INT. J. CONTROL, 2004, VOL. 77, NO. 10, 895–906

Taylor & Francis
Taylor & Francis Group

Low dimensional modelling and Dirichlét boundary controller design for Burgers equation

MEHMET ÖNDER EFE†* and HİTAY ÖZBAY‡

Modelling and boundary control for the Burgers equation is studied in this paper. Modelling has been done via processing of numerical observations through proper orthogonal decomposition (POD) with Galerkin projection. This results in a set of spatial basis functions together with a set of ordinary differential equations (ODEs) describing the temporal evolution. Since the dynamics described by the Burgers equation are non-linear, the corresponding reduced-order dynamics turn out to be non-linear. The presented analysis explains how the free boundary condition appears as a control input in the ODEs and how controller design can be accomplished. The issues of control system synthesis are discussed from the point of practicality, performance and robustness. The numerical results obtained are in good compliance with the theoretical claims. A comparison of various different approaches is presented.

1. Introduction

An important problem in the aerospace community is closed-loop aerodynamic flow control. The underlying physical system is governed by Navier–Stokes equations. The efforts towards understanding the behaviour of such systems have already started with simpler systems, e.g. the Burgers equation, a one-dimensional 'cartoon' of Navier–Stokes equations. The reason for this is the fact that the Burgers equation provides the same kind of non-linearity but without turbulence. The difficulty here is the infinite dimensionality, due to which the classical approaches of the control theory are difficult to apply directly. The Burgers equation has previously been considered for modelling and control design purposes (Krstić 1999, Liu and Krstić 2000, 2001, Vedantham 2000, Burns *et al.* 2002 a,b, Hinze and Volkwein 2002, Park and Jang 2002, Efe and Özbay 2003 a). This paper approaches the modelling and control problem from a control specialist's point of view, i.e. a suitable model reduction followed by a controller design under the presence of several performance metrics, such as settling time and percent overshoot.

Modelling and control of such a system contains three major issues that need to be studied carefully. The first issue is the modelling, i.e. collecting the representative data and exploiting several techniques to come up with a set of ODEs. The second issue is to separate the effect of external stimuli from the other terms by using the boundary conditions. The third issue is to design a controller that meets a set of prescribed performance criteria. The process is continuous over a physical domain (Ω), the boundaries of which are the possible entries of external stimuli. Choosing an adequately dense grid, say Ω_d, lets us obtain a finite size representation of the process $u(x,t)$ over Ω_d. When the content of the observed data, say $u(x,t)$, is decomposed into spatial and temporal constituents ($u(x,t) \approx \langle \Phi(x), \alpha(t) \rangle_{\Omega_d}$), the essence of spatial behaviour appears as a set of spatially varying gains ($\Phi(x) = \{\Phi_1(x), \Phi_2(x), \ldots, \Phi_{R_L}(x)\}$), and the essence of temporal evolution, $\alpha(t)$, appears as the solution of a set of ODEs obtained after utilizing the orthogonality properties of the spatial basis functions. Having this picture in front of us, the closed-loop control goal is to observe a predefined behaviour at a set of physical locations by altering the boundary condition(s) appropriately.

When the low dimensional modelling issue is taken into consideration, proper orthogonal decomposition (POD), or singular value decomposition (SVD) in cooperation with Galerkin projection are the popular approaches utilized frequently in the literature (see Ravindran 2000, Ly and Tran 2001, Singh *et al.* 2001, Efe and Özbay 2003 a, b) and the references therein). In Gugercin and Antoulas (2000), a good comparison of model reduction techniques is presented. The decomposition based methods use a library of solutions from the process, and separate the content of the data in such a way that the spatial components (basis functions) display certain orthogonality properties and the temporal components synthesize the time evolution over those spatial basis functions. The decomposition yields meaningful information as long as the data contains coherent modes. One has to know that the result of POD or SVD schemes will be a set of basis functions accompanied by a set of autonomous ODEs.

Received 1 September 2003. Revised and accepted 8 June 2004.

* Author for Correspondence. e-mail: onderefe@ieee.org,
† TOBB University of Economics and Technology, Department of Electrical and Electronics Engineering, Ankara, Turkey.
‡ Department of Electrical and Electronics Engineering, Bilkent University, Bilkent, TR-06800, Ankara, Turkey; on leave from Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210, USA.

International Journal of Control ISSN 0020–7179 print/ISSN 1366–5820 online © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/00207170412331270532

The next issue, which is the separation of boundary condition(s) (or the control input(s)) from the remaining terms, is a key step. For example Krstić describes a neatly selected Lyapunov function, and the expression in its time derivative enables us to apply integration by parts, then the boundary condition emerges in an explicit manner (Krstić 1999). Although the approach lets the designer manipulate Dirichlét and Neumann type boundary conditions on the Burgers equation, it is still tedious to follow the same procedure for more complicated PDEs. This can be because of the high dimensionality of the partial differential equation (PDE) in particular, and difficulty in finding an appropriate Lyapunov function in general. Therefore, utilizing the numerical techniques is a practical alternative to describe reduced-order models for complicated systems of PDEs. A key contribution of this paper is to explain how the issue of control separation is handled in numerical data based modelling approaches.

The third stage is the design of a suitable controller. In Hinze and Volkwein (2002), a receding horizon optimal control approach is studied for the Burgers equation with control input explicitly available in the PDE. The works presented by Burns *et al.* (2002 a,b) demonstrate the stabilization by feedback control. More explicitly, if $u(x, t)$ is the variable of interest, a control signal of the form $\gamma(t) = -\int_{x\in\Omega} k(x)u(x, t)\,\mathrm{d}x$ is suggested in Burns *et al.* (2002 a,b) to minimize a particularly defined quadratic cost function. In Efe and Özbay (2003 a,b), it is discussed that an integral controller of the form $\gamma(t) = K_i(x_m)\int_0^t (u_d(x_m, \rho) - u(x_m, \rho))\,\mathrm{d}\rho$, can be obtained upon linearization. The aim is the tracking based on the feedback signal obtained from a given measurement point x_m. The way in which the gain $K_i(x_m)$ is selected is based on the gain margin analysis, yet the model in Efe and Özbay (2003 a,b) is valid only for low frequencies. This restricts the operating conditions to low frequencies. This problem is addressed in the present paper.

An alternative approach to the design of an optimal controller for the Burgers equation is presented in Vedantham (2000). This reference demonstrates the Cole–Hopf transformation to obtain a linear diffusion type problem. The drawback of this approach is twofold: first it converts the cost function into a relatively complicated equivalent one; second, the applicability of the technique is highly dependent on the structure of the governing PDE. For this reason, approaches such as the one in Park and Jang (2002) are developed to extract valuable information from numerical data. A quadratic cost function is defined and the process of minimization is achieved through the conjugate gradient method. In Liu and Krstić (2000), backstepping control is presented with the assumption that actuators are available at the boundaries. In Liu and Krstić (2001), the viscosity coefficient is assumed to be unknown, and an adaptive control scheme has been presented particularly for the Burgers equation.

Aside from the several novelties highlighted above, the motivation of this paper is to draw a clear path between a given PDE system and closed-loop controller design alternatives. With this in mind, the paper is organized as follows. The second section presents briefly the POD technique and its relevance to the modelling strategy. In the third section, development of the reduced-order model for the Burgers equation is analysed. Section 4 demonstrates the modelling issues with an example. The fifth section presents the design of controllers, and simulation results. In the sixth section, a short discussion on the relationship with flow control is presented. Concluding remarks are given at the end of the paper.

2. Proper orthogonal decomposition

Consider the ensemble $U_i(x)$, $i = 1, 2, \ldots, N_s$, where N_s is the number of elements. Every element of this set corresponds to a snapshot observed from a process, say for example, the one-dimensional Burgers equation

$$u_t(x, t) = \epsilon u_{xx}(x, t) - u(x, t)u_x(x, t) \tag{1}$$

where ϵ is a known constant, and the subscripts x and t refer to the partial differentiation with respect to x and time, respectively. The continuous time process takes place over the physical domain $\Omega := \{x \mid x \in [0, 1]\}$ and the solution is obtained on a grid denoted by Ω_d, which describes the coordinates of the pixels of every snapshot in the ensemble.

The goal is to find an orthonormal basis set letting us write the solution as

$$u(x, t) \approx \hat{u}(x, t) = \sum_{i=1}^{R_L} \alpha_i(t)\Phi_i(x) \tag{2}$$

where $\alpha_i(t)$ is the temporal part, $\Phi_i(x)$ is the spatial part, $\hat{u}(x, t)$ is the finite element approximate of the infinite dimensional PDE and R_L is the number of independent basis functions that can be synthesized from the given ensemble, or equivalently that spans the space described by the ensemble. It will later be clear that if the basis set $\{\Phi_i(x)\}_{i=1}^{R_L}$ is an orthonormal set, then the modelling task can exploit the Galerkin projection technique.

Let us summarize the POD procedure.

Step 1. Start calculating the $N_s \times N_s$ dimensional correlation matrix L, the (ij)th entry of which is $L_{ij} := \langle U_i, U_j\rangle_{\Omega_d}$, where $\langle ., .\rangle_{\Omega_d}$ is the inner product operator defined over the chosen spatial grid Ω_d.

Step 2. Find the eigenvectors (denoted by v_i) and the associated eigenvalues (λ_i) of the matrix L.

Sort them in descending order in terms of the magnitudes of λ_i. Note that every v_i is an $N_s \times 1$ dimensional vector satisfying $v_i^{\mathrm{T}} v_i = 1/\lambda_i$. Here, for simplicity of the exposition, we assume that the eigenvalues are distinct.

Step 3. Construct the basis set by using

$$\Phi_i(x) = \sum_{j=1}^{N_s} v_{ij} U_j(x) \tag{3}$$

where v_{ij} is the jth entry of the eigenvector v_i, and $i = 1, 2, \ldots, R_L$, where $R_L = \mathrm{rank}(L)$. It can be shown that $\langle \Phi_i(x), \Phi_j(x) \rangle_{\Omega_d} = \delta_{ij}$ with δ_{ij} being the Kronecker delta function. Notice that the basis functions are admixtures of the snapshots (Ly and Tran 2001, Efe and Özbay 2003 b).

Step 4. Calculate the temporal coefficients. Taking the inner product of both sides of (2) with $\Phi_i(x)$, the orthonormality property leads to

$$\alpha_i(t_0) = \langle \Phi_i(x), \hat{u}(x, t_0) \rangle_{\Omega_d} \approx \langle \Phi_i(x), U_{t_0} \rangle_{\Omega_d}. \tag{4}$$

Without loss of generality, an element of the ensemble $\{U_i(x)\}_{i=1}^{N_s}$ may be $U(x, t_0)$. Therefore, in order to generate the temporal gain, $\alpha_k(t)$, of the spatial basis $\Phi_k(x)$, one would take the inner product of $\Phi_k(x)$ with the elements of the ensemble as given below:

$$\begin{aligned} \langle U_1(x), \Phi_k(x) \rangle_{\Omega_d} &\approx \alpha_k(t_1) \\ \langle U_2(x), \Phi_k(x) \rangle_{\Omega_d} &\approx \alpha_k(t_2) \\ &\vdots \\ \langle U_{N_s}(x), \Phi_k(x) \rangle_{\Omega_d} &\approx \alpha_k(t_{N_s}). \end{aligned} \tag{5}$$

Note that the temporal coefficients satisfy orthogonality properties over the discrete set $t \in \{t_1, t_2, \ldots, t_{N_s}\}$ (see (6)). For a more detailed discussion on the POD method, the reader is referred to (Ravindran 2000, Ly and Tran 2001, Singh *et al.* 2002, Efe and Özbay 2003 b) and the references therein

$$\sum_{i=1}^{N_s} \langle U_i(x), \Phi_k(x) \rangle_{\Omega_d}^2 \approx \sum_{i=1}^{N_s} \alpha_i^2(t_i) = \lambda_k. \tag{6}$$

Standing Assumption: The majority of works dealing with POD and model reduction applications presume that the flow is dominated by coherent modes and the quantities on both sides of $u(x, t)$ and $\hat{u}(x, t)$ are indistinguishable (Ravindran 2000, Ly and Tran 2001, Singh *et al.* 2002, Efe and Özbay 2003 a,b). Because of the dominance of coherent modes, the typical spread of the eigenvalues of the correlation matrix turns out to be logarithmic and the terms decay very rapidly in magnitude. This fact further enables us to assume that a reduced-order representation, say with M modes ($M \leq \min(R_L, N_s)$) can also be written as an equality

$$\hat{u}(x, t) = \sum_{i=1}^{M} \alpha_i(t) \Phi_i(x), \tag{7}$$

and the reduced-order model is derived under the assumption that (7) satisfies the governing PDE. Unsurprisingly, such an assumption results in a model having uncertainties. However, one should keep in mind that the goal is to find a model, which matches the infinite dimensional system in some sense of approximation with typically $M \ll R_L \leq N_s$. To represent how good such an expansion is, a percent energy measure is defined as

$$E = 100 \frac{\sum_{i=1}^{M} \lambda_i}{\sum_{i=1}^{R_L} \lambda_i} \tag{8}$$

where the tendency of $E \to 100\%$ means that the model captures the dynamical information contained in the snapshots well. Conversely, an insufficient model will be obtained if E is far below 100%. In the next section, we demonstrate how the boundary condition is transformed to an explicit control input in the ODEs.

3. Low dimensional modelling of Burgers system

In this section, we apply the POD technique to the viscous Burgers equation described by

$$u_t(x, t) = \epsilon u_{xx}(x, t) - u(x, t) u_x(x, t) \tag{9}$$

where $\epsilon = 4$ is a known process parameter, $x \in \Omega$ and $\Omega = [0, 1]$. The problem is specified with the initial condition $u(x, 0) = 0 \; \forall x$, the homogeneous boundary condition at $x = 0$ as $u(0, t) = 0$ and Dirichlét boundary condition at $x = 1$ as $u(1, t) = \gamma(t)$, where $\gamma(t)$ is the free external input (boundary condition, or the control input) of the system. Since the POD scheme yields the decomposition in (2), according to the standing assumption, inserting $\hat{u}$ in the place of u in (9) results in

$$\sum_{i=1}^{M} \dot{\alpha}_i(t) \Phi_i(x) = \sum_{i=1}^{M} \alpha_i(t) \epsilon \frac{\partial^2 \Phi_i(x)}{\partial x^2} - \left(\sum_{i=1}^{M} \sum_{j=1}^{M} \alpha_i(t) \alpha_j(t) \Phi_i(x) \frac{\partial \Phi_j(x)}{\partial x} \right). \tag{10}$$

Taking the inner product of both sides of (10) with $\Phi_k(x)$, which corresponds to the Galerkin projection,

results in the equality in (11)

$$\dot{\alpha}_k(t) = \sum_{i=1}^{M} \alpha_i(t)\epsilon\langle \Phi_k(x), \zeta_i(x)\rangle_{\Omega_d} - \left(\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i(t)\alpha_j(t)\langle \Phi_k(x), \Phi_i(x)\beta_j(x)\rangle_{\Omega_d}\right) \quad (11)$$

where $\zeta_i(x) := \partial^2\Phi_i(x)/\partial x^2$ and $\beta_i(x) := \partial\Phi_i(x)/\partial x$. As mentioned earlier, the effects of the external stimulus are implicit in the above equation. For this reason, choose the grid $\Omega_d^\circ := \{x \mid x \in \bigcup_{i=0}^{S-2}(i\Delta x)\}$, where S is the number of grid points considered for the numerical solution satisfying $(S-1)\Delta x = 1$. Clearly, $\Omega_d^\circ \bigcup 1 \equiv \Omega_d$, or equivalently the boundary $\partial\Omega_d := \{x \mid x = 1\}$. According to these definitions, $\langle f(x), g(x)\rangle_\Omega = \langle f(\boldsymbol{x}), g(\boldsymbol{x})\rangle_{\Omega_d} = \langle 1/N_s\rangle f(\boldsymbol{x})^{\mathrm{T}} g(\boldsymbol{x})$, where the column vector $\boldsymbol{x}$ contains the elements of Ω_d in ascending order. In a similar fashion, one can define $\boldsymbol{x}^\circ$ as the column vector containing the elements of Ω_d° in the same way. Taking the above partitioning into account, and rewriting (11) yields

$$\begin{aligned} N_s\dot{\alpha}_k(t) &= \sum_{i=1}^{M} \alpha_i(t)\epsilon\Phi_k^{\mathrm{T}}(\boldsymbol{x})\zeta_i(\boldsymbol{x}) - \left(\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i(t)\alpha_j(t)\Phi_k^{\mathrm{T}}(\boldsymbol{x})(\Phi_i(\boldsymbol{x}) \star \beta_j(\boldsymbol{x}))\right) \\ &= \sum_{i=1}^{M} \alpha_i(t)\epsilon\Phi_k^{\mathrm{T}}(\boldsymbol{x}^\circ)\zeta_i(\boldsymbol{x}^\circ) + \sum_{i=1}^{M} \alpha_i(t)\epsilon\Phi_k(1)\zeta_i(1) \\ &\quad - \left(\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i(t)\alpha_j(t)\Phi_k^{\mathrm{T}}(\boldsymbol{x}^\circ)(\Phi_i(\boldsymbol{x}^\circ) \star \beta_j(\boldsymbol{x}^\circ))\right) \\ &\quad - \left(\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i(t)\alpha_j(t)\Phi_k(1)\Phi_i(1)\beta_j(1)\right) \end{aligned} \quad (12)$$

where $\star$ denotes the elementwise product operator.

Since the external inputs are not seen explicitly in (12), in what follows, the terms will be manipulated such that the two dynamics, namely the one entered directly with the boundary condition and the one governed by the PDE along the spatial direction, are separated properly. The driving point is to notice that the solution in (7) must be satisfied at the boundaries as well. This gives the information

$$u(1,t) = \gamma(t) = \sum_{i=1}^{M} \alpha_i(t)\Phi_i(1). \quad (13)$$

Or $\alpha_k(t)\Phi_k(1) = \gamma(t) - \sum_{i=1}^{M}(1-\delta_{ik})\alpha_i(t)\Phi_i(1)$. Inserting this into the second summation in (12) yields

$$\begin{aligned} &\sum_{i=1}^{M} \alpha_i(t)\epsilon\Phi_k(1)\zeta_i(1) \\ &= \alpha_k(t)\epsilon\Phi_k(1)\zeta_k(1) + \sum_{i=1}^{M}(1-\delta_{ik})\alpha_i(t)\Phi_k(1)\zeta_i(1) \\ &= \gamma(t)\epsilon\zeta_k(1) + \sum_{i=1}^{M} \alpha_i(t)\epsilon(\Phi_k(1)\zeta_i(1) - \Phi_i(1)\zeta_k(1)). \end{aligned} \quad (14)$$

Similarly, considering (13) for the last term of (12), we can perform the rearrangement

$$\begin{aligned} &\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i(t)\alpha_j(t)\Phi_k(1)\Phi_i(1)\beta_j(1) \\ &\quad = \Phi_k(1)\sum_{i=1}^{M} \alpha_i(t)\Phi_i(1)\sum_{j=1}^{M} \alpha_j(t)\beta_j(1) \\ &\quad = \Phi_k(1)\gamma(t)\sum_{j=1}^{M} \alpha_j(t)\beta_j(1). \end{aligned} \quad (15)$$

Summing up the four terms of (12) results in

$$\begin{aligned} N_s\dot{\alpha}_k(t) &= \left(\sum_{i=1}^{M} \alpha_i(t)\epsilon\Phi_k^{\mathrm{T}}(\boldsymbol{x}^\circ)\zeta_i(\boldsymbol{x}^\circ)\right) + \gamma(t)\epsilon\zeta_k(1) \\ &\quad + \sum_{i=1}^{M} \alpha_i(t)\epsilon(\Phi_k(1)\zeta_i(1) - \Phi_i(1)\zeta_k(1)) \\ &\quad - \left(\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i(t)\alpha_j(t)\Phi_k^{\mathrm{T}}(\boldsymbol{x}^\circ)(\Phi_i(\boldsymbol{x}^\circ) \star \beta_j(\boldsymbol{x}^\circ))\right) \\ &\quad - \left(\Phi_k(1)\gamma(t)\sum_{j=1}^{M} \alpha_j(t)\beta_j(1)\right) \\ &= \left(\sum_{i=1}^{M} \alpha_i(t)\epsilon(\Phi_k^{\mathrm{T}}(\boldsymbol{x})\zeta_i(\boldsymbol{x}) - \Phi_i(1)\zeta_k(1))\right) \\ &\quad - \left(\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i(t)\alpha_j(t)\Phi_k^{\mathrm{T}}(\boldsymbol{x}^\circ)(\Phi_i(\boldsymbol{x}^\circ) \star \beta_j(\boldsymbol{x}^\circ))\right) \\ &\quad + \left(\epsilon\zeta_k(1) - \Phi_k(1)\sum_{j=1}^{M} \alpha_j(t)\beta_j(1)\right)\gamma(t). \end{aligned} \quad (16)$$

Defining the state vector as $\alpha = (\alpha_1 \quad \alpha_2 \quad \cdots \quad \alpha_M)^{\mathrm{T}}$, it becomes obvious that the above model implies the following dynamical system for temporal components of the POD

$$\dot{\alpha} = A\alpha - C(\alpha) + (B - D\alpha)\gamma \quad (17)$$

where $A, D \in \Re^{M\times M}$ and $B, C \in \Re^{M}$. We have

$$(A)_{ki} = \frac{1}{N_s}\epsilon\left(\Phi_k^{\mathrm{T}}(\underline{x})\zeta_i(\underline{x}) - \Phi_i(1)\zeta_k(1)\right) \tag{18}$$

$$C(\alpha) = \begin{pmatrix}\alpha^{\mathrm{T}}C_1\alpha & \alpha^{\mathrm{T}}C_2\alpha & \cdots & \alpha^{\mathrm{T}}C_M\alpha\end{pmatrix}^{\mathrm{T}} \tag{19}$$

where $(C_k)_{ij} = (1/N_s)\Phi_k^{\mathrm{T}}(\underline{x}^{\circ})(\Phi_i(\underline{x}^{\circ}) \star \beta_j(\underline{x}^{\circ}))$

$$(B)_k = \frac{1}{N_s}\epsilon\zeta_k(1) \tag{20}$$

and

$$(D)_{ki} = \frac{1}{N_s}\beta_i(1)\Phi_k(1) \tag{21}$$

with $\alpha(0) = 0$. In the next section, the issues of modelling are discussed with numerical results.

4. Modelling results

In this section, we discuss the important issues in the modelling phase and demonstrate that the developed model performs well under the chosen conditions. In the model derivation stage, we chose $\epsilon = 4$ and collected the snapshots according to the following procedure: the end time is 1 sec and the time step is 1 msec, i.e. we have 1001 snapshots at each solution trial, and the nth trial is performed with $u(x, 0) = 0$, $u(0, t) = 0$,

$$u(1, t) = \gamma(t) = \sin\left(2\pi\frac{2n-1}{1.024}t\right)$$

where $n = 1, 2, \ldots, 101$. Such a boundary excitation scheme covers frequencies approximately up to 196 Hz. The snapshot collection scheme linearly samples ten snapshots from each trial to build up the ensemble of snapshots, $\{U_i(x)\}_{i=1}^{N_s}$. During this procedure, the Crank–Nicholson method is used as the numerical solver with $S = 100$ spatial grid points, i.e. $\Delta x = 1/99$ (Farlow 1993). Running this kind of boundary regime excites a reasonably large number of dynamical constituents of the Burgers system. In Efe and Özbay (2003 b), the locality of the POD models has been emphasized and this problem is alleviated by maintaining the spectral diversity in the snapshots. In other words, executing the above-described experiments gives us a model capturing those frequencies to some extent. Nevertheless, reliability of the resulting dynamical model should be viewed as a decreasing quantity as the operating conditions become dissimilar from the model generation conditions.

The simulations have shown that for $M < 5$, the energy content is insufficient to rebuild the numerical data. On the other hand, for $M > 7$, the basis functions are steeper and numerical differentiation errors become significant. For this reason, we set $M = 5$, which captures $E = 99.9875\%$ of the total energy. One should note that the application of the above-described procedure gives us the terms seen in (17) as well as the basis functions of the output equation in (7). Our expectation is to have a good match between the response to a test signal obtained from the numerical solver and from the dynamical system. For this purpose, we choose a chirp signal added to a 10 Hz sinusoid given by (22) and illustrate the relevant subdomain of fast Fourier transform (FFT) magnitude components in figure 1. It should be visible that the chosen signal contains an admissible spectral richness to validate that the model works appropriately nearly below 200 Hz. It has a distinguishable low frequency component to excite the diffusion term ϵu_{xx}, where the larger the ϵ the more diffusive the behaviour along the x-direction

$$\gamma(t) = \sin(2\pi 150(1-t)t) + \sin(2\pi 10t) \quad \text{for } 0 \le t \le 1.023\,\text{sec}. \tag{22}$$

In order to obtain the α values of the numerical solution, we applied the POD scheme with linearly sampled 256 snapshots out of 1024. The original α values and those obtained from the dynamical model have been illustrated in figure 2. The subplots in the first column depict the original and reconstructed signals together, whereas the second column illustrates the difference between them. The results here suggest that the dynamically important modes are synthesized precisely, whereas the modes having higher indices are not. Since the latter modes have little effect on the overall result, any mismatch after the third mode seems tolerable. The mean squared error over 102 400 points (due to (22), 1024 points along time, and $S = 100$ points along the spatial direction) has the value 0.0148, which is quite reasonable to utilize this model for control system design purposes. A comparison of the numerical and approximate solutions is depicted in figure 3.

5. Controller design

Design of a suitable feedback controller for the plant, whose dynamics are governed by the Burgers equation, is discussed in this section. We start with a simple controller, namely the one based on Ziegler–Nichols PID tuning using the step response method. The design of this does not necessitate the reduced-order model, yet at the fine tuning stage, the information contained in the basis functions helps significantly. The second one is based on the linearization of the reduced-order model around $u = 0$ or equivalently $\alpha = 0$. At the controller design stage, we consider the local system obtained for $x_m = 0.5$ and design a root-locus based controller. Then the fine tuning is performed to meet the performance specifications uniformly over the spatial domain. The third scheme is based on the expert knowledge. A fuzzy controller admitting the

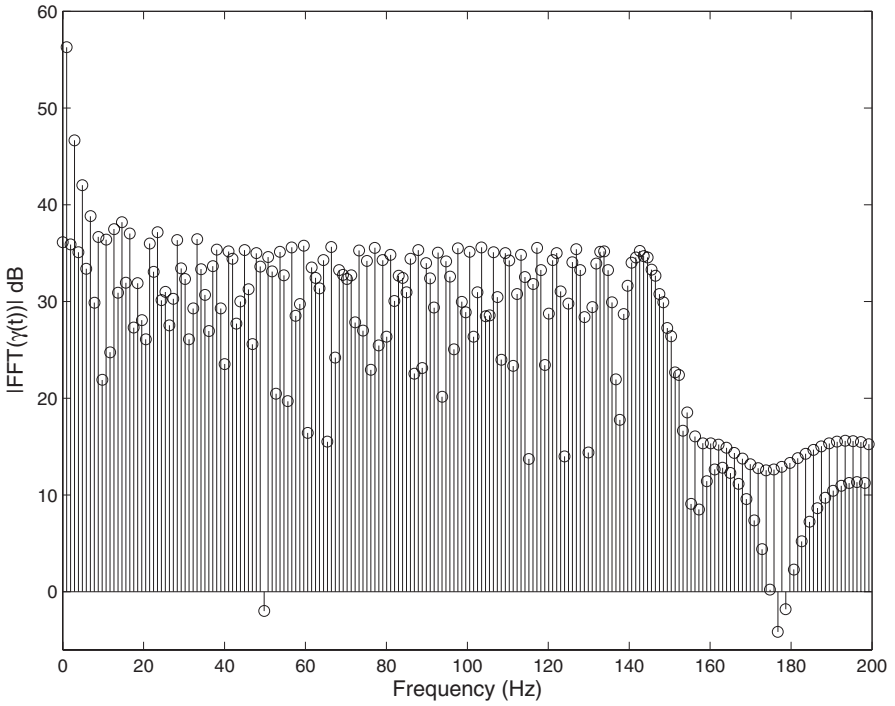

Figure 1. FFT magnitude view of the $\gamma(t)$ for frequencies below 200 Hz.

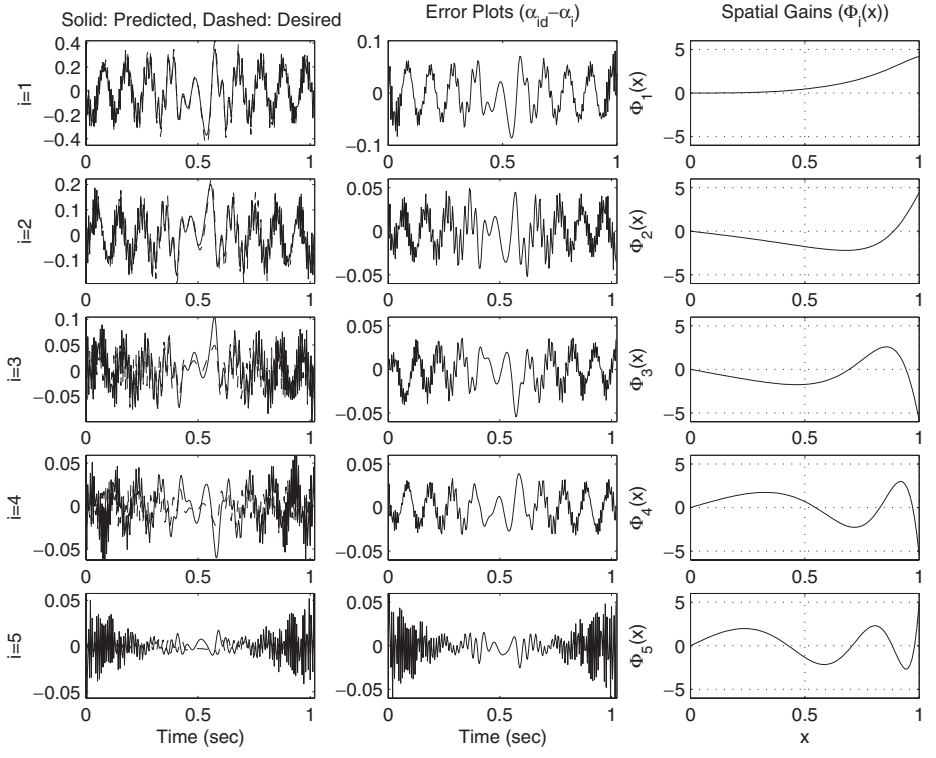

Figure 2. Obtained temporal information and the basis set.

measurement point ($x = x_m$) and the error as the inputs and outputting $\dot{\gamma}(t)$ is constructed. Again, the hidden spatial implications in $\{\Phi_i(x)\}_{i=1}^{R_L}$ have been exploited. The controller is operated non-adaptively to provide a fair comparison between the three approaches. In all three cases, once the design is completed, the tests are carried out on the embedded finite element Burgers equation solver in Matlab/Simulink environment.

5.1. *A Ziegler–Nichols based PI controller*

Due to their widespread use, it is a natural starting point to perform the first trials with a PID type controller. In this subsection, we follow the step response method for a PI controller. Since the derivative action introduces undesired effects particularly for noisy observations, we omit the derivative action and apply a unit step signal from $x=1$ and choose a measurement point $x=x_m$. According to the procedure described by Aström and Hägglund (1995), the point at which the slope is maximum is found. The tangent line is drawn and the intersection of this line with the horizontal axis is determined. The distance between this point and the origin is denoted by $\ell(x_m)$. The tangent line is further drawn to observe the intersection with the vertical axis. The distance between the intersection point and the origin is called $a(x_m)$. According to this, defining $e(x_m,t)$ as the discrepancy between a desired signal and $u(x_m,t)$, the controller is $\gamma(t)=K_p(x_m)e(x_m,t)+K_i(x_m)\int_0^t e(x_m,\sigma)d\sigma$ and its gains are given by

$$K_p(x_m)=\frac{0.9}{a(x_m)} \quad \text{and} \quad K_i(x_m)=\frac{0.3}{a(x_m)\ell(x_m)}. \tag{23}$$

This formulation clearly implies that the controller gains are functions of x_m. Repeating this procedure for all the nodes considered in the spatial grid, one may have a biased initial selection for the controller gains for every $x_m\in[0,1]$.

Apparently, the design presented so far is not involved with the information provided by the low dimensional modelling effort, but the vector norm of the basis functions guides the designer in fine-tuning of the gains in (23). In what follows, it will be clear that the vector norm is a good indicator of how the instantaneous power of $\gamma(t)$ spreads over the spatial direction. For this purpose define the instantaneous power of $\gamma(t)$ (or equivalently $u(1,t)$) as

$$P(1,t):=\frac{1}{2}\gamma(t)^2=\frac{1}{2}\left(\sum_{i=1}^{M}\alpha_i(t)\Phi_i(1)\right)^2 \tag{24}$$

and that of $u(x_m,t)$ as

$$P(x_m,t):=\frac{1}{2}u(x_m,t)^2=\frac{1}{2}\left(\sum_{i=1}^{M}\alpha_i(t)\Phi_i(x_m)\right)^2. \tag{25}$$

Utilizing these quantities, one can define a power transfer ratio as given in (26)

$$T(x_m,t):=\frac{P(x_m,t)}{P(1,t)}=\frac{\alpha(t)^{\mathrm{T}}\Psi(x_m)\Psi(x_m)^{\mathrm{T}}\alpha(t)}{\alpha(t)^{\mathrm{T}}\Psi(1)\Psi(1)^{\mathrm{T}}\alpha(t)} \tag{26}$$

where $\Psi(x_m):=(\Phi_1(x_m)\quad\Phi_2(x_m)\quad\cdots\quad\Phi_M(x_m))^{\mathrm{T}}$. Clearly, when the time is frozen, the above expression puts an upper bound to the power that can be transferred from boundary input to the spatial location $x=x_m$. The maximum value can be given as

$$\begin{aligned}T(x_m,t)&\le\frac{\|\alpha(t)\|_2^2\lambda_{\max}\{\Psi(x_m)\Psi(x_m)^{\mathrm{T}}\}}{\alpha(t)^{\mathrm{T}}\Psi(1)\Psi(1)^{\mathrm{T}}\alpha(t)}\\&=\frac{\|\alpha(t)\|_2^2\Psi(x_m)^{\mathrm{T}}\Psi(x_m)}{\alpha(t)^{\mathrm{T}}\Psi(1)\Psi(1)^{\mathrm{T}}\alpha(t)}\\&=\left(\frac{\|\alpha(t)\|_2^2}{\alpha(t)^{\mathrm{T}}(\Psi(1)\Psi(1)^{\mathrm{T}}/\Psi(1)^{\mathrm{T}}\Psi(1))\alpha(t)}\right)\\&\quad\times\left(\frac{\Psi(x_m)^{\mathrm{T}}\Psi(x_m)}{\Psi(1)^{\mathrm{T}}\Psi(1)}\right)=F(t)K_n(x_m).\end{aligned} \tag{27}$$

The interpretation of the last expression is interesting: no matter what the instantaneous power of the boundary input is, its distribution over Ω is managed by the basis functions. The function $K_n(x_m)$ is close to unity if the feedback is obtained from nodes close to the $x=1$ boundary. On the other hand, it decreases as x_m gets closer to the $x=0$ boundary, at which $K_n(0)=0$. The information contained in $K_n(x_m)$ lets us use a controller globally as it has a normalization effect on the spatial distribution of power. In other words, before performing the fine tuning of the controller gains, one should modify (23) to

$$\begin{aligned}K_p(x_m)&=K_n^{-1}(x_m)\frac{0.9}{a(x_m)}\\\text{and}\quad K_i(x_m)&=K_n^{-1}(x_m)\frac{0.3}{a(x_m)\ell(x_m)}.\end{aligned} \tag{28}$$

It can now be claimed that the PI controller gains are valid on Ω and are ready for fine tuning. We choose the signal $u_d(x_m,t)=\mathrm{sign}(\sin(2\pi ft))$ as the reference signal to be tracked at $x=x_m$. In order to test the designed controller, we embedded the Crank–Nicholson PDE solver into the Matlab/Simulink environment which lets us maintain realistic test conditions. We chose $f=0.2\,\mathrm{Hz}$ and ended up with final gains given in (29) and the results given in the second column of table 1:

$$\begin{aligned}K_p(x_m)&=0.001K_n^{-1}(x_m)\frac{0.9}{a(x_m)}\\\text{and}\quad K_i(x_m)&=0.01K_n^{-1}(x_m)\frac{0.3}{a(x_m)\ell(x_m)}.\end{aligned} \tag{29}$$

In order to see the robustness against load disturbances, we have applied the signal shown at the left subplot of figure 4 at the control input. In the right subplot of figure 4, the behaviour of $K_n^{-1}(x_m)$ is demonstrated. The trend suggests that observing a desired behaviour is possible with relatively smaller control effort as x_m approaches 1-boundary. Conversely, enforcing the same task will necessitate a higher control effort if x_m is close to 0-boundary.

	Ziegler–Nichols Based PI	Root locus	Fuzzy logic
Region of validity x-min	0.10	0.15	0.15
Region of validity x-max	0.99	0.99	0.99
2% Settling time at $x=0.2$ ($u_d=1$)	0.274 sec	0.487 sec	0.095 sec
2% Settling time at $x=0.9$ ($u_d=1$)	0.134 sec	0.354 sec	0.032 sec
Overshoot at $x=0.2$, positive step at $t=0$	0% (well-damped)	0% (well-damped)	0.04%
Overshoot at $x=0.9$, positive step at $t=0$	0% (well-damped)	0% (well-damped)	3.21%
Overshoot at $x=0.2$, negative step at $t=0$	0% (well-damped)	0% (well-damped)	33.48%
Overshoot at $x=0.9$, negative step at $t=0$	0% (well-damped)	0% (well-damped)	0.36%
Overall disturbance rejection at $x=0.2$	Average	Very good	Very good
Overall disturbance rejection at $x=0.9$	Average	Very good	Very good
Ringing during transitions	Low	High	Medium
Practicality (computational simplicity)	High	Medium	Medium
Interpretability (understandability)	Medium	Low	High

Table 1. A comparison of the results.

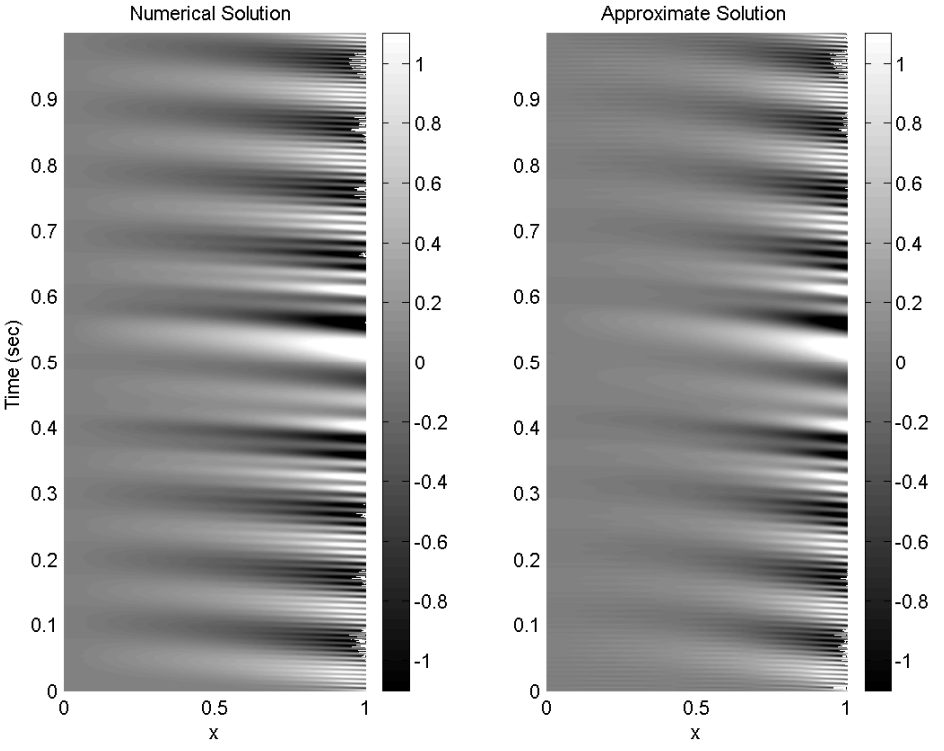

Figure 3. Numerical and approximate solutions.

As highlighted by Aström and Hägglund (1995), and as seen from the comparison of (28) and (29), the step response method overestimates the parameters of the controller gains. As an alternative method, it is recommended to utilize frequency response methods, yet, since the system considered is a non-linear one, those approaches are therefore not applicable. For a better transient behaviour, one should figure out the internal dynamical composition of the system at hand. The next section presents the controller design problem from the root-locus point of view, which lets us modify the modal components of the response.

5.2. *A root-locus based controller*

In this section, we design a root-locus based controller for the system

$$\dot{\alpha}(t) = A\alpha(t) + B\gamma(t)$$

$$\hat{u}(x_m, t) = \Psi(x_m)^{\mathrm{T}}\alpha(t) \tag{30}$$

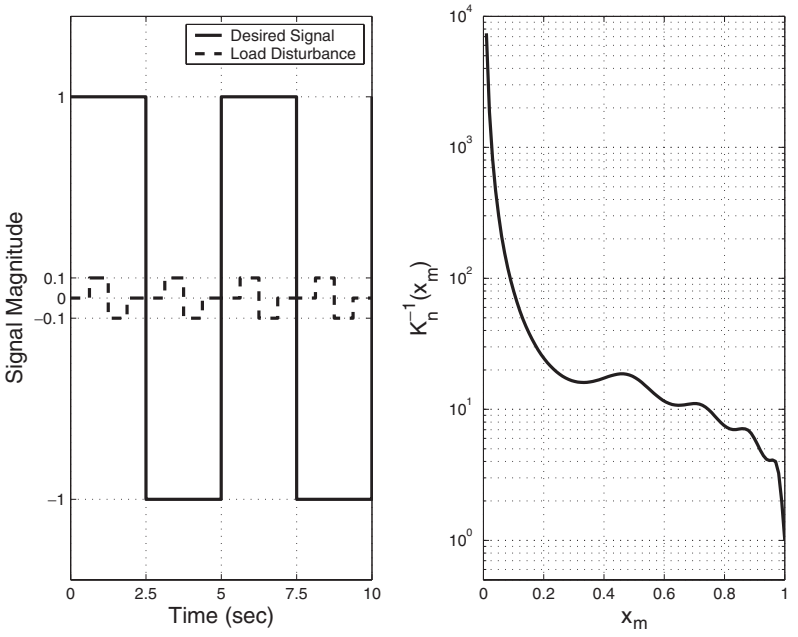

Figure 4. Desired trajectory and the load disturbance (left), function $K_n^{-1}(x_m)$ (right).

which can be obtained upon linearization of the dynamics in (17) around $\alpha = 0$. We consider $x_m = 0.5$ and design a controller such that percent overshoot is less than 4% and settling time is less than 0.1 sec. The transfer function is

$$G(s)|_{x_m=0.5} = -80.1153\frac{(s-220.3)(s+171.9)(s^2-261.7s+1.716e5)}{(s+234.7)(s+106.6)(s+19.08)(s^2+259.1s+2.195e6)} \tag{31}$$

and the controller is

$$H(x_m,s) = K_H(x_m)\frac{(s+19.08)(s^2+145.7s+9494)(s^2+258.7s+2.194e6)}{s(s+191.6)(s+162.5)(s^2+533.9s+7.175e4)} \tag{32}$$

where $K_H(0.5) \cong 0.323$. According to the specifications, the best performance has been obtained with adding a pole at $s = 0$, and cancelling several stable poles of the plant and adding new poles in such a way that the above criteria are met at a simultaneously good level. The linear system settles down in 0.128 sec but the non-linear system, as table 1 indicates, has a longer time to settle down. The fine-tuned controller gain is given as

$$K_H(x_m) = 0.075K_n^{-1}(x_m). \tag{33}$$

In figure 5, we depict the overall shape of the locus, and the dominant poles are depicted in figure 6. The design specifications are depicted as fade-out lines and the chosen configuration is good enough to try with the non-linear system, i.e. the embedded numerical solver.

We further investigated the behaviour of the Nyquist plot for the $x_m = 0.5$ case and justified that it has very good stability and robustness margins. Due to the space limitations, this is skipped. One should note that the system in (31) is complicated enough and so is the controller (see (32)). This is a visible difficulty to follow a rigorous design approach. Instead of this, we followed a trial-and-error based strategy to place the necessary controller poles and zeros so that the dominant dynamics approach the desired one.

5.3. *A non-adaptive fuzzy controller*

The design experience one gains through the PI controller and root locus based controller design examples contains a good expert knowledge about the controller. A very rough look at the results shows that an integral action is required to make the open-loop system Type-1. We adopt a fuzzy gain scheduling mechanism here, and depending on the value of the error signal and the measurement location (x_m), we set the gain of a pure integral controller.

The triangular membership functions are used at the fuzzification stage. These are described as:

$$\left.\begin{aligned}
\mu_{\mathcal{N}}(e) &= \max(0, \min(1, -5e)) \\
\mu_{\mathcal{Z}}(e) &= \max(0, \min(5(e+0.2), 5(-e+0.2))) \\
\mu_{\mathcal{P}}(e) &= \max(0, \min(1, 5e)) \\
\mu_{\mathcal{L}}(x_m) &= \max(0, \min(1, -2x_m+1)) \\
\mu_{\mathcal{M}}(x_m) &= \max(0, \min(2x_m, -2x_m+2)) \\
\mu_{\mathcal{H}}(x_m) &= \max(0, \min(1, 2x_m-1))
\end{aligned}\right\} \tag{34}$$

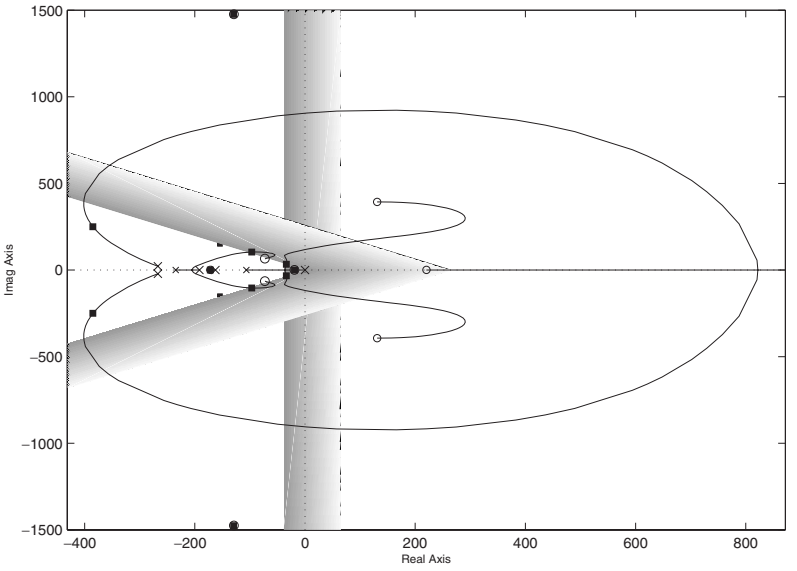

Figure 5. Complete view of the root locus of $H(0.5, s)G(s)$ for $x_m = 0.5$.

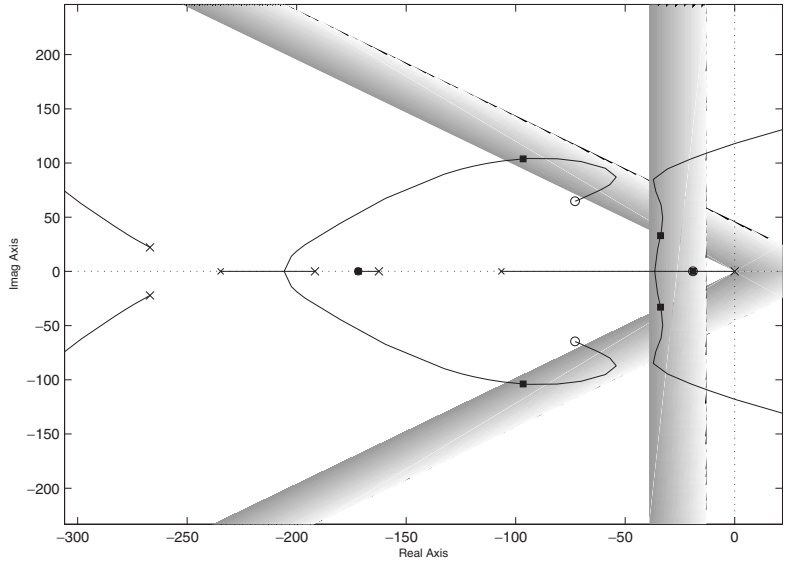

Figure 6. Zoomed view of the root locus of $H(0.5, s)G(s)$ for $x_m = 0.5$.

where $e(x_m, t) = u_d(x_m, t) - u(x_m, t)$ and $\mathcal{N}$, $\mathcal{Z}$, $\mathcal{P}$, $\mathcal{L}$, $\mathcal{M}$, and $\mathcal{H}$ stand for the linguistic labels negative, zero, positive, low, medium and high, respectively. The parameterization of the above membership functions has been performed by trial and error. The membership functions are organized symmetrically with respect to the centre point of the relevant domain, and the value of the sum of them at every point in the domain is unity. Having this picture in front of us, it can fairly be claimed that the choices made at this subsection are absolutely up to what the designer expects. If the operating conditions are too demanding, one may increase the resolution of fuzziness by increasing the number of rules, thereby ending up with some extra computational burden. Furthermore, one may of course improve the fuzzy decision mechanism with an adaptive one yet the goal of this paper is not to prove how superior the fuzzy control schemes are, rather, the goal is to exploit the experience to get an expert control system.

The fuzzy controller used here runs the algebraic product inference mechanism and weighted sum (or equivalently the weighted average due to the fact mentioned above) defuzzification scheme. The defuzzifier has the singletons, which are described in table 2. The rule structure is clearly like IF e is Z_e^r AND x_m is $Z_{x_m}^r$ THEN $\tau = \tau_r$, where r is the rule number, τ is the conclusion, τ_r is the conclusion value for the rth rule (see table 2), and Z_e and Z_{x_m} are the linguistic labels

	e is $\mathcal{N}$	e is $\mathcal{Z}$	e is $\mathcal{P}$
x is $\mathcal{L}$	-100	0	20
x is $\mathcal{M}$	-70	0	80
x is $\mathcal{H}$	-50	0	150

Table 2. Fuzzy rule base and defuzzifier parameters.

quantifying e and x_m, respectively. For a detailed discussion on fuzzy logic and fuzzy controllers, the reader is referred to Jang *et al.* (1997) and Passino and Yurkovich (1998) and references therein.

The same input and disturbance signals have been used in the simulations and the third column of table 1 is filled out. Depending on the priorities of an infinite dimensional control problem, several entries determine the overall cost dominantly. We believe that table 1 conveys adequately informative knowledge to a control specialist.

Lastly in this section, we mention why we confined ourselves to these three approaches only. When the output equation (7) is rewritten as

$$\dot{\hat{\boldsymbol{u}}}(x_m, t) = \Psi^{\mathrm{T}}(x_m)(A\alpha(t) - C(\alpha(t))) + \Psi^{\mathrm{T}}(x_m)(B - D\alpha(t))\gamma(t) \tag{35}$$

the low-order model exhibits orthogonality in between the vectors $\Psi(x_m)$ and $\alpha(t)$ for some t and x_m values. This clearly means that the control signal $\gamma(t)$ has no effect on the output dynamics of the low dimensional model at those (t, x_m) pairs. Just to exemplify, design of feedback linearization or variable structure control techniques based on the low dimensional model would be potentially dangerous from the stability point of view. For this reason, we followed methods that do not require inversion of such potentially dangerous terms and that utilize input/output readings only.

As an overall assessment, since the procedure described here is based on the numerical observations obtained from a process, one should see that the approaches discussed in this paper can enjoy a variety of PDE systems that are currently being researched. From this point of view, it is our understanding that the overall feasibility is quite promising.

6. Relationship with flow control

This paper is related to the notion of flow control in such a way that it describes a way to obtain a representative dynamic model, and demonstrates that the well-known techniques are straightforward to apply afterwards. The authors' primary goal is to apply these techniques to the real-time cavity flow control problem. A detailed discussion on this matter describing the available experimental facility, modelling efforts and the most recent results can be found in Samimy *et al.* (2004) and the references therein. Briefly, after collecting a set of snapshots, the POD technique is to be applied and the control separation technique presented here will be exploited for building the dynamic model. If necessary, interpolation techniques for local models will be investigated. Next, the boundary control techniques and the associated performances will be assessed.

Although the model reduction by POD is one alternative towards modelling, there are other alternatives for the same purpose (see, e.g. Gugercin and Antoulas 2000, Baramov *et al.* 2002 a,b, Lassaux and Willcox 2003 and references therein). Likewise, once the model is developed, it is possible to utilize quadratic optimal control techniques (Bewley 2001, Ravindran 2000, Ly and Tran 2001), $\mathcal{H}_\infty$ control (Baramov *et al.* 2002 a,b) and the techniques presented here. It is noteworthy to emphasize that a significant amount of reported flow control studies deal with the optimality due to the state disturbances and measurement noise; and robustness against dynamics that are lost upon model reduction. Parallel to the literature, this paper assesses the results from a set of performance measures and describes the way to implement such techniques in real-time. The contribution of the research presented in this paper is to develop a useful model with an explicit control input (boundary excitations), and to establish a link between the classical finite dimensional tools of control theory and infinite dimensional systems in general, and aerodynamic cavity flows in particular. In doing this, we figure out how the power transfer ratio normalizes the controller gains. In all stages, the practical implementability has been the central point as our ultimate goal is to apply these approaches in real time.

7. Conclusions

It is a well-known fact that the modelling is a core issue in all control system design problems. Particularly for systems having spatial continuum, the dynamics are governed by PDEs and the standard tools of classical control theory are very difficult to apply directly. In such instances, one strategy is the reduce-then-design approach that is adopted in this paper. As a test example, the Burgers equation is chosen in one dimension, and its solutions under various boundary regimes have been decomposed into its dominant components. This is achieved by a successful application of the POD technique. This yields a set of orthonormal basis functions that enables us to devise the autonomous model. One important contribution of this paper is to clarify how the design can proceed from the available autonomous model to a model having the control input(s) explicitly. Having obtained the dynamical

model, three controller design alternatives have been considered: Ziegler–Nichols based PI controller, root-locus based controller and a fuzzy logic controller. A power transfer ratio concept is developed to describe how the power of the boundary signal is distributed over the physical domain, and how this information can be used to generalize controllers devised for a single point.

The results obtained have shown that all three approaches are successful depending on what the designer's priorities are. The PI controller is simple, valid on a comparably larger domain and highly practical. The root-locus based controller performs very well in terms of rejecting the disturbances. These two approaches result in minimal overshoot, which is a very desirable objective. The fuzzy logic controller is very promising in terms of settling time, interpretability and disturbance rejection, but it comes with some overshoot. This set of overall assessment items demonstrates that the application of any of them will come with a tradeoff.

Finally, it should be emphasized that the reduced-order model can display several undesired properties that are not available in the original system dynamics, and this fact may limit the available design alternatives as emphasized in the fifth section.

Acknowledgements

This work was supported in part by the AFRL/VA and AFOSR through the Collaborative Center of Control Science at the Ohio State University (Contract F33615-01-2-3154).

The authors would like to thank Prof. Mo Samimy, Dr J. H. Myatt, Dr J. DeBonis, Dr R. C. Camphouse, Dr P. Yan, X. Yuan and E. Caraballo for fruitful discussions in devising the presented work.

References

Aström, K. J., and Hägglund, T. H., 1995, *PID Controllers: Theory, Design and Tuning* (ISA — The Instrumentation, Systems, and Automation Society, North Carolina), pp. 134–136.

Baramov, L., Tuty, O. R., and Rogers, E., 2002 a, $\mathcal{H}_\infty$ control of non-periodic two dimensional channel flow. *IEEE Transactions on Control Systems Technology*, **12,** 111–122.

Baramov, L., Tuty, O. R., and Rogers, E., 2002 b, Robust control of linearized Poiseuille flow. *AIAA Journal of Guidance, Dynamics, and Control*, **25,** 145–151.

Bewley, T. R., 2001, Flow control: new challenges for a new Renaissance. *Progress in Aerospace Sciences*, **37**(1), 21–58.

Burns, J. A., King, B. B., Rubio, A. D., and Zietsman, L., 2002 b, Functional gain computations for feedback control of a thermal fluid. *Proceedings of the 3rd Theoretical Fluid Mechanics Meeting*, St. Louis, USA, June 24–26, AIAA, pp. 2002–2992.

Burns, J. A., King, B. B., and Zietsman, L., 2002 a, On the computation of singular functional gains for linear quadratic optimal boundary control problems. *Proceedings of the 3rd Theoretical Fluid Mechanics Meeting*, St. Louis, USA, June 24–26, AIAA, pp. 2002–3074.

Efe, M. Ö., and Özbay, H., 2003 a, Integral action based Dirichlét boundary control of Burgers equation. *Proceedings of the IEEE International Conference on Control Applications*, (CCA'2003), Istanbul, Turkey, June 23–25, pp. 1267–1272.

Efe, M. Ö., and Özbay, H., 2003 b, Proper orthogonal decomposition for reduced order modelling: 2D heat flow. *Proceedings of the IEEE International Conference on Control Applications* (CCA'2003), Istanbul, Turkey, June 23–25, pp. 1273–1278.

Farlow, S. J., 1993, *Partial Differential Equations for Scientists and Engineers* (New York: Dover Publications), pp. 317–322.

Gugercin, S., and Antoulas, A. C., 2000, A comparative study of 7 model reduction algorithms. *Proceedings of the 39th IEEE Conference on Decision and Control*, Sydney, Australia.

Hinze, M., and Volkwein, S., 2002, Analysis of instantaneous control for Burgers equation. *Nonlinear Analysis*, **50,** 1–26.

Jang, J.-S. R., Sun, C. -T., and Mizutani, E., 1997, *Neuro-Fuzzy and Soft Computing* (Upper Saddle River, NJ: Prentice Hall).

Krstić, M., 1999, On global stabilization of Burgers equation by boundary control. *Systems and Control Letters*, **37,** 123–141.

Lassaux, G., and Willcox, K., 2003. Model reduction for active control design using multiple-point Arnoldi methods. *41st Aerospace Sciences Meeting Exhibit*, Reno, NV, USA, AIAA, pp. 2003–0616.

Liu, W.-J., and Krstić, M., 2000, Backstepping boundary control of Burgers equation with actuator dynamics. *Systems and Control Letters*, **41,** 291-303.

Liu, W.-J., and Krstić, M., 2001, Adaptive control of Burgers equation with unknown viscosity. *International Journal Adaptive Control and Signal Processing*, **15,** 745–766.

Ly, H. V., and Tran, H. T., 2001, Modeling and control of physical processes using proper orthogonal decomposition. *Mathematical and Computer Modelling of Dynamical Systems*, **33,** 223–236.

Park, H. M., and Jang, Y. D., 2002, Control of Burgers equation by means of mode reduction. *International Journal of Engineering Science*, **38,** 785–805.

Passino, K., and Yurkovich, S., 1998, *Fuzzy Control* (Menlo Park, CA: Addison Wesley Longman).

Ravindran, S. S., 2000, A reduced order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, **34,** 425–488.

Samimy, M., Debiasi, M., Caraballo, E., Malone, J., Little, J., Özbay, H., Efe, M. Ö., Yan, P., Yuan, X., Debonis, J., Myatt, J. H., and Camphouse, R. C., 2004. Exploring strategies for closed-loop cavity flow control. *42nd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, USA, January 5–8, AIAA, pp. 2004–0576.

Singh, S. N., Myatt, J. H., Addington, G. A., Banda, S. and Hall, J.K., 2001, Optimal feedback control of vortex shedding using proper orthogonal decomposition models. *Transactions of the ASME: Journal of Fluids Engineering*, **123,** 612–618.

Vedantham, R., 2000, Optimal control of the viscous Burgers equation using an equivalent index method. *Journal of Global Optimization*, **18,** 255–263.