01 Apr 2009

A Snapshot Algorithm for Linear Feedback Flow Control Design

Benjamin T. Dickinson

Belinda A. Batten

John R. Singler
Missouri University of Science and Technology, singlerj@mst.edu

Recommended Citation

B. T. Dickinson et al., "A Snapshot Algorithm for Linear Feedback Flow Control Design," *AIAA Unmanned, Unlimited Conference*, American Institute of Aeronautics and Astronautics (AIAA), Apr 2009.
The definitive version is available at https://doi.org/10.2514/6.2009-1961

A Snapshot Algorithm for Linear Feedback Flow Control Design

Benjamin T. Dickinson[*] and Belinda A. Batten[†]

Oregon State University, Corvallis, OR, 97331

John R. Singler[‡]

Missouri University of Science and Technology, Rolla, MO 65409

The control of fluid flows has many applications. For micro air vehicles, integrated flow control designs could enhance flight stability by mitigating the effect of destabilizing air flows in their low Reynolds number regimes. However, computing model based feedback control designs can be challenging due to high dimensional discretized flow models. In this work, we investigate the use of a snapshot algorithm proposed in Ref. 1 to approximate the feedback gain operator for a linear incompressible unsteady flow problem on a bounded domain. The main component of the algorithm is obtaining solution snapshots of certain linear flow problems. Numerical results for the example flow problem show convergence of the feedback gains.

I. Introduction

Controlling fluid flows has many potential applications. For example, robust feedback control of the air flow around micro air vehicles could lead to enhanced flight performance, stability, and maneuverability. Recent research has shown that a linear feedback controller (or a nonlinear extension thereof) has the potential to delay or even eliminate the onset of turbulence (e.g., see Refs. 2–11). Furthermore, there is evidence that it is beneficial to use a linear feedback controller as a nominal stabilizing controller, which is then extended to further treat nonlinear effects (see, e.g., Refs. 12–15).

In this work, we consider the problem of computing an optimal feedback control law for a linear incompressible flow problem on a bounded domain. The spatial discretization of flow problems often leads to a very large system of equations. Standard algorithms to compute the feedback control gain are only feasible for small systems of equations. Much recent research has focused on solving the resulting large-scale matrix equations (see, e.g., Ref. 16 and the references therein), however there are still many difficulties and open questions. First, approximating discretization matrices needed for existing numerical algorithms can be difficult (if not impossible) to extract from existing simulation code. Also, the incompressibility condition requires special numerical methods. Little is known about how such methods affect the convergence of existing control gain algorithms as the computational mesh is refined. Furthermore, there is no known method to adaptively refine the mesh to ensure accuracy.

An alternate approach to computing feedback control laws for distributed parameter systems is to first reduce the model and then solve the resulting low order matrix equation to construct the feedback gain. Proper orthogonal decomposition is a model reduction procedure that has been used for this purpose (see, e.g., 12, 15, 17–20), however there are no known guarantees of accuracy or convergence for feedback gain computations.

We investigate the use of a snapshot algorithm proposed in Ref. 1 to approximate the feedback gain operator for a linear flow problem. The algorithm is related to snapshot-based balanced model reduction methods proposed by Wilcox and Peraire[21] and Rowley[22] for finite dimensional systems. The main computational cost of the algorithm is computing solution snapshots of linear unsteady flow problems. These

[*] Graduate Student, School of Mechanical Industrial and Manufacturing Engineering, AIAA Student Member.

[†] Professor and School Head, School of Mechanical Industrial and Manufacturing Engineering, AIAA Member.

[‡] Assistant Professor, Department of Mathematics and Statistics.

computations can be performed with existing software and one can also take advantage of existing techniques such as special discretization schemes, domain decomposition methods, adaptive mesh refinement, and parallel algorithms. Also, since the algorithm is based on simulation data, we bypass the potential difficulty of extracting matrices from existing simulation code.

The snapshot algorithm is also similar in spirit to computing feedback gains for infinite dimensional control problems using the Chandrasekhar equations (see, e.g., Refs. 23–26), which are a nonlinear system of differential equations whose solution approaches the gain when integrated to steady state. In contrast, the snapshot algorithm considered here computes the gain using a sequence of *linear* differential equations arising from the Lyapunov equations in the Newton-Kleinman iteration for the relevant operator Riccati equation. As discussed in Ref. 27, it can be difficult to compute the gain accurately when integrating the Chandrasekhar equations to steady state; therefore, we expect that the snapshot algorithm discussed here may be preferable for many problems. We note however that the Chandrasekhar equations have been used to compute feedback gains for linear flow problems (see, e.g., Ref. 23); also, they can be used to provide a good stabilizing initial guess for the Newton-Kleinman iteration.[27]

The remainder of this work proceeds as follows. We begin by describing the linear unsteady flow control problem. In Section III, we discuss the snapshot algorithm to compute the feedback gain and its implementation for the flow problem. We then present numerical results in Section IV, and close with conclusions and avenues for future work.

II. Problem Description

We consider the control of an unsteady Stokes flow in a lid driven cavity with an open bottom. The equations of motion are given by

$$v_t = -\nabla p + \mu \Delta v + b\, u(t), \quad \nabla \cdot v = 0, \tag{1}$$

where $v = [v_1(t,x,y), v_2(t,x,y)]^T$ is the flow velocity vector, $p = p(t,x,y)$ is the pressure, $b = [b_1(x,y), b_2(x,y)]^T$ is a given control distribution function, and $u(t)$ is a scalar control input. We consider the following boundary and initial conditions

$$\begin{aligned} v_1 = 1,\ v_2 = 0 \qquad & \text{on} \qquad \Gamma_t \times (0,T], \\ v = 0 \qquad & \text{on} \qquad \Gamma_{l,r} \times (0,T], \\ -p\, n + \mu \frac{\partial v}{\partial n} = 0 \qquad & \text{on} \qquad \Gamma_b \times (0,T], \\ v(0,x,y) = v_0(x,y) \qquad & \text{in} \qquad \Omega, \end{aligned} \tag{2}$$

where Ω is the unit square $[0,1] \times [0,1]$, and Γ_t, Γ_l, Γ_r, and Γ_b denote the top, left, right, and bottom walls of Ω. Also, n denotes the outward normal to the boundary.

The goal of the control is to bring the state $(v,\, p)$ to an equilibrium state, $(V,\, P)$, faster than would occur in the uncontrolled system. We assume the equilibrium state $(V,\, P)$ is a solution of (1), and we define the velocity and pressure fluctuations by $(v',\, p') = (v - V,\, p - P)$. Making this change of variables transforms the above Stokes flow system (1) and (2) to the following "fluctuation" Stokes equations

$$v'_t = -\nabla p' + \mu \Delta v' + b\, u(t), \quad \nabla \cdot v' = 0, \tag{3}$$

with boundary and initial conditions

$$\begin{aligned} v' = 0 \qquad & \text{on} \qquad \Gamma \times (0,T], \\ -p\, n + \mu \frac{\partial v'}{\partial n} = 0 \qquad & \text{on} \qquad \Gamma_b \times (0,T], \\ v'(0,x,y) = v'_0(x,y) \qquad & \text{in} \qquad \Omega, \end{aligned} \tag{4}$$

where Γ is the union of Γ_t, Γ_l, and Γ_r.

II.A. An Abstract Formulation

For the control problem and algorithm considered below, we place the above fluctuation Stokes problem in an abstract form. Our presentation follows Ref. 28, which considers the Dirichlet problem. See Ref. 29 for variational formulations of the related Navier-Stokes equations with both Dirichlet and outflow boundary conditions.

First, we define the function spaces relevant to the problem. Let $L^2(\Omega)$ be the Hilbert space of square integrable vector-valued functions over Ω with standard inner product

$$(f, g) = \int_\Omega f(x, y) \cdot g(x, y)\, dx\, dy,$$

and corresponding norm $\|f\| = (f, f)^{1/2}$. Define X to be the Hilbert space of weakly divergence free functions (with the above L^2 inner product and norm) given by

$$X = \left\{ f \in L^2(\Omega) \,:\, \nabla \cdot f = 0 \text{ in } \Omega,\ f \cdot n = 0 \text{ on } \Gamma \right\}.$$

Also let $H^m(\Omega)$ be the Hilbert space of functions in $L^2(\Omega)$ with m distributional derivatives that are all square integrable. Finally, let V be the Hilbert space

$$V = \left\{ f \in X \,:\, f \in H^1(\Omega),\ f = 0 \text{ on } \Gamma \right\},$$

with inner product $(f, g)_V = \sum(\nabla f_i, \nabla g_i)$ and norm $\|f\|_V = (f, f)_V^{1/2}$.

Now we place the fluctuation Stokes system (3) and (4) in a variational form. Taking the inner product of the fluctuation Stokes equations (3) with any vector φ in V gives

$$\frac{\partial}{\partial t}(v', \varphi) = -\mu(v', \varphi)_V + (b, \varphi)u(t).$$

This can be derived by integrating by parts as follows:

$$-(\nabla p, \varphi) + \mu(\Delta v', \varphi) = \int_{\partial\Omega} \left(-pn + \mu \frac{\partial v'}{\partial n} \right) \cdot \varphi \, dx\, dy + (p, \nabla \cdot \varphi) - \mu(v', \varphi)_V = -\mu(v', \varphi)_V.$$

The boundary integral is zero since φ is zero on Γ and due to the boundary condition on Γ_b in Eq. (4); the term $(p, \nabla \cdot \varphi)$ must also be zero since φ is in V and therefore must be divergence free.

Define the operator $A : D(A) \subset X \to X$ as follows:

$$Af = g \quad \text{if} \quad (g, \varphi) = -\mu(f, \varphi)_V \quad \text{for all } \varphi \in V.$$

Here, the set $D(A)$ consists of all functions f in V so that Af is in X. Roughly, for $f \in D(A)$, Af is the projection of $\mu\Delta f$ onto X, and functions in $D(A)$ are twice differentiable, divergence free, and satisfy the boundary conditions of the fluctuations Stokes problem. The control input operator $B : \mathbb{R} \to X$ is defined by

$$[Bu](x, y) = b(x, y)u.$$

With these operators, the above fluctuation Stokes system (3) and (4) can be written abstractly as the following differential equation over the Hilbert space X:

$$\dot{w}(t) = Aw(t) + Bu(t), \quad w(0) = w_0, \tag{5}$$

where $w(t) = v'(t, \cdot, \cdot)$ is a function in X for each t.

II.B. The Control Problem

Now we consider a specific control objective, namely to find $u \in L^2(0, \infty)$ that minimizes the cost function

$$J = \int_0^\infty [Dw]^2(t) + u^2(t)\, dt, \tag{6}$$

where $w(t)$ satisfies the abstract fluctuation Stokes system (5). Here, the controlled output operator $D : X \to \mathbb{R}$ is defined by $Dw = (w, d)$, where d is a state weighting vector in X.

Under certain assumptions, the solution to the above LQR problem is given by the feedback control law

$$u(t) = -K\, w(t), \quad K = B^*\Pi, \tag{7}$$

where $\Pi : X \to X$ is the minimal, nonnegative definite, self-adjoint solution of the algebraic Riccati equation

$$A^*\Pi + \Pi A - \Pi BB^*\Pi + D^*D = 0. \tag{8}$$

Here, the asterisk $(^*)$ denotes the Hilbert adjoint operator.

In this work, we focus on computing the feedback gain operator $K : X \to \mathbb{R}$.

III. Computational Approach

We now describe the snapshot algorithm to compute feedback gains for infinite dimensional systems. We provide a description of the snapshot algorithm in a general infinite dimensional setting and then provide implementation details for the current problem.

III.A. A Snapshot Algorithm for Feedback Gains

Consider the approximation of the feedback gain operator $K = B^*\Pi$, where $\Pi : X \to X$ is the solution of the algebraic Riccati equation (8). We consider the following general framework. Let X be a Hilbert space X with real-valued inner product $(\cdot,\cdot)$ and corresponding norm $\|x\| = (x,x)^{1/2}$. Assume the operator $A : D(A) \subset X \to X$ generates a C_0-semigroup, and the control input operator $B : \mathbb{R}^m \to X$, and the controlled output operator $D : X \to \mathbb{R}^p$ are both bounded and finite rank.

The assumptions on B and D imply that the operators must take the form

$$Bu = \sum_{j=1}^{m} u_j b_j, \quad Dx = [\,(x,d_1),\ldots,(x,d_p)\,]^T,$$

for some vectors $b_1,\ldots,b_m$ and $d_1,\ldots,d_p$ in X (see [30, Theorem 6.1]). For simplicity we focus on the case of a single input and single output, i.e., $m=1$ and $p=1$; the algorithms are easily modified for $m>1$ and $p>1$. As with most large-scale algorithms for feedback control gain computations, the snapshot algorithms require m and p to be relatively small.

For the case $m=1$, we have $Bu = bu$ where b is a vector in X. This assumption implies that the feedback operator $K : X \to \mathbb{R}$ given by $K = B^*\Pi$ has the representation $Kx = (x,k)$, where $k = \Pi b$ is a vector in X known as a functional gain. This representation holds since $B^*x = (x,b)$ and therefore $Kx = B^*\Pi x = (\Pi x, b) = (x, \Pi b)$, since Π is self-adjoint. Below, we concentrate on approximating this functional gain.

We first apply a Newton-Kleinman iteration as modified by Banks and Ito[27] to obtain a sequence of Lyapunov equations. The solutions to the Lyapunov equations are then approximated using a snapshot algorithm. The details are as follows.

Modified Newton-Kleinman iteration[27] for the algebraic Riccati equation (8)

1. Chose an initial guess K_0 so that $A - BK_0$ generates an exponentially stable C_0-semigroup.

2. Compute $K_1 = B^*S_0$, where S_0 solves the Lyapunov equation

$$(A - BK_0)^*S_0 + S_0(A - BK_0) + K_0^*K_0 + C^*C = 0.$$

3. For $i = 1$ until convergence, compute $K_{i+1} = K_i - B^*S_i$, where S_i solves the Lyapunov equation

$$(A - BK_i)^*S_i + S_i(A - BK_i) + E_i^*E_i = 0, \tag{9}$$

and $E_i = K_i - K_{i-1}$.

This algorithm is a reformulation of the standard Newton-Kleinman iteration, which is known to converge with a quadratic rate for the class of infinite dimensional problems considered here.[31]

In the above modified Newton-Kleinman iterations, we need to compute $K_1 = B^*S_0$ and $K_{i+1} = K_i - B^*S_i$ for $i \geq 1$. In the same manner as above, these operators can be represented as follows: $K_ix = (x,k_i)$, where $k_1 = S_0b$ and $k_{i+1} = k_i - S_ib$ for $i \geq 1$. Therefore, in each iteration we do not need to compute the entire Lyapunov solution S_i, we only need the product S_ib. We compute this product using a snapshot algorithm below.

Consider a general infinite dimensional Lyapunov equation

$$A^*S + SA + C^*C = 0, \tag{10}$$

where we assume $C : X \to \mathbb{R}$ is given by $Cx = (x,c)$ with $c \in X$. It is well known that the solution $S : X \to X$ is given by

$$Sx = \int_0^\infty e^{A^*t}C^*Ce^{At}x\,dt.$$

Using the above representation of C, it can be shown[1, 32] that the solution may also be represented by

$$Sx = \int_0^\infty (x, z(t))z(t)\, dt, \tag{11}$$

where $z(t) = e^{A^* t}c$ is the solution of the infinite dimensional linear differential equation

$$\dot{z}(t) = A^* z(t), \quad z(0) = c. \tag{12}$$

This representation leads to the following snapshot algorithm.

Snapshot algorithm[1, 32] to approximate Sx, where S solves the Lyapunov equation (10)

1. Compute an approximation $z^N(t)$ of the solution $z(t)$ of the differential equation (12).
2. Replace $z(t)$ with $z^N(t)$ in the integral representation of Sx in (11) and approximate the integral (by quadrature or some other method).

In Ref. 1 it is shown that if $\int_0^\infty \|z^N(t) - z(t)\|^2\, dt \to 0$, then the resulting approximation converges to Sx .

The approximate solution $z^N(t)$ of the differential equation (12) need not be stored to approximate Sx. Instead, a time stepping method can be used to approximate the differential equation and the approximation to the integral can be updated while simultaneously integrating the differential equation. For example, using a piecewise linear approximation to $z(t)$ in time leads to the trapezoid rule to time step the differential equation and the following approximation to the integral.

Trapezoid snapshot algorithm[1] to approximate Sx, where S solves the Lyapunov equation (10)

1. Approximate the solution of the differential equation (12) with the trapezoid rule:

$$(I - \Delta t A^*/2)z_{n+1} = (I + \Delta t A^*/2)z_n,$$

where I is the identity operator.

2. Update the approximation to Sx:

$$\begin{aligned}[Sx]_{n+1} = [Sx]_n &+ \Delta t[(x, z_{n+1})/3 + (x, z_n)/6]z_{n+1} \\ &+ \Delta t[(x, z_{n+1})/6 + (x, z_n)/3]z_n.\end{aligned}$$

This updating procedure can be stopped when the norm of the update to Sx (possibly unscaled by Δt) is below a certain tolerance. We note that we used a constant time step for simplicity; this is not necessary in general.

For the computations presented below, we used a "stabilized" trapezoid rule[33, 34] which starts with two backward Euler steps and continues with the standard trapezoid rule. For the two backward Euler steps, we updated Sx as follows:

$$\begin{aligned}[Sx]_1 &= \Delta t(x, z_1)z_1, & (I - \Delta t A^*)z_1 &= c, \\ [Sx]_2 &= [Sx]_1 + \Delta t(x, z_2)z_2, & (I - \Delta t A^*)z_2 &= z_1.\end{aligned}$$

III.B. Implementation Details for the Stokes Control Problem

To use the above snapshot algorithm to approximate the solution of the Lyapunov equations (9) arising in the modified Newton iteration for the Riccati equation, we must approximate differential equations of the form

$$\dot{z}(t) = (A - BK)^* z(t), \quad z(0) = z_0.$$

We now present details on approximating the solution of this differential equation in the context of the above Stokes problem. Approximating the solution can be done using a variety of methods; here, we first discretize in time using the trapezoid rule and then discretize in space using a mixed finite element method.

For the above Stokes problem, A is the Stokes operator, B is the control input operator given by $Bu = bu$ for $b \in X$, and K is of the form $Kx = (x, k)$ for some $k \in X$. Since $A = A^*$, we have $(A - BK)^* = A - K^* B^*$, where $K^*u = ku$ and $B^*x = (x, b)$. Therefore, the above abstract differential equation is a representation of the following partial differential equation

$$z_t = -\nabla q + \mu \Delta z - k(z, b), \quad \nabla \cdot z = 0, \tag{13}$$

with boundary conditions

$$z = 0 \text{ on } \Gamma, \quad -qn + \mu \frac{\partial z}{\partial n} = 0 \text{ on } \Gamma_b. \tag{14}$$

As described above, we use the trapezoid rule for the time integration to obtain

$$[I - (\Delta t/2)(A - BK)^*]\, z_n = [I + (\Delta t/2)(A - BK)^*]\, z_{n-1},$$

where $z_n \approx z(t_n)$. This can be rewritten as

$$(A_s - B_s K_s)\, z_n = [I + (\Delta t/2)(A - BK)^*]\, z_{n-1},$$

where $A_s = I - (\Delta t/2)A^*$, $B_s = -(\Delta t/2)K_i^*$, and $K_s = B^*$. Then

$$z_n = (A_s - B_s K_s)^{-1} g, \quad g = [I + (\Delta t/2)(A - BK)^*]\, z_{n-1}.$$

To compute this inverse, we formally apply the Sherman-Morrison-Woodbury formula (see, e.g., Ref. 35):

$$(A_s - B_s K_s)^{-1} g = (I + A_s^{-1} B_s (I - K_s A_s^{-1} B_s)^{-1} K_s) A_s^{-1} g.$$

Since $B^*x = (x, b)$ and $K^*u = ku$, the above inverse can be computed once we approximate $A_s^{-1}g$ and $A_s^{-1}k$. Thus, we need to solve the problems $A_s y_i = f_i$, for $i = 1, 2$, where $f_1 = g$ and $f_2 = k$. In the context of the above Stokes problem, these abstract steady problems take the form

$$y_i - \frac{\Delta t}{2}\big(-\nabla p_i + \mu \Delta y_i \big) = f_i, \quad \nabla \cdot y_i = 0, \tag{15}$$

with boundary conditions

$$y_i = 0 \text{ on } \Gamma, \quad -p_i n + \mu \frac{\partial y_i}{\partial n} = 0 \text{ on } \Gamma_b, \tag{16}$$

where $f_2 = k$, and $f_1 = g$ is given by

$$f_1 = g = [I + (\Delta t/2)(A - K^* B^*)]\, z_{n-1} = z_{n-1} + \frac{\Delta t}{2}\big(-\nabla q_{n-1} + \mu \Delta z_{n-1} - k(z_{n-1}, b) \big). \tag{17}$$

Here, $q_{n-1} \approx q(t_{n-1})$ and $q = q(t, x, y)$ is the pressure in the above PDE (13) and (14).

For the spatial discretization of the above steady problems, we used a mixed formulation. The approximate pressures will be constructed in the Hilbert space $X_0 = L^2(\Omega)$ of scalar-valued square integrable functions. The approximate velocities will be in the Hilbert space of vector-valued functions V_0 defined by

$$V_0 = \left\{ f \in H^1(\Omega) \,:\, f = 0 \text{ on } \Gamma \right\}.$$

Note that unlike the function space V considered in Section II.A, the vector-valued functions in the space V_0 are not required to be weakly divergence free.

The above steady problem (15) and (16) can be formulated weakly as follows: Find $y_i \in V_0$ and $p_i \in X_0$ such that

$$(y_i, \phi) - \frac{\Delta t}{2}\big((p_i, \nabla \cdot \psi) - \mu (y_i, \phi)_V \big) = (f_i, \psi), \quad (\nabla \cdot y_i, \chi) = 0,$$

for all $\psi \in V_0$ and all $\chi \in X_0$. Here, $(\cdot, \cdot)$ denotes the scalar-valued or vector-valued L^2 inner product, and $(\cdot, \cdot)_V$ denotes the V inner product defined in Section II.A. Also, recall $f_2 = k$, and for $f_1 = g$ we reformulate (f_1, ψ) weakly using Eq. (17) as follows:

$$(z_{n-1}, \psi) + \frac{\Delta t}{2}\big((q_{n-1}, \nabla \cdot \psi) - \mu (z_{n-1}, \psi)_V - (k, \psi)(z_{n-1}, b) \big).$$

The above variational problems were discretized with the Taylor-Hood finite element pair. This finite element pair satisfies the inf-sup condition, is second order accurate in the velocity variables, and is first order accurate in the pressure variables.

IV. Numerical Results

For the numerical experiments of the Stokes flow problem we set $\mu = 1$ and applied control to the bottom half of the domain in the horizontal velocity component by taking

$$b_1(x,y) = \begin{cases} 100, & \text{for } y \leq 0.5 \\ 0, & \text{otherwise} \end{cases}, \quad \text{and} \quad b_2(x,y) \equiv 0.$$

In the performance index, the state weight function d was also applied to the bottom half of the domain:

$$d_1(x,y) = d_2(x,y) = \begin{cases} 5, & \text{for } y \leq 0.5, \\ 0, & \text{otherwise.} \end{cases}$$

All computations were performed in FreeFem++, a free two-dimensional finite element package available online.[36] The cavity domain was discretized with a uniform triangulation containing 32 elements in each coordinate direction. This corresponds to 4225 and 1089 nodes in the velocity and pressure grids for a total of 9539 degrees of freedom. We set the time step size to $\Delta t = 10^{-4}$. The tolerance for convergence of the modified Newton-Kleinman algorithm and the snapshot Lyapunov solution were both set to 10^{-4}. For the initial Newton iteration, we chose initial guess $K_0 = 0$.

Six Newton iterations were required for convergence of the functional gain. The number of time steps required for the corresponding snapshot Lyapunov solution is listed in Table 1.

Table 1. Lyapunov iteration number and time steps for convergence of corresponding Lyapunov solutions

Lyap. Iter.	Time Steps
1	1426
2	302
3	368
4	279
5	33
6	2

Figures 1 and 2 contain contour plots of the functional gain for the horizontal and vertical velocity components, respectively. We demonstrated the convergence of the functional gain by repeating the above experiment on a grid of 64 elements in each coordinate direction (for a total of 37507 degrees of freedom) with a time step of 10^{-5}. The resulting functional gain changed on the order of 10^{-2} by measure of the global relative norm.

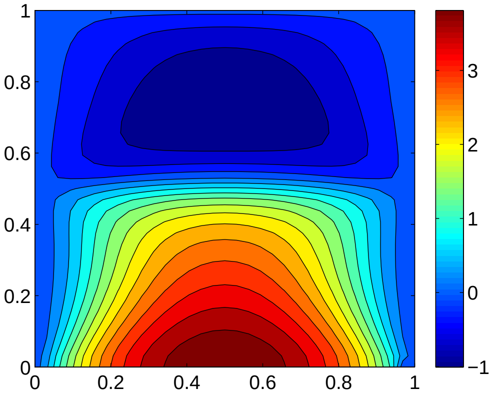

Figure 1. Functional gain for horizontal velocity, k_1

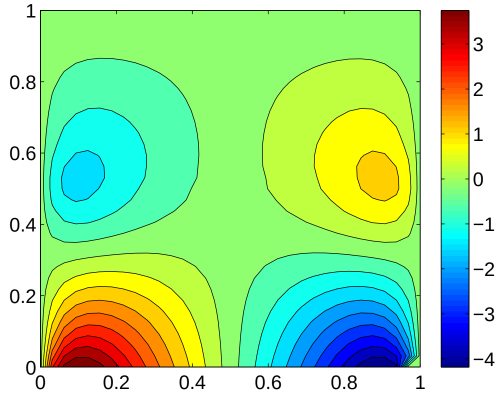

Figure 2. Functional gain for vertical velocity, k_2

We note that the small time step was likely required due to the nonsmooth nature of the functions b and d. An adaptive time stepping algorithm may be advantageous to use for these computations. This will be considered in future work. Also, numerical experiments on a less complex partial differential equation control problem showed that the computational speed could be improved with a good initial guess K_0 to the Newton-Kleinman iteration.[37] One can use the result of one Newton iteration as the initial guess in another Newton iteration with a finer spatial grid (see, e.g., Ref. 38) or time step for the snapshot algorithm. Also, as mentioned in the introduction, the Chandrasekhar equations can also be used to provide an initial guess.

V. Summary

We determined the feedback control gain operator for a linear incompressible flow problem using a snapshot Lyapunov equation solver in conjunction with a modified Newton-Kleinman iteration for the operator Riccati equation. The main computational cost of this approach was the numerical approximation of solutions of linear unsteady flow problems. With a sufficiently refined grid and time step, the algorithm produced a converged functional gain for the linear flow problem.

This preliminary work was intended as a proof-of-principle for computing control operators for linear flow problems without using matrix approximations of the infinite dimensional operators. In future work we will consider the performance of the closed loop system. Preliminary numerical experiments show that, as expected, the solution of the closed loop system is regulated to the equilibrium flow faster than the uncontrolled system. Other remaining problems are to consider control inputs on the boundary, include sensor measurements, and develop robust low order feedback controllers for the linearized Navier-Stokes equations.

Acknowledgments

This research is supported in part by the Air Force Office of Scientific Research through grants FA9550-05-1-0041 and FA9550-07-1-0540.

References

[1] Singler, J. R., "Convergent snapshot algorithms for infinite dimensional Lyapunov equations," submitted for publication.

[2] Barbu, V., Lasiecka, I., and Triggiani, R., "Tangential Boundary Stabilization of Navier-Stokes Equations," *Mem. Amer. Math. Soc.*, Vol. 181, No. 852, 2006.

[3] Burns, J. A. and Singler, J. R., "Feedback control of low dimensional models of transition to turbulence," *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference*, 2005, pp. 3140 – 3145.

[4] Burns, J. A. and Singler, J. R., "Modeling transition: New scenarios, system sensitivity and feedback control," *Transition and Turbulence Control*, World Scientific, 2006, pp. 1–37.

[5] Cortelezzi, L. and Speyer, J., "Robust reduced-order controller of laminar boundary layer transitions," *Physical Review E.*, Vol. 58, No. 2-A, 1998, pp. 1906 – 1910.

[6] Högberg, M., Bewley, T. R., and Henningson, D. S., "Linear feedback control and estimation of transition in plane channel flow," *J. Fluid Mech.*, Vol. 481, 2003, pp. 149–175.

[7] Kim, J., "Control of turbulent boundary layers," *Phys. Fluids*, Vol. 15, No. 5, 2003, pp. 1093–1105.

[8] Kim, J. and Bewley, T. R., "A linear systems approach to flow control," *Annual Review of Fluid Mechanics. Vol. 39*, Annual Reviews, Palo Alto, CA, 2007, pp. 383–417.

[9] Lee, K., Cortelezzi, L., Kim, J., and Speyer, J., "Application of reduced-order controller to turbulent flows for drag reduction," *Physics of Fluids*, Vol. 13, No. 5, 2001, pp. 1321 – 1330.

[10] Or, A., Cortelezzi, L., and Speyer, J., "Robust feedback control of Rayleigh-Benard convection," *J. Fluid Mech.*, Vol. 437, 2001, pp. 175 – 202.

[11] Raymond, J.-P., "Feedback boundary stabilization of the two-dimensional Navier-Stokes equations," *SIAM J. Control Optim.*, Vol. 45, No. 3, 2006, pp. 790–828.

[12] Atwell, J. A., Borggaard, J. T., and King, B. B., "Reduced order controllers for Burgers' equation with a nonlinear observer," *Int. J. Appl. Math. Comput. Sci.*, Vol. 11, No. 6, 2001, pp. 1311–1330.

[13] Burns, J. A. and King, B. B., "A reduced basis approach to the design of low-order feedback controllers for nonlinear continuous systems," *J. Vib. Control*, Vol. 4, No. 3, 1998, pp. 297–323.

[14] Burns, J. A., King, B. B., and Rubio, D., "Feedback control of a thermal fluid using state estimation," *Int. J. Comput. Fluid Dyn.*, Vol. 11, No. 1-2, 1998, pp. 93–112.

[15] Lee, C. H. and Tran, H. T., "Reduced-order-based feedback control of the Kuramoto-Sivashinsky equation," *J. Comput. Appl. Math.*, Vol. 173, No. 1, 2005, pp. 1–19.

[16] Simoncini, V., "A new iterative method for solving large-scale Lyapunov matrix equations," *SIAM J. Sci. Comput.*, Vol. 29, No. 3, 2007, pp. 1268–1288.

[17] Atwell, J. A. and King, B. B., "Proper orthogonal decomposition for reduced basis feedback controllers for parabolic equations," *Math. Comput. Modelling*, Vol. 33, No. 1-3, 2001, pp. 1–19.

[18] Atwell, J. A. and King, B. B., "Reduced order controllers for spatially distributed systems via proper orthogonal decomposition," *SIAM J. Sci. Comput.*, Vol. 26, No. 1, 2004, pp. 128–151.

[19] Banks, H. T., del Rosario, R. C. H., and Smith, R. C., "Reduced-order model feedback control design: numerical implementation in a thin shell model," *IEEE Trans. Automat. Control*, Vol. 45, No. 7, 2000, pp. 1312–1324.

[20] Kepler, G. M., Tran, H. T., and Banks, H. T., "Reduced order model compensator control of species transport in a CVD reactor," *Optimal Control Appl. Methods*, Vol. 21, No. 4, 2000, pp. 143–160.

[21] Willcox, K. and Peraire, J., "Balanced model reduction via the proper orthogonal decomposition," *AIAA Journal*, Vol. 40, No. 11, 2002, pp. 2323–2330.

[22] Rowley, C. W., "Model reduction for fluids, using balanced proper orthogonal decomposition," *Internat. J. Bifur. Chaos Appl. Sci. Engrg.*, Vol. 15, No. 3, 2005, pp. 997–1013.

[23] Borggaard, J., Burns, J., and Zietsman, L., "Computational challenges in control of partial differential equations," *Proceedings of the 2nd AIAA Flow Control Conference*, 2004, AIAA 2004-2526.

[24] Burns, J. A. and Hulsing, K. P., "Numerical methods for approximating functional gains in LQR boundary control problems," *Math. Comput. Modelling*, Vol. 33, No. 1-3, 2001, pp. 89–100.

[25] Ito, K. and Powers, R. K., "Chandrasekhar equations for infinite-dimensional systems," *SIAM J. Control Optim.*, Vol. 25, No. 3, 1987, pp. 596–611.

[26] Ito, K. and Powers, R. K., "Chandrasekhar equations for infinite-dimensional systems. II. Unbounded input and output case," *J. Differential Equations*, Vol. 75, No. 2, 1988, pp. 371–402.

[27] Banks, H. T. and Ito, K., "A numerical algorithm for optimal feedback gains in high-dimensional linear quadratic regulator problems," *SIAM J. Control Optim.*, Vol. 29, No. 3, 1991, pp. 499–515.

[28] Ito, K. and Kappel, F., "The Trotter-Kato theorem and approximation of PDEs," *Math. Comp.*, Vol. 67, No. 221, 1998, pp. 21–44.

[29] Heywood, J. G., Rannacher, R., and Turek, S., "Artificial boundaries and flux and pressure conditions for the incompressible Navier-Stokes equations," *Internat. J. Numer. Methods Fluids*, Vol. 22, No. 5, 1996, pp. 325–352.

[30] Weidmann, J., *Linear Operators in Hilbert Spaces*, Springer-Verlag, New York, 1980.

[31] Burns, J. A., Sachs, E. W., and Zietsman, L., "Mesh independence of Kleinman–Newton iterations for Riccati equations in Hilbert space," *SIAM J. Control Optim.*, Vol. 47, No. 5, 2008, pp. 2663–2692.

[32] Singler, J. R., "Approximate low rank solutions of Lyapunov equations via proper orthogonal decomposition," *Proceedings of the 2008 American Control Conference*, 2008, pp. 267–272.

[33] Rannacher, R., "Finite element solution of diffusion problems with irregular data," *Numer. Math.*, Vol. 43, No. 2, 1984, pp. 309–327.

[34] Hansbo, A., "Nonsmooth data error estimates for damped single step methods for parabolic equations in Banach space," *Calcolo*, Vol. 36, No. 2, 1999, pp. 75–101.

[35] Golub, G. H. and Van Loan, C. F., *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 1996.

[36] Hecht, F., Pironneau, O., Le Hyaric, A., and Ohtsuka, K., *FreeFem++*, 2008, http://www.freefem.org/ff++.

[37] Singler, J. R. and Batten, B. A., "A comparison of balanced truncation methods for closed loop systems," *Proceedings of the American Control Conference*, 2009.

[38] Morris, K. and Navasca, C., "Solution of algebraic Riccati equations arising in control of partial differential equations," *Control and Boundary Analysis*, Chapman & Hall/CRC, Boca Raton, FL, 2005, pp. 257–280.