Learning-based Reduced Order Model Stabilization for Partial Differential Equations: Application to the Coupled Burgers' Equation

Mouhacine Benosman, Boris Kramer, Petros T. Boufounos, Piyush Grover

***Abstract*— We present results on stabilization for reduced order models (ROM) of partial differential equations using learning. Stabilization is achieved via closure models for ROMs, where we use a model-free extremum seeking (ES) dither-based algorithm to optimally learn the closure models' parameters. We first propose to auto-tune linear closure models using ES, and then extend the results to a closure model combining linear and nonlinear terms, for better stabilization performance. The coupled Burgers' equation is employed as a test-bed for the proposed tuning method.**

I. INTRODUCTION

The problem of reducing a partial differential equation (PDE) to a system of finite dimensional ordinary differential equations (ODE) has significant applications in engineering and physics, where solving a PDE model is often too time consuming. Reducing the PDE model to a simpler representation, without loosing the main characteristics of the original model, such as stability and prediction precision, is appealing for real-time model-based computations. However, this problem remains challenging, since model reduction can introduce stability loss and prediction degradation. To remedy these problems, many methods have been developed aiming at what is known as stable model reduction.

In this paper, we focus on additive terms called *closure models* and their application in reduced order model (ROM) stabilization. We develop a learning-based method for stabilization of the ROM, applying extremum-seeking (ES) methods to automatically and optimally tune the free parameters of the closure models.

Our work extends some of the existing results in the field. For instance, a reduced order modelling method is proposed in [1] for stable model reduction of Navier-Stokes flow models. The authors propose stabilization by adding a nonlinear viscosity stabilizing term to the reduced order model. The coefficients of this term are identified using a variational data-assimilation approach, based on solving a deterministic optimization. In [2], [3], a Lyapunov-based stable model reduction is proposed for incompressible flows. The approach is based on an iterative search of the projection modes satisfying a local Lyapunov stability condition.

An example of stable model reduction for the Burger's equation using approximate inertial manifold (AMI) was presented in [4], and using closure models in [5], [6]. These closure models modify some stability-enhancing coefficients of the reduced order ODE model using either constant additive terms, such as the constant eddy viscosity model, or time and space varying terms, such as Smagorinsky models. The added terms' amplitudes are tuned in such a way to stabilize the reduced order model. However, such tuning is not always straightforward. Our work addresses this issue and achieves optimal tuning using learning-based approaches.

This paper is organized as follows: Section II establishes our notation and some necessary definitions. Section III introduces the problem of PDE model reduction and the closure model-based stabilization, and presents the main result of this paper. An example using the coupled Burgers' equation is treated in Section IV. Finally, Section V provides some discussion on our approach and concludes.

II. BASIC NOTATIONS AND DEFINITIONS

Throughout the paper we will use $\|.\|$ to denote the Euclidean vector norm; i.e., for $x \in \mathbb{R}^n$ we have $\|x\| = \sqrt{x^T x}$. The Kronecker delta function is defined as: $\delta_{ij} = 0,$ for $i \neq j$ and $\delta_{ii} = 1$. We will use $\dot{f}$ for the short notation of time derivative of f, and x^T for the transpose of a vector x. A function is said analytic in a given set, if it admits a convergent Taylor series approximation in some neighborhood of every point of the set. We consider the Hilbert space $\mathcal{Z} = L^2([0,1])$, which is the space of Lebesgue square integrable functions, i.e., $f \in \mathcal{Z}$, iff $\int_0^1 |f(x)|^2 dx < \infty$. We define on $\mathcal{Z}$ the inner product $\langle\cdot,\cdot\rangle_{\mathcal{Z}}$ and the associated norm $\|.\|_{\mathcal{Z}}$, as $\|f\|^2_{\mathcal{Z}} = \int_0^1 |f(x)|^2 dx$, and $\langle f,g\rangle_{\mathcal{Z}} = \int_0^1 f(x)g(x)dx$, for $f, g \in \mathcal{Z}$. A function $\omega(t,x)$ is in $L^2([0,T];\mathcal{Z})$ if for each $0 \leq t \leq T$, $\omega(t,\cdot) \in \mathcal{Z}$, and $\int_0^T \|\omega(t,\cdot)\|^2_{\mathcal{Z}} dt \leq \infty$. Finally, in the remaining of this paper by stability we mean stability of dynamical systems in the sense of Lagrange, e.g., [7][1]

III. ES-BASED PDES STABLE MODEL REDUCTION

We consider a stable dynamical system modeled by a nonlinear partial differential equation of the form

$$\dot{z} = \mathcal{F}(z) \in \mathcal{Z}, \tag{1}$$

where $\mathcal{Z}$ is an infinite-dimension Hilbert space. Solutions to this PDE can be obtained through numerical discretization, using, e.g., finite elements, finite volumes, finite differences etc. Unfortunately, these computations are often very expensive and not suitable for online applications such as analysis, prediction and control. However, solutions of the original PDE often exhibit low rank representations in an 'optimal' basis [8]. These representation can be exploited to reduce the PDE to an ODE of significantly lower order.

In particular, dimensionality reduction follows three steps: The first step is to discretize the PDE using a finite number of basis functions, such as piecewise linear or higher order polynomials or splines. In this paper we use the well-established finite element method (FEM), and refer the reader to the large literature, e.g., [9], [10] for details. We denote the approximation of the PDE solution by $z_n(t,x) \in \mathbb{R}^n$, where t denotes the scalar time variable, and x denotes the multidimensional space variable, i.e., x is scalar for a one dimensional space, a vector of two elements in a two dimensional space, etc. We consider the one-dimensional case, where x is a scalar in a finite interval, chosen as $\Omega = [0,1]$ without loss of generality. By a standard abuse of notation, $x \in \mathbb{R}^n$ also denotes the discretization of the spatial domain at equidistant space

M. Benosman (m_benosman@ieee.org) is with Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA. B. Kramer is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, 02139. P. Boufounos and P. Grover are with MERL.

[1]A system $\dot{x} = f(t,x)$ is said to be Lagrange stable if for every initial condition x_0 associated with the time instant t_0, there exists $\epsilon(x_0)$, such that $\|x(t)\| < \epsilon$, $\forall t \geq t_0 \geq 0$.

points, $x_i = i \cdot \Delta x$, for $i = 1, \ldots, n$, and some spatial distance Δx. In this notation, x is an n-dimensional vector.

In a second step one determines a set of spatial basis vectors $\{\phi_i\}_{i=1}^r \in \mathbb{R}^n$ for the discretized problem, which are optimal with respect to a specified criterion. The basis is used to approximate the discretized PDE solution as

$$P_n z(t,x) \approx \Phi z_r(t) = \sum_{i=1}^{r} z_{r\,i}(t)\phi_i(x) \in \mathbb{R}^n, \tag{2}$$

where P_n is the projection of $z(t,x)$ onto $\mathbb{R}^n$. Here, Φ is a $n \times r$ matrix containing the basis vectors ϕ_i as column vectors. In many situations, $r \ll n$, i.e. the dimension of the high fidelity discretization of the PDE is much larger than the dimension r of the optimal basis set.

The third step employs a Galerkin projection, a classical nonlinear model reduction technique, to obtain a ROM of the form

$$\dot{z}_r(t) = F(z_r(t)) \in \mathbb{R}^r. \tag{3}$$

The function $F: \mathbb{R}^r \to \mathbb{R}^r$ is obtained from the weak form of the original PDE, through the Galerkin projection.

The main challenge in this approach lies in selecting the basis matrix Φ, and the criterion of optimality used. Many model reduction methods to find basis functions for nonlinear systems exist, such as proper orthogonal decomposition (POD) [11], dynamic mode decomposition (DMD) [12], and reduced basis methods (RB) [13].

Remark 1: In the remainder, we present the idea of closure models in the framework of POD. However, the derivation it is not limited to a particular type of ROM. Indeed, closure modeling - and more generally stabilization of ROMs - can be applied to other ROMs, such as DMD.

A. Proper Orthogonal Decomposition (POD)

We briefly review the necessary steps for computing POD reduced order models, described in detail in [11], [8]. Models based on POD select basis functions that capture an maximal amount of energy of the original system. In particular, the POD basis functions are computed from a collection of snapshots from the dynamical system over a finite time interval. These snapshots are usually obtained from a discretized approximation of the PDE model. This approximation could be obtained using a numerical method, such as FEM, or using direct measurements of the system modeled by the PDE, if feasible. In this paper, the POD basis is computed from snapshots of approximate numerical solutions of the partial differential equation.

To this end, we compute a set of s snapshots of approximate solutions as

$$\mathcal{S} = \{z_n(t_1,\cdot), ..., z_n(t_s,\cdot)\} \subset \mathbb{R}^n, \tag{4}$$

where n is the selected number of FEM basis functions, and t_i are time instances at which snapshots are recorded (do not have to be uniform). Next, we define the *correlation matrix* K with elements

$$K_{ij} = \frac{1}{s}\langle z_n(t_i,\cdot), z_n(t_j,\cdot)\rangle,\ i,j = 1,...,s. \tag{5}$$

The normalized eigenvalues and eigenvectors of K are denoted by λ_i and v_i, respectively. Note that the λ_i are also referred to as the *POD eigenvalues*. The i^{th} *POD basis function* is given by

$$\phi_i(x) = \frac{1}{\sqrt{s}\sqrt{\lambda_i}}\sum_{j=1}^{s} v_{i,j} z_n(t_j,x),\ i = 1,...,r, \tag{6}$$

where $r \le \min\{s,n\}$, the number of retained POD basis functions, depends on the application. An important property of the POD basis functions is their orthonormality:

$$\langle \phi_i, \phi_j\rangle = \int_0^1 \phi_i(x)\phi_j(x)dx = \delta_{ij} \tag{7}$$

where δ_{ij} denotes the Kronecker delta function. In this new basis, the solution of the PDE (1) can then be approximated by

$$z_n^{pod}(t,x) = \sum_{i=1}^{r} q_i(t)\phi_i(x) \ \in \mathbb{R}^n, \tag{8}$$

where $q_i,\ i = 1,...,r$ are the POD projection coefficients (which play the role of the $z_{r,i}(t)$ in the ROM (3)). To find the coefficients $q_i(t)$, the model (1) is projected on the r-dimensional POD subspace using Galerkin projection. To this end, $z(t,x)$ is replaced by $z_n^{pod}(t,x)$ in equation (1), and subsequently both sides are projected (via the inner product) onto the basis $\{\phi_i\}_{i=1}^r$. Using the orthonormality of the POD basis, equation (7), yields a system of r ODEs

$$\dot{q}(t) = F(q(t)) \in \ \mathbb{R}^r. \tag{9}$$

The Galerkin projection preserves the nonlinear structure of the original PDE.

B. Closure models for ROM stabilization

We start with presenting the problem of stable model reduction in its general form, i.e., without specifying a particular type of PDE. To this end, we highlight the dependence of the general PDE (1), on a single physical parameter μ by

$$\dot{z} = \mathcal{F}(z,\mu) \in \mathcal{Z}. \tag{10}$$

The parameter $\mu \in \mathbb{R}$ is assumed to be critical for the stability and accuracy of the model; changing the parameter can either make the model unstable, or inaccurate for prediction. As an example, since we are interested in fluid dynamical problems, we use μ to denote a viscosity coefficient. The corresponding reduced order POD model takes the form of (8) and (9):

$$\dot{q}(t) = F(q(t),\mu). \tag{11}$$

As we explained earlier, the issue with this straightforward Galerkin POD ROM (denoted POD ROM-G) is that the norm of z_n^{pod} might become unbounded over a finite time support, despite the fact that the solution of (10) is bounded. This can be reasoned by the fact that the discarded modes in the reduced order models contributed to energy dissipation.

One of the main ideas behind the closure models approach is that the viscosity coefficient μ in (11) can be substituted by a virtual viscosity coefficient μ_{cl}, whose form is chosen to stabilize the solutions of the POD ROM (11). Furthermore, a penalty term $H(\cdot,\cdot)$ is added to the original POD ROM-G, as follows

$$\dot{q}(t) = F(q(t),\mu) + H(t,q(t)). \tag{12}$$

The term $H(\cdot,\cdot)$ is chosen depending on the structure of $F(\cdot,\cdot)$ to stabilize the solutions of (12). For instance, one can use the Cazemier penalty model described in [6].

C. Examples of Closure Models

We present various closure models reported in the literature, to illustrate the principles behind closure modeling, and motivate our proposed method. Throughout, r denotes the total number of modes retained in the ROM, and $i \in \{1,\ldots,r\}$ the index of a basis function. Moreover, μ is the nominal value of the viscosity coefficient in (10), and $\mu_e(i)$ is the viscosity term added to the truncated modes, providing the stabilizing effect and improving accuracy. We refer the reader to the cited references for further results on stability guarantees, and heuristics/theoretic analogies of choosing such closure terms.

1) Closure models with constant eddy viscosity coefficients: We start with the simplest case of closure models based on constant stabilizing eddy viscosity coefficients.

-ROM-H model: The first eddy viscosity model, known as the Heisenberg ROM (ROM-H), [14], [6], is simply given by the constant viscosity coefficient

$$\mu_{cl} = \mu + \mu_e. \tag{13}$$

-ROM-R model: This model is introduced in [15], with a mode dependent viscosity coefficient

$$\mu_{cl}(i) = \mu + \mu_e \frac{i}{r}. \tag{14}$$

-ROM-RQ model: Proposed in [6], this model is a quadratic version of the ROM-R, which we refer to as ROM-RQ. It is given by the coefficients

$$\mu_{cl}(i) = \mu + \mu_e \left(\frac{i}{r}\right)^2. \tag{15}$$

-ROM-RQ model: This model proposed in [6], is a root-square version of the ROM-R; we use ROM-RS for reference. It is given by

$$\mu_{cl}(i) = \mu + \mu_e \sqrt{\frac{i}{r}}. \tag{16}$$

-ROM-T model: Known as spectral vanishing viscosity model, is similar to the ROM-R in the sense that the amount of induced damping changes as function of the mode index. This concept has been introduced by Tadmor in [16], and so these closure models are referred to as ROM-T. These models are given by

$$\begin{cases} \mu_{cl}(i) = \mu & \text{for } i \leq m, \\ \mu_{cl}(i) = \mu + \mu_e & \text{for } i > m, \end{cases} \tag{17}$$

where $m \leq r$ is the index of modes above which a nonzero damping is introduced.

-ROM-SK model: Introduced by Sirisup and Karniadakis in [17], falls into the class of vanishing viscosity models. We use ROM-SK for reference; it is given by

$$\begin{cases} \mu_{cl}(i) = \mu + \mu_e e^{\frac{-(i-r)^2}{(i-m)^2}} & \text{for } i \leq m, \\ \mu_{cl}(i) = \mu & \text{for } i > m,\ m \leq r. \end{cases} \tag{18}$$

2) Closure models with time and space varying eddy viscosity coefficients: Several (time and/or space) varying viscosity terms have been proposed in the literature. For instance, [6] describes the Smagorinsky nonlinear viscosity model. However, the model requires online computation of some nonlinear closure terms at each time step, which in general makes it computationally consuming. In [1], a nonlinear (in the state variables) viscosity model is proposed. This requires explicitly rewriting the ROM model (11), to separate the linear viscous term as follows

$$\dot{q}(t) = F(q(t), \mu) = \tilde{F}(q(t)) + \mu\ Dq(t), \tag{19}$$

where $D \in \mathbb{R}^{r \times r}$ represents a constant viscosity damping matrix, and the function $\tilde{F}(\cdot)$ represents the remainder of the ROM model, i.e., the part without damping. Based on equation (19), we can write the nonlinear eddy viscosity model denoted by $H_{nev}(.)$, as

$$H_{nev}(\mu_e, q(t)) = \mu_e \sqrt{\frac{V(q(t))}{V_\infty(\lambda)}} \mathrm{diag}(d_{11}, ..., d_{rr}) q(t), \tag{20}$$

where $\mu_e > 0$ is the amplitude of the closure model, the $d_{ii},\ i = 1, \ldots, r$ are the diagonal elements of the matrix D, and $V(q),\ V_\infty(\lambda)$ are defined as follows

$$V(q) := \frac{1}{2}\sum_{i=1}^{r} {q_i}^2, \qquad V_\infty(\lambda) := \frac{1}{2}\sum_{i=1}^{r} \lambda_i, \tag{21}$$

where the λ_i are the selected POD eigenvalues (as defined in Section III-A). Compared to the previous closure models, the nonlinear term H_{nev} does not just act as a viscosity, but is rather added directly to the right-hand side of the reduced order model (19), as an additive stabilizing nonlinear term. The stabilizing effect has been analyzed in [1] based on the decrease over time of an energy function along the trajectories of the ROM solutions, i.e., a Lyapunov-type analysis.

All these closure models share several characteristics, including a common challenge, among others [1], [5]: the selection and tuning of their free parameters, such as the closure models amplitude μ_e. In the next section, we show how ES can be used to auto-tune the closure models' free coefficients and optimize their stabilizing effect.

D. Main result: ES-based closure models auto-tuning

As mentioned in [5], the tuning of the closure model parameter is important to achieve stabilization of the ROM. We use model-free ES optimization algorithms to optimally tune the coefficients of the closure models presented in Section III-B. The advantage of using ES is the auto-tuning capability that such algorithms allow. Moreover, in contrast to manual off-line tuning approaches, the use of ES allows us to constantly tune the closure model, even in an online operation of the system. Indeed, ES can be used off-line to tune the closure model, but it can also be connected online to the real system to continuously fine-tune the closure model coefficients, such as the amplitudes of the closure models. Thus, the closure model can be valid for a longer time interval compared to the classical closure models with constant coefficients, which are usually tuned off-line over a fixed finite time interval.

We start by defining a suitable learning cost function. The goal of the learning (or tuning) is to enforce Lagrange stability of the ROM model (11), and to ensure that the solutions of the ROM (11) are close to the ones of the original PDE (10). The later learning goal is important for the accuracy of the solution. Model reduction works toward obtaining a simplified ODE model which reproduces the solutions of the original PDE (the real system) with much less computational burden, i.e., using the lowest possible number of modes. However, for model reduction to be useful, the solution should be accurate.

We define the learning cost as a positive definite function of the norm of the error between the approximate solutions of (10) and the ROM (11), as follows

$$\begin{aligned} Q(\hat{\mu}) &= \tilde{H}(e_z(t, \hat{\mu})), \\ e_z(t, \hat{\mu}) &= z_n^{pod}(t, x, \hat{\mu}) - z_n(t, x, \mu), \end{aligned} \tag{22}$$

where $\hat{\mu} \in \mathbb{R}^d$, for some dimension d, denotes the learned parameter(s), and $\tilde{H}$ is a positive definite function of e_z. Note that the error e_z could be computed off-line using solutions of the ROM (11), and approximate solutions of the PDE (10). The error could be also computed online where the $z_n^{pod}(t,x,\hat{\mu})$ is obtained from solving the model (11), but the $z_n(t,x,\mu)$ is obtained from real measurements of the system at selected space points x.

A more practical way of implementing the ES-based tuning of $\hat{\mu}$, is to start with an off-line tuning of the closure model. Then, the obtained ROM, i.e., the computed optimal value of $\hat{\mu}$, is used in an online operation of the system, e.g., control and estimation. One can then fine-tune the ROM online by continuously learning the best value of $\hat{\mu}$ at any give time during the operation of the system. To derive formal convergence results, we use some classical assumptions of the solutions of the original PDE, and on the learning cost function.

Assumption 1: The solutions of the original PDE model (10), are assumed to be in $L^2([0,\infty);\mathcal{Z})$, $\forall\mu$.

Assumption 2: The cost function Q in (22) has a local minimum at $\hat{\mu}=\mu^*$.

Assumption 3: The cost function Q in (22) is analytic and its variation with respect to μ is bounded in the neighborhood of μ^*, i.e., $\|\frac{\partial Q}{\partial\mu}(\tilde{\mu})\|\leq\xi_2$ for all μ and $\xi_2>0,\ \tilde{\mu}\in\mathcal{V}(\mu^*)$, where $\mathcal{V}(\mu^*)$ denotes a compact neighborhood of μ^*.

Under these assumptions, the following lemma follows.

Lemma 1: Consider the PDE (10), under Assumption 1, together with its ROM model (11), where the viscosity coefficient μ is substituted by μ_{cl}. Let μ_{cl} take the form of any of the closure models in (13) to (18), where the closure model amplitude μ_e is tuned based on the following ES algorithm

$$\begin{aligned}\dot{y}&=a\ \sin(\omega t+\frac{\pi}{2})Q(\hat{\mu}_e),\ y(0)=0,\\ \hat{\mu}_e&=y+a\ \sin(\omega t-\frac{\pi}{2}),\end{aligned}$$

where $\omega>\omega^*$, ω^* large enough, and Q is given by (22). Under Assumptions 2, and 3, there exists an $\xi_1>0$ such that error w.r.t. the optimal value of μ_e, $e_\mu=\mu^*-\hat{\mu}_e(t)$ admits the following bound

$$\|e_\mu(t)\|\leq\frac{\xi_1}{\omega}+a,\quad t\to\infty,$$

and the learning cost function approaches its optimal value within the following upper-bound

$$\|Q(\hat{\mu}_e)-Q(\mu^*)\|\leq\xi_2(\frac{\xi_1}{\omega}+a),\quad t\to\infty$$

where $\xi_2:=\max\limits_{\mu\in\mathcal{V}(\mu^*)}\|\frac{\partial Q}{\partial\mu}\|$.

Proof 1: Please refer to [18].

Where the linear terms of the PDE are dominant, e.g., in short-time scales, closure models based on constant linear eddy viscosity coefficients can be a good solution to stabilize ROMs and preserve the intrinsic energy properties of the original PDE. However, in many cases with nonlinear energy cascade, these closure models are unrealistic; linear terms cannot recover the nonlinear energy terms lost during the ROM computation. For this reason, many researchers have tried to come up with nonlinear stabilizing terms for instable ROMs. An example of such a nonlinear closure model is the one given by equation (20), and proposed in [19] based on finite-time thermodynamics (FTT) arguments and in [20] based on scaling arguments.

We now introduce a combination of both linear and nonlinear closure models, wich can lead to a more efficient closure model. In particular, this combination can efficiently handle linear energy terms, that are typically dominant for small time scales and handle nonlinear energy terms, which are typically more dominant for large time-scales and in some specific PDEs/boundary conditions. Furthermore, we propose to auto-tune this closure model using ES algorithms, which provides an automatic way to select the appropriate term to amplify. It can be either the linear part or the nonlinear part of the closure model, depending on the present behavior of the system, e.g., depending on the test conditions. We summarize this result in the following Lemma.

Lemma 2: Consider the PDE (10) under Assumption 1, together with its stabilized reduced order model

$$\begin{aligned}\dot{q}(t)&=F(q(t),\mu)=\tilde{F}(q(t))+\mu_{lin}\ Dq(t)+H_{nl}(q(t),\mu_{nl}),\\ H_{nl}&=\mu_{nl}\sqrt{\frac{V(q)}{V_\infty(\lambda)}}\mathrm{diag}(d_{11},...,d_{rr})q(t),\end{aligned}$$

where $V(q)$ and $V_\infty(\lambda)$ are defined in equation (21). The linear viscosity coefficient μ_{lin} is substituted by $\mu_{cl}=\mu+\mu_e$ chosen from any of the constant closure models (13) to (18). The closure model amplitudes $\mu_e,\ \mu_{nl}$ are then tuned based on the following ES algorithm

$$\begin{array}{l}\dot{y}_1=a_1\ \sin(\omega_1 t+\frac{\pi}{2})Q(\hat{\mu}_e,\hat{\mu}_{nl}),\\ \hat{\mu}_e=y_1+a_1\ \sin(\omega_1 t-\frac{\pi}{2}),\\ \dot{y}_2=a_2\ \sin(\omega_2 t+\frac{\pi}{2})Q(\hat{\mu}_e,\hat{\mu}_{nl}),\\ \hat{\mu}_{nl}=y_2+a_2\ \sin(\omega_2 t-\frac{\pi}{2}),\end{array}\tag{23}$$

where $y_1(0)=y_2(0)=0$, $\omega_{max}=\max\{\omega_1,\omega_2\}>\omega^*$, ω^* large enough, and Q is given by (22), with $\hat{\mu}=(\hat{\mu}_e,\hat{\mu}_{nl})$. Let $e_\mu(t):=[\mu_e^*-\hat{\mu}_e(t),\mu_{nl}^*-\hat{\mu}_{nl}(t)]^T$ be the error between the computed and optimal values of the tuning parameters. Under Assumptions 2 and 3, the norm of the error vector admits the following bound

$$\|e_\mu(t)\|\leq\frac{\xi_1}{\omega_{max}}+\sqrt{a_1^2+a_2^2},\quad t\to\infty,$$

with $a_1,\ a_2>0,\ \xi_1>0$, and the learning cost function approaches its optimal value within the following upper-bound

$$\|Q(\hat{\mu}_e,\hat{\mu}_{nl})-Q({\mu_e}^*,{\mu_{nl}}^*)\|\leq\xi_2(\frac{\xi_1}{\omega}+\sqrt{a_1^2+a_2^2}),\quad t\to\infty,$$

where $\xi_2:=\max\limits_{(\mu_1,\mu_2)\in\mathcal{V}(\mu^*)}\|\frac{\partial Q}{\partial\mu}\|$.

Proof 2: Please refer to [18].

IV. Example: The coupled Burgers' equation

As an example application of our approach, we consider the coupled Burgers' equation, (e.g., see [21]), of the form

$$\left\{\begin{array}{l}\frac{\partial w(t,x)}{\partial t}+w(t,x)\frac{w(t,x)}{\partial x}=\mu\frac{\partial^2 w(t,x)}{\partial x^2}-\kappa T(t,x)\\ \frac{\partial T(t,x)}{\partial t}+w(t,x)\frac{\partial T(t,x)}{\partial x}=c\frac{\partial^2 T(t,x)}{\partial x^2}+f(t,x),\end{array}\right.\tag{24}$$

where $T(\cdot,\cdot)$ represents the temperature, $w(\cdot,\cdot)$ represents the velocity, κ is the coefficient of the thermal expansion, c the heat diffusion coefficient, μ the viscosity (inverse of the Reynolds number Re), $x\in[0,1]$ is the one dimensional space variable, $t>0$, and $f\in L^2((0,\infty),\mathcal{Z})$, $\mathcal{Z}=L^2([0,1])$ is the external forcing term. The boundary conditions are imposed as

$$\begin{array}{ll}w(t,0)=w_L, & \frac{\partial w(t,1)}{\partial x}=w_R,\\ T(t,0)=T_L, & T(t,1)=T_R,\end{array}\tag{25}$$

where w_L,w_R,T_L,T_R are positive constants, and L and R denote left and right boundary, respectively. The initial conditions are imposed as

$$\begin{aligned}w(0,x)&=w_0(x)\in L^2([0,1]),\\ T(0,x)&=T_0(x)\in L^2([0,1]),\end{aligned}$$

and are specified below. We assume that $\mu>0$ throughout. Following a Galerkin projection onto the subspace spanned by the

POD basis functions, the coupled Burgers' equation is reduced to a POD ROM with the following structure (e.g., see [22])

$$\begin{bmatrix}\dot{q}_w \\ \dot{q}_T\end{bmatrix} = B_1(t) + \mu B_2 + \mu\ D\ q + \tilde{D}q + Cqq^T,$$

where the matrix-valued function $B_1(t)$ is due to the projection of the forcing term f, matrix B_2 is due to the projection of the boundary conditions, matrix D is due to the projection of the viscosity damping term $\mu\frac{\partial^2 w(t,x)}{\partial x^2}$, matrix $\tilde{D}$ is due to the projection of the thermal coupling and the heat diffusion terms $-\kappa T(t,x)$, $c\frac{\partial^2 T(t,x)}{\partial x^2}$, and the matrix C is due to the projection of the gradient-based terms $w\frac{w(t,x)}{\partial x}$, and $w\frac{\partial T(t,x)}{\partial x}$. The solutions in the n-dimensional space are expressed in the POD basis as

$$w_n^{pod}(x,t) = w_{av}(x) + \sum_{i=1}^{r} \phi_{wi}(x) q_{wi}(t),$$

$$T_n^{pod}(x,t) = T_{av}(x) + \sum_{i=1}^{r} \phi_{Ti}(x) q_{Ti}(t),$$

Here, $\phi_{wi}(x),\ \ q_{wi}(t), i = 1, ..., r_w$, and $\phi_{Ti}(x),\ \ q_{Ti}(t), i = 1, ..., r_T$, are the spatial basis functions and time-dependent coefficients, for the velocity and temperature, respectively. The terms $w_{av}(x)$, $T_{av}(x)$ represent the mean values (over time) of w and T, respectively.

A. Burgers equation ES-based POD ROM stabilization

Due to space limitation, we only report results related to Lemma 2[2], where we test the auto-tuning results of ES on the combination of a linear constant viscosity and a nonlinear closure model. We consider the coupled Burgers' equation (24), with the parameters $Re = 1000$, $\kappa = 5 \times 10^{-4}$, $c = 1 \times 10^{-2}$, the trivial boundary conditions $w_L = w_R = 0$, $T_L = T_R = 0$, a simulation time-length $t_f = 1s$ and zero forcing, $f = 0$. We use ten POD modes for both variables (temperature and velocity). For the choice of the initial conditions, we follow [6], where the simplified Burgers' equation has been used in the context of POD ROM stabilization. Indeed, in [6] the authors propose two types of initial conditions for the velocity variable, which led to instability of the nominal POD ROM, i.e., the basic Galerkin POD ROM (POD ROM-G) without any closure model. Accordingly, we choose the following initial conditions:

$$w(x,0) = \begin{cases} 1, & \text{if } x \in\ [0,\ 0.5] \\ 0, & \text{if } x \in\ ]0.5,\ 1], \end{cases}$$

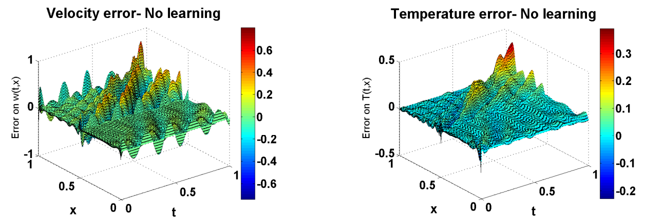
$$T(x,0) = \begin{cases} 1, & \text{if } x \in\ [0,\ 0.5] \\ 0, & \text{if } x \in\ ]0.5,\ 1], \end{cases}$$

We apply Lemma 2, with the Heisenberg linear closure model given by (13). The two closure model amplitudes μ_e and μ_{nl} are tuned using the discrete version of the ES algorithm (23), given by

$$\begin{aligned}
&y_1(k+1) = y_1(k) + a_1\ t_f\ \sin(\omega_1 t_f k + \tfrac{\pi}{2}) Q(\hat{\mu}_e, \hat{\mu}_{nl}), \\
&\hat{\mu}_e(k+1) = y_1(k+1) + a_1\ \sin(\omega_1 t_f k - \tfrac{\pi}{2}), \\
&y_2(k+1) = y_2(k) + a_2\ t_f\ \sin(\omega_2 y_2(k+1) + \tfrac{\pi}{2}) Q(\hat{\mu}_e, \hat{\mu}_{nl}), \\
&\hat{\mu}_{nl}(k+1) = y_2(k+1) + a_2\ \sin(\omega_2 t_f k - \tfrac{\pi}{2}),
\end{aligned}$$

where $y_1(0) = y_2(0) = 0$, and $k = 0, 1, 2, ...$ is the number of the learning iterations. We use the parameters' values: $a_1 =$

[2]A longer version with more tests has been posted on the arXiv version [18].



(a) Error between the true velocity and the learning-free POD ROM velocity profile

(b) Error between the true temperature and the learning-free POD ROM temperature profile

Fig. 1. Errors between the nominal POD ROM and the true solutions- No learning

$6 \times 10^{-6}\ [-]$, $\omega_1 = 10\ [\frac{rad}{sec}]$, $a_2 = 6 \times 10^{-6}\ [-]$, $\omega_2 = 15\ [\frac{rad}{sec}]$. The learning cost function is chosen as

$$Q(\mu_e, \mu_{nl}) = \int_0^{t_f} \langle e_T, e_T \rangle dt + \int_0^{t_f} \langle e_w, e_w \rangle dt. \tag{26}$$

Moreover, $e_T := T_n(\mu) - T_n^{pod}(\mu, \mu_e, \mu_{nl})$ and $e_w := w_n(\mu) - w_n^{pod}(\mu, \mu_e, \mu_{nl})$ define the errors between the true model solution and the POD ROM solution for temperature and velocity, respectively. We report the errors between the true solutions and the POD ROM-G solutions, in Figure 1.

Next, we show the profile of the learning cost function over the learning iterations in Figure 2(a). We can see a quick decrease of the cost function within the first 20 iterations. This means that the ES manages to improve the overall solutions of the POD ROM very fast. The associated profiles for the two closure models' amplitudes learned values $\hat{\mu}_e$ and $\hat{\mu}_{nl}$ are reported in figures 2(b), and 2(c). We can see that even though the cost function value drops quickly, the ES algorithm continues to fine-tune the values of the parameters $\hat{\mu}_e$, $\hat{\mu}_{nl}$ over the iterations, and they eventually reach optimal values of $\hat{\mu}_e \simeq 0.3$, and $\hat{\mu}_{nl} \simeq 0.76$.

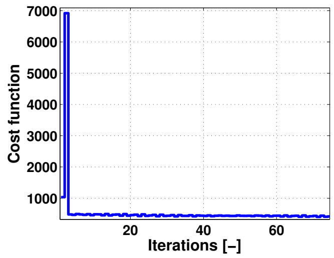
Figure 3 demonstrates the improvement in solution accuracy through learning of the closure terms for the POD ROM models. The reader should note the difference in scaling compared to the errors with no learning in Figure 1.

V. Conclusion

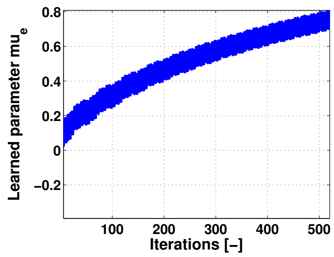
In this work, we explore the problem of stabilization of reduced order models for partial differential equations, focusing on the closure model-based ROM stabilization approach. It is well known that tuning the closure models' gains is an important part in obtaining good stabilizing performances. Thus, we propose a learning ES-based auto-tuning method to optimally tune the gains of linear and nonlinear closure models, and achieve an optimal stabilization of the ROM. We validate our method using the coupled Burgers' equation as an example, demonstrating significant improvements in error performance. The results are encouraging. We defer to future publications verifying our approach on more challenging higher dimensional cases. Our results also raise the prospect of developing new nonlinear closure models, together with their auto-tuning algorithms using extremum-seeking, as well as other machine learning techniques.

References

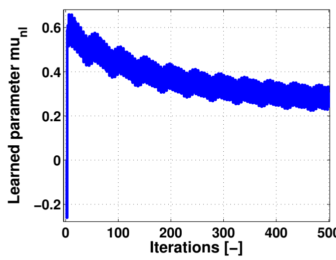
[1] L. Cordier, B. Noack, G. Tissot, G. Lehnasch, J. Delville, M. Balajewicz, G. Daviller, and R. K. Niven, "Identification strategies for model-based control," *Experiments in Fluids*, vol. 54, no. 1580, pp. 1–21, 2013.



(a) Learning cost function vs. number of iterations



(b) Learned parameter $\hat{\nu_e}$ vs. number of iterations



(c) Learned parameter $\hat{\nu_{nl}}$ vs. number of iterations

Fig. 2. Learned parameters and learning cost function- Stabilization with learning- Nonlinear closure model

[2] M. Balajewicz, E. Dowell, and B. Noack, "Low-dimensional modelling of high-reynolds-number shear flows incorporating constraints from the NavierStokes equation," *Journal of Fluid Mechanics*, vol. 729, no. 1, pp. 285–308, 2013.

[3] M. Balajewicz, "Lyapunov stable galerkin models of post-transient incompressible flows," arXiv.org/physics /arXiv:1312.0284, Tech. Rep., December 2013.

[4] J. Baker, A. Armaou, and P. Christofides, "Nonlinear control of incompressible fluid flow: application to burger's equation and 2d channe flow," *Journal of Mathematical Analysis and Applications*, vol. 252, pp. 230–254, 2000.

[5] O. San and J. Borggaard, "Basis selection and closure for POD models of convection dominated Boussinesq flows," in *21st International Symposium on Mathematical Theory of Networks and Systems*, Groningen-The Netherlands, July 2014, pp. 132–139.

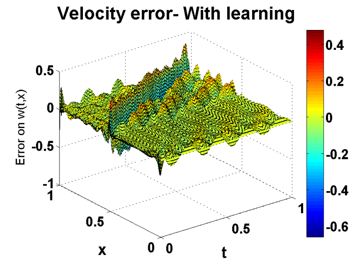
[6] O. San and T. Iliescu, "Proper orthogonal decomposition closure models for fluid flows: Burgers equation," *International Journal of Numerical Analyis and Modeling*, vol. 1, no. 1, pp. 1–18, 2013.

[7] W. Haddad and V. S. Chellaboina, *Nonlinear dynamical systems and control: a Lyapunov-based approach*. Princeton University Press, 2008.

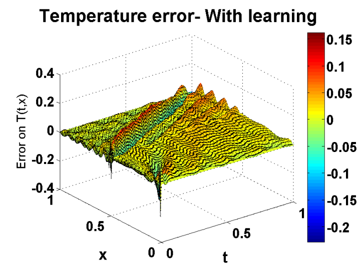
[8] P. Holmes, J. L. Lumley, and G. Berkooz, *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 1998.

[9] O. Sordalen, "Optimal thrust allocation for marine vessels," *Control Engineering Practice*, vol. 15, no. 4, pp. 1223–1231, 1997.

[10] C. A. J. Fletcher, "The group finite element formulation," *Computer Methods in Applied Mechanics and Engineering*, vol. 37, pp. 225–244,



(a) Error between the true velocity and the learning-based POD ROM velocity profile



(b) Error between the true temperature and the learning-based POD ROM temperature profile

Fig. 3. Errors between the learning-based POD ROM and the true solutions-Stabilization with learning- Nonlinear closure model

1983.

[11] K. Kunisch and S. Volkwein, "Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics," *SIAM Journal on Numerical Analysis*, vol. 40, no. 2, pp. 492–515, 2007.

[12] B. Kramer, P. Grover, P. Boufounos, M. Benosman, and S. Nabi, "Sparse sensing and dmd based identification of flow regimes and bifurcations in complex flows," *arXiv*, 2015.

[13] K. Veroy and A. Patera, "Certified real-time solution of the parametrized steady incompressible navier-stokes equations: rigorous reduced-basis a posteriori error bounds," *International Journal for Numerical Methods in Fluids*, vol. 47, no. 8, pp. 773–788, 2005.

[14] Z. Wang, I. Akhtar, J. Borggaard, and T. Iliescu, "Proper orthogonal decomposition closure models for turbulent flows: A numerical comparison," *Computer Methods in Applied Mechanics and Engineering*, vol. 237-240, pp. 10–26, 2012.

[15] D. Rempfer, "Koharente strukturen und chaos beim laminar-turbulenten grenzschichtumschlag," PhD, University of Stuttgart, 1991.

[16] E. Tadmor, "Convergence of spectral methods for nonlinear conservation laws," *SIAM Journal on Numerical Analysis*, vol. 26, no. 1, pp. 30–44, 1989.

[17] S. Sirisup and G. E. Karniadakis, "A spectral viscosity method for correcting the long-term behavior of pod models," *Journal of computational Physics*, vol. 194, no. 1, pp. 92–116, 2004.

[18] M. Benosman, B. Kramer, P. Boufounos, and P. Grover, "Learning-based reduced order model stabilization for partial differential equations: Application to the coupled burgers equation," *arXiv:1510.01728*, 2015.

[19] B. Noack, M. Schlegel, B. Ahlborn, G. Mutschke, M. Morzynski, P. Comte, and G. Tadmor, "A finite-time thermodynamics of unsteady fuid fows," *Journal of Non-Equilibrium Thermodynamics*, vol. 33, no. 2, pp. 103–148, 2008.

[20] B. R. Noack, M. Morzynski, and G. Tadmor, *Reduced-Order Modelling for Flow Control*, 1st ed., ser. 528. Springer-Verlag Wien, 2011.

[21] B. Kramer, "Model reduction of the coupled Burgers' equation in conservation form," Masters Thesis, Virginia Polytechnic Institute and State University, 2011.

[22] ——, "Model reduction for control of a multiphysics system: Coupled Burgers' equation," in *American Control Conference*. IEEE, 2016, to appear.