INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2004; **45**:21–42 (DOI: 10.1002/fld.651)

Numerical approximation of optimal control of unsteady flows using SQP and time decomposition

S. S. Ravindran*,†

Department of Mathematical Sciences, The University of Alabama in Huntsville, Alabama, AL 35899, U.S.A.

SUMMARY

In this paper, we present numerical approximations of optimal control of unsteady flow problems using sequential quadratic programming method (SQP) and time domain decomposition. The SQP method is considered superior due to its fast convergence and its ability to take advantage of existing numerical techniques for fluid flow problems. It iteratively solves a sequence of linear quadratic optimal control problems converging to the solution of the non-linear optimal control problem. The solution to the linear quadratic problem is characterized by the Karush–Kuhn–Tucker (KKT) optimality system which in the present context is a formidable system to solve. As a remedy various time domain decompositions, inexact SQP implementations and block iterative methods to solve the KKT systems are examined. Numerical results are presented showing the efficiency and feasibility of the algorithms. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: optimal control; time domain decomposition; iterative methods; inexact SQP; flowcontrol

1. INTRODUCTION

Optimal control of unsteady flows is critical to the design and performance of fluid dynamical systems. The invention of micro electro mechanical systems (MEMS) and other micro-devices has generated considerable interest in active control of fluid dynamical systems. The application of MEMS technology to control the boundary layer over aerodynamic surfaces has the potential of accomplishing high maneuverability of aircraft with larger flight envelope. Synthetic jets and suction/injection devices distributed over the wings of the aircraft can be used to modify the flow field in the boundary layer and prevent separation and transition to turbulence, and thus can modify the lift and drag characteristics of the

*Correspondence to: S. S. Ravindran, Department of Mathematical Sciences, University of Alabama in Huntsville, Huntsville, Alabama 35899, U.S.A.
†E-mail: ravindra@ultra.math.uah.edu, ultra.math.uah.edu/ravindra

Received 18 February 2003
Revised 7 August 2003

aerodynamic surfaces. These effects could result in reduced fuel consumption and increased range.

The past two decades have seen significant progress in control of computational fluid dynamics; see References [1–6 and 29] and the references therein. Experimental results using micro actuators and sensors have also demonstrated the potential of control of fluids to increase the performance of high-lift wings, cavity noise and other flow systems [7, 8]. Despite this progress, numerical computation of optimal control for unsteady flows remains a challenging problem. Optimal control of fluid flow is a non-linear optimization problem that is constrained by the Navier–Stokes equations that govern the underlying fluid flow. Therefore solving the optimal control problem can be orders of magnitude more costlier computationally than solving the Navier–Stokes equations.

Numerical methods for solving optimal control problems can be divided into two classes depending on whether the state variables are considered as an independent variables or function of control variables. In the former case the Navier–Stokes equation is an explicit constraint in the non-linear optimization (e.g. References [3, 4, 9, 10]) whereas in the latter the Navier–Stokes equations merely act to describe the state as a function of control and therefore the resulting problem is that of unconstrained minimization (e.g. References [11, 12]). The latter can be very inefficient as the simulation problem has to be solved at each optimization iteration. In the former case, however, one can design sequential quadratic programming (SQP) algorithms that simultaneously converge to the state and control variables.

The SQP methods are second order and considered superior for non-linear optimal control problems; see for e.g. References [13, 14] for the finite dimensional case and References [3, 4, 9, 10, 15] for the infinite dimensional case. The SQP methods solve non-linear optimal control problem by a sequence of linear quadratic subproblems. In other words the SQP method requires satisfaction of only a linear approximation of the state constraint avoiding the need to converge fully. Thus, the state equations are satisfied as the control values converge to their optimal values. Its application to solve optimal control of steady Navier–Stokes flow has been demonstrated in References [3, 4, 10, 16, 17] and [9]. The numerical resolution of the optimal control of unsteady flow problem requires solving the Navier–Stokes equations with an initial condition and the adjoint equations with a final condition. This abnormal coupling makes the unsteady flow control problems challenging in terms of both the storage and computer time required to carry out the simulations.

In this article, we present numerical approximations of optimal control of unsteady flows using SQP method and time domain decomposition. The linear quadratic subproblem that arises during the SQP iteration is solved by solving the corresponding first order necessary condition of optimality known as the Karush–Kuhn–Tucker (KKT) condition. Efficient solution of KKT system of partial differential equations is one of the important factors affecting the robustness of the SQP algorithm. We will investigate several time-domain decomposition approaches and inexact SQP implementation in this article. We present numerical implementation and results for two unsteady control problems, a channel flow and a cavity flow, when the control is velocity (suction/blowing) on part of the boundary.

The article is organized as follows. In Section 2 we state the optimal control problem. The SQP methods for solving the optimal control problems associated with the unsteady Navier–Stokes equations are presented in Section 3. Time decomposition approaches for efficient implementation of the SQP methods are discussed in Section 4. Section 5 reports numerical experiments of the methods and approaches, followed by conclusion in Section 6.

2. OPTIMAL CONTROL PROBLEMS

In optimal control problems, the objective of interest is represented as a cost functional to be minimized. In the test problems to be discussed in this article, the control $\mathbf{g}$ is to be applied to either track a desired flow field or to reduce the size of wake spread in the flow domain over the time horizon $[0, T]$ using the least amount of control effort possible. A relevant cost functional for such a task can be written in a generic form as

$$\mathscr{T}(\mathbf{u}, \mathbf{g}) = \int_0^T \mathscr{F}(\mathbf{u})\, \mathrm{d}t + \frac{\gamma}{2} \int_0^T \int_{\Gamma_c} |\mathbf{g}|^2\, \mathrm{d}\mathbf{x}\, \mathrm{d}t$$

The first term in the above cost functional is exactly that quantity we would like to minimize and the second term is a measure of the magnitude of the control which is included in the cost functional to limit the size of the control. The parameter γ is a positive constant that adjust the relative weight of the two terms in the functional.

The optimal control problem is to find the control $\mathbf{g}$ that minimizes the cost functional $\mathscr{T}$ subject to the constraint that the fluid obeys the equations of motion and conservation of mass, namely, the Navier–Stokes equations

$$\begin{aligned}
&\mathbf{u}_t - \frac{1}{Re} \nabla^2 \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{0} \quad \text{in } \Omega \times (0, T] \\
&\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times (0, T] \\
&\mathbf{u} = 0 \quad \text{on } \Gamma_u \times [0, T], \qquad \mathbf{u} = \mathbf{g} \quad \text{on } \Gamma_c \times [0, T] \\
&\text{and} \quad \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \quad \text{in } \Omega
\end{aligned} \tag{1}$$

where Ω is the flow domain with boundary $\Gamma = \Gamma_u \cup \Gamma_c$. Moreover the velocity $\mathbf{u}$, the pressure p, the time t and the spatial variable $\mathbf{x}$ are in non-dimensional form. The Reynolds' number Re is defined as $Re = \rho u_{ave} L/\mu$, where ρ is the density, u_{ave} is the average inflow velocity, L is the characteristic length and μ is the kinematic viscosity. The flow is controlled through suction and blowing on the boundary which takes the form $\mathbf{u} = \mathbf{g}$ on $\Gamma_c \times [0, T]$.

For spatial discretization, we will use a mixed finite-element method. For this we need a weak form of the Navier–Stokes equation (1). A weak formulation is given by: find $\mathbf{u} \in L^2(0, T; \mathbf{H}^1(\Omega))$, $\mathbf{u}|_{\Gamma_u} = 0$ and $p \in L^2(\Omega)$ that satisfies

$$\begin{aligned}
&(\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u}, \mathbf{w}) + \frac{1}{Re}(\nabla \mathbf{u}, \nabla \mathbf{w}) - (p, \nabla \cdot \mathbf{w}) + \frac{1}{\varepsilon}(\mathbf{u} - \mathbf{g}, \mathbf{w})_{\Gamma_c} = 0 \quad \forall \mathbf{w} \in \mathbf{V} \\
&(\nabla \cdot \mathbf{u}, q) = 0 \quad \forall q \in L^2(\Omega) \\
&\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \quad \text{for } \mathbf{x} \in \Omega
\end{aligned} \tag{2}$$

where $\mathbf{V} = \{\mathbf{w} | \mathbf{w} \in \mathbf{H}^1(\Omega),\ \mathbf{w}|_{\Gamma_u} = 0\}$. In the above weak-form, we employed a penalty method and reformulated our boundary condition for control as $-p\mathbf{n} + (1/Re)(\partial\mathbf{u}/\partial\mathbf{n}) + \mathbf{u}/\varepsilon = \mathbf{g}/\varepsilon$ on $\Gamma_c \times [0, T]$ which reduces to the original form as $\varepsilon \to 0$; see [3, 4].

The optimal control problems we consider can be described in a general manner as follows:

seek the boundary velocity control $\mathbf{g}$ and state pair $(\mathbf{u}, \mathbf{p})$ such

that the cost functional $\mathscr{T}(\cdot, \cdot)$ is minimized subject to the constraint (P)

that the flow field satisfy the Navier–Stokes equations (2) over $[0, T]$.

3. SQP METHOD FOR UNSTEADY OPTIMAL CONTROL

The SQP method solves a quadratic subproblem at each iteration. Each subproblem minimizes a quadratic model of a modified Lagrangian subject to linearized constraints. A globalization strategy is used to guarantee convergence from initial guess. The SQP method does not satisfy the non-linear constraints at each iteration except as the optimal solution is approached. We will in the sequel denote the state variables by $\mathbf{v}$, the adjoint variables by $\boldsymbol{\lambda}$, the current iterate by $(\mathbf{v}^c, \boldsymbol{\lambda}^c) = (\mathbf{u}^c, p^c, \mathbf{g}^c, \boldsymbol{\mu}^c, \sigma^c)$, the new iterate by $(\mathbf{v}^+, \boldsymbol{\lambda}^+) = (\mathbf{u}^+, p^+, \mathbf{g}^+, \boldsymbol{\mu}^+, \sigma^+)$. Moreover, the Lagrangian $\mathscr{L}$ of the optimal control problem is given by

$$\mathscr{L} = \mathscr{T}(\mathbf{u}, \mathbf{g}) - (\mathbf{u}_t + \mathbf{u} \cdot \nabla\mathbf{u}, \boldsymbol{\mu}) - \frac{1}{Re}(\nabla\mathbf{u}, \nabla\boldsymbol{\mu}) + (p, \nabla \cdot \boldsymbol{\mu}) - \frac{1}{\varepsilon}(\mathbf{u} - \mathbf{g}, \boldsymbol{\mu})_{\Gamma_c} - (\nabla \cdot \mathbf{u}, \sigma)$$

and its first and second derivatives, respectively, are given by

$$\mathscr{L}_{\mathbf{v}}(\mathbf{v}^c, \boldsymbol{\lambda}^c)[\Delta_{\mathbf{v}}], \quad \mathscr{L}_{\mathbf{v},\mathbf{v}}(\mathbf{v}^c, \boldsymbol{\lambda}^c)[\Delta_{\mathbf{v}}, \Delta_{\mathbf{v}}]$$

where $\Delta_{\mathbf{v}} = \mathbf{v} - \mathbf{v}^c$ is the search direction. In the SQP method the new iterate is computed as the solution of the following quadratic subproblem (QP):

$$\text{Minimize}\{\nabla_{\mathbf{v}}\mathscr{T}(\mathbf{v}^c)[\Delta_{\mathbf{v}}] + \tfrac{1}{2}\mathscr{L}_{\mathbf{vv}}(\mathbf{v}^c, \boldsymbol{\lambda}^c)[\Delta_{\mathbf{v}}, \Delta_{\mathbf{v}}]\}$$

subject to the linearized state equations

$$(\mathbf{u}_t + \mathbf{u}^c \cdot \nabla\mathbf{u} + \mathbf{u} \cdot \nabla\mathbf{u}^c, \mathbf{w}) + \frac{1}{Re}(\nabla\mathbf{u}, \nabla\mathbf{w}) - (p, \nabla \cdot \mathbf{w}) + \frac{1}{\varepsilon}(\mathbf{u} - \mathbf{g}, \mathbf{w})_{\Gamma_c}$$
$$= (\mathbf{u}^c \cdot \nabla\mathbf{u}^c, \mathbf{w}) \quad \forall \mathbf{w} \in \mathbf{V} \tag{3}$$

$$(\nabla \cdot \mathbf{u}, q) = 0 \quad \forall q \in L^2(\Omega)$$

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \quad \text{for } \mathbf{x} \in \Omega$$

The statement of the SQP method is given in Algorithm 1.

Algorithm 1 [SQP]

Given $(\mathbf{v}^{(0)}, \boldsymbol{\lambda}^{(0)})$ sufficiently close to the solution $(\mathbf{v}^*, \boldsymbol{\lambda}^*)$ and set $k = 0$

Step 1. Form and solve the quadratic subproblem (QP) to determine $\Delta_{\mathbf{v}}^{(k)}$ and let $\boldsymbol{\lambda}^{(k+1)}$ be the corresponding optimal Lagrange multipliers.

Step 2. Set $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta_{\mathbf{v}}^{(k)}$.

Step 3. Stop if converged.

Step 4. Set $k \to k+1$, go to Step 1.

The execution of Step 2, that is a solution to the linear quadratic subproblem (QP) is obtained by solving the following first order necessary conditions of optimality:

$$\mathscr{L}_{\mathbf{v}}^{\mathrm{s}} = 0, \quad \mathscr{L}_{\boldsymbol{\lambda}}^{\mathrm{s}} = 0, \quad \mathscr{L}_{\mathbf{g}}^{\mathrm{s}} = 0$$

where $\mathscr{L}^{\mathrm{s}}$ is the Lagrangian corresponding to the linear quadratic subproblem. Thus, given the current iterate $(\mathbf{v}^{\mathrm{c}}, \boldsymbol{\lambda}^{\mathrm{c}})$, one obtains the new iterate $(\mathbf{v}^{+}, \boldsymbol{\lambda}^{+})$ by solving the system of linearized state equations (3) and

$$-(\boldsymbol{\mu}_t, \mathbf{w}) + \frac{1}{Re}(\nabla\boldsymbol{\mu}, \nabla\mathbf{w}) + (\mathbf{u}^{\mathrm{c}} \cdot \nabla\mathbf{w}, \boldsymbol{\mu} - \boldsymbol{\mu}^{\mathrm{c}}) + (\mathbf{w} \cdot \nabla\mathbf{u}^{\mathrm{c}}, \boldsymbol{\mu} - \boldsymbol{\mu}^{\mathrm{c}}) - (\sigma, \nabla \cdot \mathbf{w}) + (\mathbf{u} \cdot \nabla\mathbf{w}, \boldsymbol{\mu}^{\mathrm{c}})$$

$$+ (\mathbf{w} \cdot \nabla\mathbf{u}, \boldsymbol{\mu}^{\mathrm{c}}) + \frac{1}{\varepsilon}(\boldsymbol{\mu}, \mathbf{w})_{\Gamma_{\mathrm{c}}} = (\mathscr{F}_{\mathbf{u}}(\mathbf{u}^{\mathrm{c}}), \mathbf{w}) + (\mathscr{F}_{\mathbf{uu}}(\mathbf{u}^{\mathrm{c}})(\mathbf{u} - \mathbf{u}^{\mathrm{c}}), \mathbf{w}) \quad \forall \mathbf{w} \in \mathbf{V}$$

$$(\nabla \cdot \boldsymbol{\mu}, q) = 0 \quad \forall q \in L^2(\Omega)$$

$$\gamma(\mathbf{g}, \mathbf{z})_{\Gamma} + \frac{1}{\varepsilon}(\boldsymbol{\mu}, \mathbf{z})_{\Gamma_{\mathrm{c}}} = 0 \quad \forall \mathbf{z} \in L^2(\Gamma_{\mathrm{c}})$$

$$\boldsymbol{\mu} = 0 \quad \text{on } \Gamma_{\mathrm{u}} \times [0, T]$$

$$\boldsymbol{\mu}(\mathbf{x}, T) = 0 \quad \text{in } \Omega$$

The SQP method described above is only locally convergent and its ultimate success depends on how fast the outer iterations converge. Its global convergence is guaranteed only if the Jacobian of the constraints is nonsingular for all the iterates of the optimization variables. But this is too strong a requirement for highly non-linear systems like the Navier–Stokes equations. Commonly used globalization methods include line search and trust region algorithms [18]. The trust region method in the context of SQP methods is discussed in Reference [19]. The line search methods require choosing a good merit function that allows unit step length close to a solution so that the quadratic convergence of the SQP method can be observed. Popular choices for merit function are l_1 merit function with second order corrections and augmented Lagrangian. Another alternative for globalization in the context of optimal control of fluid flow systems is to use continuation on a parameter that scales the non-linearity. Such parameters are common in fluid flow systems: Reynolds number in incompressible flows, Mach number in compressible flows and Hartman number in MHD. Continuation generates good initial

guesses and globalizes quite naturally when the optimal solutions are away from the limit and bifurcations points.

In our computations, we use a continuation method on the Reynolds number as globalization strategy. It proceeds as follows: Set $\eta = 1$ in the SQP algorithm. Let Re_{d} the desired Reynolds number at which we want to solve the optimal control problem. We start with $Re = Re_0$ and define a sequence of optimal control problems by increasing the Reynolds number with increment ΔRe. For the subsequent optimization problems we choose sufficiently small ΔRe such that solution obtained by solving the ith sequence in the optimization problem with Reynolds number defined by $Re_0 + \mathrm{i}\Delta Re$ can be used as initial iterate for the $(i+1)$st problem to achieve convergence. The statement of the resulting globalized algorithm is given in Algorithm 2.

Algorithm 2 [Globalized SQP]

Choose $(\mathbf{v}^{(0)}, \boldsymbol{\lambda}^{(0)})$, $k_{\max}$; set $k = 0$, $Re = Re_0$

Step 1. Form and solve the quadratic subproblem (QP) to determine $\Delta_{\mathbf{v}}^{(k)}$ and let $\boldsymbol{\lambda}^{(k+1)}$ be the corresponding optimal Lagrange multipliers.

Step 2. Set $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta_{\mathbf{v}}^{(k)}$.

Step 3. If converged, stop and go to Step 6.

Step 4. If $k = k_m$ reduce Re and go to Step 1.

Step 5. Set $k \leftarrow k + 1$, go to Step 1.

Step 6. If $Re = Re_{\mathrm{d}}$ stop else set $Re = Re + \Delta Re$ and go to Step 1.

The algorithm uses two level iterations. The outer iteration involves incrementing the continuation parameter until the desired parameter is reached. The inner iteration involves solving the optimality system of the linear quadratic subproblem for a fixed continuation parameter. Due to the large-scale nature of the unsteady flow control problems, the inner iterations cannot be solved by direct methods. In the next section we will look at some time domain decomposition approaches to deal with this issue.

4. TIME DOMAIN DECOMPOSITION TECHNIQUES

The first approach is an iterative method to efficiently solve the optimality system of the linear quadratic subproblem arising in the SQP method. An iterative method similar to the one discussed here has been presented in Reference [20] to solve linear quadratic optimal control problems. The second approach uses a receding horizon idea and replaces the optimal control problem on the full time horizon by a sequence of optimal control problems on short control horizons. The time discretization of the Navier–Stokes equations is carried out using the fully implicit backward Euler method and the spatial discretization uses a mixed finite-element method. Let $\mathscr{K}_h$ be a standard finite element triangulation of Ω, where h is the maximal length of all the triangulation edges in $\mathscr{K}_h$. Let $\mathscr{P}^k$ to be the space of all polynomials of degree less than or equal to k and

$$\mathbf{V}_h = \{\mathbf{v}_h \mid \mathbf{v}_h \in C^0(\bar{\Omega}) \times C^0(\bar{\Omega}),\ \mathbf{v}_h \mid_K \in \mathscr{P}^2 \times \mathscr{P}^2\ \forall K \in \mathscr{K}_h\}$$
$$P_h = \{q_h \mid q_h \in C^0(\bar{\Omega}),\ q_h \mid_K \in \mathscr{P}^1\ \forall K \in \mathscr{K}_h\}$$

4.1. Technique I

The time interval $[0, T]$ is partitioned with a uniform subdivision with step-size $\Delta t = T/N$ as

$$0 = t_0 < t_1 \cdots < t_N = T \qquad (\mathscr{P}_1)$$

Let $\mathbf{u}^n$ denote the approximation of $\mathbf{x} \to \mathbf{u}(\mathrm{x}, n\Delta t)$, and similarly $p^n, \mathbf{g}^n$. Approximating the time integral in the cost functional by the right-endpoint rule we get the following fully discrete optimal control problem:

$$\text{Minimize } \mathscr{T}_h(\mathbf{u}_h^n, \mathbf{g}_h^n) = \Delta t \sum_{n=1}^{N} \mathscr{F}(\mathbf{u}_h^n) + \frac{\gamma \Delta t}{2} \sum_{n=1}^{N} \int_{\Gamma_c} |\mathbf{g}_h^n|^2 \, \mathrm{d}\mathbf{x}$$

subject to

$$\left.\begin{aligned} &\left(\frac{\mathbf{u}_h^n - \mathbf{u}_h^{n-1}}{\Delta t} + \mathbf{u}_h^n \cdot \nabla \mathbf{u}_h^n, \mathbf{w}\right) + \frac{1}{Re}(\nabla \mathbf{u}_h^n, \nabla \mathbf{w}) - (p_h^n, \nabla \cdot \mathbf{w}) \\ &\quad + \frac{1}{\varepsilon}(\mathbf{u}_h^n - \mathbf{g}_h^n, \mathbf{w})_{\Gamma_c} = 0 \quad \forall \mathbf{w} \in \mathbf{V}_h \\ &(\nabla \cdot \mathbf{u}_h^n, q) = 0 \quad \forall q \in P_h \end{aligned}\right\} \quad \text{for } n = 1, \ldots, N \qquad (4)$$

Application of Algorithm 2 to the optimal control problem (4) requires computing the solutions of the optimality system for the associated quadratic subproblem. This system can be rearranged to have a block-structure if we eliminate the control variable $\mathbf{g}$ and reorder the other variables as

$$(\mathbf{u}_h^1, p_h^1, \boldsymbol{\mu}_h^1, \sigma_h^1 | \ldots, | \mathbf{u}_h^N, p_h^N, \boldsymbol{\mu}_h^N, \sigma_h^N)^{\mathrm{T}}$$

Thus, given the current iterate $(\mathbf{u}_h^{1,\mathrm{c}}, p_h^{1,\mathrm{c}}, \boldsymbol{\mu}_h^{1,\mathrm{c}}, \sigma_h^{1,\mathrm{c}} | \ldots, | \mathbf{u}_h^{N,\mathrm{c}}, p_h^{N,\mathrm{c}}, \boldsymbol{\mu}_h^{N,\mathrm{c}}, \sigma_h^{N,\mathrm{c}})^{\mathrm{T}}$, one obtains the new iterate $(\mathbf{u}_h^{1,+}, p_h^{1,+}, \boldsymbol{\mu}_h^{1,+}, \sigma_h^{1,+} | \ldots, | \mathbf{u}_h^{N,+}, p_h^{N,+}, \boldsymbol{\mu}_h^{N,+}, \sigma_h^{N,+})^{\mathrm{T}}$, by solving the discrete the optimality system: for $n = 1, \ldots, N$

$$\mathbf{u}_h^0 = \mathbf{u}_{h0}, \quad \boldsymbol{\mu}_h^{N+1} = 0$$

$$\left(\frac{\mathbf{u}_h^n - \mathbf{u}_h^{n-1}}{\Delta t}, \mathbf{w}\right) + \frac{1}{Re}(\nabla \mathbf{u}_h^n, \nabla \mathbf{w}) + (\mathbf{u}_h^{n,\mathrm{c}} \cdot \nabla \mathbf{u}_h^n, \mathbf{w}) + (\mathbf{u}_h^n \cdot \nabla \mathbf{u}_h^{n,\mathrm{c}}, \mathbf{w}) - (p_h^n, \nabla \cdot \mathbf{w})$$

$$+ \frac{1}{\varepsilon}\left(\mathbf{u}_h^n + \frac{1}{\Delta t \gamma \varepsilon} \boldsymbol{\mu}_h^n, \mathbf{w}\right)_{\Gamma_c} = (\mathbf{u}_h^{n,\mathrm{c}} \cdot \nabla \mathbf{u}_h^{n,\mathrm{c}}, \mathbf{w}) \quad \forall \mathbf{w} \in \mathbf{V}_h$$

$$(\nabla \cdot \mathbf{u}_h^n, q) = 0 \quad \forall q \in P_h$$

$$\left(\frac{\boldsymbol{\mu}_h^n - \boldsymbol{\mu}_h^{n+1}}{\Delta t}, \mathbf{w}\right) + \frac{1}{Re}(\nabla \boldsymbol{\mu}_h^n, \nabla \mathbf{w}) + (\mathbf{u}_h^{n,c} \cdot \nabla \mathbf{w}, \boldsymbol{\mu}_h^{n,c} - \boldsymbol{\mu}_h^{n,c}) + (\mathbf{w} \cdot \nabla \mathbf{u}_h^{n,c}, \boldsymbol{\mu}_h^n - \boldsymbol{\mu}_h^{n,c})$$

$$-(\sigma_h^n, \nabla \cdot \mathbf{w}) + (\mathbf{u}_h^n \cdot \nabla \mathbf{w}, \boldsymbol{\mu}_h^{n,c}) + (\mathbf{w} \cdot \nabla \mathbf{u}_h^n, \boldsymbol{\mu}_h^{n,c}) + \frac{1}{\varepsilon}(\boldsymbol{\mu}_h^n, \mathbf{w})_{\Gamma_c}$$

$$= \Delta t(\mathscr{F}_{\mathbf{u}_h^n}(\mathbf{u}_h^{n,c}), \mathbf{w}) + \Delta t(\mathscr{F}_{\mathbf{u}_h^n \mathbf{u}_h^n}(\mathbf{u}_h^{n,c})(\mathbf{u}_h^n - \mathbf{u}_h^{n,c}), \mathbf{w}) \quad \forall \mathbf{w} \in \mathbf{V}_h$$

$$(\nabla \cdot \boldsymbol{\mu}_h^n, q) = 0 \quad \forall q \in P_h$$

Some comments on the structure of the discrete optimality system are in order. This is a formidable system of $(dN_v + N_p)N$ linear equations with $(dN_v + N_p)N$ unknowns, where N_v denotes the number of velocity unknowns, N_p denotes the number of pressure unknowns and d denotes the space dimension. In matrix form, the resulting linear algebraic system $A\mathbf{X} = \mathbf{b}$ can be written as

$$\begin{bmatrix} A_{11} & A_{12} & 0 & \cdots & \cdots & \cdots & 0 \\ A_{21} & A_{22} & A_{23} & \ddots & & & \vdots \\ 0 & A_{32} & A_{33} & A_{34} & \ddots & & \vdots \\ \vdots & \ddots & A_{43} & A_{44} & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & & \ddots & A_{N-1N-2} & A_{N-1N-1} & A_{N-1N} \\ 0 & \cdots & \cdots & \cdots & 0 & A_{NN-1} & A_{NN} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ \vdots \\ \vdots \\ X_{N-1} \\ X_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ \vdots \\ \vdots \\ b_{N-1} \\ b_N \end{bmatrix} \qquad (\mathrm{OSQP}_1)$$

where

$$(\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_N)^{\mathrm{T}} = (\mathbf{U}^1, P^1, M^1, S^1 | \mathbf{U}^2, P^2, M^2, S^2 | \cdots | \mathbf{U}^N, P^N, M^N, S^N)^{\mathrm{T}}$$

and $\mathbf{U}, P, M, S$ denote the nodal values of $\mathbf{u}, p, \boldsymbol{\mu}, \sigma$ in the space-time grid, respectively. The success of Algorithm 2 depends on efficient and fast computation of this linear system of equations. Our approach to solve this formidable linear system is to exploit its block structure and design a Gauss–Seidel iterative method and its variants. An outline of the SQP

Block–Gauss–Seidel algorithm is given by the following:

Algorithm 3:[SQP Block-Gauss-Seidel]
Choose $\mathbf{X}^{(0)}$, $k_{\max}$; set $k=0$, $Re=Re_0$.
Step 1. Form the optimality system ($OSQP_1$) of the quadratic subproblem.
Step 2: Solve ($OSQP_1$) by the block-Gauss–Seidel method:
Step 2(a). Set $m=1$, $X_i^{[0]}=\mathbf{X}_i^{(k)}$, $i=1,2,\ldots,N$.
Step 2(b). Form and solve
$$A_{ii}X_i^{[m]}=b_i-A_{ii-1}X_{i-1}^{[m]}-A_{ii+1}X_{i+1}^{[m-1]},\ i=1,2,\ldots,N$$
by a linear algebraic solver.
Step 2(c). If converged, set $\mathbf{X}^*=\mathbf{X}^{[m]}$ and go to Step 3.
Step 2(d). Set $m\leftarrow m+1$ and go to Step 2(b).
Step 3. Set $\mathbf{X}^{(k+1)}=\mathbf{X}^*$.
Step 4(a). Stop if converged and go to Step 6.
Step 4(b). If $k=k_{\max}$ reduce Re and go to Step 1.
Step 5. Set $k\leftarrow k+1$ and go to Step 1.
Step 6. If $Re=Re_{\mathrm{d}}$ stop else set $Re=Re+\Delta Re$ and go to Step 1.

In our implementation of Algorithm 3, we use a banded Gaussian elimination with partial pivoting in Step 2(b) but of course one can instead use an iterative linear solver. Step 2 of this algorithm can instead be carried out using a block Jacobi type iteration or successive over-relaxation (SOR) type iteration. In a Jacobi iteration one replaces Step 2(b) with

$$A_{ii}X_i^{[m]}=b_i-A_{ii-1}X_{i-1}^{[m-1]}-A_{ii+1}X_{i+1}^{[m-1]},\quad i=1,2,\ldots,N$$

and in SOR iteration one replaces Step 2(b) with

$$\mathbf{X}^*=\omega\mathbf{X}^{[m]}+(1-\omega)\mathbf{X}^{[m-1]}$$

where $0<\omega<2$ is a fixed relaxation parameter. One known drawback of these iterative methods is their slow convergence. However, our numerical implementation reported in Section 5 indicate that they can still be efficient if used in the context of inexact SQP methods. The idea behind the inexact SQP methods is to terminate the iterative linear solver such as Gauss-Seidel or Jacobi before convergence and approximately compute the SQP step.

4.2. Technique II

In the receding horizon control strategies, the optimal control problem (P) is decomposed into sequence of same type of optimal control problems posed on shorter time interval. The time interval $[0,T]$ is first partitioned into

$$0=t_0^*<t_1^*<\cdots<t_N^*=T \qquad (\mathscr{P}_2)$$

and the short time interval optimal control problems are defined by restricting the original control problem to the time intervals $\{t_{i-1}^*,t_i^*]$. These control problems can then be solved sequentially. To compute the optimal control $\hat{g}^i$ in the ith short time interval, we assume the optimal control $\hat{g}^{i-1}$ and the corresponding states $(\hat{\mathbf{u}}^{i-1},\hat{p}^{i-1})$ have been computed. We compute the optimal control $\hat{g}^i$ by applying the SQP numerical optimization technique to the

optimal control problem defined on the interval $[t_{i-1}^*, t_i^*]$. The control (suboptimal) $\hat{g}$ for the original optimal control problem (P) over $[0, T]$ is defined by patching together the piecewise optimal controls $\hat{g}^i,\ i = 1, \ldots, N$. There are two ways to select the partitioning $\mathscr{P}_2$ of the control horizon $[0, T]$. One way is to select the partition ($\mathscr{P}_2$) such that it coincides with the partition ($\mathscr{P}_1$) used for time integration. This particular selection is called 'instantaneous control techniques'; see Reference [21–23]. The other way is to select the partition ($\mathscr{P}_2$) such that $\Delta t^* = t_i^* - t_{i-1}^* > \Delta t = t_i - t_{i-1}$. This selection therefore does not require that the step-size Δt^* be sufficiently small unlike in the instantaneous control case where the step-size $\Delta t = \Delta t^*$ has to be sufficiently small.

The optimal control problem on the short horizon $[t_{i-1}^*, t_i^*]$ is defined as: *find a solution* $(\hat{\mathbf{u}}^i \hat{p}^i, \hat{\mathbf{g}}^i)$ *on the interval* $[t_{i-1}^*, t_i^*]$ *which minimizes the functional*

$$\mathscr{T}_{(t_{i-1}^*, t_i^*)}(\mathbf{u}, \mathbf{g}) = \int_{t_{i-1}^*}^{t_i^*} \mathscr{F}(\mathbf{u})\, \mathrm{d}t + \frac{\gamma}{2} \int_{t_{i-1}^*}^{t_i^*} \int_{\Gamma_c} |\mathbf{g}|^2\, \mathrm{d}\mathbf{x}\, \mathrm{d}t$$

subject to

$$(\mathbf{u}_t, \mathbf{w}) + \frac{1}{Re}(\nabla \mathbf{u}, \nabla \mathbf{w}) + (\mathbf{u} \cdot \nabla \mathbf{u}, \mathbf{w}) - (p, \nabla \mathbf{w})$$

$$+ \frac{1}{\varepsilon}(\mathbf{u} - \mathbf{g}, \mathbf{w})_{\Gamma_c} = \mathbf{0} \quad \forall \mathbf{w} \in \mathbf{V}, \quad \forall t \in (t_{i-1}^*, t_i^*]$$

$$(\nabla \cdot \mathbf{u}, q) = 0 \quad \forall q \in L^2(\Omega), \quad \forall t \in (t_{i-1}^*, t_i^*]$$

$$\mathbf{u}(\mathbf{x}, t_{i-1}) = \hat{\mathbf{u}}^{(i-1)} \quad \text{in } \Omega \tag{5}$$

To solve the optimal control problem (5) for each n, we use the so-called discretize-then-optimize approach. The time integral in the cost functional $\mathscr{T}_{(t_{i-1}^*, t_i^*)}$ is discretized using the right-endpoint rectangle rule with step-size $\Delta t = t_i - t_{i-1} = \Delta t^*/2$ and the Navier–Stokes equations is discretized in time using the backward Euler method with the same step-size Δt. The SQP iterates are computed by solving the optimality system for the associated quadratic subproblem. Spatial discretization of this system is by the finite elements as described in the previous section. The resulting finite dimensional system $A\mathbf{X} = \mathbf{b}$ can be written as

$$\begin{bmatrix} A_{11}^i & A_{12}^i \\ A_{21}^i & A_{22}^i \end{bmatrix} \begin{bmatrix} \mathbf{X}_1^i \\ \mathbf{X}_2^i \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1^i \\ \mathbf{b}_2^i \end{bmatrix} \tag{OSQP$_2$}$$

where

$$(\mathbf{X}_1^i, \mathbf{X}_2^i)^{\mathrm{T}} = (\mathbf{U}^1, P^1, M^1, S^1, \mathbf{U}^2, P^2, M^2, S^2)^{\mathrm{T}}$$

and $\mathbf{U}, P, M, S$ denote the nodal values of $\mathbf{u}, p, \boldsymbol{\mu}, \sigma$, respectively. The above linear system is solved by employing a banded Gaussian elimination with pivoting but of course one can instead use an iterative linear solver. A better, although not ideal approach, may be to use block Gauss–Seidal method. An outline of the SQP-Receding-Horizon algorithm is given by

the following:

Algorithm 4: [*SQP-Receding-Horizon*]
Given the optimal control $\hat{\mathbf{g}}^0$ and the corresponding states $(\hat{\mathbf{u}}^0, \hat{\mathbf{p}}^0)$ and set $i = 0$.
Step 1. Form the optimality system (OSQP_2) of the quadratic subproblem.
Step 2. Solve (OSQP_2) by a linear algebraic solver for $(\mathbf{X}_1^i, \mathbf{X}_2^i)^{\mathrm{T}}$.
Step 3. Stop if $i\Delta t^* = T$.
Step 4. Set $i \leftarrow i + 1$ and go to Step 1.

In order to derive the SQP instantaneous control algorithm one first replaces (OSQP_2) with

$$A^i \mathbf{X}^i = \mathbf{b}^i \qquad (\mathrm{OSQP}_3)$$

where $\mathbf{X}^i = (\mathbf{U}^1, P^1, M^1, S^1)^{\mathrm{T}}$ and $\mathbf{U}, P, M, S$ denote the nodal values of $\mathbf{u}, p, \boldsymbol{\mu}, \sigma$, respectively. Then in Step 2 above we solve (OSQP_3) for $\mathbf{X}^i$.

5. NUMERICAL COMPUTING AND RESULTS

In this section we will present the results of numerical experiments with the techniques described in the previous section. All the simulations presented in this study are performed with finite element discretization in space and finite difference in time for the Navier–Stokes equations in primitive variable form. The finite element grid for the flow domain uses triangles and a triangular grid is generated as follows. The domain is first divided into squares and then each square is subdivided into two triangles by cutting from bottom right to top left. The shape functions for the velocity $\mathbf{v}$ and adjoint velocity $\boldsymbol{\mu}$ are piecewise quadratic polynomials, and that for the pressure p and adjoint pressure σ are piecewise linear polynomials. All four variables are defined on the same triangle and the degrees of freedom for the quadratic elements are the function values at the vertices and midpoints of each edge; the degrees of freedom for linear elements are the function values at the vertices. This choice of finite element spaces have been widely used in simulation (e.g. References [24, 25]) and control [3, 4, 10] of incompressible flow, and compiles the div-stability conditions needed to avoid spurious oscillations.

5.1. Test I

In this test we consider the problem of tracking an unsteady velocity field in a square cavity by controlling the boundary injection/suction of the fluid. This reflects the desire to steer, over time, a candidate velocity field $\mathbf{u}$ to a given target velocity field $\mathbf{u}_d$ by appropriately controlling the velocity along the boundary of the flow domain. Specifically, the control problem solved is as follows: Minimize the cost functional

$$\mathscr{J}(\mathbf{u}, \mathbf{g}) = \frac{\delta}{2} \int_0^T \int_\Omega |\mathbf{u} - \mathbf{u}_d|^2 \, \mathrm{d}\mathbf{x} \, \mathrm{d}t + \frac{\gamma}{2} \int_0^T \int_{\Gamma_c} |\mathbf{g}|^2 \, \mathrm{d}\mathbf{x} \, \mathrm{d}t$$

subject to the constraints (2).

Here we will compare the three algorithms that we proposed: SQP–Gauss–Seidel, SQP–Receding–Horizon and SQP–Instantaneous. In all cases, at $t = 0$ the fluid inside the cavity is

at rest and the control is applied on the boundary to drive the flow field to the target unsteady velocity field

$$\mathbf{u}_d(\mathbf{x},t)=\begin{pmatrix}\sin(2\pi y+\pi t)(\cos(2\pi x)-1)\\ 2\sin(2\pi x)\sin(\pi y+\pi t)\sin(\pi y)\end{pmatrix}$$

The flow domain is taken to be a unit square and the computational grid is non-uniform in both spatial directions. The time domain is $[0,0.5]$ with the time step size of $\Delta t=\frac{1}{40}$ and the parameters ε,δ and γ are $\varepsilon=10^{-4}$, $\delta=10^3$ and $\gamma=1$, respectively. The control $\mathbf{g}$ is a four-sided control and that covers the whole boundary $\Gamma=\Gamma_c$. In Figures 1 and 2 we present the results from the SQP–Gauss–Seidel algorithm implementations. It shows the evolution of the controlled and target flow field. The controlled flow is presented in the first column on the left, the target flow is presented in the second column on the right and all the figures are normalized. As seen in the figures, at $t=0.25$ the controlled flow reaches the optimal flow and follows the motion of the target velocity field. In Figure 3 we present the evolution of the cost functional ($\|\mathbf{u}-\mathbf{u}_d\|$) using both technique I (block Gauss–Seidel) and technique II (receding Horizon and instantaneous). As seen in the graphs, the rate of decrease of the functional is independent of the step-size. This seem to confirm the theoretical results proved in Reference [26]. It also shows that the optimal control from the SQP–Gauss–Seidel algorithm equi-distributes the reduction over the time whereas the other two try to match the desired state at every instant of time. In Table I we present results obtained by using the three different algorithms; specifically we give the values of the cost functional at the final time. For means of comparison of the speed of the algorithms we give the CPU time generated by performing the computations for Reynolds number $Re=20$ on a SUN Ultra60 machine with a 19×19 spatial grid and no continuation. All the implementations used the same grid and linear algebraic solver. The CPU timing listed in the table indicates that the instantaneous approach is the fastest albeit suboptimal. However, the speed of approach I can be improved by implementing an efficient iterative method for the linear solver.

Our computational experiments reported in Table I indicate that among these algorithms the SQP–Gauss–Seidel algorithm being the only genuinely optimal control algorithm gives the best reduction in the cost functional. In Figure 4, the evolution of the cost functional as a function of time during the SQP outer iterations are documented with Gauss–Seidel inner iterations fixed at 5. It can be gleaned from the graphs that there is no significant change after the third outer iteration.

Let us turn our discussion to implementing an inexact SQP method that avoids fully converging to the KKT system during inner iterations. We want to stop the Gauss–Seidel or Jacobi inner iterations prematurely and study its effect on the convergence of SQP iterations with technique I. We do this by presetting the maximum number of Gauss–Seidel or Jacobi iterations to a small fixed number and use the resulting approximate solution in the SQP iterations. In Figures 5 and 6 we present results of this case study for $Re=20$ and for a 10×10 grid with no continuation. These results indicate that only few Gauss–Seidel iterations are enough to get the SQP outer iterations to converge and doing so results in minimum total computational time. It also indicates that as we increase the total number of allowed Gauss–Seidel iterations, the total number of SQP iterations decreases and eventually stays at about 6. The optimal number of iterations required for the Gauss–Seidel and Jacobi inner iterations to achieve minimum CPU time is 1 and 3, respectively; see Figures 5 and 6. This suggests

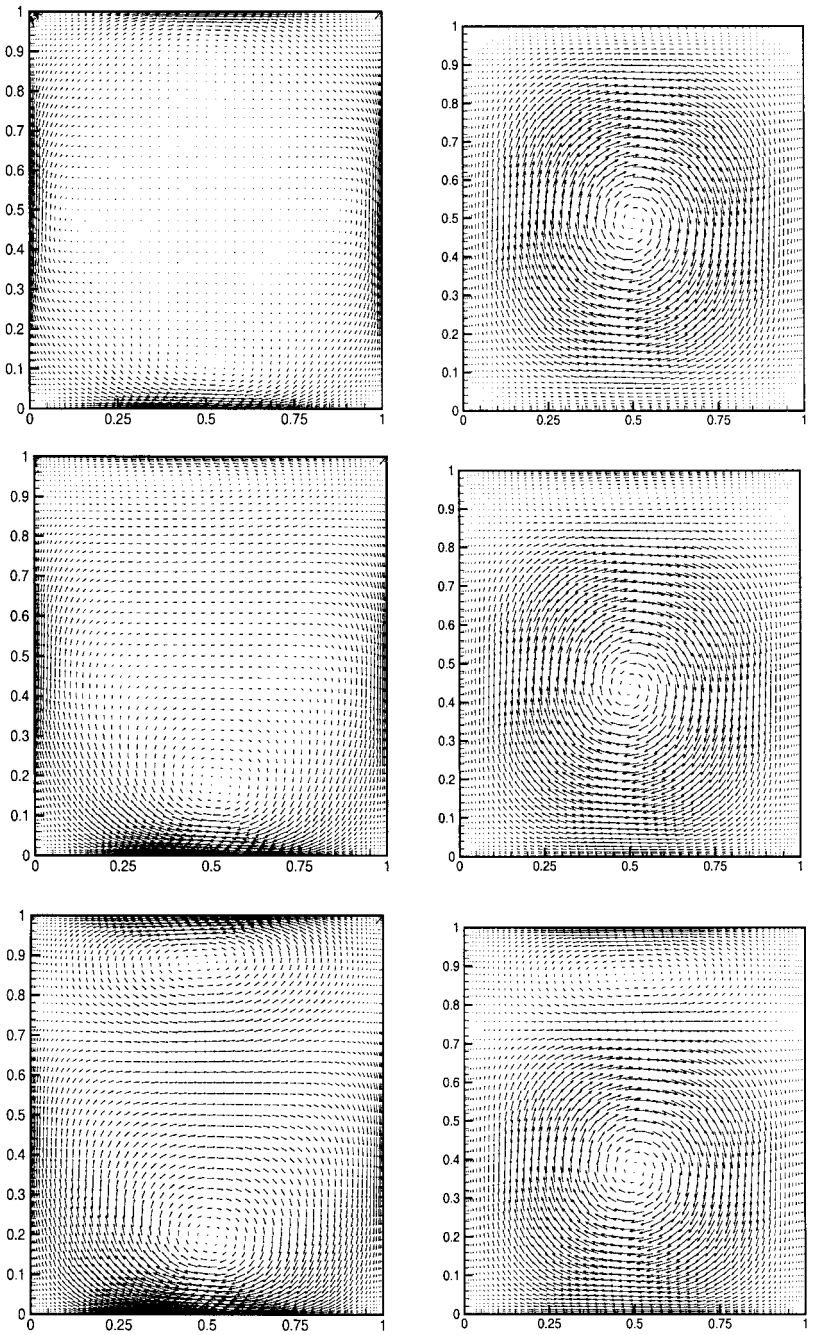

Figure 1. Controlled (first column) and target (second column) flows at $t = 0.025$ (first row), $t = 0.125$ (second row) and $t = 0.25$ (third row).

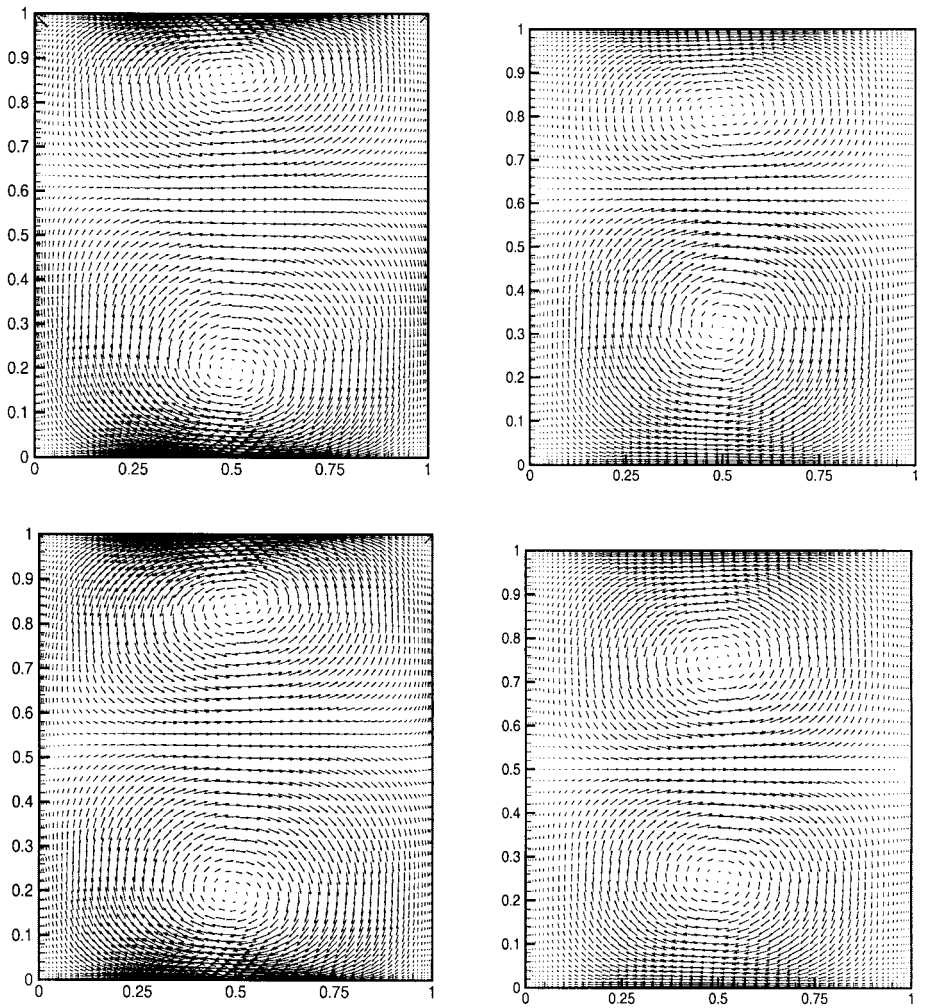

Figure 2. Controlled (first column) and target (second column) flows at $t = 0.375$ (first row) and $t = 0.5$ (second row).

implementing the inexact SQP method that avoids fully converging to the KKT system during the inner iterations is indeed effective.

Table II gives statistics for the high Reynolds number case using the continuation technique. Continuation was used at every Reynolds number listed in the first column starting at $Re = 100$. For example, the solutions with Reynolds number of 50 was used as the initial guess for the optimal control computation with Reynolds number of 100. We report the number of SQP iterations taken to fully converge in the inexact setting by prematurely stopping the Gauss–Seidel iterations; see the last row in Table II. The CPU time listed are in hours and shows again that the inexact SQP implementation reduces the computational time.

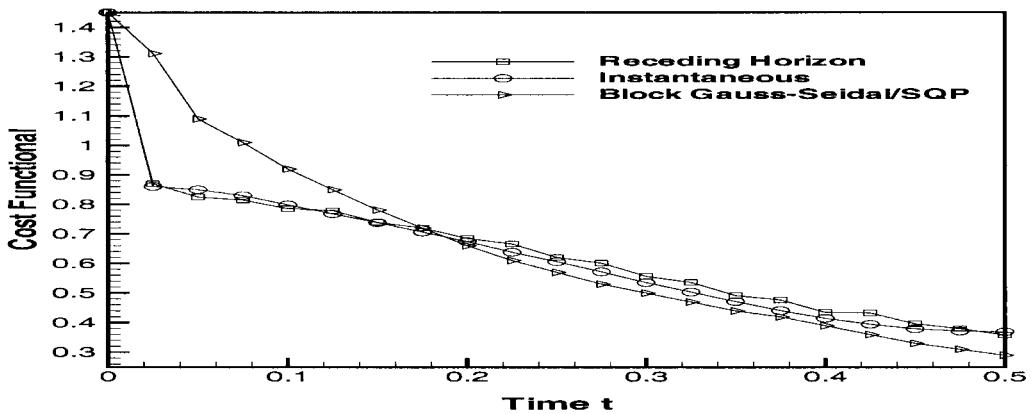

Figure 3. Evolution of cost functional with time t.

Table I. Comparison of optimal control approaches for Test I.

	Receding Horizon	Instantaneous	Block Gauss–Seidal
CPU Time	4.11 h	9 min	45 min
$\|\mathbf{u} - \mathbf{u}_d\|$ at time $\mathbf{T}$	0.358	0.369	0.29

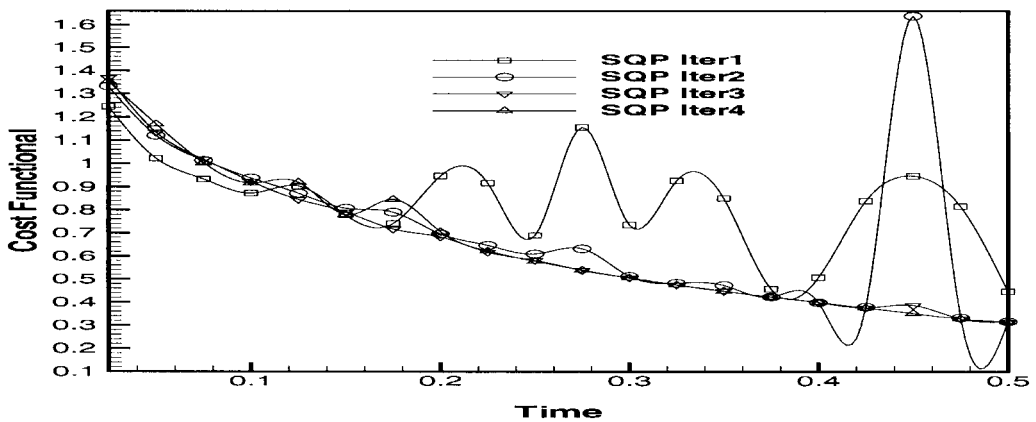

Figure 4. Evolution of cost functional during SQP iterations of approach I.

5.2. Test II

In this test case we consider the problem of reducing the flow separation and wake spread in channel flows using boundary control. Reducing the separation and wake spread in flow domain is of interest in flow past bluff bodies. The flow configuration considered is a backward-facing step channel shown in Figure 7. In the schematic of the geometry, the downstream channel is defined to have unit height L with step height and inlet height $L/2$, and step length

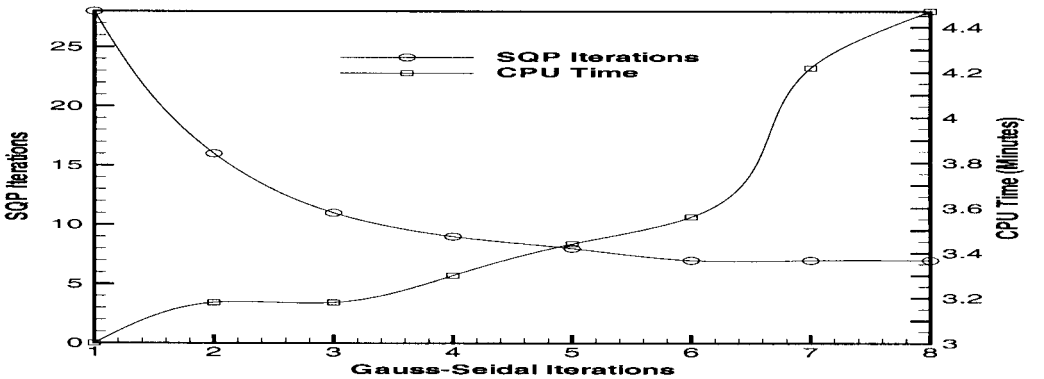

Figure 5. SQP iterations and CPU time as a function of Gauss–Seidel iterations.

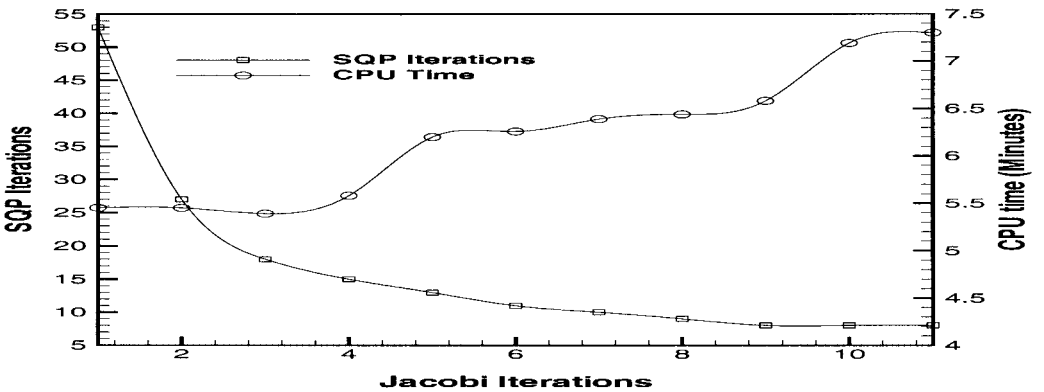

Figure 6. SQP iterations and CPU times as a function of Jacobi iterations.

Table II. Performance of Inexact implementation of the SQP method that avoids fully converging to the KKT system during inner iterations.

Re	sqp iter	sqp iter	sqp iter	sqp iter	sqp iter	sqp iter	sqp iter
50	30	15	12	8	7	6	5
100	25	13	9	7	7	5	6
150	23	12	9	7	6	5	5
200	23	12	8	7	6	5	5
250	22	11	8	7	6	5	4
300	21	10	7	6	5	5	4
350	19	10	7	6	5	4	4
400	19	10	7	6	5	4	4
Total CPU	7.09	7.13	7.25	7.26	7.42	7.27	7.34
GS iter	1	2	3	4	5	6	7

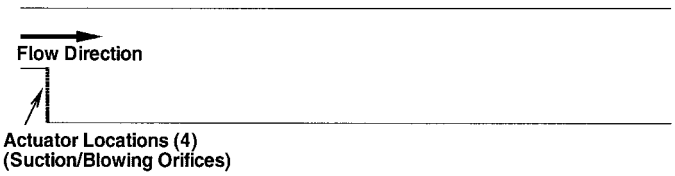

Figure 7. Computational domain for the backward-facing-step channel problem.

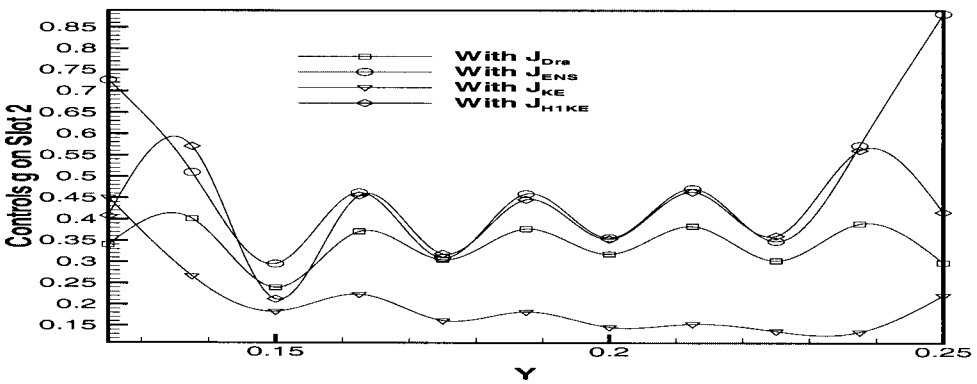

Figure 8. Computed control g for three different cost functionals as a function of y for slot 2 at time $t = 10$.

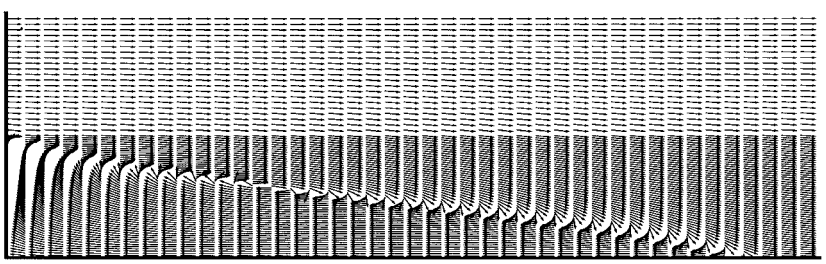

Figure 9. Baseline velocity field behind the step of the channel ($(x, y) \in (L/4, 3) \times (0, L)$).

$L/4$. The downstream channel length was taken as $x = 8L$. At $t = 0$, fluid is injected into the channel through the opening on the left. At the enlargement, the flow velocity is suddenly reduced and as a consequence the pressure is increased (see Figure 8). This causes the fluid particles near the lower wall to separate and form a recirculation bubble, just downstream of the sudden enlargement; see Figure 9. The length of the recirculation region is a function of the channel width ratio and Reynolds number. The objective of the optimal control is to reduce the size of the recirculation and hence the length of re-attachment for a fixed Reynolds number and channel width ratio. The control is effected by wall normal velocity (suction/blowing) on a small slot in the vertical part of the step; see Figure 7.

The choice of cost functional to be minimized to achieve our objective is not trivial. Since our goal is to reduce the recirculation region in the flow, a straight forward choice is to regulate the square of the vorticity (the enstrophy). Alternatively, one can minimize the H^1-norm of the velocity field. Other less direct choices are the minimization of the viscous dissipation or the drag on the surface and the minimization of the kinetic energy. We therefore consider four different cost functionals namely the cost functional for the enstrophy regulation

$$\mathscr{T}_{\mathrm{Ens}}(\mathbf{u},\mathbf{g}) = \frac{1}{2}\int_0^T \int_\Omega \alpha|\nabla \times \mathbf{u}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T \int_{\Gamma_c} |\mathbf{g}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t$$

the cost functional for the kinetic energy

$$\mathscr{T}_{\mathrm{KE}}(\mathbf{u},\mathbf{g}) = \frac{1}{2}\int_0^T \int_\Omega \alpha|\mathbf{u}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T \int_{\Gamma_c} |\mathbf{g}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t$$

the cost functional for the dissipation which for incompressible flow is equivalent to the drag on the surface

$$\mathscr{T}_{\mathrm{Dra}}(\mathbf{u},\mathbf{g}) = \frac{1}{2}\int_0^T \int_\Omega \alpha|\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathrm{T}}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T \int_{\Gamma_c} |\mathbf{g}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t$$

and the cost function for the kinetic energy in H^1-norm

$$\mathscr{T}_{\mathrm{HIKE}}(\mathbf{u},\mathbf{g}) = \frac{1}{2}\int_0^T \int_\Omega \alpha\{|\mathbf{u}|^2 + |\nabla\mathbf{u}|^2\} \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T \int_{\Gamma_c} |\mathbf{g}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t$$

with $\alpha(x, y)$ being a weight function to reduce the physical quantity of interest in the corner region $\Omega^* = (L/4, 2) \times (0, L/2)$ of the channel Ω and defined to be

$$\alpha(\mathbf{x}) = \begin{cases} 10 & \text{if } \mathbf{x} \in \Omega^* \\ 0 & \text{if } \mathbf{x} \in \Omega \backslash \Omega^* \end{cases}$$

The only non-dimensional parameter of interest, the Reynolds' number is defined by $Re = \rho u_{\mathrm{ave}} L/\mu$. At the inflow boundary a parabolic velocity profile is prescribed, i.e. $u(x = 0,\ L/2 \leqslant y \leqslant L) = 24(y - L/2)(L - y)$, $v(x = 0,\ L/2 \leqslant y \leqslant L) = 0$, which produces a maximum inflow velocity of $u_{\max} = \frac{3}{2}$ and an average velocity of $u_{\mathrm{ave}} = 1$. On the solid walls the no-slip condition ($\mathbf{u} = 0$) is imposed. At the outflow, we apply the pseudo stress-free condition [27]

$$-p + \frac{1}{Re}\frac{\partial u}{\partial x} = 0 \quad \text{and} \quad \frac{\partial v}{\partial x} = 0$$

The numerical simulations for this test were performed for the Reynolds number of 200, $\varepsilon = 10^{-4}$, $\gamma = 1.0$ and for the control horizon of $[0, 10]$. The computational grid was uniform in both horizontal and vertical directions with finer grid behind the step where separation bubble forms. We report only the final results with 79×79 grid and a time step size of $\Delta t = \frac{1}{50}$ but calculations with varying mesh sizes have been performed. We implemented the SQP instantaneous algorithm with the continuation technique that uses $\Delta Re = 50$ as the Reynolds number increment.

The optimal location of actuator, where control is applied, is crucial for the effectiveness of control. We address this problem by the following simple approach. We first choose to

place the actuator on the vertical part of the step. This choice is motivated by the fact that if one wants maximum influence in the flow, then the control has to be applied in that vicinity. Then we find where exactly on this part of the channel the actuator has to be placed. To this end we divide this portion of the step Γ_s into four parts of equal size and name them as Slot 1 (Γ_s^1: $0 \leqslant y \leqslant L/8$), Slot 2 ($\Gamma_s^2$: $L/8 \leqslant y \leqslant L/4$), Slot 3 ($\Gamma_s^3$: $L/4 \leqslant y \leqslant 3L/8$) and Slot 4 ($\Gamma_s^4$: $3L/8 \leqslant y \leqslant L/2$) with $\Gamma_s = \bigcup_{i=1}^{4} \Gamma_s^i$. We define the control on these slots as follows

$$\text{Slot 1:} \quad \mathbf{u}(L/4, y) = \begin{cases} (g, 0) & \text{if } y \in \Gamma_s^1 \\ (0, 0) & \text{if } y \in \Gamma_s \backslash \Gamma_s^1 \end{cases}$$

$$\text{Slot 2:} \quad \mathbf{u}(L/4, y) = \begin{cases} (g, 0) & \text{if } y \in \Gamma_s^2 \\ (0, 0) & \text{if } y \in \Gamma_s \backslash \Gamma_s^2 \end{cases}$$

$$\text{Slot 3:} \quad \mathbf{u}(L/4, y) = \begin{cases} (g, 0) & \text{if } y \in \Gamma_s^3 \\ (0, 0) & \text{if } y \in \Gamma_s \backslash \Gamma_s^3 \end{cases}$$

$$\text{Slot 4:} \quad \mathbf{u}(L/4, y) = \begin{cases} (g, 0) & \text{if } y \in \Gamma_s^4 \\ (0, 0) & \text{if } y \in \Gamma_s \backslash \Gamma_s^4 \end{cases}$$

In order to determine the best location for the actuator, we compute the control for each of these cases and compare the results. The optimal control formulation for this test was based on the enstrophy cost functional $\mathscr{T}_{\text{Ens}}$. Computed controls g at various actuator positions indicated that it is blowing on Slot 2 and Slot 3 and suction on Slot 4. The resulting velocity fields with the optimal control action is given in Figure 10. As shown in Figure 10(a) when the control is on Slot 1 the flow separates both at the corner of the step and at the bottom wall further downstream. Although the separation at the corner of the step causes a marginal effect, the separation on the bottom wall causes a substantial separation bubble. The control action on Slot 2 provides the best performance with almost complete suppression of the separation bubble. As indicated by the flow fields in Figure 10(b) the wake spread has been effectively eliminated by the optimal control (blowing) and the re-attachment length has been reduced by more than 99% compared to the uncontrolled case. When the control is on Slot 3, the separation at the corner is eliminated but it separates below Slot 3 with the associated bubble albeit small is seen in Figure 10(c). The control action on Slot 4 effectively eliminates the flow separation at the corner of the step but it makes the flow to separate from the top wall and form a substantial separation bubble on the top wall as shown in Figure 10(d). Therefore the best location for control is on Slot 2 and the optimal control is blowing. These findings are in agreement with our earlier findings reported in Reference [28] using reduced-order control design.

The choice of the cost functional to be minimized is crucial to achieve the control objective. Figure 11 show the performance of the optimal controls based on minimization of $\mathscr{T}_{\text{Ens}}, \mathscr{T}_{\text{KE}}$ and $\mathscr{T}_{\text{Dra}}$, respectively, and when the actuator is on Slot 2. Shown in the figure are u-velocity at various stations downstream of the channel. As seen in the figure, the optimal control based on the $\mathscr{T}_{\text{Ens}}$ formulation performs the best. The optimal control based on the $\mathscr{T}_{\text{HIKE}}$

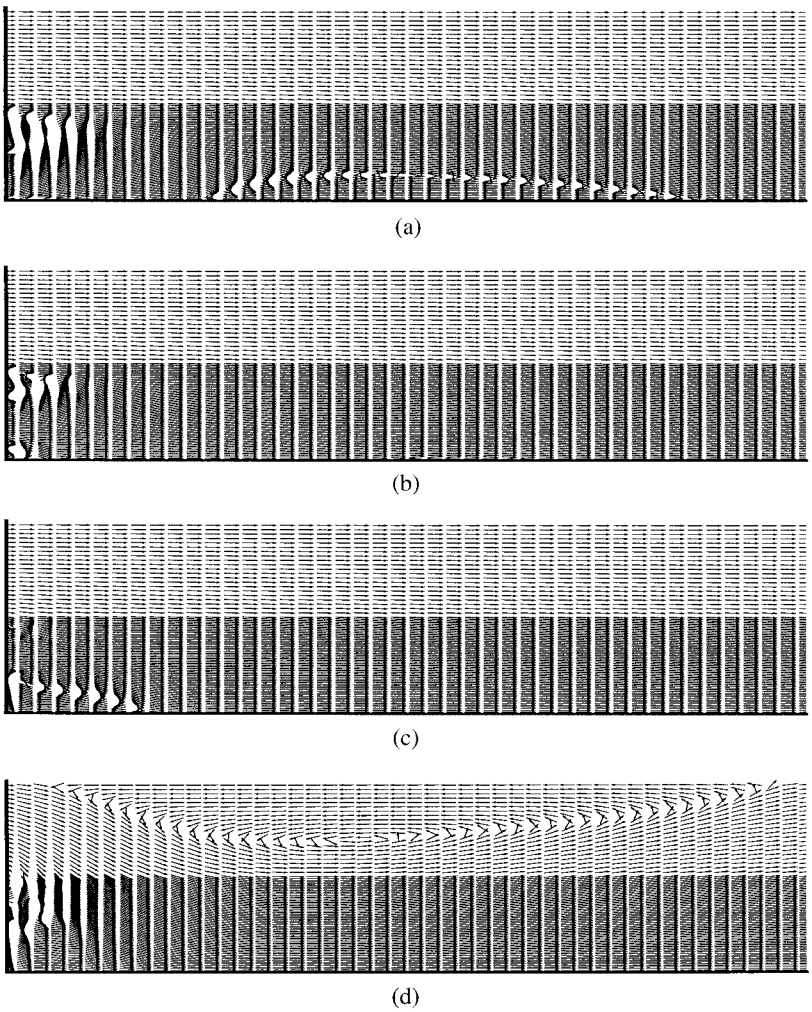

Figure 10. Controlled velocity field behind the step of the channel ($(x, y) \in (L/4, 3) \times (0, L)$). (a) Controlled velocity field when control is on Slot 1. (b) Controlled velocity field when control is on Slot 2. (c) Controlled velocity field when control is on Slot 3. (d) Controlled velocity field when control is on Slot 4.

formulation performs well in reducing the recirculation region, but it is not as good as the one corresponds to enstrophy formulation. As expected, the optimal controls with other two formulations were not effective.

6. CONCLUSION

The optimal control techniques for non-linear unsteady flow control problems are complex and computationally demanding. The SQP method provides a fast and efficient way to solve

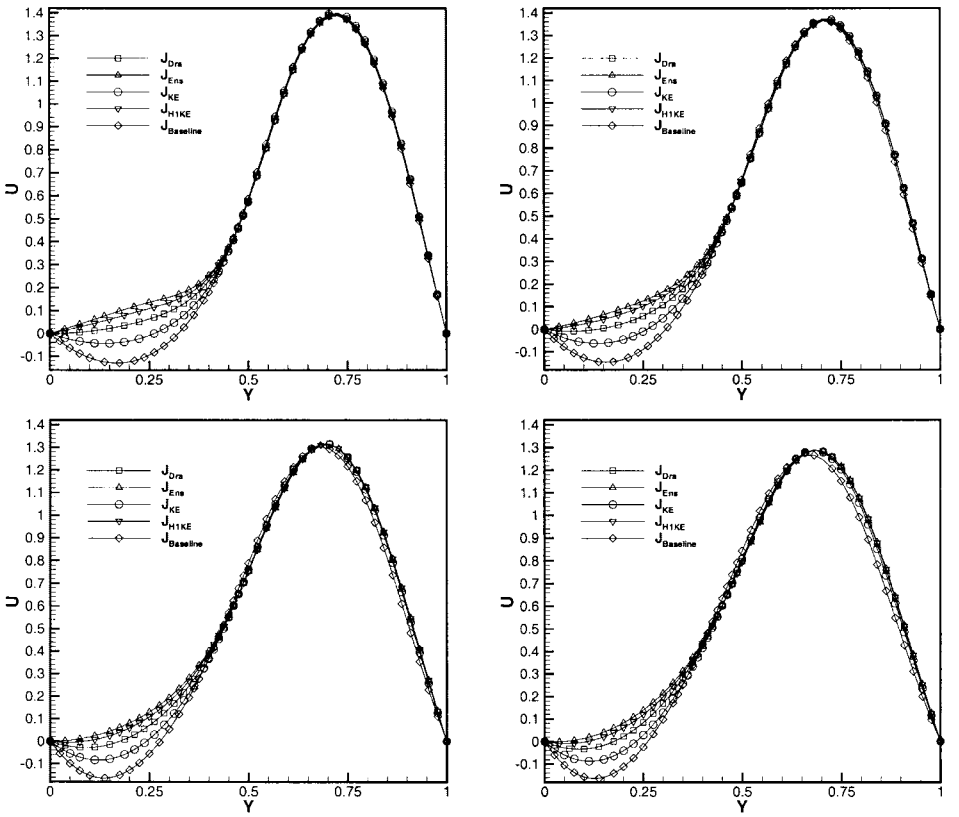

Figure 11. Performance of optimal controls when placed on Slot 2 for four different cost functionals. Shown are u-velocity at several stations in the channel. The baseline u-velocity is shown for comparison.

non-linear optimal control problems. It iteratively solves a sequence of linear quadratic optimal control problems converging to the optimal solution. The solution to the linear quadratic problem is characterized by the Karush–Kuhn–Tucker (KKT) optimality system which in the present context is a formidable system to solve. As a remedy various time domain decompositions, inexact SQP implementations and block iterative methods to solve the KKT systems were examined. The numerical experiments reported demonstrates that inexact SQP implementations substantially accelerates the convergence. Our study confirms that SQP method is indeed a robust algorithm for solving unsteady flow control problems.

REFERENCES

1. Gunzburger MD (ed.). *Flow Control*. Springer Verlag: New York, 1995.
2. Sritharan SS (ed.). *Optimal Control of Viscous Flows*. SIAM: Philadelphia, 1998.
3. Hou LS, Ravindran SS. A penalized Neumann control approach for solving an optimal Dirichlet control problem for the Navier–Stokes equations. *SIAM Journal on Control and Optimization* 1997; **36**(5):1795–1814.
4. Hou LS, Ravindran SS. Numerical approximation of optimal flow control problems by a penalty method: error estimates and numerical results. *SIAM Journal on Scientific Computing* 1999; **20**(5):1753–1777.

5. Ito K, Ravindran SS. Optimal control of thermally convected fluid flows. *SIAM Journal on Scientific Computing* 1998; **19**(6):1847–1869.
6. Barbu V, Sritharan SS. H^{∞}-control theory of fluid dynamics. *The Royal Society of London Proceedings, Series A, Mathematical, Physical and Engineering Sciences* 1998; **454**:3009–3033.
7. Pack LG, Joslin RD. Overview of active flow control at NASA Langley Research Center. In *SPIE's 5th annual International Symposium on Smart Structures and Materials*, San Diego, CA, 1998.
8. Seifert A, *et al*. Oscillatory blowing, a tool to delay boundary layer separation. *AIAA Journal* 1994; **31**:2052.
9. Ravindran SS. Numerical approximation of optimal flow control problems by SQP method. In *Optimal Control of Viscous Flows*, Sritharan SS (ed.). SIAM: Philadelphia, 1998; 181–198.
10. Hou LS, Ravindran SS. Computations of boundary optimal control problems for an electrically conducting fluid. *Journal of Computational Physics* 1996; **128**(2):319–330.
11. Gunzburger MD, Manservisi S. Analysis and approximation of the velocity tracking problem for Navier–Stokes flow with distributed control. *SIAM Journal on Numerical Analysis* 2000; **37**(5):1481–1512.
12. Joslin RD, Gunzburger MD, Nicolaides R, Erlebacher G, Hussaini MY. A self-contained, automated methodology for optimal flow control validated for transition delay. *AIAA Journal* 1997; **35**:816–824.
13. Fletcher R. *Practical Methods of Optimization* (2nd edn). Wiley: New York, 1987.
14. Gill PE, Murray W, Wright MH. *Practical Optimization*. Academic Press: London-New York, 1989.
15. Hinze M, Kunisch K. Second order methods for optimal control of time-dependent fluid flow. *SIAM Journal of Control and Optimization* 2001; **40**:925–946.
16. Heinkenschloss M. Formulation and analysis of a SQP method for the optimal Dirichlet boundary control of Navier–Stokes flow. In *Optimal Control*: *Theory*, *Algorithms*, *and Applications*, Hager WW, Pardalos PM (eds). Kluwer Academic Publishers: Dordrecht, 1998; 178–203.
17. Ghattas O, Bark JJ. Optimal control of two and three dimensional incompressible Navier–Stokes flows. *Journal of Computational Physics* 1997; **136**:231–244.
18. Nocedal J, Wright SJ. *Numerical Optimization*. Springer-Verlag: New York, 1999.
19. Conn AR, Gould NIM, Toint PhL. *Trust Region Methods*. SIAM: Philadelphia, 2000.
20. Heinkenschloss M. Time-domain decomposition iterative methods for the solution of distributed linear quadratic optimal control problems. *Technical Report TR00-31*, Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005-1892, 2000.
21. Ravindran SS. Numerical solution of optimal control for thermally convected fluid flow. *International Journal for Numerical Methods in Fluids* 1997; **25**(2):205–223.
22. Choi H, Temam R, Moin P, Kim J. Feedback control for unsteady flow and its application to the stochastic Burgers equation. *Journal of Fluid Mechanics* 1993; **253**:509–543.
23. Bewley TR, Moin P, Temam R. Control of turbulent flows. In *Proceedings of the 18th IFIP TC7 Conference on System Modeling and Optimization*, Detroit, MI, 1997.
24. Brezzi F, Fortin M. *Mixed and Hybrid Finite Element Methods*. Springer: New York, 1991.
25. Gunzburger MD. *Finite Element Methods for Viscous Incompressible Flows*. Academic Press: Boston, 1989.
26. Hou LS, Yan Y. Dynamics and approximation of a velocity tracking problem for the Navier–Stokes flows with piecewise distributed controls. *SIAM Journal on Control and Optimization* 1997; **35**(6):1847–1885.
27. Sani RL, Gresho PM. Resume and remarks on the open boundary condition mini-symposium. *International Journal for Numerical Methods in Fluids* 1994; **18**:983–1008.
28. Ravindran SS. Reduced-order adaptive controllers for fluid flows using POD. *Journal of Scientific Computing* 2000; **15**(4):457–478.
29. Sritharan SS. Dynamic programming of the Navier–Stokes equations. *Systems & Control Letters* 1991; **16**: 299–307.

