue box: Balajewicz, <u>Tezaur</u>, Dowell.



Mode Truncation Instability



- Projection-based MOR necessitates *truncation*.
- POD is, by definition and design, biased towards the *large, energy producing* scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a <u>data compression</u> point of view (i.e., small POD eigenvalues) but are crucial for the <u>dynamical</u> <u>equations</u>.
- For fluid flow applications, higher-order modes are associated with energy <u>dissipation</u>

 \Rightarrow low-dimensional ROMs (Galerkin <u>and</u> Petrov-Galerkin) can be <u>inaccurate</u> and <u>unstable</u>.



Proposed new approach*: basis rotation

Instead of modeling truncation via additional linear term, model the truncation <u>a priori</u> by "rotating" the projection subspace into a more dissipative regime

*M. Balajewicz, E. Dowell. Stabilization of projection-based reduced order models of the Navier-Stokes equation. *Nonlinear Dynamics* **70** (2), 1619-1632 (2012).

Proposed new approach*: basis rotation

Instead of modeling truncation via additional linear term, model the truncation <u>a priori</u> by "rotating" the projection subspace into a more dissipative regime

Illustrative example

- <u>Standard approach</u>: retain only the most energetic POD modes, i.e., U_1 , U_2 , U_3 .
- <u>Proposed approach</u>: add some higher order basis modes to increase dissipation, i.e., $a_1U_1 + b_1U_6 + c_1U_8$, $a_2U_2 + b_2U_{11} + c_2U_{18}$, $a_3U_3 + b_3U_{21} + c_3U_{28}$

Proposed new approach*: basis rotation

Instead of modeling truncation via additional linear term, model the truncation <u>a priori</u> by "rotating" the projection subspace into a more dissipative regime

Illustrative example

- <u>Standard approach</u>: retain only the most energetic POD modes, i.e., U₁, U₂, U₃.
- <u>Proposed approach</u>: add some higher order basis modes to increase dissipation, i.e., $a_1U_1 + b_1U_6 + c_1U_8$, $a_2U_2 + b_2U_{11} + c_2U_{18}$, $a_3U_3 + b_3U_{21} + c_3U_{28}$
- <u>More generally</u>: approximate the solution using a linear superposition of n + p (with p > 0) most energetic modes:

$$\widetilde{\boldsymbol{U}}_{i} = \sum_{j=1}^{n+p} X_{ij} \, \boldsymbol{U}_{j}, \quad i = 1, \dots, n,$$
(3)

where $X \in \mathbb{R}^{(n+p) \times n}$ is an orthonormal ($X^T X = I_{n \times n}$) "rotation" matrix.

*M. Balajewicz, E. Dowell. Stabilization of projection-based reduced order models of the Navier-Stokes equation. *Nonlinear Dynamics* **70** (2), 1619-1632 (2012).



Find X such that:

Find **X** such that:

1. New modes $\widetilde{\phi}$ remain <u>good approximations</u> of the flow

 \rightarrow minimize the "rotation" angle, i.e., minimize $\|X - I_{(M+P),M}\|_F$





Find **X** such that:

- 1. New modes $\tilde{\phi}$ remain <u>good approximations</u> of the flow \rightarrow minimize the "rotation" angle, i.e., minimize $\|X - I_{(M+P),M}\|_F$
- New modes produce <u>stable</u> and <u>accurate</u> ROMs.
 → ensure appropriate balance between energy production and energy dissipation.



Find **X** such that:

- 1. New modes $\tilde{\phi}$ remain <u>good approximations</u> of the flow \rightarrow minimize the "rotation" angle, i.e., minimize $\|X - I_{(M+P),M}\|_F$
- New modes produce <u>stable</u> and <u>accurate</u> ROMs.
 → ensure appropriate balance between energy production and energy dissipation.

• Once X is found, the result is a system of the form (3) with:

$$Q^{(i)}_{jk} \leftarrow \sum_{s,q,r=1}^{M+P} X_{si} Q^{(s)}_{qr} X_{qr} X_{rk}, \quad \boldsymbol{L} \leftarrow \boldsymbol{X}^T \boldsymbol{L} \boldsymbol{X}, \quad \boldsymbol{C} \leftarrow \boldsymbol{X}^T \boldsymbol{C}^*$$

Minimal subspace rotation



(5)

• Trace minimization problem on the Stiefel manifold:

minimize_{$$X \in \mathcal{V}_{(M+P),M}$$} - tr $(X^T I_{(M+P) \times M})$
subject to tr $(X^T L X) = \eta$

- $\mathcal{V}_{(M+P),M} \in \{ X \in \mathbb{R}^{(M+P) \times M} : X^T X = I_M, P > 0 \}$ is the <u>Stiefel manifold</u>.
- Constraint is traditional <u>linear eddy-viscosity closure model ansatz</u> \rightarrow involves overall balance between <u>linear energy production</u> and <u>dissipation</u> / vanishing of <u>averaged total power</u> (= tr($X^T L X$) + energy transfer).
 - $\eta \in \mathbb{R}$: proxy for the balance between linear energy production and energy dissipation (calculated iteratively using modal energy).
- Equation (5) is solved efficiently offline using the method of Lagrange multipliers (Manopt MATLAB toolbox).
- See (Balajewicz, <u>IKT</u>, Dowell, 2016) and Appendix slide for Algorithm.

ndia tional poratories

- <u>Advantages of proposed approach</u>:
 - Retains <u>consistency</u> between ROM and Navier-Stokes equations → no additional turbulence terms required.

- <u>Advantages of proposed approach</u>:
 - Retains <u>consistency</u> between ROM and Navier-Stokes equations → no additional turbulence terms required.
 - Inherently a <u>nonlinear</u> model → should be expected to outperform linear models.

- <u>Advantages of proposed approach</u>:
 - Retains <u>consistency</u> between ROM and Navier-Stokes equations → no additional turbulence terms required.
 - 2. Inherently a <u>nonlinear</u> model \rightarrow should be expected to outperform linear models.
 - 3. Works with *any* basis and Petrov-Galerkin projection.

- <u>Advantages of proposed approach</u>:
 - Retains <u>consistency</u> between ROM and Navier-Stokes equations → no additional turbulence terms required.
 - Inherently a <u>nonlinear</u> model → should be expected to outperform linear models.
 - 3. Works with *any* basis and Petrov-Galerkin projection.
- <u>Disadvantages of proposed approach</u>:
 - 1. Off-line calibration of free parameter η is required.

- <u>Advantages of proposed approach</u>:
 - Retains <u>consistency</u> between ROM and Navier-Stokes equations → no additional turbulence terms required.
 - Inherently a <u>nonlinear</u> model → should be expected to outperform linear models.
 - 3. Works with *any* basis and Petrov-Galerkin projection.
- <u>Disadvantages of proposed approach</u>:
 - 1. Off-line calibration of free parameter η is required.
 - 2. Stability cannot be proven like for incompressible case.

Numerical results: low Re number cavity



Flow over square cavity at Mach 0.6, Re = 1453.9, Pr = $0.72 \Rightarrow$ M = 4 ROM (91% snapshot energy).



• Above: domain and mesh for viscous channel driven cavity problem.



- Figure (a) shows evolution of modal energy. Standard ROM is unstable.
- Figure (b) shows phase plot of first and second temporal basis $a_1(t)$ and $a_2(t)$. Stabilized ROM computes stable limit cycle; standard ROM computes unstable spiral.
- Figure (c) is an illustration of the stabilizing rotation matrix. Rotation is small: $\frac{\|X - I_{(M+P),M}\|_F}{M} = 0.188, X \approx I_{(M+P),M}$

Numerical results: low Re number cavity

Sandia

National



• Pressure power spectral density (PSD) at location x = (2, -1).

Numerical results: moderate Re number cavity

Flow over square cavity at Mach 0.6, Re = 5452.1, Pr = 0.72 $\Rightarrow M = 20$ ROM (71.8% snapshot energy).



• Above: domain and mesh for viscous channel driven cavity problem.

Numerical results: moderate Re number cavity



Sandia

- Figure (a) shows evolution of modal energy. Stabilized ROM energy closer to FOM.
- Figure (b) illustrates stabilizing rotation matrix. Rotation is small: $\frac{\|X I_{(M+P),M}\|_F}{M} = 0.038, X \approx I_{(M+P),M}$

Numerical results: moderate Re number avity



stabilized ROM (M=P=20)

- DNS

• Figures show pressure cross PSD of of $p(x_1, t)$ and $p(x_2, t)$ where $x_1 = (2, -0.5)$, $x_2 = (0, -0.5)$. Left: power; right: phase lag.

Power and phase lag at fundamental frequency, and first two super harmonics are predicted accurately using the fine-tuned ROM (Δ = stabilized ROM, \Box = DNS)

Future work (basis rotation)



- Application to *higher Reynolds number* problems.
- Extension of the proposed approach to problems with <u>generic nonlinearities</u>, where the ROM involves some form of hyper-reduction (e.g., DEIM, gappy POD).
- Extension of the method to *minimal-residual-based* nonlinear ROMs.
- Extension of the method to <u>predictive applications</u>, e.g., problems with varying Reynolds number and/or Mach number.
- Selecting different <u>goal-oriented</u> objectives and constraints in our optimization problem:

minimize_{$X \in \mathcal{V}_{(M+P),M}$} f(X)subject to g(X, L) = 0

e.g.,

• Maximize parametric robustness:

$$f = \sum_{i=1}^k \beta_i \| \boldsymbol{U}^*(\boldsymbol{\mu}_i) \boldsymbol{X} - \boldsymbol{U}^*(\boldsymbol{\mu}_i) \|_F.$$

• ODE constraints: $g = ||\boldsymbol{a}(t) - \boldsymbol{a}^*(t)||$.

Appendix: Continuous vs. discrete Galerkin projection

Continuous Projection

Discrete Projection



* Continuous function space defined using e.g., finite elements.

Appendix: Section 3.3.4 : DS*-POD-ROM

- Original DS models are in LES, where it is considered *state-of-the-art*.
- Derivation requires precise definition of *filtering operation*, unlike other models, which were phenomenological.
 - *LES:* filtering operation effected by convolving flow variables with a rapidly-decaying spatial filter.
 - **POD:** filtering operation is effected by using the POD Galerkin projection (Sec. 3.1).
- Apply *filtering* operation (14) to:

$$\frac{\partial \mathbf{u}_r}{\partial t} - Re^{-1}\Delta \mathbf{u}_r + \nabla \cdot (\mathbf{u}_r \mathbf{u}_r) + \nabla p = 0.$$
(49)

((49) is equivalent to momentum equation (1) since $\nabla \cdot \boldsymbol{u}_r = 0$), one obtains:

$$\frac{\partial \bar{\mathbf{u}}_r}{\partial t} - Re^{-1}\Delta \bar{\mathbf{u}}_r + \nabla \cdot (\overline{\mathbf{u}_r \mathbf{u}_r}) + \nabla \bar{p} = 0.$$
(50)

(assuming differentiation and POD filtering commute).

- Remarks:
 - If filtering and differentiation do not commute, one has to estimate commutation error [67-69].
 - Since POD filtering is Galerkin projection (14), we have that $\overline{u}_r = u_r$.