LOW-DIMENSIONAL CHARACTERIZATION AND CONTROL OF NON-LINEAR FLOW PHENOMENA

Eric Gillies and John Anderson

Department of Aerospace Engineering,
University of Glasgow, Glasgow, G12 8QQ, UK

Abstract

The spatio-temporal characteristics of many complex fluid flows can often be described in terms of the evolution of a relatively small set of coherent structures, or modes, of the velocity field. This simplified representation of the flow dynamics forms the basis of an adaptive non-linear flow control scheme for a class of unstable wake flows. An extension to the method of proper orthogonal decomposition, which accounts for the influence of external forcing, is used to construct an appropriate set of modes from high-resolution off-line measurements of the velocity field. On-line estimates of the time-varying mode amplitudes are derived from a modal filter based on a limited number of velocity measurements. A feedforward neural network, trained to emulate the observed modal response of the flow to control forcing, is used in conjunction with an adaptive neural network controller to minimize flow fluctuation. The utility of the control scheme is investigated using data for a low Reynolds number, two-dimensional flow behind a circular cylinder exhibiting vortex shedding. Successful feedback control of a simplified model of the unsteady wake flow demonstrates the feasibility of the control scheme.

Introduction

Active control of unsteady flow phenomena attributable to flow instability has attracted much interest in recent years. The potential benefits of managing and controlling unsteady flows of this kind that occur in engineering applications are known to be significant. The design of an appropriate flow control strategy depends, to some extent, on the nature of the instability mechanisms responsible for the observed flow behaviour and on the characterization of the system to be controlled[1,2]. To date, attention has focused on the control of *convectively* unstable flows, such as boundary-layer flows[3,4,5]. These flows are noise amplifiers and have been controlled successfully using linear control methods based on wave superposition techniques. Feedback control of flows exhibiting *absolute* instabilities, such as wake flows, is more difficult[1,6]. Unstable flows of this kind are self-excited, and persist in the absence of external disturbances. They are characterized by *regions* of instability and, in general, are not susceptible to linear control. In addition, the complex spatial structure of such flows limits the effectiveness of linear control schemes based on point actuation and sensing devices[7,8]. Control schemes which address the inherent distributed and non-linear nature of wake-like flows are therefore of interest. Model-based flow control strategies derived directly from the Navier-Stokes equations are conceptually appealing. While some attempts have been made to address the issues associated with the control of non-linear evolution problems governed by partial differential equations[9], the complexity of many engineering flow systems makes this approach unsuitable for real-time applications.

For many dissipative fluid systems, organized structures play an important role in the characterization of the flow[10]. The behaviour of such flows can often be described in terms of the evolution of a relatively small set of spatial modes which are, in some sense, representative of the aggregate flow dynamics. This observation permits a dramatic simplification in the mathematical representation of flows of this kind. Several investigators[10,11,12] have adopted this approach in the characterization of stationary laminar and turbulent flows. Identification of an appropriate set of modes is based on the method of proper orthogonal decomposition[13,14,15] in which the velocity field is represented by a finite-dimensional basis corresponding to local statistical maxima of the flow energy. The flow dynamics is described by a reduced-order model derived from a Galerkin projection of the Navier-Stokes equations onto the optimal basis. An extension of the method of proper orthogonal decomposition to account for non-stationary flow behaviour resulting from the influence of external forcing[16] offers scope for the characterization of controlled flows by similar reduced-order flow models. However, while the analytical form of the reduced-order flow model is known for autonomous flows, the structure of the reduced equation set for controlled flows depends, intimately, on the nature of the control process which, itself, may be difficult to model. Nevertheless, the notional existence of an 'optimal' reduced-order model of the flow in the

presence of a control is appealing. Recently, neural networks[17,18] have been applied to non-linear control problems[19,20] where analytical representation of the system has proved difficult or impractical. For the flow control problem, a neural network can be trained to emulate the modal dynamics of the flow under the action of an applied control, and an adjoining neural network designed to control the flow. The principal advantage of this approach is that it circumvents the need for explicit representation of the reduced-order flow model.

The aim of this paper is to describe a neural network flow control scheme based on a characterization of the flow dynamics by the method of proper orthogonal decomposition. A brief account of the decomposition method for autonomous and non-autonomous flows is presented. This is followed by a description of a generic feedback flow control model incorporating limited flowfield measurements. Application of the control scheme to a prototype flow control problem is then used to demonstrate the feasibility of the approach.

Characterization of Stationary Flows

Modal Decomposition of the Velocity Field

Proper orthogonal decomposition is a method for the characterization of an ensemble of data by an 'optimal' orthonormal basis corresponding to local statistical extrema of the ensemble. To identify an appropriate basis for the unsteady velocity field, it is helpful to search for a *fixed* set of orthogonal vectors (compatible with the spatial representation of the velocity field) which has a structure statistically typical of an ensemble of velocity field measurements taken at different times[10,13].

The discretized velocity field is represented as the sum of the mean (time-average) flow and a time-varying part, expressed as a concatenated vector of local Cartesian velocity components. For a two-dimensional flow, the discretized velocity field is expressed in the form

$$\mathbf{V}(t) = \overline{\mathbf{V}} + \mathbf{V}'(t) \tag{1}$$

where

$$\mathbf{V}'(t) = \begin{bmatrix} \mathbf{v}(x_1, y_1, t) \\ \mathbf{v}(x_2, y_2, t) \\ \vdots \\ \mathbf{v}(x_P, y_P, t) \end{bmatrix} \tag{2}$$

and

$$\mathbf{v}(x_i, y_i, t) = \begin{bmatrix} v_x(x_i, y_i, t) \\ v_y(x_i, y_i, t) \end{bmatrix} \tag{3}$$

Selection of an 'optimal' fixed basis for $\mathbf{V}'(t)$ entails maximization, in some average sense, of the projection

$$(\boldsymbol{\psi} \cdot \mathbf{V}'(t)) = \boldsymbol{\psi}^T \mathbf{V}'(t) \tag{4}$$

for each base vector $\boldsymbol{\psi}$.

To select an appropriate base vector from the *ensemble* of velocity field measurements, the quantity

$$E\{(\boldsymbol{\psi} \cdot \mathbf{V}'(t))^2\} = \lambda \geq 0 \tag{5}$$

is maximized subject to the constraint $(\boldsymbol{\psi} \cdot \boldsymbol{\psi}) = 1$. It is readily shown[13] that extremal $\boldsymbol{\psi}$ correspond to eigensolutions of the algebraic eigenvalue problem

$$\mathbf{R}\boldsymbol{\psi} = \lambda \boldsymbol{\psi} \tag{6}$$

where

$$\mathbf{R} = E\{\mathbf{V}'(t)\mathbf{V}'(t)^T\} \tag{7}$$

Here, $\mathbf{R}$ is the time-averaged spatial correlation matrix of the ensemble of velocity field measurements. The matrix eigenproblem defined by (6) yields an orthogonal set of eigenvectors which characterizes the spatial structure of the flow. The eigenvectors, or modes, can be recognized as 'directions' in R^{2P} along which the variance of the discretized velocity field has local maxima. The fraction of kinetic energy of the velocity field captured by an eigenmode is proportional to its associated eigenvalue. For dissipative fluid flows, most of the system energy is captured by only a few dominant modes of the velocity field. Consequently, modes associated with small eigenvalues can be neglected, and the velocity field characterized, approximately, by a relatively small number of modes.

In general, solution of the eigenproblem (6) is a difficult task if the number of measurement points, P, is large - for flows comprising two space variables, the problem is of order $(2 \times P)^2$. The effective order of the problem can be reduced, however, using the method of 'snapshots'[10,15].

For a sufficient number of time-sampled velocity fields, $\mathbf{V}'(t_k), k = 1, \ldots, M$, the correlation matrix is approximated as

$$\mathbf{R} \approx \frac{1}{M} \sum_{k=1}^{M} \mathbf{V}'(t_k)\mathbf{V}'(t_k)^T \tag{8}$$

The approximate correlation (8) is symmetric and non-negative and, as such, its eigenvectors are of the form

$$\boldsymbol{\psi} = \frac{1}{M} \sum_{k=1}^{M} A(k)\mathbf{V}'(t_k) \tag{9}$$

where

$$A(k) = (\boldsymbol{\psi} \cdot \mathbf{V}'(t_k)) \tag{10}$$

Substitution of relation (9) into the eigenproblem (6) gives

$$\mathbf{CA} = \lambda \mathbf{A} \tag{11}$$

where

$$C_{k\ell} = \frac{1}{M}(\mathbf{V}'(t_k) \cdot \mathbf{V}'(t_\ell)) \tag{12}$$

and

$$\mathbf{A} = (A(1), \ldots, A(M))^T$$

The eigenvectors of the matrix **C** are M-dimensional and are used to form the eigenvectors of the correlation matrix **R** via the relation (9). For an ensemble of M realizations of a particular flow, there exist M eigenvectors of the correlation matrix **R**.

The mode index is arranged such that the eigenvalues decrease monotonically with increasing index number. In practice, many of the modes can be neglected. The N most significant modes are used to reconstruct the flow velocity field; that is,

$$\mathbf{V}(t) \approx \overline{\mathbf{V}} + \sum_{i=1}^{N} A_i(t) \boldsymbol{\psi}_i \tag{13}$$

where

$$A_i(t) = (\boldsymbol{\psi}_i \cdot \mathbf{V}'(t)) \tag{14}$$

The modes developed by the method of proper orthogonal decomposition are optimal, in that they capture more energy, for a given N, than any other expansion.

Mode Evolution and Reduced-Order Flow Models

Evolution equations for the mode coefficients, $A_i(t)$, are obtained from a Galerkin projection of the governing flow equations. In the absence of external forcing, the Navier-Stokes equations for incompressible flow can be written, symbolically, as[10,12]

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbf{N}(\mathbf{V}) \quad \text{in } \boldsymbol{\Omega} \tag{15}$$

$$\nabla \cdot \mathbf{V} = 0 \quad \text{in } \boldsymbol{\Omega}$$

subject to appropriate boundary conditions on $\partial\boldsymbol{\Omega}$.

The modes are, by construction, incompressible and satisfy the boundary conditions. Projection of the Navier-Stokes equation onto the space spanned by the eigenmodes,

$$(\boldsymbol{\psi}_i \cdot (\frac{\partial \mathbf{V}}{\partial t} - \mathbf{N}(\mathbf{V}))) = 0 \qquad i = 1, \ldots, N \tag{16}$$

yields, by virtue of the mode orthogonality, a set of non-linear ordinary differential equations (ODEs) for the mode amplitudes, $A_i(t)$,

$$\frac{d\mathbf{A}(t)}{dt} = \mathbf{F}(\mathbf{A}(t)) \tag{17}$$

where

$$\mathbf{A}(t) = (A_1(t), \ldots, A_N(t))^T$$

In conjunction with the relation (13) and an appropriate set of initial conditions, equations (17) provide a succinct description of the spatio-temporal characteristics of the flow for a fixed set of flow parameters.

A Generic Flow Control Model

Non-Stationary Flow Behaviour

The generation of the eigenvectors of the velocity correlation produces a relatively small set of modes which characterizes the large scale spatial structure of the flow. An implicit assumption incorporated in the construction of the eigenmodes is that the flow is statistically stationary which is not a reasonable assumption for a flow subject to time-varying forcing[16]. Transient, or forced flows, will, in general, contain richer spatial structures than statistically stationary flows. Nevertheless, a set of modes can be formed from an average of the correlation matrices from several transient or forced time-series.

The averaged correlation matrix (which is the counterpart of the matrix **C** in (12)) is formed from a set of Q transient time-series, each comprising M velocity field realizations; that is,

$$\overline{C}_{k\ell} = \frac{1}{QM} \sum_{m=1}^{Q} (\mathbf{V}'(t_k)_m \cdot \mathbf{V}'(t_\ell)_m) \tag{18}$$

The 'non-stationary' modes derived from the eigenvectors of $\overline{\mathbf{C}}$ capture less energy than stationary modes as a result of the averaging process in (18) and the corresponding approximation of the velocity field is sub-optimal. Consequently, a greater number of non-stationary modes is necessary to characterize a forced or transient flow than a stationary flow[16].

Since the non-stationary eigenmodes retain their orthogonality properties and automatically satisfy the flow boundary conditions, the effect of a control (which may involve boundary or interior control) is to modify the form of the reduced-order model describing the evolution of the mode amplitudes.

The reduced-order model appropriate to a flow subject to an external control is of the general form,

$$\frac{d\mathbf{A}(t)}{dt} = \mathbf{G}(\mathbf{A}(t), \mathbf{u}(t)) \qquad (19)$$

Here, $\mathbf{u}(t)$ represents a time-dependent, possibly multi-variable, control.

Emulation of Forced Flow Using Neural Networks

Explicit representation of the reduced-order model (19) is generally not possible. The functional form of $\mathbf{G}$ depends, implicitly, on the nature of the external control and is generally unknown. The discrete-time mapping between the applied control and the modal amplitudes can, however, be emulated by a neural network which can be trained with experimental data. Analytic determination of the reduced-order model (19) is therefore unnecessary. Historically, neural networks have been used to approximate non-linear mappings[17,18], and to determine relationships between sets of data, in situations where traditional modelling methods have proved difficult to apply. Neural networks are also applicable to the emulation of dynamical systems, and they have been applied successfully to predict the response of dynamical systems to control inputs[19,20].

The basic processing unit of a neural network is the neuron, shown in Figure 1. The neuron inputs are weighted and summed and the resulting value is passed to an activation function as an argument.

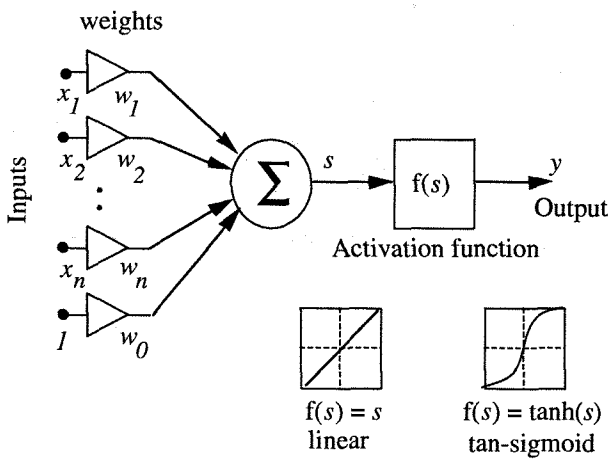


Figure 1. Neuron Model.

A multi-layer feedforward neural network (or perceptron) is formed by interconnecting layers of neurons (Figure 2). If the network contains enough non-linear neurons in its first two hidden layers, and has a linear output layer, then the network can approximate any non-linear function between its inputs and outputs[17,18]. A number of learning algorithms, which adjust the network connection strengths, or weights, have been developed to train neural networks. One such learning algorithm is the backpropagation method - a supervised learning algorithm[18] in which the error between the network's output and the desired system output is minimized by iterative calculation of the equivalent errors of each of the neurons in the network.

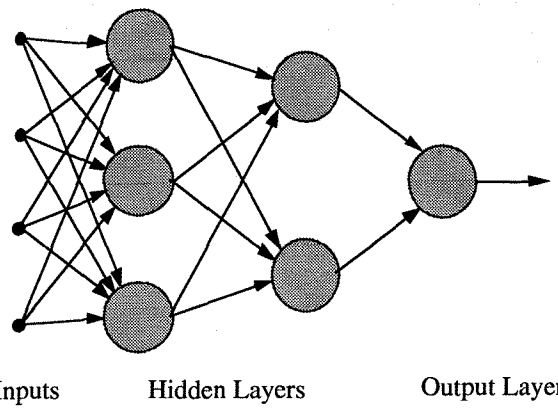


Figure 2. Example of a Network Architecture.

Essentially, the multi-layer perceptron is a *static* network and can be trained to approximate any *algebraic* mapping. To accommodate neural emulation of the mode-amplitude evolution, the reduced-order flow model (19) is represented by an equivalent discrete-time process described by a non-linear regressive model of order p,

$$\mathbf{A}(k+1) = \mathbf{\Gamma}(\mathbf{A}(k), \mathbf{A}(k-1), \ldots, \mathbf{A}(k-p), \mathbf{u}(k)) \qquad (20)$$

If the output of a static network, trained to approximate the process (20), is delayed and fed back to the network input, then the network follows a discrete-time trajectory and behaves like a dynamical system. However, small errors in the network approximation of (20) accumulate after each time-step, so that the trajectory of the network output soon differs from that of the actual system (which is governed by a continuous-time differential equation). The relationship between present and past system states, and the future (one-step) state, is, however, predicted accurately if the network is supplied with the actual present and past system states. The neural emulator then has the form of a one-step predictor,

$$\hat{\mathbf{A}}(k+1) = \hat{\mathbf{\Gamma}}(\mathbf{A}(k), \mathbf{A}(k-1), \ldots, \mathbf{A}(k-p), \mathbf{u}(k)) \qquad (21)$$

which is trained to minimize the error measure

$$\left\| \mathbf{A}(k+1) - \hat{\mathbf{A}}(k+1) \right\|^2 \qquad (22)$$

In this way, a neural emulator is trained to estimate the future state of the fluid system (described by the mode amplitudes), given the present system state and the control input.

Modal Filtering

Neural emulation of the reduced flow model relies on the input of the mode amplitudes measured from the actual flow. A linear neural network can be trained to output the mode amplitudes given a velocity field measurement of the flow. The linear network emulates the projection (14). However, there may exist situations where only a partial set of velocity field data is measurable.

An incomplete velocity field measurement is represented by a vector of the form,

$$\mathbf{V}^*(t_k) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \overline{\mathbf{v}}_r + \mathbf{v}_r(t_k) \\ \overline{\mathbf{v}}_s + \mathbf{v}_s(t_k) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{23}$$

That is, $\mathbf{V}^*(t_k)$ is a vector of the same dimension as $\mathbf{V}^*(t_k)$ The non-zero entries in $\mathbf{V}^*(t_k)$ are the measurable velocity components which include the contribution from the mean flow velocity. The projection of $\mathbf{V}^*(t_k) - \overline{\mathbf{V}}^*$ onto the modal basis results in a set of M simultaneous equations for the M unknown modal amplitudes at each time point,

$$((\mathbf{V}^*(t_k) - \overline{\mathbf{V}}^*) \cdot \boldsymbol{\psi}_j) = \sum_{i=1}^{M} A_i(t_k)(\boldsymbol{\psi}_i^* \cdot \boldsymbol{\psi}_j^*)$$

$$\text{for } j = 1, \cdots, M \tag{24}$$

Here, $\boldsymbol{\psi}_i^*$ is a vector, compatible with $\mathbf{V}^*(t_k)$ whose non-zero entries correspond to those of $\boldsymbol{\psi}_i$. In this case, $(\boldsymbol{\psi}_i^* \cdot \boldsymbol{\psi}_j^*) \neq \delta_{ij}$.

In principle, the system of linear equations (24) can be solved by a standard algebraic technique. Alternatively, a single layer linear neural network can be trained to estimate the mode amplitudes given a partial velocity field, $\mathbf{V}^*(t_k)$, $k = 1, 2, \ldots$, as an input, and the actual mode amplitudes (calculated by either a full projection or direct solution of (24)) as a training reference. The linear network is computationally fast so, once trained, performs well in estimating the mode amplitudes from an incomplete velocity field. The velocity at each measurement point is accompanied by a measurement error. However, this error becomes less significant as the number of measurement points increases. If a set of modes that adequately characterizes the flow (with little error) is developed off-line using full velocity field information, then the modal amplitudes can be determined on-line from an incomplete velocity field measurement.

Design of an Adaptive Controller

The neural emulator provides a method of estimating the dynamical response of the fluid (in terms of the mode amplitudes) to a control input. The emulator forms the basis for the design of a neural control algorithm[5,19,20]. The current mode amplitudes are input to a controller neural network, which provides a control output to the emulator and the actual fluid system. The response of the fluid system to the applied control at the next time step is predicted by the neural emulator. The error between the predicted response and the control goal is the *predicted* control system error. However, the error at the *controller* output (rather than the emulator output) is needed to train the controller[19]. Because the emulator is a neural network, the predicted control system error can be backpropagated through the emulator (with the emulator weights held fixed) to give an equivalent error at the controller output. This equivalent error is used to adjust the controller weights so that the control system error is reduced at the next time step[19,20]. This control strategy is shown, schematically, in Figure 3.

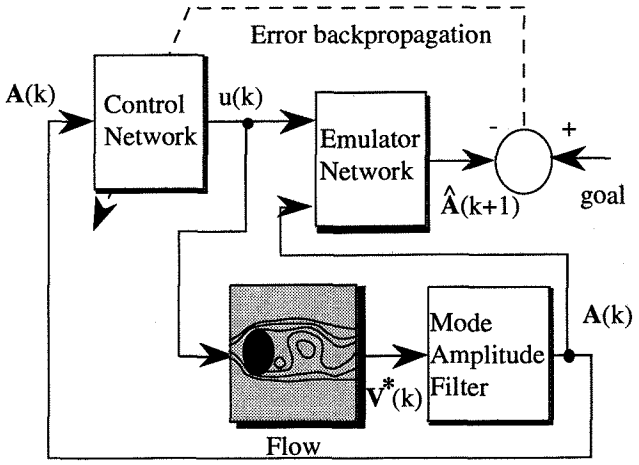


Figure 3. Neural Flow Control Scheme.

A Prototype Flow Control Problem

Basic Flow Model

To illustrate the utility of the aforementioned control scheme, a numerical simulation of the control of the unsteady velocity field of a low Reynolds number two-dimensional flow past a circular cylinder was undertaken. The cylinder flow was chosen as a prototype flow because of the spatial and temporal complexity of the downstream vortex street that forms as a result of a near wake absolute instability[7,12].

Simulated velocity data for an uncontrolled stationary flow at Re=100 was generated by means of a finite-volume solution of the two-dimensional incompressible Navier-Stokes equations. The computational domain, comprising 6600 cells, extended approximately 35 cylinder diameters in the downstream flow direction and 10 cylinder diameters in the transverse direction. Outflow boundary conditions were specified as a zero gradient for all flow variables normal to the boundary. After some experimentation, the time-step adopted for simulation was 600 steps per cycle of vortex shedding oscillation. This arrangement yielded a satisfactory Strouhal number for the flow. Orthogonal decomposition of the limit cycle velocity field data produced a set of modes, whose structures were similar to the vortex structures in the wake. An ensemble of twenty realizations of the velocity field taken from a single flow transient was used for orthogonal decomposition resulting in twenty mode/eigenvalue pairs. The spatial structures (contours of velocity magnitude) of the first four most energetic modes for the prototype flow are shown in Figure 4. The higher order modes are omitted because their eigenvalues are very small, and so they contribute little to the flow energy The modal distribution of energy is shown in Figure 5.

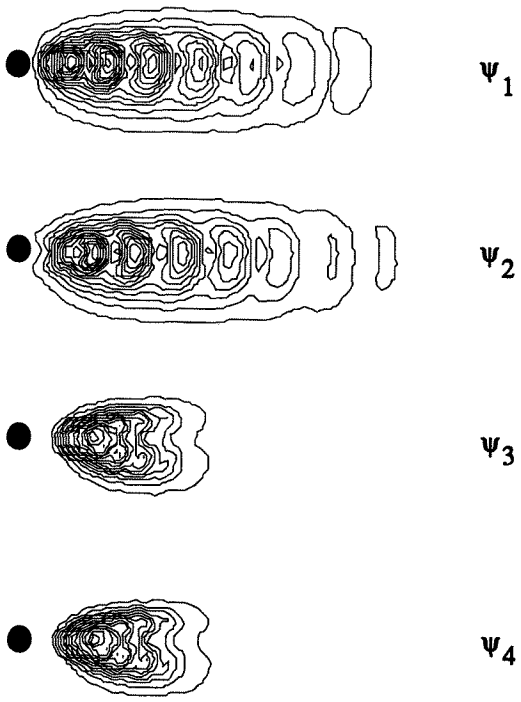


Figure 4. Spatial Structure of the First Four Circular Cylinder Modes, Developed at Re=100.

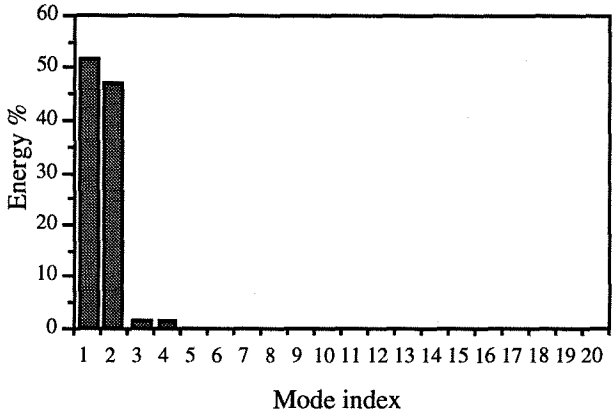


Figure 5. Circular Cylinder Mode Energies.

The reduced-order flow model derived from the orthogonal decomposition of the velocity field is represented by a set of *quadratic* ordinary differential equations for the mode amplitudes

$$\frac{dA_k}{dt} = c_k^0 + \sum_{i=1}^{M} c_k^i A_i + \sum_{i=1}^{M}\sum_{j=1}^{M} c_k^{ij} A_i A_j \qquad (25)$$

The constant coefficients in (25) are formed from the various inner products in (16). Several of the coefficients, however, rely on the calculation of spatial derivatives of the velocity field and modes. Errors in the evaluation of the spatial derivatives resulting from irregular sampling of the velocity field lead to inaccuracies in the model coefficients. To overcome this difficulty, the model structure defined by equation (25) is assumed and the coefficients identified from a time-series of the modal amplitudes $A_i(t_k)$, $k=1,\ldots,n$ derived from the projection operation defined by equation (16). Estimates of the time-derivatives of the mode amplitudes at the sampled times are obtained by means of a cubic spline fit to the mode amplitude histories. The projection operation for the mode amplitudes acts, essentially, as a noise filter, so that the mode amplitude histories are generally quite smooth, and their numerical differentiation is therefore straightforward.

Substitution of the resulting derivative and amplitude values in equation (25) yields a set of nM equations for the $(M+M^2+\frac{1}{2}M^2(M+1))$ unknown coefficients, from which least squares estimates of the coefficients are obtained (the system of equations corresponding to the first four modes is presented in the Appendix). The asymptotic limit cycle behaviour predicted by the reduced-order flow model is shown in Figure 6. The amplitude and phase of the modal response predicted by the reduced-order model compares well with the mode amplitude response calculated directly from the projection (16) (the reduced-order model captures approximately 99.9% of the kinetic energy of the original CFD flow model).

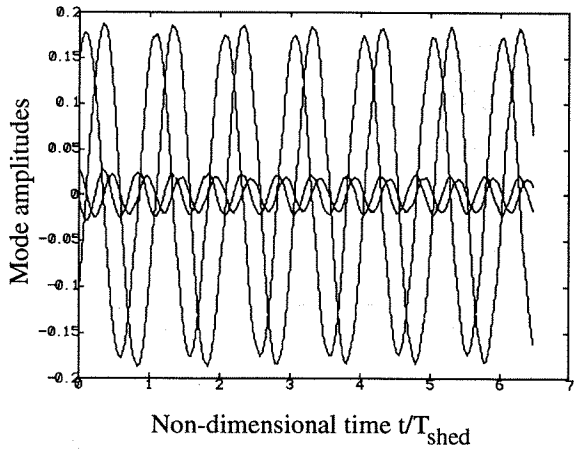


Figure 6. Limit Cycle Behaviour of ODE Model.

Simulation of Adaptive Flow Control

The numerical generation of a large ensemble of transient flow velocity fields (in order to develop non-stationary modes applicable to a forced flow) is computationally prohibitive. In addition, training a neural network emulator with the backpropagation method involves many cycles through the (possibly large) training data set of forced velocity fields. To facilitate a simple exposition of the flow control strategy developed in earlier sections, an *ad hoc* model of a controlled flow is presented in which it is assumed that the modes developed from a single unforced flow transient are similar in structure to the true non-stationary modes developed from a large ensemble of forced flow transients. To maintain consistency with the flowfield boundary conditions embodied in the original stationary modes determined for the unforced flow, the control is assumed to derive from a time-varying point source of strength $u(t)$ in the interior of the flow domain. The effects of such a control are modelled qualitatively by the addition of an appropriate source term in the Navier-Stokes equations; that is,

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbf{N}(\mathbf{V}) + \mathbf{U}u(t) \quad \text{in } \boldsymbol{\Omega} \qquad (26)$$

subject to appropriate boundary conditions on $\partial\boldsymbol{\Omega}$.

The corresponding reduced-order flow model derived from the Galerkin projection of equation (26) is then of the form

$$\frac{dA_k}{dt} = c_k^0 + \sum_{i=1}^{M} c_k^i A_i + \sum_{i=1}^{M}\sum_{j=1}^{M} c_k^{ij} A_i A_j + (\boldsymbol{\Psi}_k \cdot \mathbf{U})u(t) \quad (27)$$

This simple model of the control input/mode amplitude interaction was used to investigate the behaviour of the neural control scheme. Approximation of the temporal evolution of the flow dynamics was achieved by integration of the equation set (27) subject to specified initial conditions. Although somewhat artificial, reduced models of this kind are believed to be capable of modelling the main qualitative features of the flow dynamics.

To train the neural emulator for the flow system, the network was presented with the current system state, $\mathbf{A}(t_k)$, sampled from the output response of (27), and a uniformly distributed random burst signal as a control input. To simulate the effect of measurement uncertainty, noise was added to the sampled system state components. A three layer network of four tan-sigmoid neurons in the first hidden layer, eight tan-sigmoid neurons in the second hidden layer and four linear output neurons provided an adequate approximation of the input/output relationship for the flow. The network performed well in predicting the system states when totally different control forces were applied. Numerical experimentation suggested that the largest sample interval for an acceptable level of accuracy was $0.02T_{shed}$ (where T_{shed} is the period of vortex shedding). As can be seen in Figure 7, the emulator predicted the four mode amplitudes during a random forcing period with reasonable accuracy.

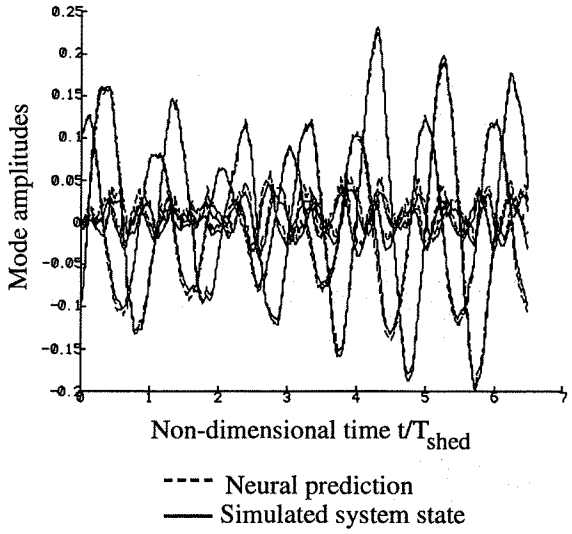
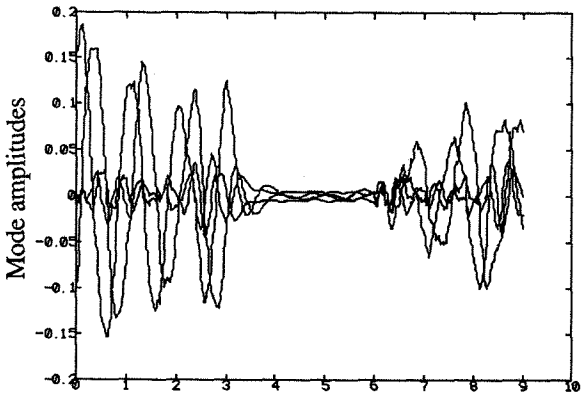


Figure 7.Neural Emulator Performance During Random Forcing Cycle.

The neural controller comprised a three layer fully non-linear network. To simulate the inability of the controller to exceed a physical control limit, gains were added to the output to limit the control amplitude to a certain range. A network with eight first layer neurons, eight second layer neurons and one output neuron was found to be adequate in controlling the reduced-order flow model. The desired performance objective of reduced residual mode amplitude was achieved rapidly after switching on controller learning. The 'on-line' operation of the control scheme is depicted in Figure 8. In order to ensure that the untrained controller did not produce a large initial control which would have made

achievement of the control goal more difficult, the initial weights were made very small, so that the initial controller output was near zero.

Preliminary observations of the control system performance indicate that the steady-state behaviour of the control is periodic but not harmonic. The steady-state mode amplitude/phase relationships of the modal responses also indicate suppression of vortex shedding in the reconstructed velocity field despite the persistence of temporal fluctuations in the flowfield.



(Control on at 3T and off at 6T)

Figure 8. Performance of the Neural Controller.

Conclusions

The design of an active flow control strategy necessitates the characterization of both the spatial, and temporal, non-linear features of the flow. Proper orthogonal decomposition provides an efficient means of characterizing the spatial structure of the flow by an orthogonal set of modes. The modes generated by the decomposition method satisfy the flow boundary conditions. Modifications to the boundary conditions introduced by a control are automatically accounted for using 'non-stationary' modes derived from a transient flow generated by the control. Temporal characterization of the flow is possible using an reduced-order ODE model to describe the evolution of the mode amplitudes. In general, however, explicit representation of the relationship between an applied control force and the mode amplitudes is difficult to realize.

Neural networks are readily applicable to systems where analytical determination of the system model is difficult or impractical. A neural network, exposed to appropriate experimental data sets of control input/mode amplitude histories can be trained to emulate the required input/output relationship between the control and the fluid state (as represented by the mode amplitudes). A second neural network can then be trained to control the flow. Application of an adaptive neural control scheme to a simplified model of a wake flow indicates the feasibility of the scheme. More realistic simulations, however, involving the generation of data for neural network training by CFD methods, are likely to be computationally intensive. It is expected that future research effort will focus on experimental implementation of the control scheme and the development of an adaptive neural network modal filter which performs the dual role of mode estimation and filtering under variation of flow parameters.

Acknowledgements

The first author (EG) gratefully acknowledges the financial support of the Engineering and Physical Sciences Research Council of the UK during the tenure of a postgraduate research studentship.

References

1 P. Heurre and P. A. Monkewitz, 'Local and Global Instabilities in Spatially Developing Flows', *Ann.Rev. Fluid Mech.*, **22**, 473-537 (1990).

2 P. A. Monkewitz, 'Feedback Control of Global Oscillations in Fluid Systems', *AIAA 89-0991* (1989).

3 R. Metcalfe, C. Rutland, J. Duncan and J. Riley, 'Numerical Simulations of Active Stabilization of Laminar Boundary Layers', *AIAA Journal* **24**(9), 1494-1501 (1986).

4 E. Laurien and L. Kleiser, 'Numerical Simulation of Boundary Layer Transition and Transition Control', *J. Fluid Mech* . **199**, 403-441 (1989).

5 S. Jacobson and W. Reynolds, 'Active Control of Boundary Layer Shear Stress Using Self Learning Neural Networks', *AIAA Paper No. 93-3272* (1993).

6 K. Roussopoulos, 'Feedback Control of Vortex Shedding at Low Reynolds Numbers', *J. Fluid Mech.* **248**, 267-297 (1993).

7 G. Karniadakis and G. S. Triantafyllou, Frequency Selection and Asymptotic States in Laminar Wakes', *J. Fluid Mech.* **199**, 441-469 (1989).

8 J. Ffowcs Williams and B. Zhao, 'The Active Control of Vortex Shedding', *J.Fluids and Structures* **3**, 115-122 (1989).

9 H. Choi, R Teman, P. Moin and J. Kim, 'Feedback Control for Unsteady Flow and its Application to the Stochastic Burgers Equation', *J. Fluid Mech.* **253**, 509-543 (1993).

10 L. Sirovich, 'Turbulence and the Dynamics of Coherent Structures, Parts I,II,III', *Quarterly of Applied Mathematics* **XLV**, 561-587 (1987).

11 N. Aubry, P. Holmes, J. L. Lumley and E. Stone, 'The Dynamics of Coherent Structures in the Wall Region of a Boundary Layer', *J. Fluid Mech.* **192**, 115-173 (1988).

12 A. Deane, I. Kevrekidis, G. Karniadakis and S. Orszag, 'Low Dimensional Models for Complex Geometry Flows: Application to Grooved Channels and Circular Cylinders', *Phys. Fluids A* **3**(10), 2337-2354 (1991).

13 J. L. Lumley, *Stochastic Tools in Turbulence,* Academic Press, New York (1970).

14 G. Berkooz, P. Holmes and J. L. Lumley, 'The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows', *Ann. Rev. Fluid Mech.* **25**, 539-575 (1993).

15 L. Sirovich and M. Kirby, 'Low Dimensional Procedure for the Characterization of Human Faces', *J . Opt . Soc. America A* **4**(3), 519-526 (1987).

16 A. Glezer, Z. Kadioglu and A. Pearlstein, 'Development of an Extended Proper Orthogonal Decomposition and its Application to a Time Periodically Forced Plane Mixing Layer', *Phys. Fluids A* **1**(8), 1363-1373 (1989).

17 S. Haykin, *Neural Networks : A Comprehensive Foundation,* Macmillan, New York (1994).

18 D. Hush and B. G. Horne, 'Progress in Supervised Neural Networks', *IEEE Signal Processing Magazine*, pp8-39, Jan. (1993).

19 D. Nguyen and B. Widrow, 'Neural Networks for Self Learning Control Systems', *IEEE Control Systems Magazine*, pp19-23, April (1990).

20 G. Lightbody, Q. Wu and G. Irwin, 'Control Applications for Feedforward Networks', in *Neural Networks For Control and Systems* , (Eds. K. Warwick, G.Irwin and K. Hunt), pp51-69, Peregrinus, London (1992).

Appendix

Four Mode Reduced-Order Model of 2-D Cylinder Flow

$$\begin{aligned}
\frac{dA_1}{dt} &= -1.66290429E\text{-}04 + 2.50169474E\text{-}03A_1 \\
&\quad + 1.42487956E\text{-}01A_2 + 5.18359463E\text{-}02A_3 \\
&\quad - 2.58162018E\text{-}02A_4 + 3.56429413E\text{-}03A_1A_1 \\
&\quad + 1.70567213E\text{-}02A_1A_2 - 2.67112058E\text{-}02A_1A_3 \\
&\quad + 8.70818833E\text{-}02A_1A_4 + 1.49046298E\text{-}03A_2A_2 \\
&\quad - 8.42083454E\text{-}02A_2A_3 + 1.57876233E\text{-}02A_2A_4 \\
&\quad + 1.26628945E\text{-}02A_3A_3 + 2.42933636E\text{-}02A_3A_4 \\
&\quad + 1.58450628E\text{-}02A_4A_4 + 9.998E\text{-}03u(t)
\end{aligned}$$

$$\begin{aligned}
\frac{dA_2}{dt} &= 2.90175296E\text{-}05 - 1.31744505E\text{-}01A_1 \\
&\quad + 6.23852739E\text{-}04A_2 + 6.02492351E\text{-}03A_3 \\
&\quad + 3.63752811E\text{-}02A_4 + 2.57676693E\text{-}02A_1A_1 \\
&\quad - 1.28802553E\text{-}02A_1A_2 + 5.86899819E\text{-}02A_1A_3 \\
&\quad + 3.70566031E\text{-}02A_1A_4 - 1.57004226E\text{-}02A_2A_2 \\
&\quad + 2.42364347E\text{-}02A_2A_3 - 9.81767262E\text{-}02A_2A_4 \\
&\quad - 7.09051882E\text{-}03A_3A_3 - 1.29932356E\text{-}02A_3A_4 \\
&\quad + 5.18047642E\text{-}03A_4A_4 + 9.866E\text{-}03u(t)
\end{aligned}$$

$$\begin{aligned}
\frac{dA_3}{dt} &= 7.85160994E\text{-}04 - 8.40895701E\text{-}04A_1 \\
&\quad - 5.52677989E\text{-}06A_2 - 3.47421084E\text{-}02A_3 \\
&\quad - 3.15618142E\text{-}01A_4 + 2.29766347E\text{-}02A_1A_1 \\
&\quad + 6.70135725E\text{-}04A_1A_2 + 2.52875366E\text{-}03A_1A_3 \\
&\quad - 3.00029222E\text{-}02A_1A_4 - 2.42665864E\text{-}02A_2A_2 \\
&\quad - 8.54086958E\text{-}03A_2A_3 - 3.55181276E\text{-}02A_2A_4 \\
&\quad - 9.76471889E\text{-}03A_3A_3 - 3.76874519E\text{-}02A_3A_4 \\
&\quad + 3.24691326E\text{-}03A_4A_4 + 6.0577E\text{-}03u(t)
\end{aligned}$$

$$\begin{aligned}
\frac{dA_4}{dt} &= 5.53803810E\text{-}05 + 2.65916998E\text{-}04A_1 \\
&\quad + 7.16616167E\text{-}04A_2 + 2.43268601E\text{-}01A_3 \\
&\quad - 2.92261771E\text{-}02A_4 - 5.46440118E\text{-}03A_1A_1 \\
&\quad - 3.81723387E\text{-}02A_1A_2 + 9.67050500E\text{-}03A_1A_3 \\
&\quad + 2.17240367E\text{-}02A_1A_4 - 3.27980557E\text{-}03A_2A_2 \\
&\quad + 5.05461587E\text{-}02A_2A_3 + 3.35363453E\text{-}03A_2A_4 \\
&\quad + 2.94750410E\text{-}02A_3A_3 - 1.51851997E\text{-}02A_3A_4 \\
&\quad - 8.30075377E\text{-}03A_4A_4 + 6.6324E\text{-}03u(t)
\end{aligned}$$