



LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Using adaptive proper orthogonal decomposition to solve the reaction-diffusion equation

M. A. Singer, W. H. Green

December 4, 2007

Applied Numerical Mathematics

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.

Using adaptive proper orthogonal decomposition to solve the reaction-diffusion equation

Michael A. Singer^{a,*}, William H. Green^b

^a *Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94550, USA*

^b *Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*

Abstract

We introduce an adaptive POD method to reduce the computational cost of reacting flow simulations. The scheme is coupled with an operator-splitting algorithm to solve the reaction-diffusion equation. For the reaction sub-steps, locally valid basis vectors, obtained via POD and the method of snapshots, are used to project the minor species mass fractions onto a reduced dimensional space thereby decreasing the number of equations that govern combustion chemistry. The method is applied to a one-dimensional laminar premixed CH₄-air flame using GRI_{mech} 3.0; with errors less than 0.25%, a speed-up factor of 3.5 is observed. The speed-up results from fewer source term evaluations required to compute the Jacobian matrices.

Key words: Model reduction; Premixed flame; Proper orthogonal decomposition; Strang splitting

1 Introduction

The simulation of chemically reacting flows with detailed chemistry is expensive computationally. When using kinetic mechanisms for fuels containing hundreds of species and thousands of reactions, large systems of stiff differential equations must be solved; obtaining accurate solutions to these equations often consumes the majority of CPU time. As a result, model reduction

* Corresponding author.

Email address: msinger@llnl.gov (Michael A. Singer).

strategies that reduce solution time yet maintain accuracy requirements are of interest to combustion modelers. Here we introduce a model reduction scheme based on proper orthogonal decomposition (POD).

2 Time-splitting algorithm

For simplicity, we consider an ideal gas mixture on the one-dimensional domain $x \in [0, 1]$. The mixture contains n_s chemical species and remains at a constant and uniform pressure. Extension to higher spatial dimensions and variable pressure flows follows the same construction. The time-dependent state of the mixture is specified by the thermochemical composition vector,

$$\boldsymbol{\phi}(x, t) = [Y_1, Y_2, Y_3, \dots, Y_{n_s}, T]^T,$$

where Y_i is the mass fraction of species i , T is temperature, and the superscript T denotes transpose. The composition evolves according to the model equation

$$\frac{\partial \boldsymbol{\phi}}{\partial t} = \mathbf{S}(\boldsymbol{\phi}) + \Gamma \frac{\partial^2 \boldsymbol{\phi}}{\partial x^2}, \quad (1)$$

where $\mathbf{S}(\boldsymbol{\phi})$ is the reaction source term; Γ is the constant and uniform molecular diffusivity and is the same for all species and temperature. Equation 1 provides a suitable test for our POD-based algorithm because it captures the essential physics for which our method is designed to treat efficiently: nonlinear coupled chemical reactions.

Given an initial composition $\boldsymbol{\phi}^n = \boldsymbol{\phi}(x, t^n)$, equation 1 is solved numerically using Strang splitting [8]:

$$\frac{\partial \boldsymbol{\phi}^{(1)}}{\partial t} = \mathbf{S}(\boldsymbol{\phi}^{(1)}), \quad \boldsymbol{\phi}^{(1)}(x, 0) = \boldsymbol{\phi}^n \quad \text{on} \quad [0, \Delta t/2] \quad (2)$$

$$\frac{\partial \boldsymbol{\phi}^{(2)}}{\partial t} = \Gamma \frac{\partial^2 \boldsymbol{\phi}^{(2)}}{\partial x^2}, \quad \boldsymbol{\phi}^{(2)}(x, 0) = \boldsymbol{\phi}^{(1)}(x, \Delta t/2) \quad \text{on} \quad [0, \Delta t], \quad (3)$$

$$\boldsymbol{\phi}^{(2)}(0, t) = \boldsymbol{\phi}^L(t), \quad \boldsymbol{\phi}^{(2)}(L, t) = \boldsymbol{\phi}^R(t) \quad (4)$$

$$\frac{\partial \boldsymbol{\phi}^{(3)}}{\partial t} = \mathbf{S}(\boldsymbol{\phi}^{(3)}), \quad \boldsymbol{\phi}^{(3)}(x, 0) = \boldsymbol{\phi}^{(2)}(x, \Delta t) \quad \text{on} \quad [0, \Delta t/2]. \quad (5)$$

The solution at $t^{n+1} = t^n + \Delta t$ is $\boldsymbol{\phi}^{n+1} = \boldsymbol{\phi}(x, t^{n+1}) = \boldsymbol{\phi}^{(3)}(x, \Delta t/2)$. In the present work, the diffusion sub-step (i.e., equation 3) is solved using a second-order Crank-Nicolson scheme. For combustion simulations with detailed chemistry, the most CPU-intensive steps are the chemistry sub-steps (equations 2 and 5), which require solving large systems of stiff ODEs. Frequently, these

equations are solved using ODE integrators that are specialized for stiff systems, e.g., DVODE [1]. Here we present an adaptive model reduction scheme, based on POD, that can be combined with an ODE integrator to reduce the CPU time required to solve the equations that govern combustion chemistry.

3 Proper orthogonal decomposition

Details of the POD method and the use of snapshots are described elsewhere [6,3,2]. In general, POD works as follows: The full system of equations is advanced in time to obtain m snapshots of the solution vector: $\phi_1, \phi_2, \dots, \phi_m$. Then, the correlation matrix \mathbf{R} is formed where $R_{ij} = \phi_i^T \phi_j$. The eigenvalues (λ_i) and corresponding eigenvectors (α^i) of \mathbf{R} are computed, and the POD basis vectors are linear combinations of the snapshots,

$$\mathbf{a}_i = \sum_{j=1}^m \alpha_j^i \phi_j. \quad (6)$$

The most energetic basis vectors (i.e., those corresponding to the largest eigenvalues) form the reduced basis onto which the governing equations are projected. If \mathbf{A}_p contains the first p eigenvectors (those with the p largest eigenvalues), then $\mathbf{Q}_p = \mathbf{A}_p \mathbf{A}_p^T$ is the optimal projection in the least squares sense. In addition, the fraction of “energy” captured by \mathbf{A}_p is $\sum_{i=1}^p \lambda_i / \sum_{i=1}^m \lambda_i$.

4 Adaptive POD algorithm

The objective of the present work is to use POD to solve equation 1 efficiently and accurately. To this end, we apply POD to the reaction sub-steps at a fixed spatial location, i.e.,

$$\frac{d\phi}{dt} = \mathbf{S}(\phi(x, t)), \quad (7)$$

with appropriate initial conditions. The transport sub-step is unchanged because it consumes an order of magnitude less CPU time than the chemistry sub-steps, but adaptive POD could also be applied to speed transport computations. Equation 7 represents a systems of $n_s + 1$ coupled ODEs that must be solved at all spatial locations for each time-step.

The fundamental idea is to project the n_m minor species in $\phi(x, t)$, denoted $\phi^{\min}(x, t)$, onto a n_b dimensional subspace

$$\boldsymbol{\phi}^r(x, t) = \mathbf{A}\boldsymbol{\phi}^{\text{min}}(x, t), \quad (8)$$

where \mathbf{A} is a $n_b \times n_m$ orthogonal projection matrix, and $\boldsymbol{\phi}^r(x, t)$ is a n_b dimensional vector of pseudo-species. The matrix \mathbf{A} , which captures the dynamics of the minor species as they undergo chemical reactions, is constructed using POD and snapshots from the reaction sub-steps. The calculation of \mathbf{A} occurs as a preprocessing step during which equation 1 is solved with the full chemical mechanism; the snapshots used to construct \mathbf{A} are obtained as the ODE integrator solves the chemistry sub-steps. This preprocessing step takes negligible time to compute compared to the full calculation. Due to the wide range of mass fractions in $\boldsymbol{\phi}(x, t)$, only the minor species are included in the projections. By excluding the major species and temperature, \mathbf{A} captures the dynamics of a lower dimensional space in which the range of numerical quantities is smaller.

Therefore, denoting the $n_j = n_s - n_m + 1$ dimensional vector of major species and temperature by $\boldsymbol{\phi}^{\text{maj}}(x, t)$, equation 7 becomes

$$\frac{d}{dt} \begin{bmatrix} \boldsymbol{\phi}^r \\ \boldsymbol{\phi}^{\text{maj}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{S} \left(\begin{bmatrix} \mathbf{A}^T \boldsymbol{\phi}^r & \boldsymbol{\phi}^{\text{maj}} \end{bmatrix} \right), \quad (9)$$

where \mathbf{I} is the $n_j \times n_j$ identity matrix. Equation 9 is a system of $n_b + n_j$ equations that is solved implicitly using DVODE [1]. When $n_b \ll n_m$, few POD basis vectors are required to capture the evolution of the minor species, thereby reducing the number of governing equations appreciably.

Because reaction activity can vary significantly throughout a computational domain (e.g., reactants, products, flame), specialized POD projections are constructed from snapshots in different regions of the domain. During a computation, selection of a projection matrix is based on locating one that includes the temperature of interest; hence, the method is called adaptive POD (aPOD).

5 Results

To test the algorithm, we use equation 1 to model a freely propagating stoichiometric methane-air flame with $\Gamma = 0.77 \text{ cm}^2/\text{sec}$. The combustion chemistry is modeled using GRIMech 3.0 (53 species and 325 reactions) [7]. This model problem provides a flame front and propagation mechanism (i.e., reaction-diffusion) that is characteristic of premixed flames, but the quantitative features of the flame (e.g., thickness, flame speed) are different than a real premixed methane-air flame. Nonetheless, the model problem serves to demon-

strate the capabilities of the aPOD algorithm in the context of premixed flames.

The computational domain contains 512 grid points and $\Delta t \approx 7.63 \times 10^{-6}$ sec (based on convergence studies). POD snapshots are obtained from 30 time-steps using full chemistry. The domain is then partitioned into 6 sub-domains (figure 1), and POD basis vectors are computed from an eigenvalue analysis. A total of 5 projection matrices are computed, one for each sub-domain where POD is used. For simplicity, and to demonstrate the feasibility of aPOD, sub-domain boundaries for the current work are chosen based on temperature gradients. In a more general context (e.g., three dimensional unsteady flows), a user could create a library of projection matrices that project the state (or a portion thereof) onto a reduced dimensional space. The region of phase space over which the projection is valid would be determined by the region of phase space covered by the snapshots used to construct the projection. Then, during a simulation, a projection matrix from the library could be used if the current state (at a particular spatial location) falls within the valid range of a projection in the library. If no such projection exists, the full model could be used to ensure accuracy.

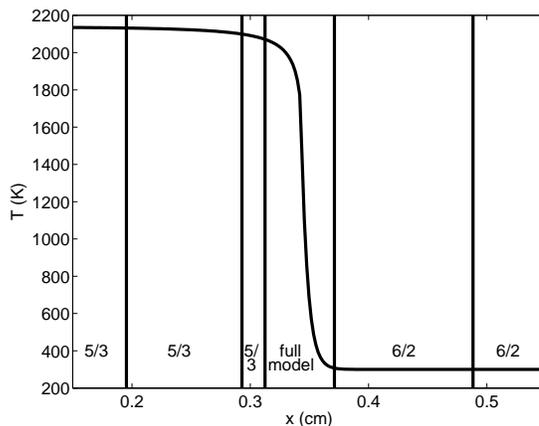


Fig. 1. Partitioned domain near the flame front. The numbers of major species and POD basis vectors, for each sub-domain, are shown.

In the present work, each projection matrix captures 99.99% of the “energy” in its sub-domain, yet only requires 2–3 basis vectors and excludes 5–6 major species (figure 1); hence, using aPOD reduces the number of equations that are solved from ≈ 50 to ≈ 10 . Figure 1 illustrates a portion of the partitioned domain and the number of POD basis vectors and major species (i.e., O_2 , CO_2 , H_2O , CO , N_2 , CH_4) in each sub-domain. Note that all major species are included in the reactant sub-domains, while CH_4 is absent throughout the product sub-domain. More basis vectors are needed in the product sub-domains to capture the chemical dynamics of active chemical processes in the product zones [5].

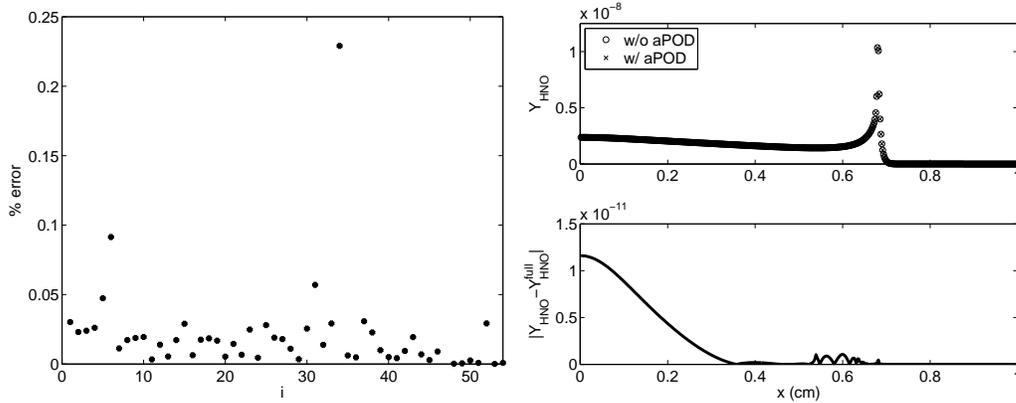


Fig. 2. Solution error due to aPOD: (a) error for each species ($i = 1, \dots, 53$) and temperature ($i = 54$); (b) comparison of HNO mass fractions with aPOD (Y_{HNO}) and without aPOD ($Y_{\text{HNO}}^{\text{full}}$).

The aPOD scheme exhibits second-order temporal convergence (not shown) and small errors relative to full chemistry solutions. After propagating the flame ≈ 3000 time-steps, errors in $\phi(x, t)$ (relative to a full chemistry solution) are less than 0.25% (figure 2a); major species and temperature errors are non-zero due to transport coupling. The species with the largest error, HNO, has an error of 0.23% (figure 2b).

Figure 3 demonstrates the performance of the ODE solver at a representative instant in time. From figure 3a, the number of DVODE iterations for both schemes is nearly identical both near and afar the flame; the former region requires the greatest number of iterations due to the small sub-steps required to maintain the specified error tolerances. As a result, the number of Jacobian evaluations per iteration is nearly identical with and without aPOD. The second reaction sub-step (corresponding to equations 5) requires more iterations due to the proceeding transport sub-step.

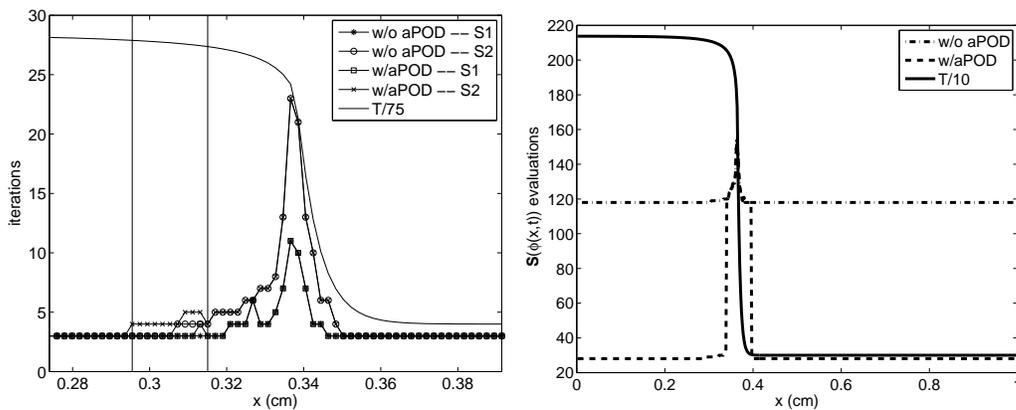


Fig. 3. Performance of the ODE solver: (a) number of iterations for the first (S1) and second (S2) reaction sub-steps in a portion of the computational domain near the flame front; (b) number of $\mathbf{S}(\phi(x, t))$ evaluations.

Figure 3b shows the number of source term evaluations per reaction time-step (i.e., $\Delta t/2$). As seen, aPOD requires fewer $\mathbf{S}(\phi(x, t))$ evaluations away from the flame front due to the smaller Jacobians (computed via finite differences) that result from the smaller systems of equations. The CPU time per $\mathbf{S}(\phi(x, t))$ evaluation (not shown) is nearly identical with and without aPOD; hence, the CPU time per iteration is smaller with aPOD than without, and the computational expense of the matrix-vector multiplications required by aPOD to evaluate the nonlinear source term (equation 9) is negligible compared to the overall time required to evaluate $\mathbf{S}(\phi(x, t))$. In addition, the smaller Jacobians require less memory. In all, aPOD reduces the number of $\mathbf{S}(\phi(x, t))$ evaluations by a factor of 3.5, and the overall speed-up factor due to aPOD is also 3.5.

6 Conclusions

Adaptive POD (aPOD) can be used to reduce the computational cost of reacting flow simulations. Here, aPOD is coupled with an operator-splitting algorithm to solve the reaction-diffusion equation. For the reaction sub-steps, locally valid projection matrices containing 2–3 basis vectors are obtained via POD and the method of snapshots; these are used to project the minor species mass fractions onto a reduced dimensional space thereby decreasing the number of equations that govern combustion chemistry. The method is applied to a one-dimensional laminar premixed CH_4 -air flame using GRImech 3.0. With errors less than 0.25%, a speed-up factor of 3.5 is observed. The speed-up results from fewer source term evaluations required to compute the Jacobian matrices. Application of aPOD to multi-dimensional problems involving larger kinetic mechanisms may result in further speed-ups. The aPOD method may also be extended to include projection matrices that cover a user-specified region of phase space. Additional speed-up may also result from coupling aPOD with other methods of model reduction (e.g., adaptive chemistry [4]).

Acknowledgements

MAS acknowledges helpful discussions with Professor Karen Willcox. This work was supported by the U.S. Department of Energy Office of Basic Energy Sciences through grant DE-FG02-98ER14914.

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

References

- [1] P.N. Brown, G.D. Byrne, A.C. Hindmarsh, *SIAM J. Sci. Stat. Comput.* 10 (5) (1989) 1038-1051.
- [2] A.E. Deane, I.G. Kevrekidis, G.E. Karniadakis, S.A. Orszag, *Phys. Fluids* 3 (10) (1991) 2337-2354.
- [3] P. Holmes, J.L. Lumley, G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge University Press, Cambridge, UK, 1996.
- [4] D.A. Schwer, P. Lu, W.H. Green Jr., *Combust. Flame* 133 (4) (2003) 451-465.
- [5] M.A. Singer, S.B. Pope, H.N. Najm, *Combust. Theory Modelling* 10 (2006) 199-217.
- [6] L. Sirovich, *Quarterly of Applied Mathematics* 45 (3) (1987) 561-571.
- [7] G.P. Smith, D.M. Golden, M. Frenklach, N.W. Moriarty, B. Eiteneer, M. Goldenberg, C.T. Bowman, R.K. Hanson, S. Song, W.C. Gardiner Jr., V.V. Lissianski, Z. Qin, http://www.me.berkeley.edu/gri_mech/.
- [8] G. Strang, *SIAM J. Numer. Anal.* 5 (3) (1968) 506-517.