Active Drag Reduction Using Neural Networks

David Babcock[1], Changhoon Lee[2], Bhusan Gupta[1], John Kim[2], Rodney Goodman[1]

[1]Dept. of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125
[2]MANE, University of California Los Angeles, Los Angeles, CA 90024

Abstract

This paper presents the application of a neural network controller to the problem of active drag reduction in a fully turbulent 3D fluid flow regime. The neural network learns a function nearly identical to an analytically derived control law. We then demonstrate the ability of a neural controller to maintain a drag-reduced flow in a fully turbulent fluid simulation. Finally we examine the amount of parameter variation that may be required for a physical implementation of such a neural controller.

1: Introduction

In today's cost-conscious air transportation industry, fuel costs are a substantial economic concern. Drag reduction is an important way to increase fuel efficiency thereby reducing these costs. It is estimated that even a 5% reduction in drag can easily translate into millions of dollars in annual fuel cost savings. Neural networks have been used to actively reduce drag in 2D flow simulations [5]. In this paper we will briefly review two analytic active control laws that achieve substantial drag reduction in fully turbulent 3D flow simulations. We then present a neural network approach to this problem and compare the results of an on-line neural controller with those of the analytic schemes.

2: The cause of drag

Large skin friction drag has recently been linked to organized structures in turbulent flows which play an important role in turbulence transport. The main cause of high drag in turbulent flows is commonly observed near-wall streamwise vortices, see Figure 1. The interaction of these vortices, which appear randomly in both space and time, with the viscous layer near surfaces creates high local regions of surface shear stress which contribute to the total drag. Therefore, attempts to reduce drag in turbulent flows have focused on either preventing the formation or mitigating the strength of these vortices. The small size of these vortices, which decreases as the Reynolds number of the flow increases, has limited physical experimentation. The

0-8186-7456-3/96 $5.00

inherent complexity of the non-linear governing Navier-Stokes equations has likewise limited analytical approaches.

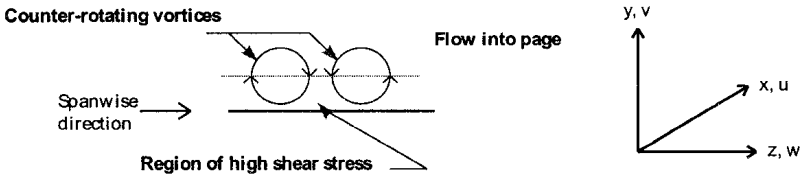

Figure 1. Diagram of the interaction between a vortex pair and a surface. x, y, and z represent the streamwise, normal, and spanwise directions with u, v, and w the respective velocity components.

3: Analytic active control

With the recent development of accurate numerical algorithms, computer simulations of turbulent flows have revolutionized the study of the basic physics of turbulence. Comparison of time-averaged statistics and instantaneous turbulence structures with existing experimental results has validated these computer-generated databases [7]. The availability of full 3D velocity, vorticity, and pressure field data provides a unique opportunity to probe the organized structures in turbulence.

Numerical experimentation has demonstrated that active feedback control can achieve significant drag reduction (≈25%) [2]. This reduction results from the suppression of the interaction of the streamwise vortices with the surface. The active control scheme used in these experiments involves blowing and suction at the surface according to the normal component of the velocity field sensed in the near-wall region directly above the surface, see Figure 2. The actuation is thus given by:

$$v_{wall} = -v(y^+ = 10) \tag{1}$$

We will refer to equation 1 as Near Wall Control (NWC). Choi, et. al. [2] also discovered that approximately 80% of the drag reduction is due to large actuations (i.e. those with magnitude greater than v_{rms}) which account for about 15% of all actuations.

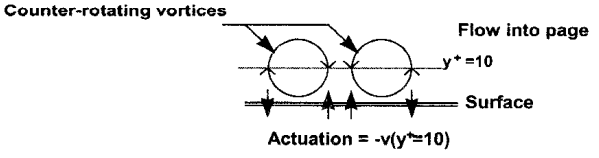

Figure 2. Near-Wall control scheme.

It is difficult and/or impractical to place sensors away from surfaces within the flow field to detect the fluid motions associated with the streamwise vortices. By

correlating the near-wall normal velocity with measurements taken via pressure and shear stress sensors at the surface, a practical version of the control law can be constructed.

The above result indicates that the streamwise vortices can be suppressed by wall blowing and suction. Suboptimal control theory has been used to derive a simple control law for the actuations based only on the surface spanwise shear stress as:

$$\hat{v}_{wall} = C\frac{1}{|k_z|}\frac{\widehat{\partial^2 w}}{\partial z \partial y}\bigg|_{wall} = C\frac{ik_z}{|k_z|}\frac{\widehat{\partial w}}{\partial y}\bigg|_{wall} \tag{2}$$

where k_z is the wave number in the spanwise direction, $\partial w/\partial y \mid_{wall}$ is the surface spanwise shear stress, C is a positive scale factor determining the amplitude of the actuation, and $\hat{(\cdot)}$ denotes a Fourier transformed quantity [8]. Equation 2 produces actuations proportional to the distribution of the *spanwise derivative* of the spanwise surface shear stress. Taking the inverse Fourier transform of $ik_z/|k_z|$ for finite maximum wave number (discretization) , the above control law can be rewritten as

$$v_{wall,k} = C\sum_{j=k-N/2}^{k+N/2} W_j\left(\frac{\partial w}{\partial y}\right)_j \tag{3}$$

where the weights are given by $W_j = A((\cos(\pi j) - 1)/j)$, N denotes the total number of weights, and k denotes the discretized location on the surface. The summation is then performed over the spanwise discretized points. We will refer to equation 3 as Spanwise Derivative Control (SDC). Lee, et. al. [8] have demonstrated that such a control scheme achieves roughly 20% drag reduction (the dashed curve in Figure 6). This result is consistent with those obtained by Choi, et. al. [3] using active control based on flow information in the near-wall region (NWC).

4: Neural network control

The above results demonstrate the existence of active control schemes that achieve appreciable drag reduction in fully turbulent flow simulations. Neural networks offer an alternative approach to active control. A neural network can be used to "learn" a correlation between the surface measurements and the desired control actuations from empirical data. Therefore no prior knowledge of the system is required, although it may be useful in guiding network architecture design.

4.1: Off-line training

As an initial experiment, we trained a neural network using the spanwise surface shear stress as input and the actuation from the NWC scheme as the desired output. The data consisted of 100 time steps of data generated from a simulated controlled flow using NWC. Each time step contains a 32x32 array of input values ($\partial w/\partial y \mid_{wall}$) and the corresponding actuations from NWC ($-v(y^+ = 10)$). Since we want the output to be based on a *local* input area, we design the network using shared weights.

The network has a single set of weights that is convolved over the entire input space to generate the output values. The template (i.e. set of network weights) attempts to find spatially invariant correlations between the input data and the desired output. For our problem we selected a two layer network with hyperbolic tangent hidden units and a linear output unit, see Figure 3.

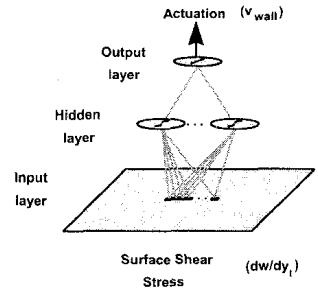

Figure 3. Neural network architecture.

The functional form of the neural network is:

$$v_{wall} = \sum_{i=1}^{N_h} W_i^1 \tanh\left(\sum_{j=-N_i/2}^{N_i/2} W_{ij}^2 \left(\frac{\partial w}{\partial y}\right)_j - W_{ib}^2 \right) - W_b^1 \tag{4}$$

where N_h is the number of hidden units, N_i is the length of the input template, and W^l are the weights for the l^{th} layer. The input template shape can be arbitrary, however based on the SDC, we chose to use a single row of inputs oriented in the spanwise direction (see Figure 3). We then trained several networks with different length input templates and different numbers of hidden units using the scaled conjugate gradient method presented by Moller [10]. Since we wanted the network to accurately predict the large actuations (which as mentioned earlier are responsible for the majority of the drag reduction), we chose to add an exponential prefactor to the standard total sum squared error measure:

$$E = \frac{1}{2} \sum_{\nu} e^{k|\bar{v}_{wall}^{\nu}|} (\bar{v}_{wall}^{\nu} - v_{wall}^{\nu})^2 \tag{5}$$

This error measure exponentially emphasizes the error contributed by the larger actuations depending on the scale parameter k.

The final input templates (normalized by the maximum weight in the template) for two networks with one hidden unit and input template lengths of 5 and 7 respectively are shown in Figure 4. The dashed curves in these figures represent the (normalized) coefficients (c_j's in equation 6) of an N-point centered difference approximation to the spanwise derivative based on Taylor series expansions given by:

$$\frac{\partial^2 w}{\partial y \partial z} \approx \left(\frac{\widetilde{\partial^2 w}}{\partial y \partial z}\right)_N = \sum_{j=-N/2}^{N/2} c_j \left(\frac{\partial w}{\partial y}\right)_j \tag{6}$$

We clearly see that the network learns an approximate spanwise derivative of the spanwise shear stress, hence it "learned" the SDC law. Increasing the number of hidden units only minimally improves performance. However, increasing template length significantly reduces the final training error. This indicates that performance is directly related to the accuracy of the derivative approximation.

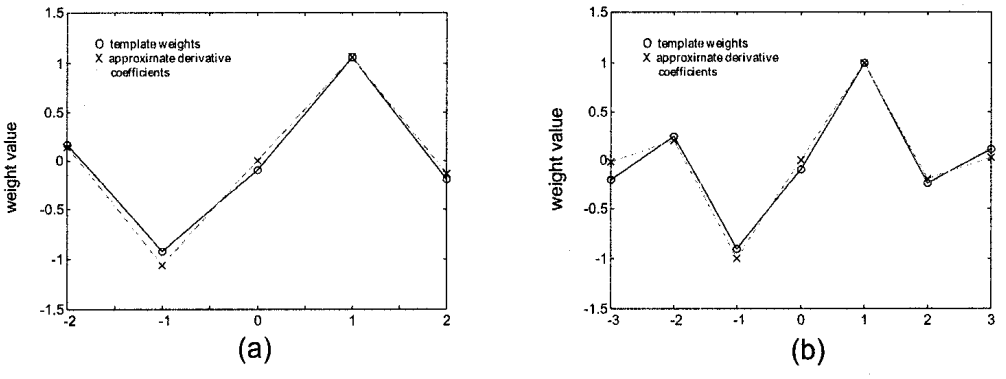

Figure 4. (a) Final template weights for length 5 input array. (b) Final template weights for length 7 input array.

4.2: On-line control

When the off-line trained network was implemented as a controller (by simply fixing the final weights) the performance was very poor. This was expected since we are trying to control a time-varying, non-linear system with a static controller. We, therefore, incorporated on-line training to allow the controller to adapt to the evolution of the fluid flow.

There are numerous implementation schemes for on-line neural control. The most direct is adaptive inverse model control [13]. This scheme is presented pictorially in Figure 5. This configuration employs a neural network to model the (possibly time-varying) inverse plant and then simply uses a copy of the model as the controller. One restriction of this technique is that it usually requires an initial model training phase using random plant inputs and corresponding plant outputs. Once the model represents a reasonably close approximation to the actual plant inverse, a copy is then implemented as a feedforward controller. We have the advantage of the off-line trained networks discussed above to represent an approximate inverse model (and controller), thus avoiding an initial on-line training phase.

The desired inputs to the controller are a fractional reduction in the shear stress from the previous time step, i.e.

$$\left(\frac{\partial w}{\partial y}\right)_{t+1} = \eta\left(\frac{\partial w}{\partial y}\right)_{t} \tag{7}$$

where $0 < \eta < 1$. The output is the predicted actuation necessary to produce this shear stress reduction. Good performance was achieved for $\eta = 0.85$.

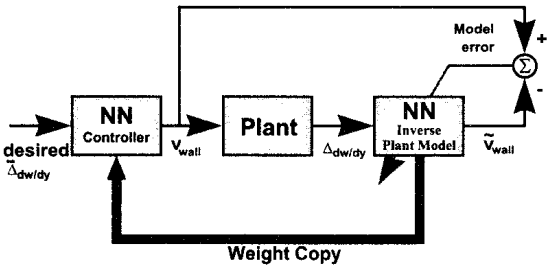

Figure 5. Schematic representation of adaptive inverse model control.

In order to verify that the approximate derivative pattern (figure 4) learned off-line generalizes to the on-line data, we allowed all the weights in the network to adapt and examined the input template pattern after each time step. There was no appreciable change in the relative magnitudes of the template weights over time, indicating that the pattern is preserved. The absolute magnitudes, however, did vary indicating the need for gain and bias adaptation for each layer.

Hence for the current on-line implementation, we fix the input template weights as the approximate derivative coefficients and use a single hidden unit network giving only four adaptable parameters (a bias and gain for each layer). This simplified network has the following functional form:

$$v_{wall} = W_0 \tanh\left(W_2\left(\frac{\widetilde{\partial^2 w}}{\partial y \partial z}\right)_N - W_3\right) - W_1 \tag{8}$$

The results are shown by the solid line in Figure 6. It can be seen that this network performs similarly to the SDC.

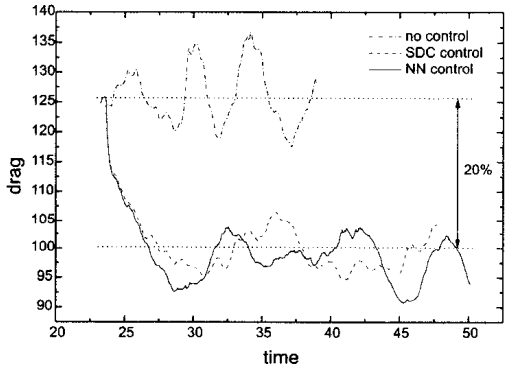

Figure 6. On-line simulation results.

5: Physical systems

The ultimate goal of these control schemes is to implement them in a physical system. An active field of research, inspired by biology, is attempting to build integrated systems utilizing micromachine technology to actively control turbulent flows. Such systems would use control laws similar to the ones discussed above while operating at the necessary physical scale determined by the desired flow conditions.

5.1: Biological inspiration

In many complex problems, one can often find inspiration by observing how nature has evolved a biological solution. For the problem of drag reduction, deep sea sharks serve as a biological model. These sharks (e.g. hammerheads) can swim up to 20 m/s in deep water. Little is known about the physiology of these species as it is difficult to replicate a deep-sea environment. Biologists have, however, been able to examine the scales (dermal denticles) that cover the shark's skin. Recently, the denticles have been found to have microscopic structure to them [1], see Figure 7. Evolutionary arguments theorize the structure of these scales assists the sharks' movement possibly indicating some type of underwater drag reduction.

The question of active control in sharks is even more of a mystery. It is speculated [11] that sharks can actively move their denticles. The indirect evidence for this is the denticles are connected to muscles underneath the shark's skin. Furthermore, the number of mechano-receptive pressure sensors (pit organs) on a shark's body is positively correlated with the speed of the species. Whether sharks actually utilize active control remains unanswered. However from this example of biology, it may be beneficial to employ microscopic actively controlled structures to reduce drag.

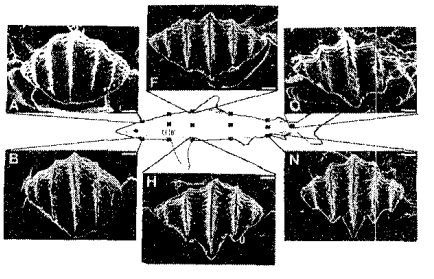

Figure 7. Shark skin microscopic structures.

5.2: Silicon implementation

For a typical airflow of 15 m/s in the wind tunnel, the Reynolds number is approximately 10^4 which corresponds to vortex streaks with statistical mean widths of about 1 mm. The length of a typical vortex streak can be about 2 cm resulting in a 20:1 aspect ratio for the streaks. The average spacing between streaks is about 2.5

mm and the frequency of appearance is about 100 Hz. Recent advances in micromachining have developed microsensors and actuators that can operate at these small scales [9, 6, 12]. Integrating these sensors and actuators with analog VLSI control circuitry (neural or otherwise) provides an exciting new direction for possible drag reduction [4]. One concern with any physical system that employs adaptive schemes is the amount of parameter variation required for different operating regimes.

To investigate this question, we constructed a single weight linear network with functional form:

$$v_{wall} = W_0 \left(\frac{\widetilde{\partial^2 w}}{\partial y \partial z} \right)_N \tag{9}$$

and trained it off-line minimizing the error at each time step. Figure 8 plots the value of W_0 at each time step for two different error scale factors (k in equation 5). We observe that the gain has a larger variation for larger scale factors. On-line training of similar networks shows that larger scale factors produce less fluctuation about the mean drag reduction (even though the average drag reduction is roughly the same for different scale factors). Therefore, there appears to be a trade-off between the amount of fluctuation in the drag and the amount of parameter variation required.

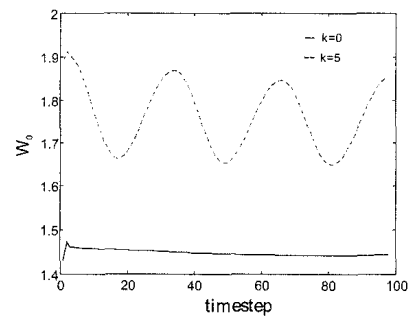

Figure 8. Gain variation for various error scale factors.

6: Remarks

We have presented a neural network that learns a function nearly identical to an analytic control law. The actuation is proportional to the approximate spanwise derivative of the spanwise shear stress. We have also demonstrated the ability of a network that explicitly computes the approximate derivative to achieve 20% drag reduction in an on-line adaptive inverse control scheme with only gain and bias adaptation. New technological advances provide the opportunity to implement such a controller in a real system provided the system can achieve the necessary range of parameter variation. Preliminary results indicate that greater parameter variation may be required if less drag fluctuation is desired.

7: Acknowledgments

This work is supported in part by the Center for Neuromorphic Systems Engineering as a part of the National Science Foundation Engineering Research Center Program under grant EEC-9402726; and by the California Trade and Commerce Agency, Office of Strategic Technology under grant C94-0165. This work is also supported in part by ARPA/ONR under grant no. N00014-93-1-0990, and by an AFOSR University Research Initiative grant no. F4962093-1-0332. Computer time has been supplied by the San Diego Supercomputer Center and by the NAS program at NASA Ames Research Center.

References

[1] D.W. Bechert, G. Hoppe, and W.-E. Reif. On the drag reduction of the shark skin. AIAA paper No. 85-0546, 1985.

[2] H. Choi, P. Moin, and J. Kim. Turbulent drag reduction: Studies of feedback control and flow over riblets. Technical Report TF-55, Stanford University, 1992.

[3] H. Choi, P. Moin, and J. Kim. Active turbulence control for drag reduction in wall-bounded flows. *Journal of Fluid Mechanics*, 262, 1994.

[4] B. Gupta, R. Goodman, F. Jiang, Y.C. Tai, S. Tung, and C.M. Ho. Analog VLSI system for active drag reduction. In *Fifth International Conference on Microelectronics for Neural Networks and Fuzzy Systems*, pages 45–51, Lausanne, Switzerland, 1996.

[5] S.A. Jacobson and W.C. Reynolds. Active control of boundary layer wall shear stress using self-learning neural networks. In *AIAA Shear Flow Conference*, pages 1–12, Orlando, 1993.

[6] F. Jiang, Y.C. Tai, J.B. Huang, and C.M. Ho. Polysilicon structures for shear stress sensors. In *Tech. Digest IEEE TENCON'95*, Hong Kong, November 1995.

[7] J. Kim, P. Moin, and R. Moser. Turbulence statistics in fully-developed channel flow at low Reynolds number. *Journal of Fluid Mechanics*, 177:133–166, 1987.

[8] C. Lee, J. Kim, D. Babcock, and R. Goodman. Application of neural networks to turbulence control for drag reduction. In *Flow Control Workshop*, Cargese, Corsica, France, 1996.

[9] C. Liu, Y.C. Tai, J.B. Huang, and C.M. Ho. Surface micromachined thermal shear stress sensor. In *ASME Application of Microfabrication to Fluid Mechanics*, pages 9–15, Chicago, 1994.

[10] M. Moller. *Efficient Training of Feed-Forward Neural Networks*. PhD thesis, Aarhus University, Denmark, 1993.

[11] W.-E. Reif and A. Dinkelacker. Hydrodynamics of the squamation in fast-swimming sharks. *Neues Jahrb. Geol. Paläontol. Abh.*, 164:184–187, 1982.

[12] T. Tsao, C. Liu, Y.C. Tai, and C.M. Ho. Micromachined magnetic actuator for active fluid control. In *ASME Application of Microfabrication to Fluid Mechanics*, pages 31–38, Chicago, 1994.

[13] B. Widrow. Adaptive inverse control. In *Second IFAC Workshop on Adaptive Systems in Control and Signal Processing*, pages 1–5, Lund, Sweden, 1986.