



LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Reduced order modelling for dynamic simulations: LDRD feasibility study final report

K. K. Chand, W. D. Henshaw, T. K. Vassilevska

November 28, 2012

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Reduced order modelling for dynamic simulations: LDRD Feasibility Study final report

Kyle K. Chand, William D. Henshaw and Tanya Kostova Vassilevska

November 26, 2012

Contents

1	Introduction	2
2	An overview of dynamic ROMs	2
3	Adaptive ROM methods	3
4	LDRD results	3
4.1	Time-composite ROM	3
4.2	Reduced Order Models on Overlapping Grids	4
4.2.1	Choice of the inner product on an overlapping grid	4
4.2.2	Reduced order models for an advection-diffusion equation on an overlapping grid . . .	5
4.2.3	Cgad: advection-diffusion of a pulse past a cylinder	6

Executive Summary

This feasibility study (12-FS-007) investigated the current state-of-the-art for the construction of reduced order models (ROMs) from dynamic PDE-based simulations. One novel result of was the demonstration of a time-partitioned ROM for the advection-diffusion equation that achieved high accuracy with a small ROM size. The time-domain of a model PDE was partitioned and a fixed-size ROM was tuned to each sub-domain resulting in an effective method for achieving an efficient and accurate ROM for a dynamic problem. This LDRD also investigated the construction ROMs for simulations that utilize overlapping grids. In this case, it was found that weighting the discrete inner product in a manner that approximates the continuous inner product yields a significantly more accurate ROM. This project demonstrated the feasibility of applying reduced order models to complex PDE-based simulation tools and explored the benefits of adaptive ROM techniques. The results of the work led to an Exploratory Research LDRD, funded for FY13, that will focus on the development of composite adaptive reduced order models suitable for use with a number of LLNL applications.

1 Introduction

Reduced-order models (ROMs) approximate the essential features of complex, large-scale, mathematical models of physical systems. ROMs are designed to be orders of magnitude more efficient (in CPU and memory) than their corresponding high-fidelity models. For example, a single high-fidelity Large Eddy Simulation of a wind turbine may contain billions of unknowns and take days on a high-performance computer while a reduced order model of the same problem would consist of only hundreds or thousands of unknowns and execute in a few seconds on a desktop (or handheld) machine. This efficiency makes ROM-based *fast running adaptive codes* (FRACs) invaluable in applications that require large numbers of simulations such as design optimization, parameter space sampling or rapid evaluation of an engineering model (e.g. optimal control).

Currently, the most popular, and computationally effective, reduced order models project a high-fidelity mathematical model into a lower dimensional subspace to produce a dynamical ODE system with many fewer degrees of freedom: this smaller system forms the *fast running code* (see, for example [1, 2, 3]). Such model order reduction techniques have been successfully applied to a number of specific areas; however, their practical application to high-fidelity, nonlinear partial-differential (PDE) and differential-algebraic (DAE) equation based simulations has been limited by a number of open research problems. Of particular importance to LLNL applications is the adaptation of the reduced model to efficiently capture highly localized spatially and temporally evolving physics with many variable parameters. Current ROMs cannot efficiently represent local features in highly complex and dynamic simulations of physical phenomenon without enlarging the entire reduced model [4, 5, 6, 2, 7, 8]. Some approaches provide a degree of dynamic adaptation by interrupting the ROM’s execution in order to reconstruct the subspace using expensive full-fidelity simulations [9, 10, 11]. While in some specific cases this may be acceptable, in general it defeats the purpose of having a *fast running code*.

2 An overview of dynamic ROMs

The construction of reduced order models generally consists of two steps: the distillation of large-scale, high-fidelity data into a lower dimensional subspace; and the projection (e.g. Galerkin) of the original governing equations into the chosen subspace. The first step usually progresses by the collection of a number of “snapshots” of the time-dependent and/or multi-parameter PDE (or DAE) solution at a number of times and parameter space locations. To be more specific, consider a high-fidelity PDE model given abstractly by

$$\partial_t u(t, x; p) = Lu(t, x; p), \quad x \in \Omega, \quad t \in [0, T], \quad (1)$$

which depends on time, t , space, x and a collection of parameters, $p \in \mathcal{P}$. The snapshots consist of solutions at some times $t_i \in \mathcal{T}$ and some parameter values $p_i \in \bar{\mathcal{P}}$ arranged into columns of the $M \times N$ snapshot matrix

$$A_{M \times N} = \left[u(t_1, x_1; p_1) \quad u(t_i, x_i; p_i) \quad \dots \quad u(t_N, x_N; p_N) \right].$$

Of course, the exact solution to the PDE is usually replaced by the approximate solution u_h resulting from the discretization of the continuous problem. This data is then used to construct a lower dimensional basis using techniques such as the proper orthogonal decomposition (POD, also known as principle component analysis or the Karhunen-Loéve decomposition) [1, 2, 3] or the reduced basis method (RBM) [1, 2, 5, 6]. In the POD approach, for instance, the N snapshots are reduced into a small subspace by computing the singular-value-decomposition (SVD) of $A_{M \times N} = U_{M \times M} \Sigma_{N \times N} V_{N \times N}^T$, and then taking the first $K \ll M$ columns of U corresponding to the K largest singular values of A . This approach is often described through the equivalent (and typically smaller) eigenvalue-eigenvector problem for the $N \times N$ covariance matrix $A^T A$. For complex problems whose solutions vary greatly in space and time, K may still need to be large in order to accurately represent the dynamics, particularly for complex physical systems, long time simulations and large problem domains, as well as for varying values of the model parameters.

Once the projection subspace has been selected, the original PDE or ODE model is projected into this lower dimensional subspace; this projected dynamical system then forms the actual reduced order model, or

“fast running code”. For example, let the semi-discrete approximation to our continuous problem be

$$\partial_t u_h(p) = L_h u_h(p) \quad (2)$$

where we have suppressed the explicit dependence on time, space and parameters for clarity and L_h is the discrete approximation to the operator L . Given a POD or RBM basis of size K , constructed from snapshots of solutions to this discrete system, we can perform a Galerkin (for example) projection to form a smaller ODE system of K unknowns, $v \in \mathbb{R}^K$,

$$\frac{dv}{dt} = L_R(p)v + F_R(p, t). \quad (3)$$

Equation (2) may have many billions of unknowns ($M = 10^9$) and take days on a high-performance computer to solve while Equation (3) is only of dimension K which can be orders of magnitude smaller. The reduced order model approximation of u_h is then $u_R^K = Qv$ where $Q_{M \times K}$ is the rectangular matrix whose columns are the POD or RBM basis vectors. The matrix Q and the projected system are dependent upon the parameters $p_i \in \bar{P}$ and times $t_i \in \mathcal{T}$ used in choosing the snapshots.

Of course, the real purpose of a ROM is to efficiently provide the data necessary to compute an objective function in an optimization procedure, the system model in a control law or even rapid results to a real-time simulation. These situations boil down to using u_R^K instead of u_h (or even the exact solution u) in a functional that computes a diagnostic quantity of interest (QOI) such as drag in an aerodynamics problem or average room temperature in a building air flow simulation. In order for u_R^K to be useful it must accurately predict the QOI while keeping the cost of evaluation to a minimum.

3 Adaptive ROM methods

The current state-of-the-art “adapts” ROMs to localized phenomenon by either enriching the subspace (i.e. increasing M) [6, 12]; or completely recomputing the subspace using the original high-fidelity model [10, 11]. Both approaches retain the “global” nature of the subspace while incurring costs that reduce the efficiency of the technology. Pope’s ISAT (in-situ adaptive tabulation) procedure[9], developed for chemical kinetics applications, does provide some degree of locality, however it still requires the ability to summon the full-scale simulation when the local approximations are deemed insufficient. ISAT-like schemes have currently only been applied to stiff forcing terms of larger (unreduced) models [9, 13, 14].

We propose the development of locally (in space, time and parameter domains) adaptive reduced order models using a constructive approach in which the ROM solution is represented as a composite collection of localized model reductions. An example of a spatially adaptive ROM would be based on partitioning the computational domain (e.g. for the flow solver) into sub-domains and using different numbers of POD vectors in the different sub-domains. A sub-region where the solution is changing rapidly could use a locally defined ROM with more degrees of freedom compared to a sub-region where the solution does not change very much. Developing rigorous and efficient mechanisms for coupling spatially and temporally partitioned ROMs, using tools such as partitions of unity, will be a novel contribution of this LDRD.

Adaptive ROMs could also be utilized as “sub-scale” models within even larger high-fidelity models. For example, a wind farm simulation might consist of a spatial domain tens of kilometers in extent and hours in time while an individual wind turbine exhibits physics on the sub-meter and sub-second space and time-scales. A ROM based on the high-fidelity simulation of a single turbine could be used as a sub-scale model in wind park simulations consisting of 10s or 100s of wind turbines.

4 LDRD results

4.1 Time-composite ROM

This LDRD has demonstrated that even a simple partitioning of the time domain of a problem using tuned reduced order models for each time-partition can produce very accurate ROMs with small system sizes.

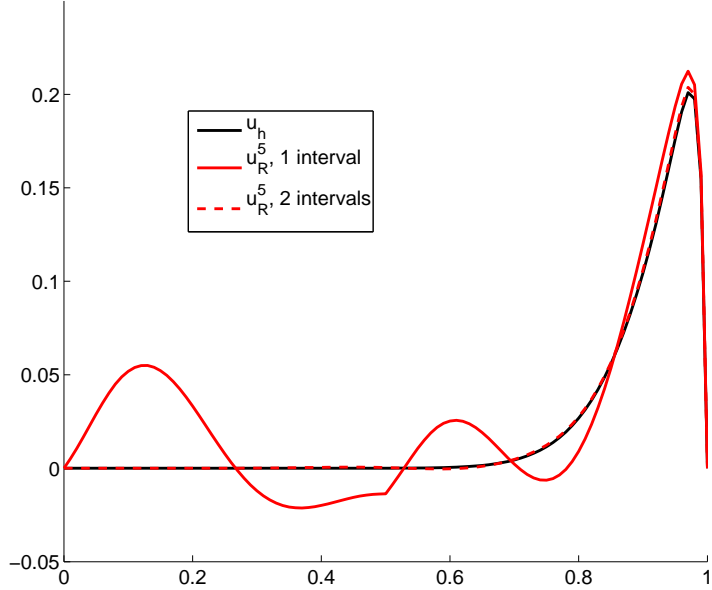


Figure 1: Even a static partitioning of the time domain produces more accurate ROMs

Consider a model problem consisting of the one dimensional advection diffusion equation:

$$u_t + au_x - \nu u_{xx} = 0 \text{ with } u(t, 0) = u(t, 1) = 0, x \in [0, 1], t \in [0, T] \quad (4)$$

$$u(t = 0, x) = \begin{cases} 1 & \text{for } 0 < x < 0.5 \\ 0 & \text{for } x = 0 \text{ and } x \geq 0.5 \end{cases} \quad (5)$$

The discrete “high-fidelity” approximation, u_h , to this PDE is obtained using second order central finite differences in space and a second order accurate predictor-corrector method in time. We a-priori break up the time domain, $[0, T]$, into S intervals and collect snapshots for each interval. A POD basis of order K is computed for each interval and is used to compute the ROM solution $u_R^{K,S}$ using a different projection and set of ROM coefficients, v^S , for each interval. Upon entering a new interval S , the change of basis from interval $S - 1$ to S is easy to accomplish using the projection matrices Q_{S-1} and Q_S

$$v^S = Q_S^T Q_{S-1} v^{S-1}. \quad (6)$$

The $K \times K$ matrix $Q_S^T Q_{S-1}$ can be computed before the ROM simulation begins and is only applied once at each transition between time partitions.

Figure 1 compares ROM solutions to u_h for the model problem for $K = 5$ with one (traditional OD) interval and two intervals. Even splitting the time domain into just two intervals improves the accuracy considerably, as shown in Table 1. Future work will include an analysis of these errors in an effort to determine efficient interval sizes and to make the process of selecting appropriate intervals dynamic and adaptive.

4.2 Reduced Order Models on Overlapping Grids

4.2.1 Choice of the inner product on an overlapping grid

The POD vectors ϕ_j computed from the SVD decomposition are orthogonal (orthonormal) with respect to the usual Euclidean inner product,

$$\phi_j^T \phi_j = \delta_{ij}. \quad (7)$$

One advantage of using this inner product is that the mass matrix M in (21) is diagonal.

intervals	K	$\mathcal{E} = u_R^K - u_h ^2$
1	5	5.50e-2
1	10	1.43e-4
2	5	7.97e-5
4	5	3.93e-6
4	3	1.45e-4
8	3	1.81e-4

Table 1: Multiple partitions can improve the answer, but only up to the resolving ability of the reduced POD basis.

We could also choose an area-weighted inner product (volume-weighted in 3D),

$$(\phi_i, \phi_j)_h = \sum_k w_k \phi_{j,k} \phi_{i,k} = \phi_i^T W \phi_j \approx \int_{\Omega} u_i(\mathbf{x}) u_j(\mathbf{x}) d\mathbf{x}. \quad (8)$$

where u_i and u_j are the continuous functions corresponding to ϕ_i and ϕ_j respectively. Here $w_k \geq 0$ are integration weights which are some approximation to the local element of area $d\mathbf{x}$. W is the diagonal matrix with entries w_k . On an overlapping grid, the weights w_k should be adjusted in regions where grids overlap to take into account the multi-valued nature of the discrete solution (i.e. we need to avoid double counting where two grids overlap). The mass matrix will no longer be diagonal using the area inner product (8)

Weighted snap-shot vectors: Instead of using the area-weighted inner product, one could instead consider weighting the snap-shot vectors $\{\mathbf{w}_n\}$ used to form the SVD. Suppose we weight the snap-shot vectors by a matrix D , $\tilde{\mathbf{w}}^n = D\mathbf{w}^n$. The SVD for the new snap-shot matrix $DA = [D\mathbf{w}^1 D\mathbf{w}^2 \dots]$ is

$$DA = \tilde{U} \tilde{\Sigma} \tilde{V}^T. \quad (9)$$

If the columns of \tilde{U} are denoted by $\tilde{\phi}_i$ then

$$\tilde{\phi}_i^T \tilde{\phi}_j = \delta_{ij}. \quad (10)$$

Note that the basis vectors we should use to represent our solution $\mathbf{u}(\mathbf{x}, t)$ are $\hat{\phi}_i = D^{-1} \tilde{\phi}_i$, since we need to scale back to the original representation. In this case we look for a ROM solution to our PDE of the form

$$\mathbf{u}^K(\mathbf{x}, t) = \sum_{i=1}^K q_i(t) D^{-1} \tilde{\phi}_i = \sum_{i=1}^K q_i(t) \hat{\phi}_i. \quad (11)$$

Note that

$$\tilde{\phi}_i^T \tilde{\phi}_j = \hat{\phi}_j^T D^T D \hat{\phi}_i = \delta_{ij}. \quad (12)$$

Therefore if we choose D to be the diagonal matrix $D = W^{1/2}$ then $\hat{\phi}_i$ are orthogonal with respect to the area-weighted inner product,

$$(\hat{\phi}_i, \hat{\phi}_j)_h = \sum_k w_k \hat{\phi}_{j,k} \hat{\phi}_{i,k} = \hat{\phi}_j^T W \hat{\phi}_i = \delta_{ij}. \quad (13)$$

Thus the mass matrix will be diagonal w.r.t. the area-weighted inner product when we weight the snap-shot vectors using $D = W^{1/2}$.

4.2.2 Reduced order models for an advection-diffusion equation on an overlapping grid

Consider solving the advection diffusion equation on some domain Ω ,

$$T_t + (\mathbf{a} \cdot \nabla) T = \kappa \Delta T + f_b(\mathbf{x}, t), \quad \text{for } \mathbf{x} \in \Omega \quad (14)$$

$$T(\mathbf{x}, 0) = T_0(\mathbf{x}), \quad \text{for } \mathbf{x} \in \Omega, \quad (15)$$

$$\alpha T + \beta \partial_n T = g(\mathbf{x}, t) \quad \text{for } \mathbf{x} \in \partial\Omega, \quad (16)$$

with some body forcing $f_b(\mathbf{x}, t)$. We will solve these equations on an overlapping grid, $\mathcal{G} = \{G_g\}_{g=1}^{\mathcal{N}_g}$, consisting of \mathcal{N}_g component grids. On each component grid G_g we define a discrete approximation, $T_i^n \approx T(\mathbf{x}_i, t^n)$.

Given a reduced (POD) basis $\{\phi_i\}_{i=1}^K$, where $\phi_i \in \mathbb{R}^M$ are vectors representing T_i^n at all grid points on all grids (there being a total of M grid points).

We look for a reduced order solution of the form

$$T^K(\mathbf{x}, t) = \sum_m g_m(t) T_m^p(\mathbf{x}, t) + \sum_{j=1}^K q_j(t) \phi_j(\mathbf{x}). \quad (17)$$

for some unknowns $\{q_i(t)\}$, and where $T_m^p(\mathbf{x}, t)$ are *particular functions*, chosen, for example, to satisfy the boundary conditions.

Substituting (17) into the advection-diffusion equation (14) gives

$$T_t^K + (\mathbf{a} \cdot \nabla) T^K = \kappa \Delta T^K + f_b(\mathbf{x}, t), \quad (18)$$

or

$$\sum_{j=1}^K \frac{dq_j}{dt} \phi_j + \sum_{j=1}^K q_j (\mathbf{a} \cdot \nabla) \phi_j = \kappa \sum_{j=1}^K q_j \Delta \phi_j + F_b(\mathbf{x}, t), \quad (19)$$

$$F_b(\mathbf{x}, t) = f_b(\mathbf{x}, t) - \sum_m \left\{ \partial_t [g_m(t) T_m^p(\mathbf{x}, t)] + g_m(t) (\mathbf{a} \cdot \nabla) T_m^p(\mathbf{x}, t) + g_m(t) \Delta T_m^p(\mathbf{x}, t) \right\}. \quad (20)$$

These last equations implicitly define (an over-determined) system of ODEs for the unknowns $\{q_i(t)\}$. We can use a Galerkin projection to define a set of N ODE equations for the N unknowns $q_i(t)$.

Let $\langle \cdot, \cdot \rangle$ denote some inner product on \mathbb{R}^M . Different choices for the inner-product will be considered.

Taking the inner product of ϕ_i with (18) (i.e. we project the equations onto the space spanned by $\{\phi_i\}$) gives

$$\sum_{j=1}^K M_{ij} \frac{dq_j}{dt} + \sum_j B_{ij} q_j = \sum_{j=1}^K K_{ij} q_j(t) + f_i, \quad (21)$$

where

$$M_{ij} = \langle \phi_i, \phi_j \rangle, \quad B_{ij} = \langle \phi_i, (\mathbf{a} \cdot \nabla) \phi_j \rangle, \quad K_{ij} = \langle \phi_i, \nu \Delta \phi_j \rangle, \quad (22)$$

$$f_i = - \langle \phi_i, F_b \rangle \quad (23)$$

Equations 21 can be written as

$$M \frac{d\mathbf{q}}{dt} + B\mathbf{q} = K\mathbf{q} + \mathbf{f}. \quad (24)$$

4.2.3 Cgad: advection-diffusion of a pulse past a cylinder

In this example we solve an advection diffusion equation (Cgad) for a pulse moving past a cylinder. The boundary conditions are Neumann (check me).

We save 200 snapshots over the time interval $[0, 2]$ so that the snap shot time interval is $\Delta t_s = .01$

We consider two ways to form the projected equations: (1) A Galerkin projection using the standard inner product and (2) a Galerkin projection using the integration weights in the inner product.

Here are the first 21 singular values:

151.71704 108.46108 79.70319 57.30043 33.02741 19.41691 10.25015 9.04892
5.27542 2.52297 1.30634 1.00365 0.48398 0.21195 0.08702 0.03895 0.02111
0.01361 0.00904 0.00629 0.00445

Figure 4 shows results using the weighted inner-product (WIP) projection. Figure 5 shows results using the standard inner-product (SIP) projection. The projected initial conditions are accurate to about $4e-3$ and $2e-9$ for 10 and 20 POD modes, for either SIP or WIP. The WIP scheme has errors of about $1.7e-4$ (10 modes) and $9.6e-4$ (20 modes) at $t=1.5$. Question: is the accuracy of the 20 mode solution being limited by the frequency of snapshots?

The SIP scheme has errors about .1 (10 modes) and .12 (20 modes) at $t=1.5$.

The WIP scheme is clearly much better in this case - but WHY?. We note that when solving the same problem on a square Cartesian grid there is little difference between WIP and SIP so that the either the overlap or the curvilinear grid has made a difference.

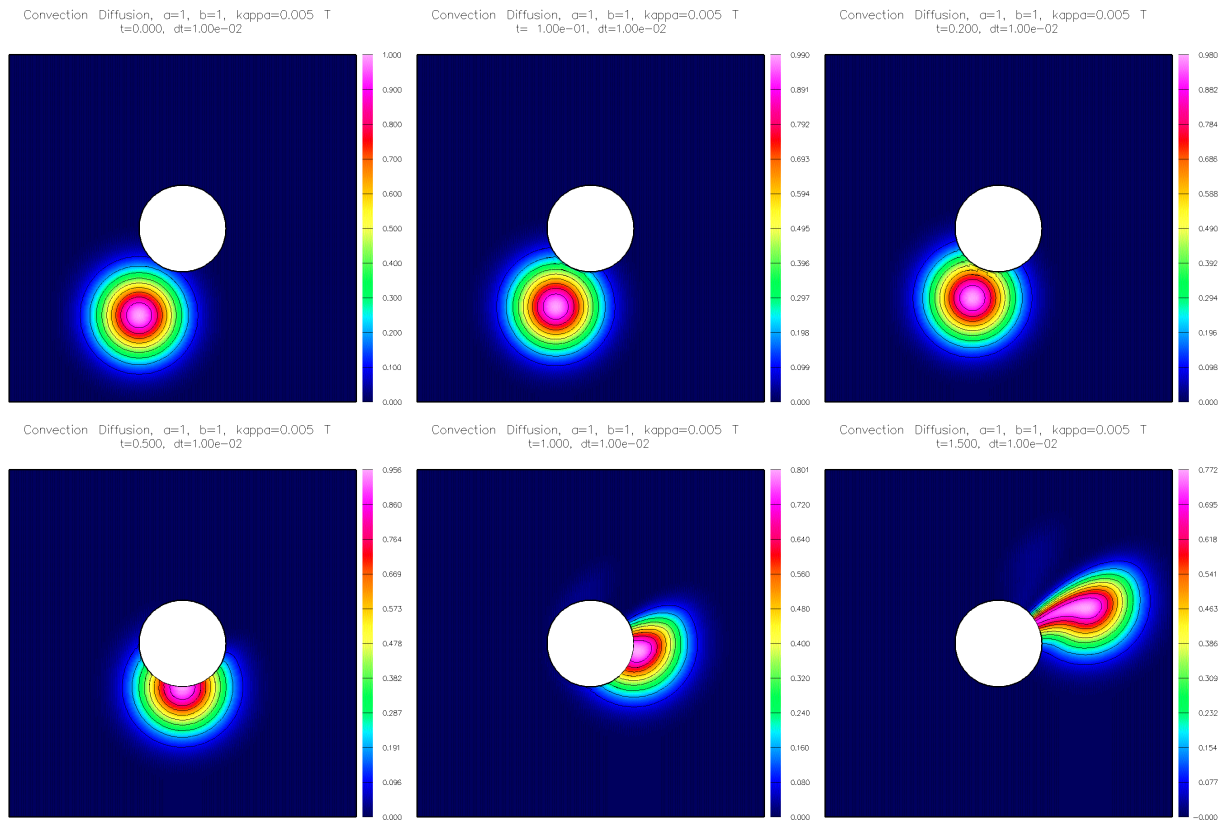


Figure 2: Advection diffusion of a pulse moving past a cylinder. Solution at various times.

References

- [1] A. Deane, I. Kevrekidis, G. Karniadakis, and S. Orszag. Low-dimensional models for complex geometry flows: applications to grooved channels and circular cylinders. *Phys. Fluids A*, 3:2337–2354, 1991.
- [2] John Burkardt, Max Gunzburger, and Hyung-Chun Lee. POD and CVT-based reduced-order modeling of navier-stokes flows. *Computer Methods in Applied Mechanics and Engineering*, 196:337–355, 2006.
- [3] Eli Shlizerman, Edwin Ding, Matthew O. Williams, and J. Nathan Kutz. The proper orthogonal decomposition for dimensionality reduction in mode-locked lasers and optical systems. *International Journal of Optics*, 2012, 2012. Artical ID 831604.
- [4] T. Braconnier, M. Ferrier, J.-C. Jouhaud, M. Montagnac, and P. Sagaut. Towards an adaptive pod/svd surrogate model for aeronautic design. *Computers & Fluids*, 40:195–209, 2011.

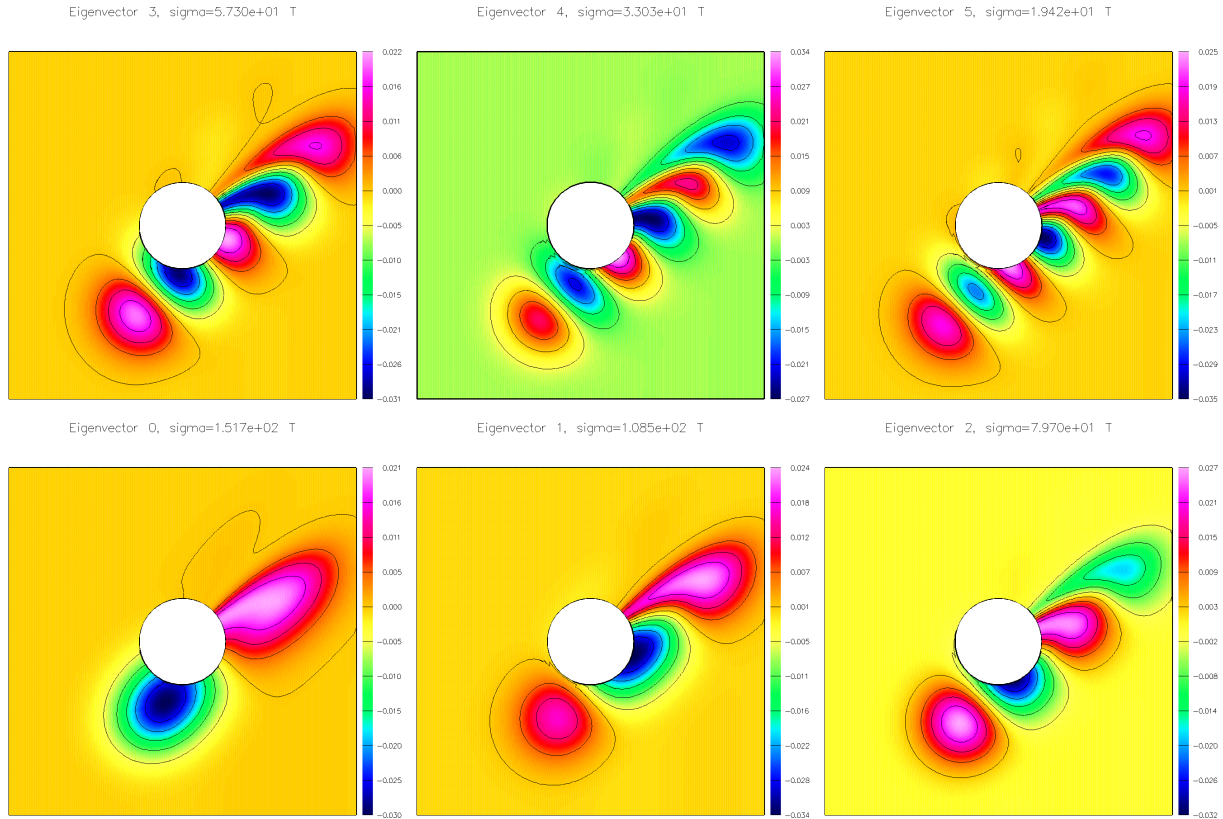


Figure 3: Advection diffusion of a pulse moving past a cylinder: first few POD vectors.

- [5] Simone Deparis and Gianluigi Rozza. Reduced basis method for multi-parameter-dependent steady navier-stokes equations: Applications to natural convection in a cavity. *Journal of Computational Physics*, 228:4359–4378, 2009.
- [6] M. Fares, J.S. Hesthaven, Y. Maday, and B. Stamm. The reduced basis method for the electric field integral equation. *Journal of Computational Physics*, 230:5532–5555, 2011.
- [7] Max D. Gunzburger, Janet S. Peterson, and John N. Shadid. Reduced-order modeling of time-dependent pdes with multiple parameters in the boundary data. *Computer Methods in Applied Mechanics and Engineering*, 196:1030–1047, 2007.
- [8] Bernard Haasdonk and Mario Ohlberger. Reduced basis method for finite volume approximations of parameterized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42:277–302, 2008.
- [9] S. B. Pope. Computationally efficient implementation of combustion chemistry using in situ adaptive tabulation. *Combustion Theory and Modelling*, 1:41–63, 1997.
- [10] E. Arian, M. Fahl, and E. W. Sachs. Trust-region proper orthogonal decomposition for flow control. Technical Report NASA/CR-2000-210124, ICASE, 2000.
- [11] Michel Bergmann, Laurent Cordier, and Jena-Pierre Brancher. *Active Flow Control*, volume 95, chapter Drag Minimization of the Cylinder Wake by Trust-Region Proper Orthogonal Decomposition, pages 309–324. Springer-Verlag, 2007.
- [12] Matthew B. Stephenson, Jin-Fa Lee, and Dan A. White. Automatic black-box model order reduction using radial basis functions. IEEE Antennas and Propagation Society, July 2011.

- [13] M. A. Singer and S. B. Pope. Exploiting ISAT to solve the reaction-diffusion equation. *Combustion Theory and Modelling*, 8:361–383, 2004.
- [14] Michael A. Singer and William H. Green. Using adaptive proper orthogonal decomposition to solve the reaction-diffusion equation. *Applied Numerical Mathematics*, 59:272–279, 2009.

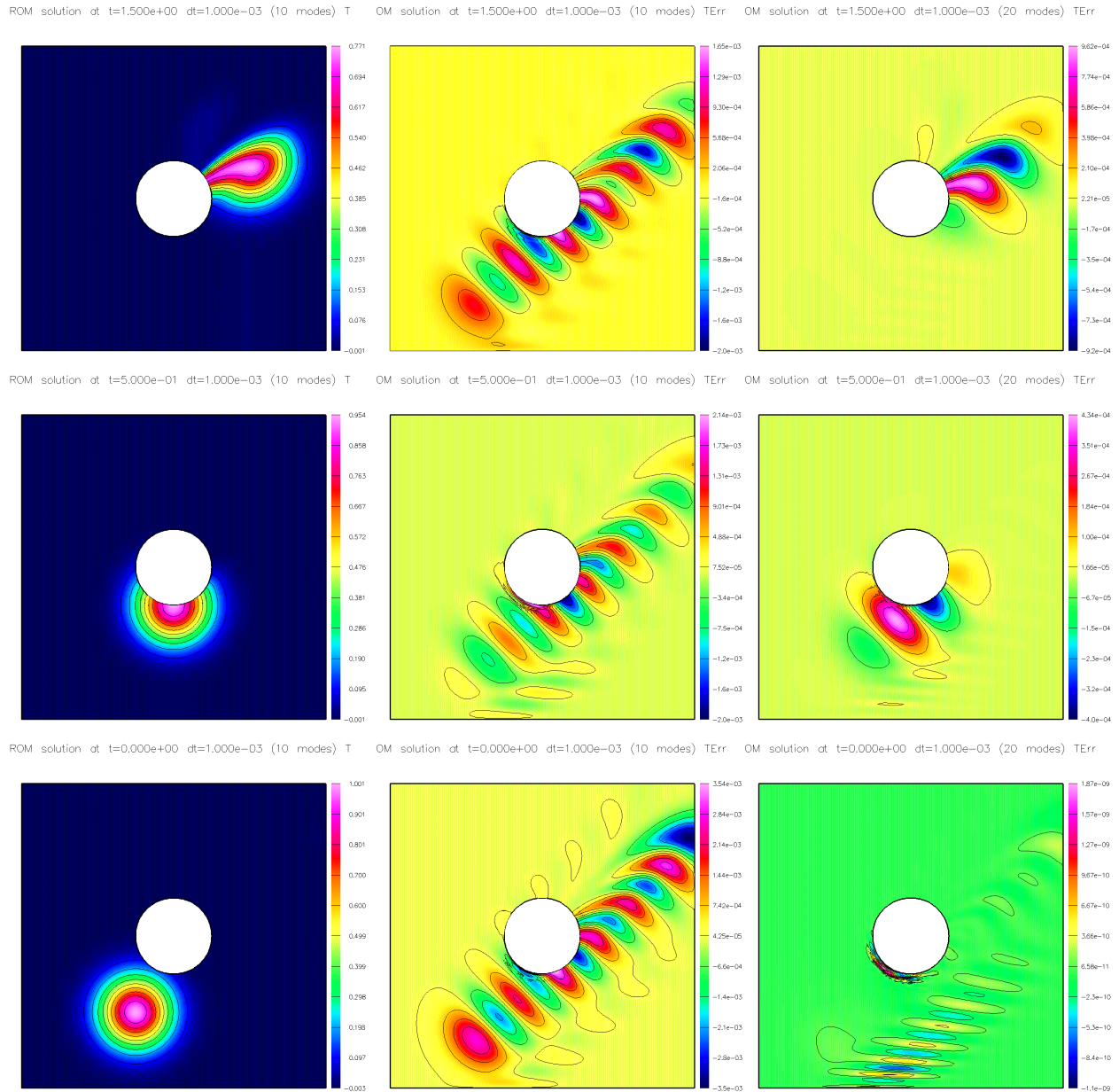


Figure 4: WIP: advection diffusion of a pulse moving past a cylinder: Galerkin projection WITH weighted inner product. Left: solution computed with 10 POD modes. Middle: error using 10 POD modes. Right: error using 20 POD modes.

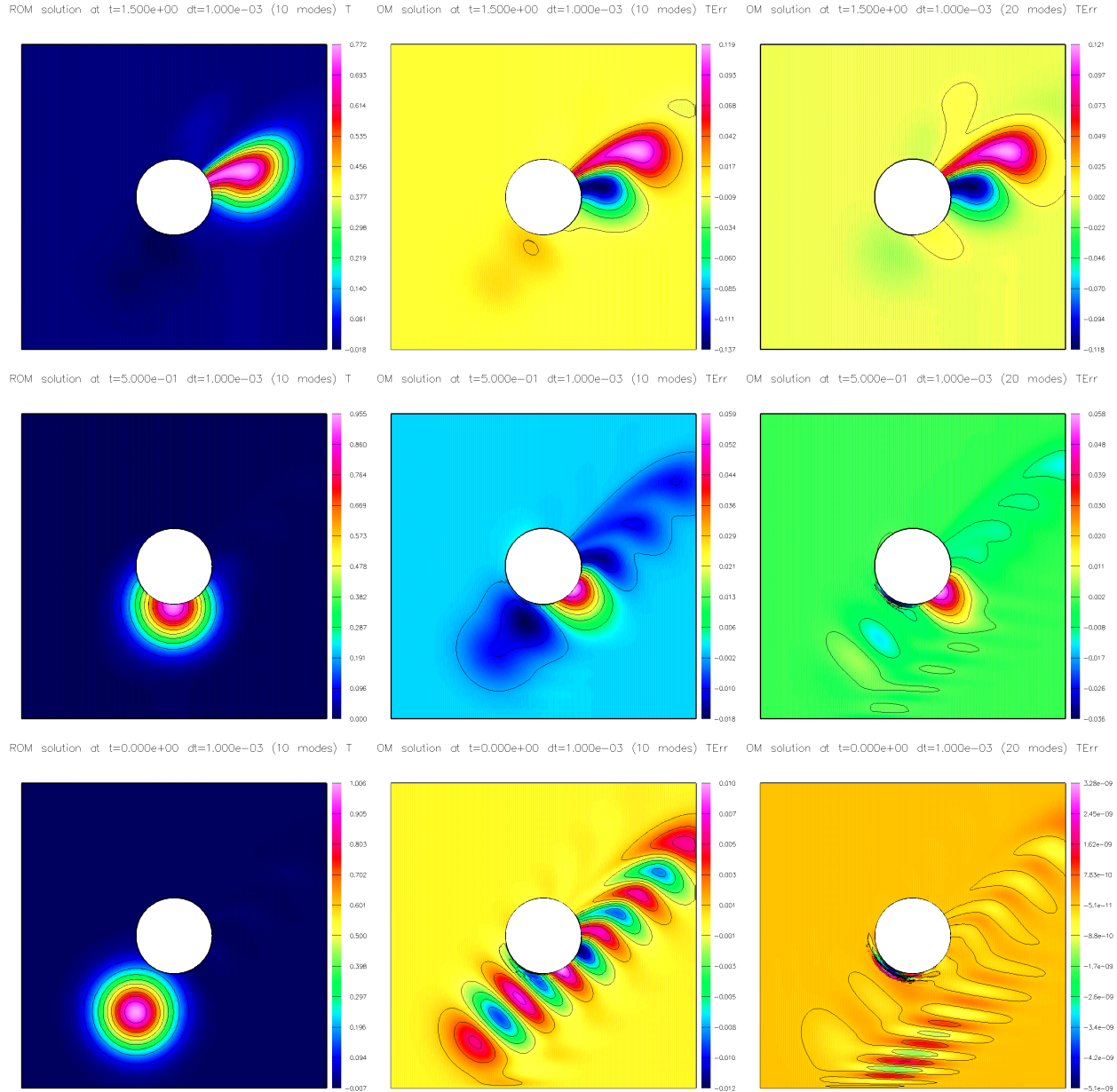


Figure 5: SIP: advection diffusion of a pulse moving past a cylinder: Galerkin projection with NO weights in the inner product. Left: solution computed with 10 POD modes. Middle: error using 10 POD modes. Right: error using 20 POD modes.