**Karl-Franzens-Universität Graz**
**Spezialforschungsbereich F 003 - Kontrolle und Optimierung**
**Projektbereich Kontinuierliche Optimierung und Kontrolle**

**Bericht Nr. 165 - September 1999**

# SECOND ORDER METHODS FOR OPTIMAL CONTROL OF TIME – DEPENDENT FLUID FLOW

MICHAEL HINZE AND KARL KUNISCH

ABSTRACT. Second order methods for open loop optimal control problems governed by the two-dimensional instationary Navier-Stokes equations are investigated. Optimality systems based on a Lagrangian formulation and adjoint equations are derived. The Newton and quasi-Newton methods as well as various variants of SQP-methods are developed for applications to optimal flow control and their complexity in terms of system solves is discussed. Local convergence and rate of convergence are proved. A numerical example illustrates the feasibility of solving optimal control problems for two-dimensional instationary Navier-Stokes equations by second order numerical methods in a standard workstation environment. Previously such problems were solved by gradient type methods.

*Date*: September 1999.

## Contents



## 1. Introduction

This research is devoted to the analysis of second methods for solving optimal control problems involving the time dependent Navier Stokes equations. Thus we consider

$$\min \; J(y,u) \text{ over } (y,u) \tag{1.1}$$

subject to

$$\begin{cases} \frac{\partial y}{\partial t} + (y\cdot\nabla)y - \nu\Delta y + \nabla p = Bu & \text{in } \; Q=(0,T)\times\Omega, \\ \operatorname{div} y = 0 & \text{in } \; Q, \\ y(t,\cdot) = 0 & \text{on } \; \Sigma = (0,T)\times\partial\Omega, \\ y(0,\cdot) = y_0 & \text{in } \; \Omega. \end{cases} \tag{1.2}$$

Here $\Omega$ is a bounded domain in $\mathbb{R}^2$, with sufficiently smooth boundary $\partial\Omega$. The final time $T>0$ and the initial condition $y_0$ are fixed. The vector valued variable $y$ and the scalar valued variable $p$ represent the velocity and the pressure of the fluid. Further $u$ denotes the control variable and $B$ the control operator. The precise functional analytic setting of problem (1.1), (1.2) will be given in Section 2. For the moment it suffices to say that typical cost functionals include tracking type functionals

$$J(y,u) = \tfrac{1}{2}\int_Q |y-z|^2 dx\, dt + \tfrac{\alpha}{2}\,|u|^2 \tag{1.3}$$

and functionals involving the vorticity of the fluid

$$J(y,u) = \tfrac{1}{2}\int_Q |\text{curl } y(t,\cdot)|^2_{\mathbb{R}^n}\, dx\, dt + \tfrac{\alpha}{2}\,|u|^2, \tag{1.4}$$

where $\alpha>0$ and $z$ are given. For the following discussion it will be convenient to formally represent all the equality constraints involved in (1.2) by $\hat{e}(y,p,u)=0$ so that (1.1), (1.2) can be expressed in the form

$$\begin{cases} \min \; J(y,u) \text{ over } (y,u) \\ \text{subject to} \\ \hat{e}(y,p,u) = 0. \end{cases} \tag{P}$$

In this form solving (1.1), (1.2) appears at first to be a standard task, see [AM, DHV, G, GT, IK, KS2, NT] and the references given there. However, the formidable size of (1.1), (1.2) and the goal of analyzing second order methods necessitate an independent analysis.

For second order methods applied to optimal control problems two classes can be distinguished depending on whether $(y,p)$ in (1.1), (1.2) are considered as independent variables or as functions of the control variable $u$. In the former case $\hat{e}(y,p,u)=0$ represents an explicit constraint for the optimization problem whereas in the latter case $\hat{e}(y(u),p(u),u)=0$ serves the purpose of describing the evaluation of $(y,p)$ as a function of $u$. In fact $(P)$ can be expressed as the reduced problem

$$(\hat{P}) \qquad \min\ \hat{J}(u) = J(y(u),u) \text{ over } u,$$

where $y(u)$ is implicitly defined via $\hat{e}(y(u),p(u),u)=0$.

To obtain a second order method in the case when $(y,p)$ are considered as independent variables one can derive the optimality system for $(P)$ and apply the Newton algorithm to the optimality system. This is referred to as the sequential quadratic programming (SQP)–method. Alternatively, if $(y,p)$ are considered as functions of $u$, then Newton's method can be applied to $(\hat{P})$ directly. The relative merits of these two approaches will be discussed in Section 4. To anticipate some of this discussion let us point out that the difference in numerical effort between these two methods is rather small. In fact, after proper rearrangements the difference in computational cost per iteration of the SQP–method for $(P)$ and the Newton method for $(\hat{P})$ consists in solving either the linearized equation (1.2) or the full nonlinear equation itself. In view of the time–dependence of either of these two equations an iterative procedure is used for their solution so that the difference between solving the linearized and nonlinear equation per sweep is not so significant. A second consideration that may influence the choice between SQP–method or Newton–method applied to $(\hat{P})$ concerns initialization. Initial guesses $(y_0,p_0)$ and $u_0$ for $(y,p,u)$ can clearly be used independently of each other in the SQP–method, where the states are decoupled from the controls. It is sometimes hinted at that this decoupling is not only important for the initialization but also during the iteration and that as a consequence the SQP–method may require fewer iterations than Newton's method for $(\hat{P})$, [H]. As we shall see below, the variables $y$ and $p$ can be initialized independently from $u_0$ also in the Newton method. Specifically, if $(y_0,p_0)$ and $u_0$ are available it is not necessary to abandon $(y_0,p_0)$ and compute $(y(u_0),p(u_0))$ from $u_0$. As for the choice of the initial guess $(y_0,p_0,u_0)$ one possibility is to rely on one of the suboptimal strategies that were developed in the recent past to obtain approximate solutions to (1.1), (1.2). We mention reduced order techniques [IR], POD–based methods [HK, KV, LT] and the instantaneous control method [CTMK, BMT, CHK]. As another possibility one can carry out some gradient steps before one switches to the Newton iteration.

Let us briefly comment on some related contributions. In [AT] optimality systems are derived for problems of the type (1.1), (1.2). A gradient technique is proposed in [GM] for the solution of (1.1), (1.2). Similarly in [B] gradient techniques are analyzed for a boundary control problem related to (1.1), (1.2). In [FGH] the authors analyze optimality systems for exterior boundary control problems. One of the few contributions focusing on second order methods for optimal control of fluids are given in [GB, H]. These works are restricted to stationary problems, however.

This paper, on the other hand focuses on second order methods for time dependent problems. We show that despite the difficulties due to the size of (1.1), (1.2) and the fact that the optimality systems contains a two point boundary value problem, forward in time for the primal- and backwards in time for the adjoint variables, second order methods are computationally feasible. We establish that the initial approximation to the reduced Hessian is only a compact perturbation of the Hessian at the minimizer. In addition we give conditions for second order sufficient optimality conditions of tracking type problems. These results imply superlinear convergence of quasi–Newton as well as SQP–methods. While the present paper focuses on distributed control problems in a future paper we plan to address the case of velocity control along the boundary.

The paper is organized as follows. Section 2 contains the necessary analytic prerequisites. First and second order derivatives of the cost functional with respect to the control are computed in Section 3. The fourth section contains a comparison of second order methods to solve (1.1), (1.2). In Section 5 convergence of quasi–Newton method and SQP–methods applied to $(\hat{P})$ is analyzed. Numerical results for the Newton–method and comparisons to a gradient method are contained in Section 6.

## 2. The optimal control problem

In this section we consider the optimal control problem (1.1), (1.2) in the abstract form

$$\left\{\begin{array}{l}\min\ J(y,u) \text{ over } (y,u)\in W\times U\\ \text{subject to } e(y,u)=0.\end{array}\right. \tag{2.1}$$

To define the spaces and operators arising in (2.1) we assume $\Omega$ to be a bounded domain in $\mathbb{R}^2$ with Lipschitz boundary and introduce the solenoidal spaces

$$H=\{v\in C_0^\infty(\Omega)^2\colon\ \text{div}\ v=0\}^{-|\cdot|_{L^2}},\ V=\{v\in C_0^\infty(\Omega)^2\colon\ \text{div}\ v=0\}^{-|\cdot|_{H^1}},$$

with the superscripts denoting closures in the respective norms. Further we define

$$W=\{v\in L^2(V)\colon v_t\in L^2(V^*)\}\quad\text{and}\quad Z:=L^2(V)\times H,$$

$W$ endowed with the norm

$$|v|_W=(|v|^2_{L^2(V)}+|v_t|^2_{L^2(V^*)})^{1/2},$$

$$H^{2,1}(Q):=\{v\in L^2(V\cap H^2(\Omega)^2); v_t\in L^2(H)\}$$

equipped with the norm

$$|v|^2_{H^{2,1}(Q)}:=|v|^2_{L^2(V\cap H^2(\Omega)^2)}+|v_t|^2_{L^2(H)},$$

and set $\langle\cdot,\cdot\rangle:=\langle\cdot,\cdot\rangle_{L^2(V^*),L^2(V)}$, with $V^*$ denoting the dual space of $V$. Here $L^2(V)$ is an abbreviation for $L^2(0,T;V)$ and similarly $L^2(V^*)=L^2(0,T;V^*)$. Recall that up to a set of measure zero in $(0,T)$ elements $v\in W$ can be identified with elements in $C([0,T];H)$ and elements $w\in H^{2,1}(Q)$ can be identified with elements in $C([0,T];V)$. In (2.1) further $U$ denotes the Hilbert space of controls and $J\colon L^2(V)\times U\to\mathbb{R}$ is the cost functional which is assumed to be bounded from below, weakly lower semi-continuous, twice Fréchet differentiable with locally Lipschitzean second derivative, and radially unbounded in $u$, i.e. $J(y,u)\to\infty$ as $|u|_U\to\infty$, for every $y\in W$. Furthermore, the control space $U$ is identified with its dual $U^*$. To simplify the notation for the second derivative we also assume that the functional $J$ can be decomposed as

$$J(y,u) = J_1(y) + J_2(u). \tag{2.2}$$

The nonlinear mapping

$$e\colon W \times U \to Z^* = L^2(V^*) \times H$$

is defined by

$$e(y,u) = (\tfrac{\partial y}{\partial t} + (y \cdot \nabla)y - \nu\Delta y - Bu, y(0) - y_0)$$

where $B \in \mathcal{L}(U, L^2(V^*))$ and $y_0 \in H$. Comparing (1.1), (1.2) to (2.1) we note that the conservation of mass, as well as the boundary condition are realized in the choice of the space $W$ while the dynamics are described by the condition $e(y,u) = 0$. In variational form the constraints in (2.1) can be equivalently expressed as:
given $u \in U$ find $y \in W$ such that $y(0) = y_0$ and

$$\langle y_t, v\rangle + \langle (y \cdot \nabla)y, v\rangle + \nu(\nabla y, \nabla v)_{L^2(L^2)} = \langle Bu, v\rangle \text{ for all } v \in L^2(V). \tag{2.3}$$

The following existence result for the Navier–Stokes equations in dimension two is well known ([CF, L, T], Chapter III).

**Proposition 2.1.** There exists a constant $C$ such that for every $u \in U$ there exists a unique element $y = y(u) \in W$ satisfying

$$e(y(u), u) = 0$$

and

$$|y|_{C(0,T;H)} + |y|_W \le C(|y_0|_H + |u|_U + |y_0|_H^2 + |u|_U^2).$$

From Proposition (2.1) we conclude that with respect to existence (2.1) is equivalent to

$$\min\ \hat{J}(u) = J(y(u), u) \text{ subject to } u \in U, \tag{2.4}$$

where $y(u) \in W$ satisfies $e(y(u), u) = 0$.

**Theorem 2.1.** Problem (2.1) admits a solution $(y^*, u^*) = (y(u^*), u^*) \in W \times U$.

**Proof.** With the above formalism the proof is quite standard and we only give a brief outline. Since $J$ is bounded from below there exists a minimizing sequence $\{(y_n, u_n)\} = \{y(u_n), u_n\}$ in $W \times U$. Due to the radial unboundedness property of $J$ in $u$ and Proposition 2.1 the sequence $\{(y_n, u_n)\}$ is bounded in $W \times U$ and hence there exists a subsequence with a weak limit $(y^*, u^*) \in W \times U$. Weak lower semi-continuity of $(y,u) \to J(y,u)$ implies that

$$J(y^*, u^*) = \inf\{J(y,u)\colon (y,u) \in W \times U,\ e(y,u) = 0\},$$

and it remains to show that $y^* = y(u^*)$. This can be achieved by passing to the limit in (2.3) with $(y,u)$ replaced by $(y(u_n), u_n)$. □

We shall also require the following result concerning strong solutions to the Navier-Stokes equation, ([T], Theorem III. 3.10).

**Proposition 2.2.** If $y_0 \in V$ and $B \in \mathcal{L}(U, L^2(H))$, then for every $u \in U$ the solution $y = y(u) \in W$ to $e(y,u) = 0$ satisfies $y \in H^{2,1}(Q)$. Moreover, for every bounded set $\mathcal{U}$ in $U$

$$\{y(u)\colon u \in \mathcal{U}\} \text{ is bounded in } H^{2,1}(Q).$$

We shall frequently refer to the linearized Navier-Stokes system and the adjoint equations given next:

$$\left\{\begin{array}{l} v_t+(v\cdot\nabla)y+(y\cdot\nabla)v-\nu\Delta v=f \quad \text{in } \Omega \text{ a.e. on } (0,T], \\ v(0)=v_0. \end{array}\right. \tag{2.5}$$

and

$$\left\{\begin{array}{l} -w_t+(\nabla y)^t w-(y\cdot\nabla)w-\nu\Delta w=g \quad \text{in } \Omega \text{ a.e. on } [0,T). \\ w(T)=0. \end{array}\right. \tag{2.6}$$

**Proposition 2.3.** Let $y\in W$, $v_0\in H$, $f\in L^2(V^*)$ and $g\in L^\alpha(V^*)\cap W^*$, with $\alpha\in[1,\frac{4}{3}]$. Then (2.5) admits a unique variational solution $v\in W$ and (2.6) has a unique variational solution $w\in L^2(V)$ with $w_t\in L^\alpha(V^*)\cap W^*$, $w\in C(H)$ and the first equation in (2.6) holding in $L^\alpha(V^*)\cap W^*$. Moreover, the following estimates hold.

i. $|v|_{L^\infty(H)}+|v|_{L^2(V)} \;\le\; C(|y|_{L^2(V)})\left\{|f|_{L^2(V^*)}+|v_0|_H\right\}$,
ii. $|v_t|_{L^2(V^*)} \;\le\; C(|y|_{L^2(V)},|y|_{L^\infty(H)})\left\{|f|_{L^2(V^*)}+|v_0|_H\right\}$,
iii. $|w|_{L^2(V)}+|w_t|_{L^\alpha(V^*)} \;\le\; C(|y|_{L^2(V)},|y|_{L^\infty(H)})\left\{|g|_{L^\alpha(V^*)}+|g|_{W^*}\right\}$.

If in addition $y\in L^\infty(V)$ and $g\in L^2(V^*)$, then $w\in W$ and

iv. $|w|_{L^2(V)}+|w_t|_{L^2(V^*)} \;\le\; C(|y|_{L^\infty(V)})\,|g|_{L^2(V^*)}$.

For $\partial\Omega\in C^2$, $y\in W\cap L^\infty(V)\cap L^2(H^2(\Omega)^2)$, $v_0\in V$ and $f,g\in L^2(H)$ the unique solutions $v$ of (2.5) and $w$ of (2.6) are elements of $H^{2,1}(Q)$ and satisfy the a-priori estimates

v. $|v|_{H^{2,1}(Q)} \;\le\; C(|y|_{L^\infty(V)},|y|_{L^2(H^2(\Omega)^2)})\left\{|f|_{L^2(H)}+|v_0|_V\right\}$

and

vi. $|w|_{H^{2,1}(Q)} \;\le\; C(|y|_{L^\infty(V)},|y|_{L^2(H^2(\Omega)^2)})\,|g|_{L^2(H)}$.

**Proof.** The proof is sketched in the Appendix. □

## 3. Derivatives

In this section representations for the first and second derivatives of $\hat{J}$ appropriate for the treatment of (2.4) by the Newton and quasi–Newton method are derived. We shall utilize the notation

$$X=W\times U \text{ and } x=(y,u) \text{ for } (y,u)\in W\times U.$$

**Proposition 3.1.** The operator $e=(e^1,e^2)\colon X\to Z^*$ is twice continuously differentiable with Lipschitz continuous second derivative. The action of the first two derivatives of $e^1$ are given by

$$\begin{aligned}\langle e^1_x(x)(w,s),\phi\rangle=\langle w_t,\phi\rangle+\langle(w\cdot\nabla)y,\phi\rangle+\langle(y\cdot\nabla)w,\phi\rangle\\ +\nu(\nabla w,\nabla\phi)_{L^2(L^2)}-\langle Bs,\phi\rangle,\end{aligned}$$

where $x=(y,u)\in X,(w,s)\in X$ and $\phi\in L^2(V)$, and

$$\begin{aligned}\langle e^1_{xx}(x)(w,s)(v,r),\phi\rangle=\langle e^1_{yy}(x)(w,v),\phi\rangle=\\ \langle(w\cdot\nabla)v,\phi\rangle+\langle(v\cdot\nabla)w,\phi\rangle=:\langle v,H(\phi)w\rangle_{W,W^*},\end{aligned} \tag{3.1}$$

where $(v,r)\in X$.

**Proof.** Since $e^2$ is linear we restrict our attention to $e^1$. Let $b\colon V \times V \times V \to \mathbb{R}$ be defined by

$$b(u, v, \phi) = \langle (u \cdot \nabla) v, \phi \rangle_{V^*, V},$$

and recall that, due to the assumption that $\Omega \subset \mathbb{R}^2$,

$$|b(u, v, \phi)|^2 \leq 2|u|_H\, |u|_V\, |v|_H\, |v|_V\, |\phi|_V^2, \tag{3.2}$$

for all $(u, v, \phi) \in V \times V \times V$, ([T], p.293). To argue local Lipschitz continuity of $e$, let $x, \tilde{x} \in X$ and $\phi \in L^2(V)$. We find

$$\begin{aligned} \langle e^1(x) - e^1(\tilde{x}), \phi \rangle &= \langle (y - \tilde{y})_t, \phi \rangle + \langle ((y - \tilde{y}) \cdot \nabla) \tilde{y}, \phi \rangle \\ &\quad + \langle (y \cdot \nabla)(y - \tilde{y}), \phi \rangle + \nu (\nabla (y - \tilde{y}), \nabla \phi)_{L^2(L^2)} + \langle B(\tilde{u} - u), \phi \rangle \\ &\leq \sqrt{2} \int_0^T |y - \tilde{y}|_H^{1/2} |y - \tilde{y}|_V^{1/2} (|\tilde{y}|_H^{1/2} |\tilde{y}|_V^{1/2} + |y|_H^{1/2} |y|_V^{1/2}) |\phi|_V\, dt \\ &\qquad + C |x - \tilde{x}|_X\, |\phi|_{L^2(V)}. \end{aligned}$$

Here and below $C$ denotes a constant independent of $x, \tilde{x}$ and $\phi$. Due to the continuous embedding of $W$ into $L^\infty(H)$ we have

$$\begin{aligned} \langle e^1(x) - e^1(\tilde{x}), \phi \rangle \leq C \Big[ & |x - \tilde{x}|_X\, |\phi|_{L^2(V)} + |y - \tilde{y}|_{L^\infty(H)}^{1/2} (|\tilde{y}|_{L^\infty(H)}^{1/2} + |y|_{L^\infty(H)}^{1/2}) \\ & + \textstyle\int_0^T |y - \tilde{y}|_V^{1/2} (|\tilde{y}|_V^{1/2} + |y|_V^{1/2}) |\phi|_V\, dt \Big]. \end{aligned}$$

Using Hölder's inequality this further implies the estimate

$$\begin{aligned} \langle e(x) - e(\tilde{x}), \phi \rangle \leq C \Big[ & |x - \tilde{x}|_X + |y - \tilde{y}|_{L^\infty(H)}^{1/2} (|\tilde{y}|_{L^\infty(H)}^{1/2} + |y|_{L^\infty(H)}^{1/2}) \\ & + \Big( \textstyle\int_0^T |y - \tilde{y}|_V (|\tilde{y}|_V + |y|_V) dt \Big)^{1/2} \Big] |\phi|_{L^2(V)} \end{aligned}$$

and consequently, after redefining $C$ one last time

$$\langle e^1(x) - e^1(\tilde{x}), \phi \rangle \leq C\, |x - \tilde{x}|_X (|y|_W + |\tilde{y}|_W) |\phi|_{L^2(V)}. \tag{3.3}$$

This estimate establishes the local Lipschitz continuity of $e$. To verify that the formula for $e_x$ given above represents the Fréchet - derivative of $e$ we estimate

$$\begin{aligned} & |e^1(\tilde{x}) - e^1(x) - e_x^1(x)(\tilde{x} - x)|_{L^2(V^*)} = \sup_{|\phi|_{L^2(V)} = 1} \textstyle\int_0^T |b(y - \tilde{y}, y - \tilde{y}, \phi)|\, dt \\ & \leq \sup_{|\phi|_{L^2(V)} = 1} \textstyle\int_0^T |y - \tilde{y}|_H\, |y - \tilde{y}|_V\, |\phi|_V\, dt \\ & \leq C |y - \tilde{y}|_W \sup_{|\phi|_{L^2(V)} = 1} \textstyle\int_0^T |y - \tilde{y}|_V |\phi|_V\, dt \;\leq\; C |y - \tilde{y}|_W^2 \end{aligned}$$

and Fréchet - differentiability of $e$ follows. To show Lipschitz continuity of the first derivative let $x, \tilde{x}$ and $(v, r)$ be in $X$ and estimate

$$\begin{aligned} & |(e_x^1(\tilde{x}) - e_x^1(x))(v, r)|_{L^2(V^*)} = \sup_{|\phi|_{L^2(V)} = 1} \textstyle\int_0^T |b(y - \tilde{y}, v, \phi) + b(v, y - \tilde{y}, \phi)|\, dt \\ & \leq 2\sqrt{2} \sup_{|\phi|_{L^2(V)} = 1} \textstyle\int_0^T |y - \tilde{y}|_H\, |y - \tilde{y}|_V\, |v|_H\, |v|_V\, |\phi|_V\, dt \\ & \leq C |y - \tilde{y}|_W\, |v|_W. \end{aligned}$$

The expression for the second derivative can be verified by an estimate analogous to the one for the first derivative. The second derivative is independent of the point at which it is taken and thus it is necessarily Lipschitz continuous. □

From (3.2) it follows that for $\phi \in L^2(V)$ and $w \in W$ the mapping

$$\sigma : v \mapsto \langle v, H(\phi)w\rangle_{W,W^*}$$

is an element of $W^*$. In Section 4 we shall use the fact that $\sigma$ can also be identified with an element of $L^4(V)^* = L^{4/3}(V^*)$.

**Lemma 3.1.** For $\phi \in L^2(V)$ and $w \in W$ the functional $\sigma$ can be identified with an element in $W^* \cap L^{4/3}(V^*)$.

**Proof.** To argue that $\sigma \in L^{4/3}(V^*)$ let $v \in L^4(V)$ and estimate using (3.2)

$$\sigma(v) = \int_0^T b(w,v,\phi) + b(v,w,\phi)\,dt \leq 2\sqrt{2k}|w|^{\frac{1}{2}}_{L^\infty(H)} \int_0^T |w|^{\frac{1}{2}}_V |v|_V |\phi|_V\,dt$$

$$\leq 2\sqrt{2k}|w|^{\frac{1}{2}}_{L^\infty(H)} |\phi|_{L^2(V)} |w|^{\frac{1}{2}}_{L^2(V)} |v|_{L^4(V)},$$

where $k$ is the embedding constant of $V$ into $H$. This gives the claim . □

**Proposition 3.2.** Let $x = (y,u) \in W \times U$. Then $e_y(x)\colon W \to Z^*$ is a homeomorphism. Moreover, if the inverse of its adjoint $e_y^{-*}(x)\colon W^* \to Z$ is applied to an element $g \in W^* \cap L^\alpha(V^*)$, $\alpha \in [1, 4/3]$ then setting $(w, w_0) := e_y^{-*}(x)g \in L^2(V) \times H$ we have $w_t \in L^\alpha(V^*)$, $w(0) = w_0$ and $w$ is the variational solution to (2.6).

**Proof.** Due to Proposition 3.1, $e_y(x)$ is a bounded linear operator. By the closed range theorem the claim follows once it is argued that (2.5) has a unique solution $v \in W$ for every $(f, v_0) \in Z^*$. This is a direct consequence of Proposition 2.3, i. and ii.. The assertion concerning the adjoint follows from the same proposition, iii. and its proof. □

As a consequence of Propositions 3.1 and 3.2 and the implicit function theorem the first derivative of the mapping $u \to y(u)$ at $u$ in direction $\delta u$ is given by

$$y'(u)\delta u = -e_y^{-1}(x)e_u(x)\delta u, \tag{3.4}$$

where $x = (y(u), u)$. By the chain rule we thus obtain

$$\langle \hat{J}'(u), \delta u\rangle_U = \langle J_u(x) - e_u^*(x)e_y^{-*}(x)J_y(x), \delta u\rangle_U.$$

Introducing the variable

$$\lambda = -e_y^{-*}(x)J_y(x) \in Z \tag{3.5}$$

we obtain utilizing Proposition 2.3 iii. with $g = -J_y(x) \in L^2(V^*)$ the Riesz representation for the first derivative of $u \to \hat{J}(u)$:

$$\hat{J}'(u) = J_u(x) + e_u^*\lambda. \tag{3.6}$$

Here $\lambda = (\lambda^1, \lambda^0) \in Z$, $\lambda^1_t \in L^{4/3}(V^*)$, $\lambda^1 \in C(H)$ and $\lambda^1$ is the variational solution of

$$\begin{cases} -\lambda^1_t + (\nabla y)^t\lambda^1 - (y\cdot\nabla)\lambda^1 - \nu\Delta\lambda^1 = -J_y(x) \\ \lambda^1(T) = 0, \end{cases} \tag{3.7}$$

where the first equation holds in $L^{4/3}(V^*) \cap W^*$.

The computation of the second derivative of $\hat{J}''(u) \in \mathcal{L}(U)$ of $\hat{J}$ is more involved. Let $(\delta u, \delta v) \in U \times U$ and note that the second derivative of $u \to y(u)$ from $U$ to $W$ can be expressed as

$$y''(u)(\delta u, \delta v) = -e_y^{-1}(x)e_{yy}(x)(y'(u)\delta u, y'(u)\delta v). \tag{3.8}$$

By the chain rule, and since $W \subset L^2(V)$ and hence $L^2(V^*) \subset W^*$, we have

$$\langle \hat{J}''(u)\delta u, \delta v\rangle_U = \langle J_{yy}(x)y'(u)\delta u, y'(u)\delta v\rangle$$
$$+\langle J_y(x), y''(u)(\delta u, \delta v)\rangle + \langle J_{uu}(x)\delta u, \delta v\rangle_U$$
$$= \langle J_{yy}(x)y'(u)\delta u, y'(u)\delta v\rangle + \langle J_y(x), e_y^{-1}e_{yy}(x)(y'(u)\delta u, y'(u)\delta v\rangle$$
$$+\langle J_{uu}(x)\delta u, \delta v\rangle_U$$
$$= \langle J_{yy}(x)y'(u)\delta u, y'(u)\delta v\rangle + \langle \lambda^1, e^1_{yy}(x)(y'(u)\delta u, y'(u)\delta v\rangle$$
$$+\langle J_{uu}(u)\delta u, \delta v\rangle_U.$$

We introduce the Lagrangian $L\colon X \times Z \to \mathbb{R}$

$$L(x,\lambda) = J(x) + \langle e(x), \lambda\rangle_{Z^*,Z} \tag{3.9}$$

and the matrix operator

$$T(x) = \begin{pmatrix} -e_y^{-1}(x)e_u(x) \\ Id_U \end{pmatrix} \in \mathcal{L}(U,X). \tag{3.10}$$

We observe that the second derivative of $L$ with respect to $x$ can be expressed as

$$L_{xx}(x,\lambda) = \begin{pmatrix} J_{yy}(x) + \quad \langle e^1_{yy}(x)(\cdot,\cdot),\, \lambda^1\rangle & 0 \\ 0 & J_{uu}(x) \end{pmatrix} \in \mathcal{L}(X,X^*).$$

The above computation for $\hat{J}''(u)$ together with (3.4) imply that

$$\hat{J}''(u) = T^*(x)L_{xx}(x,\lambda)T(x), \tag{3.11}$$

where $x = (y(u), u)$.

## 4. Second order methods

This section contains a description and a comparison of second order methods to solve (2.1). Throughout $u^*$ denotes a (local) solution to (2.1).

### 4.1. Newton–and quasi–Newton algorithm.

For the sake of reference let us specify the Newton algorithm.

**Algorithm 4.1.** *(Newton Algorithm).*

1. *Choose* $u^0 \in N(u^*)$, *set* $k = 0$.
2. *Do until convergence*
   i) *solve* $\hat{J}''(u^k)\delta u^k = -\hat{J}'(u^k)$,
   ii) *update* $u^{k+1} = u^k + \delta u^k$,
   iii) *set* $k = k + 1$.

Let us consider the linear system in 2. i). Its dimension is that of the control space $U$. From the characterization of the Hessian $\hat{J}''(u^k)$ we conclude that its evaluation requires as many solutions to the linearized Navier–Stokes equation (3.4) with appropriate right hand sides as is the dimension of $U$. If $U$ is infinite dimensional then an appropriate discretization must be carried out. Let us assume now that the dimension of $U$ is large so that direct evaluation of $\hat{J}''(u^k)$ is not feasible. In this case 2. i) must be solved iteratively, e. g. by a conjugate gradient technique. We shall then refer to 2. i) as the "inner " loop as opposed to the do–loop in 2. which is the "outer" loop of the Newton algorithm. The inner loop at iteration level $k$ of the outer loop requires to

(i) evaluate $\hat{J}'(u^k)$, i.e. given $u^k$ compute $y(u^k)$ from (1.2) and $\lambda^1$ from (3.7) with $x = (y(u^k), u^k)$,
(ii) iteratively evaluate the action of $\hat{J}''(u^k)$ on $\delta_j^k$, the $j$–th iterate of the inner loop on the $k$–th level of the outer loop.

The iterate $q = \hat{J}''(u^k)\delta_j^k$ can be evaluated by successively applying the steps

a) solve in $L^2(V^*)$ for $v \in W$

$$\begin{aligned} v_t + (v\cdot\nabla)y + (y\cdot\nabla)v - \nu\Delta v &= B\delta_j^k \\ v(0) &= 0, \end{aligned}$$

where $y = y(u^k)$,

b) evaluate $J_{yy}(x)v + \langle e^1_{yy}(x)(v,\cdot), \lambda^1\rangle$,

c) solve in $W^*$ for $w \in L^2(V)$

$$e_y(x)^* \, w \;=\; J_{yy}(x)v + \langle e^1_{yy}(x)(v,\cdot), \lambda^1\rangle,$$

d) and finally set $q := J_{uu}\delta u + B^* w$.

We recall that $\lambda^1 \in L^2(V)$ and that for $s \in W$

$$\langle e^1_{yy}(x)(v,s), \lambda^1\rangle = \int_0^T \int_\Omega ((v\cdot\nabla)s\lambda^1 + (s\cdot\nabla)v\lambda^1)dx\,dt.$$

Moreover, by Lemma 3.1 the functional appearing in b) is an element of $W^* \cap L^{4/3}(V^*)$. Hence by Proposition 2.3 the adjoint equation in c) can equivalently be rewritten as

$$\begin{aligned} -w_t + (\nabla y)^t w - (y\cdot\nabla)w - \nu\Delta w &= J_{yy}(x)v + \langle e^1_{yy}(x)(v,\cdot), \lambda^1\rangle \\ w(T) &= 0, \end{aligned}$$

where the first equation holds in $W^* \cap L^{4/3}(V^*)$. Summarizing, for the outer iteration of the Newton method one Navier–Stokes solve for $y(u^k)$ and one linearized Navier–Stokes solve for $\lambda(u^k)$ are required. For the inner loop one forward (–in time) as well as one backwards linearized Navier–Stokes solve per iteration is necessary.

Concerning initialization we observe that if initial guesses $(y_0, u_0) \in W \times U$ are available (with $y_0$ not necessarily $y(u_0)$) then alternatively to the initialization in Algorithm 4.1 this information can be used advantageously to compute the adjoint variable $\lambda^1$ required for the initial guess for the right hand side of the linear system as well as to carry out steps a) - c) for the evaluation of the Hessian. There is no necessity to recompute $y(u_0)$ from $u_0$.

To avoid the difficulties of evaluating the action of the exact Hessian in Algorithm 4.1 one can resort to quasi–Newton algorithms. Here we specify one of the most prominent candidates, the BFGS–method. For $w$ and $z$ in $U$ we define the rank–one operator $w \otimes z \in \mathcal{L}(U)$, the action of which is given by

$$(w\otimes z)(v) = \langle z, v\rangle_U \, w.$$

In the BFGS–method the Hessian $\hat{J}''$ at $u^*$ is approximated by a sequence of operators $H^k$.

**Algorithm 4.2.** *(BFGS–Algorithm)*

1. *Choose* $u^0 \in N(u^*)$, $H^0 \in \mathcal{L}(U)$ *symmetric, set* $k = 0$.

2. *Do until convergence*
   i) *solve* $H^k \delta u^k = -\hat{J}'(u^k)$,
   ii) *update* $u^{k+1} = u^k + \delta u^k$,
   iii) *compute* $\hat{J}'(u^{k+1})$,
   iv) *set* $s^k = u^{k+1} - u^k,\ d^k = \hat{J}'(u^{k+1}) - \hat{J}'(u^k)$,
   v) *update* $H^{k+1} = H^k + \frac{d^k \otimes d^k}{\langle d^k, s^k\rangle_U} - \frac{H^k s^k \otimes H^k s^k}{\langle H^k s^k, s^k\rangle_U}$,
   vi) *set* $k = k+1$.

Note that the BFGS–algorithm requires no more system solves than the gradient algorithm applied to (2.1), which is one forward solution of the nonlinear equation to obtain $y(u^k)$ and one backward solve of the linearized equation (3.7) obtain the adjoint variable $\lambda(u^k)$.

In order to compare Newton's method to the SQP method derived in the next section we rewrite the update step 2. i) in Algorithm 4.1. To begin with we observe that the right hand side in the update step can be written with the help of the adjoint variable $\lambda$ from and the operator $T(x)$ defined in (3.10) as

$$-\hat{J}_u(u) = -J_u(x) - e_u^*(x)\lambda = -T^*(x) \begin{bmatrix} 0 \\ J_u(x) + e_u^*(x)\lambda \end{bmatrix}, \tag{4.1}$$

where we dropped the iteration indices. As a consequence, with $\delta y = y'(u)\delta u$ from (3.3) the update can be written as

$$T^*(x) L_{xx}(x,\lambda) \begin{bmatrix} \delta y \\ \delta u \end{bmatrix} = -T^*(x) \begin{bmatrix} 0 \\ J_u(x) + e_u^*(x)\lambda \end{bmatrix}, \tag{4.2}$$

so that

$$L_{xx}(x,\lambda) \begin{bmatrix} \delta y \\ \delta u \end{bmatrix} + \begin{bmatrix} 0 \\ J_u(x) + e_u^*(x)\lambda \end{bmatrix} \in \mathcal{N}(T^*(x))$$

holds. Since $e_x(x) \in \mathcal{L}(X, Z^*)$ and $\mathcal{N}(e_x(x)) = \mathcal{R}(T) \subseteq X$ it follows that $\mathcal{R}(T)$ is closed and we have the sequence of identities

$$\mathcal{N}(T^*(x)) = \mathcal{R}(T(x))^\perp = \mathcal{N}(e_x(x))^\perp = \mathcal{R}(e_x^*(x)).$$

Thus there exists $\delta\lambda \in Z$ such that

$$-e_x^*(x)\delta\lambda = L_{xx}(x,\lambda) \begin{bmatrix} \delta y \\ \delta u \end{bmatrix} + \begin{bmatrix} 0 \\ J_u(x) + e_u^*(x)\lambda \end{bmatrix}.$$

Using this equation together with the definition of $\delta y$, Newton's update may be rewritten as

$$\begin{bmatrix} L_{xx}(x^k,\lambda^k) & e_{x^*}(x^k) \\ e_x(x^k) & 0 \end{bmatrix} \begin{bmatrix} \delta y \\ \delta u \\ \delta\lambda \end{bmatrix} = - \begin{bmatrix} 0 \\ J_u(x) + e_u^*(x)\lambda \\ 0 \end{bmatrix}. \tag{4.3}$$

4.2. **Basic SQP–method.** Here we regard (2.1) as a minimization problem of the functional $J$ over the space $X$ subject to the explicit constraint $e(x) = 0$. The basic SQP-algorithm consists in applying Newton's method to the first order optimality system

$$\begin{aligned} L_x(x,\lambda) &= 0 \quad \text{in } X^* \\ L_\lambda(x,\lambda) &= 0 \quad \text{in } Z^*, \end{aligned} \tag{4.4}$$

where the Lagrangian $L$ is defined in (3.9).
With $x^*$ denoting a solution to problem (P), $e_x(x^*)$ is surjective by Proposition 3.2, and hence there exists a Lagrange multiplier $\lambda^* \in Z$ which is even unique such that (4.4) holds. The SQP–method will be well defined and locally second order convergent, if in addition to the surjectivity of $e_x(x^*)$ the following second order optimality condition holds.

**(H1)** There exists $\alpha > 0$ such that
$$\langle L_{xx}(x^*,\lambda^*)x, x\rangle_{X^*,X} \geq \alpha\,|x|_X^2, \text{ for all } x \in \ker(e_x(x^*)).$$

If $(H1)$ holds then, due to the regularity properties of $e$ there exists a neighborhood $B((x^*,\lambda^*))$ such that $L_{xx}(x,\lambda)$ is uniformly positive definite on $\ker(e_x(x))$ for every $x \in B((x^*,\lambda^*))$.

**Algorithm 4.3.** *(SQP–algorithm)*

1. *Choose* $(x^0,\lambda^0) \in B((x^*,\lambda^*))$, *set* $k = 0$.
2. *Do until convergence*
   i) *solve*

$$\begin{pmatrix} L_{xx}(x^k,\lambda^k) & e_x^*(x^k) \\ e_x(x^k) & 0 \end{pmatrix} \begin{pmatrix} \delta x^k \\ \delta\lambda^k \end{pmatrix} = -\begin{pmatrix} J_x(x^k) \quad + e_x^*(x^k)\lambda^k \\ e(x^k) \end{pmatrix} \tag{4.5}$$

   ii) *update* $(x^{k+1},\lambda^{k+1}) = (x^k,\lambda^k) + (\delta x^k,\delta\lambda^k)$,
   iii) *set* $k = k+1$.

Just as for Newton's method step 2. i.) is the difficult one. While in contrast to Newton's method neither the Navier–Stokes equation nor its linearization needs to be solved, the dimension of the system matrix which is twice the dimension of the state plus the dimension of the control space is formidable for applications in fluid mechanics. In addition from experience with Algorithm 4.3 for other optimal control problems, see [KA, V] for example, it is well known that preconditioning techniques must be applied to solve (4.5) efficiently. As a preconditioner one might consider the (action of the) operator $P\colon X^* \times Z^* \to X \times Z$ given by

$$P = \begin{pmatrix} 0 & 0 & R \\ 0 & J_{uu}(x^k)^{-1} & 0 \\ R^* & 0 & 0 \end{pmatrix},$$

where $R\colon Z^* \to H$ is the inverse to the (discretized) instationary Stokes operator or the (discretized) linearization of the Navier–Stokes equation at the state $y^k$, either one with homogenous boundary conditions.

One iteration of the preconditioned version of Algorithm 4.3 therefore requires two linear parabolic solves, one forward and one backwards in time. As a consequence, even with the application of preconditioning techniques, the numerical expense counted in number of parabolic system solves is less for the SQP–method than for Newton's method. However, the number of iterations of iterative methods applied to solve the system equations in Algorithms 4.1 and 4.3 strongly depends on the system dimension, which is much larger for Algorithm 4.3 than for Algorithm 4.1.

To further compare the structure of the Newton and the SQP–methods let us assume for an instance that $x^k$ is feasible for the primal equation, i. e. $e(x^k) = 0$ and $(x^k,\lambda^k)$ is feasible for the adjoint equation (3.5), i. e. $e_y^*(x^k)\lambda^k = -J_y(x^k)$.

Then the right hand side of (4.5) has the form

$$-\begin{pmatrix} 0 \\ J_u(x^k) + e_u^*\lambda^k \\ 0 \end{pmatrix}$$

and comparing to the computation at the end of section 4.1 we observe that the linear systems describing the Newton and the SQP–methods coincide. In general the nonlinear primal and the linearized adjoint equation will not be satisfied by the iterates of the SQP–method and we therefore refer to the SQP–method as an outer or unfeasible method, while the Newton method is a feasible one.

4.3. **Reduced SQP–method.** The idea of the reduced SQP–method is to replace (4.5) with an equation in $\ker e_x(x)$, so that the reduced system is of smaller dimension than the original one. To develop the reduced system we follow the lines of [KS]. Recall the definition of $T(x)\colon U \to X$ and define $A(x)\colon Z^* \to X$ by

$$A(x) = \begin{pmatrix} e_y^{-1}(x) \\ 0 \end{pmatrix}. \tag{4.6}$$

Note that $A$ is a right–inverse to $e_x(x)$. In fact, we have

i) $\ker e_x(x) = \mathcal{R}(T(x)) = \left\{ \begin{pmatrix} -e_y^{-1}(x)e_u(x)v \\ v \end{pmatrix} : v \in U \right\}$,

ii) $e_x(x)T(x) = 0$ in $Z^*$,

iii) $e_x(x)A(x) = I_{Z^*}$.

By Proposition 3.2 and due to $B \in \mathcal{L}(U, L^2(V^*))$ the operator $T(x)$ is an isomorphism from $U$ to $\ker e_x(x)$ and hence the second equality in (4.5) given by

$$e_x(x)\delta x = -e(x)$$

can be expressed as

$$\delta x = T(x)\delta u - A(x)e(x). \tag{4.7}$$

Using this in the first equality of (4.5) we find

$$L_{xx}(x,\lambda)T(x)\delta u - L_{xx}(x,\lambda)A(x)e(x) + e_x^*(x)\delta\lambda = -(J_x(x) + e_x^*(x)\lambda).$$

Applying $T^*(x)$ to this last equation and ii) from above implies that if $\delta u$ is a solution coordinate of (4.5) then it also satisfies

$$T^*(x)L_{xx}(x,\lambda)T(x)\delta u = T^*(x)L_{xx}(x,\lambda)A(x)e(x) - T^*(x)J_x(x). \tag{4.8}$$

Once $\delta u$ is computed from (4.8) then $\delta y$ and $\delta\lambda$ can be obtained from (4.7) (which requires one forward linear parabolic solve) and the first equation in (4.5) (another backwards linear parabolic solve).

Let us note that if $x$ is feasible then the first term on the right hand side of (4.8) is zero and (4.8) is identical to step 2. i) in Newton's Algorithm 4.1.

This again reflects the fact that Newton's method can be viewed as an SQP–method that obeys the feasibility constraint $e(x) = 0$. It also points at the fact that the amount of work (measured in equation solves) for the inner loop coincides for both the Newton and the reduced SQP–methods. The significant difference between the two methods lies in the outer iteration. To make this evident we next specify the reduced SQP–algorithm.

**Algorithm 4.4.** *(Reduced SQP–algorithm).*

1. *Choose* $x^0 \in B(x^*)$*, set* $k = 0$.
2. *Do until convergence*
   i) *Lagrange multiplier update: solve*
   $$e_y^*(x^k)\lambda^k = -J_y(x^k)$$
   ii) *Solve*
   $$\alpha)\ T^*(x^k)L_{xx}(x^k,\lambda^k)T(x^k)\delta u^k \quad = T^*(x^k)L_{xx}(x^k,\lambda^k)A(x^k)e(x^k) \\ -T^*(x^k)J_x(x^k)$$
   $\beta)$ $e_y(x^k)\delta y^k = -e(x^k) - e_u(x^k)\delta u^k$,
   iii) *update*
   $$x^{k+1} = x^k + (\delta y^k, \delta u^k),$$
   iv) *set* $k = k+1$.

Note that in the algorithm that we specified we did not follow the procedure outlined above for the update of the Lagrange variable. In fact for reduced SQP–methods there is no "optimal" update strategy for $\lambda$. The two choices described above are natural and frequently used. To implement Algorithm 4.4 two linear parabolic systems have to be solved in steps 2. i) and 2. ii) $\beta$) and, in addition two linear parabolic systems are necessary to evaluate the term involving the operator $A$ on the right hand side of 2. ii) $\alpha$). In applications this term is often neglected since it vanishes at $x^*$.

The reduced SQP–method and Newton's method turn out to be very similar. Let us discuss the points in which they differ:

i) Most significantly the velocity field is updated by means of the nonlinear equation in Newton's method and via the linearized equation in the reduced SQP–method.
ii) The right hand sides of the linear systems differ due to the appearance of the term involving the operator $A$. As mentioned above this term is frequently not implemented.
iii) Formally there is a difference in the initialization procedure in that $y^0$ is chosen independently from $u^0$ in the reduced SQP–method and $y^0 = y(u^0)$ in Newton's method. However, as explained in section 4.1 above, if a good initial guess $y^0$ independent from $y(u^0)$ is available, it can be utilized in Newton's method as well.

## 5. Convergence analysis

We present local convergence results for the algorithms introduced in Section 4 for cost functionals of separable type (2.2). For this purpose it will be essential to derive conditions that ensure positive definiteness of $\hat{J}''(u^*)$ and (**H1**). The key to these conditions are the a-priori estimates of Proposition 2.3. We shall also prove that the difference $\hat{J}''(u^*) - J_{uu}(x^*)$ is compact. This property is required for the rate of convergence analysis of quasi-Newton methods. In our first result we assert positive definiteness of the Hessian provided that $J_y(x)$ is sufficiently small, a condition which is applicable to tracking-type problems.

**Lemma 5.1.** (Positive definiteness of Hessian)
Let $u \in U$ and assume that $J_{yy}(x) \in \mathcal{L}(L^2(V), L^2(V^*))$ be positive semi-definite and $J_{uu}(x) \in \mathcal{L}(U)$ be positive definite, where $x = (y(u), u)$. Then, the Hessian $\hat{J}''(u)$ is positive definite provided that $|J_y(x)|_{L^2(V^*)}$ is sufficiently small.

**Proof**: We recall from (3.11) that

$$\hat{J}''(u) = T^*(x) L_{xx}(x, \lambda) T(x),$$

where $x = (y(u), u)$ and $\lambda = \lambda(x)$ is the solution to (3.7). It follows that

$$\hat{J}''(u) = e_u^*(x) e_y^{-*}(x) J_{yy}(x) e_y^{-1}(x) e_u(x) + e_u^*(x) e_y^{-*}(x) \langle e_{yy}^1(x)(e_y^{-1}(x) e_u(x), \cdot), \lambda^1(x)\rangle + J_{uu}(x). \tag{5.1}$$

Here we note that for $\delta u \in U$ the functional

$$w \mapsto \langle e_{yy}(x)(e_y^{-1}(x) e_u(x) \delta u, w), \lambda^1\rangle$$

is an element of $W^*$. Since $J_{yy}(x)$ is assumed to be positive definite and $J_{uu}(x)$ is positive definite the result will follow provided the operator norm of

$$\mathcal{R} := e_u^*(x) e_y^{-*}(x) \langle e_{yy}^1(x)(e_y^{-1}(x) e_u(x), \cdot), \lambda^1(x)\rangle \ \in \mathcal{L}(U) \tag{5.2}$$

can be bounded by $|J_y(x)|_{L^2(V^*)}$. Straightforward estimation gives

$$\begin{aligned} \|\mathcal{R}\|_{\mathcal{L}(U)} \ \le \ & \|e_u^*(x) e_y^{-*}(x)\|_{\mathcal{L}(W^*, U)} \\ & \|\langle e_{yy}^1(x)(\cdot, \cdot), \lambda^1(x)\rangle\|_{\mathcal{L}(W, W^*)} \, \|e_y^{-1}(x) e_u(x)\|_{\mathcal{L}(U, W)} \\ = \ & \|e_y^{-1}(x) e_u(x)\|^2_{\mathcal{L}(U, W)} \, \|\langle e_{yy}^1(x)(\cdot, \cdot), \lambda^1(x)\rangle\|_{\mathcal{L}(W, W^*)}. \end{aligned} \tag{5.3}$$

From Proposition 2.3 we conclude that

$$\|e_y^{-1}(x) e_u(x)\|_{\mathcal{L}(U, W)} \ \le \ C(|y|_{L^2(V)}, |y|_{L^\infty(H)}, \|B\|_{\mathcal{L}(U, L^2(V^*))}).$$

To estimate
$\|\langle e_{yy}^1(x)(\cdot, \cdot), \lambda^1(x)\rangle\|_{\mathcal{L}(W, W^*)}$ we recall that for $g, h \in W$

$$\langle e_{yy}^1(x)(g, h), \lambda^1(x)\rangle \ = \ \int_0^T \int_\Omega (g \cdot \nabla) h \lambda^1 \ + \ (h \cdot \nabla) g \lambda^1 \, dx dt.$$

Using (3.2) and the continuity of the embedding $W \hookrightarrow L^\infty(H)$ we may estimate

$$|\langle e_{yy}^1(x)(g, h), \lambda^1(x)\rangle| \ \le \ C \, |g|_W |h|_W \, |\lambda^1|_{L^2(V)},$$

with a constant $C$ independent of $g$ and $h$. Therefore,

$$\begin{aligned} \|\mathcal{R}\|_{\mathcal{L}(U)} \ \le \ & C(|y|_{L^2(V)}, |y|_{L^\infty(H)}, \|B\|_{\mathcal{L}(U, L^2(V^*))}) \, |\lambda^1|_{L^2(V)} \\ & \le C(|y|_{L^2(V)}, |y|_{L^\infty(H)}, \|B\|_{\mathcal{L}(U, L^2(V^*))}) \, |J_y(x)|_{L^2(V^*)}, \end{aligned}$$

where we applied iii. in Proposition 2.3 to (3.7). ☐

**Lemma 5.2.** Let $x \in X$ and denote by $\lambda = \lambda(x) \in Z$ the function defined in (3.5). Then, under the assumptions of Lemma 5.1 on $J$ condition **(H1)** is satisfied with $(x^*, \lambda^*)$ replaced by $(x, \lambda)$.

**Proof.** Let $(v,u) \in \mathcal{N}(e_x(x))$. Then $v$ solves (2.5) with $v_0 = 0$ and $f = Bu$. Due to Proposition 2.3, $v \in W$ and satisfies

$$|v|_W \;\le\; C(|y|_{L^2(V)}), |y|_{L^\infty(H)}, \|B\|_{\mathcal{L}(U,L^2(V^*))})\,|u|_U. \tag{5.4}$$

Let $\delta > 0$ be chosen such that $J_{uu}(x)(u,u) \ge \delta |u|_U^2$ for all $u \in U$. We find

$$\langle L_{xx}(x,\lambda)(v,u),(v,u)\rangle_{X^*,X} = J_{yy}(x)(v,v) + \langle e^1_{yy}(x)(v,v),\lambda^1\rangle + J_{uu}(x)(u,u)$$

$$\ge \delta\,|u|_U^2 - 2\sqrt{2}\int_0^T |v|_H|v|_V|\lambda^1|_V\,dt \;\ge\; \delta\,|u|_U^2 - C|u|_U^2\,|\lambda^1|_{L^2(V)}.$$

Here and below $C$ denotes a generic constant independent of $(v,u)$ and $\lambda = \lambda(x)$. Due to (3.5) and Proposition 2.3

$$|\lambda|_{L^2(V)} \;\le\; C\,|J_y(x)|_{L^2(V^*)}.$$

These estimates imply

$$\langle L_{xx}(x,\lambda)(v,u),(v,u)\rangle_{X^*,X} \;\ge\; \left(\delta - C|J_y(x)|_{L^2(V^*)}\right)\,|u|_U^2,$$

and combined with (5.4) the claim follows. □

**Lemma 5.3.** If $B \in \mathcal{L}(U, L^2(H))$, then the difference

$$\hat{J}''(u) - J_{uu}(x)$$

is compact for every $u \in U$.

**Proof.** Utilizing (5.2) we may rewrite

$$\hat{J}''(u) - J_{uu}(x) = e_u^*(x)e_y^{-*}(x)J_{yy}(x)e_y^{-1}(x)e_u(x) \,+\, \mathcal{R}, \tag{5.5}$$

where $x = (y(u),u)$. It will be shown that both summands define compact operators on $U$. For this purpose let $\mathcal{U}$ be a bounded subset of $U$. Utilizing $B \in \mathcal{L}(U,L^2(H)) \subset \mathcal{L}(U,L^2(V^*))$ and Proposition 2.3 we conclude that

$$S = \{e_y^{-1}(x)e_u(x)\delta u : \delta u \in \mathcal{U}\}$$

is a bounded subset of $W$ and hence of $L^2(V)$. Since by assumption $J$ is twice continuously Fréchet differentiable with respect to $y$ from $L^2(V)$ to $\mathbb{R}$ it follows that $J_{yy}(S)$ is a bounded subset of $L^2(V^*)$. Proposition 2.3, iii. implies that consequently $e_y^{-*}(J_{yy}(S))$ is bounded in $W^2_{4/3}\times H$, where $W^2_{4/3} := \{v \in L^2(V) : v_t \in L^{4/3}(V^*)\}$. Since $W^2_{4/3}$ is compactly embedded in $L^2(H)$ [CF] and $B \in \mathcal{L}(U,L^2(H))$ it follows from the fact that $e_u^*(x)(z^1,z^0) = -B^*z^1$ for $z = (z^1,z^0) \in L^2(V)\times H$ that

$$\{e_u^*(x)e_y^{-*}(x)J_{yy}(x)(z) : z \in S\} \tag{5.6}$$

is pre-compact in $U$.

Let us turn to the second addend in (5.5). Due to Lemma 3.1 and its proof the set

$$\{\langle e^1_{yy}(x)(z,\cdot),\lambda^1\rangle : z \in S\}$$

is a bounded subset of $W^* \cap L^{4/3}(V^*)$. It follows, utilizing Proposition 2.3 that

$$\{e_y^{-*}(x)\langle e^1_{yy}(x)(z,\cdot),\lambda^1\rangle : z \in S\}$$

is a bounded subset of $W^2_{4/3}\times H$. As above the assumption that $B \in \mathcal{L}(U,L^2(H))$ implies that

$$\{e_u^*(x)e_y^{-*}(x)\langle e^1_{yy}(x)(z,\cdot),\lambda^1\rangle ; z \in S\}$$

is precompact in $U$ and the lemma is verified. □

The following lemma concerning the operators $T(x)$ and $A(x)$ defined in (3.10) and (4.6) will be required for the analysis of the reduced SQP-method.

**Lemma 5.4.** The mappings Let $x \mapsto A(x)$ from $X$ to $\mathcal{L}(Z^*, X)$ and $x \mapsto T(x)$ from $X$ to $\mathcal{L}(U, X)$ are Fréchet differentiable with Lipschitz continuous derivatives.

**Proof.** An immediate consequence of i., ii. in Proposition 2.3 and the identities ii) and iii) in Section (4.3) together with the differentiability properties of the mapping $x \mapsto e_x(x)$. □

We are now in the position to prove local convergence for the algorithms discussed in Section 4. Throughout we assume that $(y^*, u^*)$ is a local solution to (2.1) and set $y^* = y(u^*)$, $x^* = (y^*, u^*)$. In addition to the general conditions on $J$, $B$ and $e$ we require

**(H2)** $J_{yy}(x^*) \in \mathcal{L}(L^2(V), L^2(V^*))$ is positive semi-definite, $J_{uu}(x^*) \in \mathcal{L}(U)$ is positive definite, and $|J_y(x^*)|_{L^2(V^*)}$ is sufficiently small.

With **(H2)** holding **(H1)** is satisfied due to Lemma 5.1. In particular a second order sufficient optimality condition holds and $(y^*, u^*)$ is a strict local solution to (2.1). The following theorem follows from well known results on Newton's algorithm.

**Theorem 5.1.** If **(H2)** holds then there exist a neighbourhood $\mathcal{U}(u^*)$ such that for every $u^0 \in \mathcal{U}(u^*)$ the iterates $\{u^n\}_{n\in\mathbb{N}}$ of Newton's Algorithm 4.1 converge quadratically to $u^*$.

**Theorem 5.2.** If **(H2)** holds then there exist a neighbourhood $\mathcal{U}(u^*)$ and $\epsilon > 0$ such that for all $u^0 \in \mathcal{U}(u^*)$ and all positive definite operators $H^0 \in L(U)$ with

$$|H^0 - \hat{J}''(u^*)|_{\mathcal{L}(U)} < \epsilon,$$

the BFGS method of Algorithm 4.2 converges linearly to $u^*$. If in addition $B \in \mathcal{L}(U, L^2(H))$ and $H^0 := J_{uu}(x^*)$, then the convergence is super-linear.

**Proof**: For the first part of the theorem we refer to [GR, Section4], for example. For the second claim we observe that the difference $\hat{J}''(u^*) - J_{uu}(x^*)$ is compact by Lemma 5.3, so that the claim follows from [GR, Theorem 5.1], see also [KS1].□

**Theorem 5.3.** Assume that **(H2)** holds and let $\lambda^*$ be the Lagrange multiplier associated to $x^*$. Then there exist a neighbourhood $\mathcal{U}(x^*, \lambda^*) \subset X \times Z$ such that for all $(x^0, \lambda^0) \in \mathcal{U}(x^*, \lambda^*)$ the SQP-Algorithm 4.3 is well defined and the iterates $\{(x^n, \lambda^n)\}_{n\in\mathbb{N}}$ converge quadratically to $(x^*, \lambda^*)$.

**Proof**: Since $J$ and $e$ are twice continuously differentiable with Lipschitz continuous second derivative, $e_x(x^*)$ is surjective by Proposition 3.2 and **(H1)** is satisfied, second order convergence of the SQP-method follows from standard results, see for instance [IK]. □

We now turn to the reduced SQP-method.

**Theorem 5.4.** Assume that **(H1)** holds and let $\lambda^*$ denote the Lagrange multiplier associated to $x^*$. Then there exist a neighbourhood $\mathcal{U}(x^*) \subset X$ such that for all

$x^0 \in \mathcal{U}(x^*)$ the reduced SQP-algorithm 4.4 is well defined and its iterates $\{x^n\}_{n\in\mathbb{N}}$ converge two-step quadratically to $x^*$, i.e.

$$|x^{k+1} - x^*|_X \;\le\; C\,|x^{k-1} - x^*|_X^2$$

for some positive constant $C$ independent of $k \in \mathbb{N}$.

**Proof**: First note that **(H1)** implies positive definiteness of $T(x^*)^* L_{xx}(x^*, \lambda^*) T(x^*)$ in a neighbourhood $\tilde{\mathcal{U}}(x^*)$ of $x^*$. By Lemma 5.4 the mappings $x \mapsto T(x)$ and $x \mapsto A(x)$ are Fréchet differentiable with Lipschitz continuous derivatives. Furthermore, utilizing Proposition 2.3, iii. and Lemma A.2 it can be shown that the mapping $x \mapsto \lambda(x)$ is locally Lipschitz continuous, where $\lambda$ is defined through (3.5). This, in particular, implies for the Lagrange multiplier updates $\lambda^k$ the estimate

$$|\lambda^k - \lambda|_Z \;\le\; C\,|x^k - x^*|_X, \quad x^k \in \tilde{\mathcal{U}}(x^*),$$

where the constant $C$ is positive and depends on $x^*$ and on $\sup\{|J_{yy}(x)|_{\mathcal{L}(L^2(V),L^2(V^*)}; x \in \tilde{\mathcal{U}}(x^*)\}$. Altogether, the assumptions for Corollary 3.6 in [K] are met and there exists a neighbourhood $\hat{\mathcal{U}}(x^*)$ such that for all $x^0 \in \mathcal{U}(x^*) := \hat{\mathcal{U}}(x^*) \cap \tilde{\mathcal{U}}(x^*)$ the claim follows. □

## 6. Numerical results

Here we present numerical examples that should first of all demonstrate the feasibility of utilizing Newton's method for optimal control of the two-dimensional instationary Navier-Stokes equations in a workstation environment despite the formidable size of the optimization problem. The total number of unknowns (primal-, adjoint-, and control variables) in Example 1 below, for instance, is of order $2.2{*}10^6$. The time horizon could still be increased or the mesh size decreased by utilizing reduced storage techniques at the expense of additional cpu-time, but we shall not pursue this aspect here. The control problem is given by (1.1), (1.2) with cost function $J$ defined by

$$J(y,u) := \frac{1}{2}\int_{Q_o} |y-z|^2\,dxdt + \frac{\alpha}{2}\int_{Q_c} |u|^2\,dxdt, \tag{6.1}$$

where $Q_c := \Omega_c \times (0,T)$ and $Q_o := \Omega_o \times (0,T)$, with $\Omega_c$ and $\Omega_o$ subsets of $\Omega = (0,1)^2$ denoting the control and observation volumes, respectively. In our examples $T = 1$, $U := L^2(Q_c)$, $\nu = \frac{1}{\mathrm{Re}}$ =400 and $B$ is the indicator function of $Q_c$. The results for Newton's method will be compared to those of the gradient algorithm, which we recall here for the sake of convenience.

**Algorithm 6.1. Gradient Algorithm**

1. Set $k = 0$ and choose $u^0$,
2. Set $d := -\hat{J}'(u^k)$ and compute
$$\rho^* = \arg\min_{\rho>0} I(\rho) := \hat{J}(u^k + \rho d)$$
3. Set
$$u^{k+1} = u^k + \rho^* d$$
4. Set $k = k+1$ and goto 2.

Given a control $u$ the evaluation of the gradient of $J$ at a point $u$ amounts to solving (1.2) for the state $y$ and (3.7) for the adjoint variable $\lambda$. Implementing a stepsize rule to determine an approximation of $\rho^*$ is numerically expensive as every evaluation of the functional $J$ at a control $u$ requires solving the instationary Navier-Stokes equations with right hand side $Bu$.

We compare two possibilities for computing approximations to the optimal step size $\rho^*$. For this purpose let us consider for $u \in U$ and search direction $d \in U$ the solutions $v \in W$ and $w \in L^2(V)$ of the systems

$$\begin{aligned} v_t - \nu\Delta v + (y\cdot\nabla)v + (v\cdot\nabla)y &= Bd \\ v(0) &= 0 \end{aligned} \tag{6.2}$$

and

$$\begin{aligned} -w_t - \nu\Delta w - (y\cdot\nabla)w + (\nabla y)^t w &= J_{yy}(x)v - (v\cdot\nabla)\lambda^1 + (\nabla v)^t\lambda^1 \\ w(T) &= 0, \end{aligned} \tag{6.3}$$

where $y = y(u)$ and $\lambda \in Z$ is the associated adjoint variable.

1. For a given search direction $d \in U$ interpolate the function $I(\rho)$ by a quadratic polynomial using the values $I(0), I'(0)$ and $I''(0)$, i.e.

$$I(\rho) \doteq I(0) + I'(0)\rho + \frac{1}{2}I''(0)\rho^2$$

and use the unique zero

$$\rho_1^* = \frac{-\langle \hat{J}'(u), d\rangle_U}{\alpha|d|_U^2 + \langle B^*w, d\rangle_U} \tag{6.4}$$

of the equation $I'(\rho) = 0$ as approximation of $\rho^*$, with $w$ given by (6.3).

2. Use the linearization of the mapping $\rho \mapsto y(u+\rho d)$ at $\rho = 0$,

$$y(u+\rho d) \doteq y(u) + \rho y'(u)d,$$

in the cost functional $J$. This results in the quadratic approximation

$$I_2(\rho) := J(y(u) + \rho y'(u)d, u + \rho d)$$

of the functional $I(\rho)$. Now use the unique root

$$\rho_2^* = \frac{-\langle \hat{J}'(u), d\rangle_U}{\alpha|d|_U^2 + |v|^2_{L^2(Q_o)}} \tag{6.5}$$

of the equation $I_2'(\rho) = 0$ as approximation of $\rho^*$, with $v$ given in (6.2).

The denominator of $\rho_1^*$ equals $I''(0) = \langle \hat{J}''(u)d, d\rangle_U$. From (5.1) with $u^*$ replaced by $u$ it follows as in the proof of Theorem 5.1 that it is positive, provided that the state $y(u)$ is sufficiently close to $z$ in $L^2(H)$.

Let us note that the computation of $\rho_1^*$ requires the solution of linearized Navier-Stokes equations forward and backward in time, whereas that of $\rho_2^*$ only requires one solve of the linearized Navier-Stokes equations. In addition, a numerical comparison shows that the step-size guess $\rho_2^*$ performs better than $\rho_1^*$, both with respect to the number of iterations in the gradient method and with respect to computational time. For the numerical results presented below we therefore use the step size proposal $\rho_2^*$. Thus, every iteration of the gradient algorithm amounts to solving the nonlinear Navier-Stokes equations forward in time and the associated adjoint equations backward in time for the computation of the gradient, and to solving linearized Navier-Stokes equations forward in time for the step size proposal.

The inner iteration of Newton's method is performed by the conjugate gradient method, the choice of which is justified in a neighbourhood of a local solution $u^*$ of the optimal control problem by the positive definiteness of $\hat{J}''(u^*)$, provided the desired state $z$ is sufficiently close to the optimal state $y(u^*)$.

For the numerical tests the target flow $z$ is given by the Stokes flow with boundary condition $z_1 = 1$ in tangential direction, see Fig. 1. The termination criterion for the $j$-th iterate $u_j^k$ in the conjugate gradient method is chosen as

$$\frac{|\hat{J}''(u^k)\delta u_j^k + \hat{J}'(u^k)|}{|\hat{J}'(u^0)|} \leq \min\left\{\left(\frac{|\hat{J}'(u^k)|}{|\hat{J}'(u^0)|}\right)^{\frac{3}{2}}, 10^{-2}\frac{|\hat{J}'(u^k)|}{|\hat{J}'(u^0)|}\right\} \text{ or } j \geq 50.$$

The initialization for Newton's method was $u^0 := 0$.

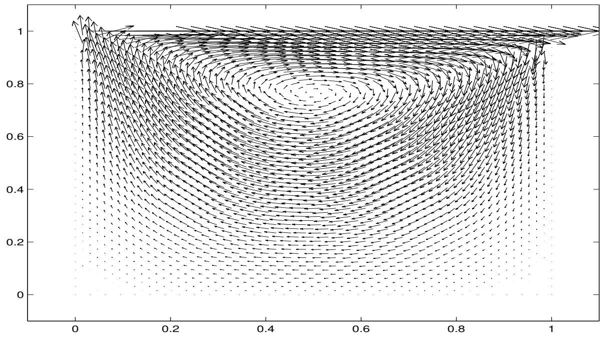


FIGURE 1. Control target, Stokes flow in the cavity

The discretization of the Navier-Stokes equations, its linearization and adjoint was carried out by using parts of the code developed by Bänsch in [BA], which is based on Taylor-Hood finite elements for spatial discretization. As time step size we took $\delta t = .00625$, which resulted in 160 grid points for the time grid, and 545 pressure and 2113 velocity nodes for the spatial discretization. All computations were performed on a DEC-ALPHA$^{TM}$station 500.

| Iteration | CG-steps | $\frac{\lvert\hat{J}'(u)\rvert}{\lvert\hat{J}'(u^0)\rvert}$ | $\frac{\lvert\delta u^k\rvert_U}{\lvert\delta u^{k-1}\rvert_U}$ | $\hat{J}(u^k)$ |
|---|---|---|---|---|
| 1 | - | 1.e0 | - | 1.196202e-2 |
| 2 | 13 | 3.358825e-1 | 1. | 3.226486e-3 |
| 3 | 11 | 5.058497e-2 | 0.492 | 1.617913e-3 |
| 4 | 18 | 8.249029e-3 | 0.422 | 1.482032e-3 |
| 5 | 17 | 1.409278e-4 | 0.079 | 1.480533e-3 |
| 6 | 19 | 4.686819e-6 | 0.032 | 1.480534e-3 |

TABLE 1. Performance of Newton's method for Example 1

**Example 1** We first present the results for $\Omega_c = \Omega_o = (0,1)^2$ and $\alpha = 10^{-2}$. Table 1 confirms super-linear convergence of the in-exact Newton method. To achieve the the same accuracy as Newton's method the gradient algorithm requires 96 iterations. The computing time with Newton's method is approximately 45 minutes whereas the gradient method requires 110 minutes. This demonstrates the superiority of Newton's method over the gradient algorithm for this example. For larger values of $\alpha$ and coarser time and space grids the difference in computing time is less drastic. In fact this difference increases with decreasing $\alpha$ and increasing mesh refinement. As expected a significant amount of computing time is spent for read-write actions of the variables to the hard-disc in the sub-problems.

In Figures 2, 3, 4 the evolution of the cost functional, the difference to the Stokes flow and the control as a function of time are documented. It can be observed that Newton's method tends to over-estimate the control in the first iteration step, whereas the gradient algorithm approximates the optimal control from below, see Figure 4. Graphically there is no significant change after the second iteration for Newton's method. These comments hold for quite a wide range of values for $\alpha$.

In Fig. 5 the uncontrolled flow together with the controlled flow and the control action at the end of the time interval are presented.

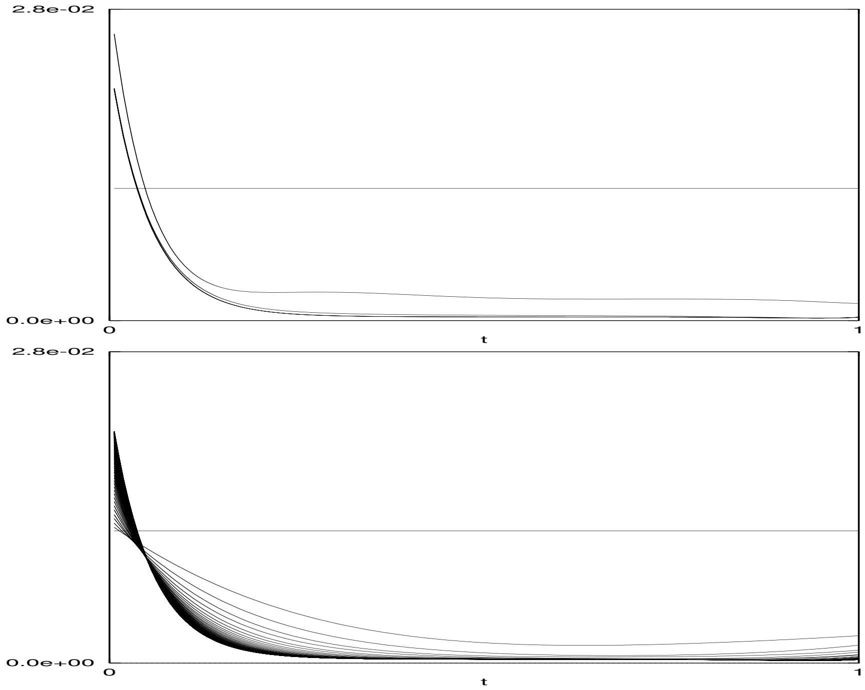


FIGURE 2. Newton's method (6 Iterations) (top) versus Gradient algorithm (96 Iterations), Re=400, $\alpha = 10^{-2}$: Evolution of cost functional for relative accuracy = 1.d-5

In the previous example the observation volume $\Omega_o$ and the control volume $\Omega_c$ each cover the whole spatial domain. From the practical point of view this is not feasible. However, from the numerical standpoint this is a complicated situation, since the inhomogeneities in the primal and adjoint equations are large.

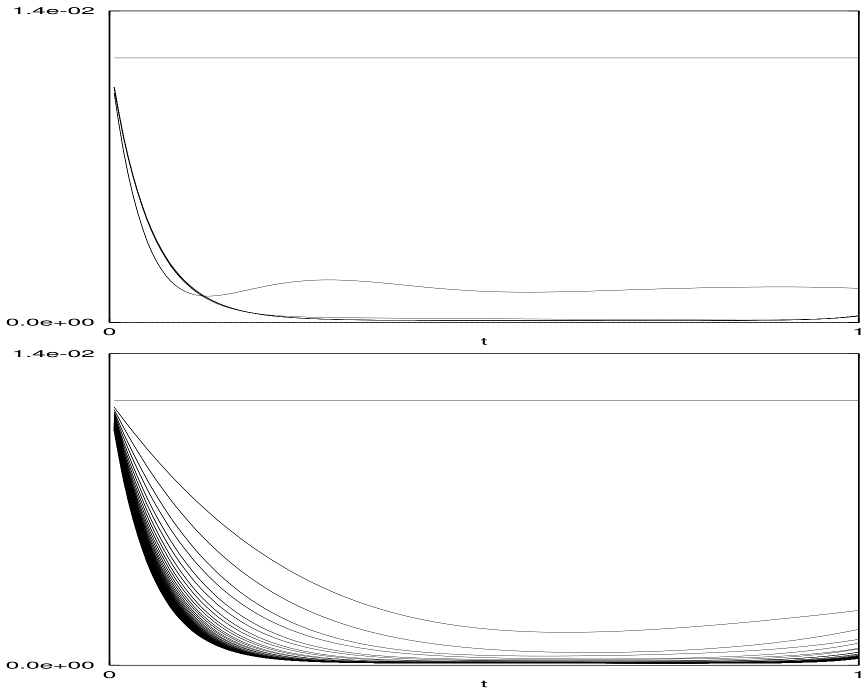


FIGURE 3. Newton's method (6 Iterations) (top) versus Gradient algorithm (96 Iterations), Re=400, $\alpha = 10^{-2}$: Evolution of difference to Stokes flow for relative accuracy = 1.d-5

We next present two numerical examples with different observation and control volumes. This results in smaller control and observation volumes than in Example 1, and thus the primal and adjoint equations are numerically simpler to solve.

**Example 2** Here $\Omega_o = (0., 1.) \times (0.85, 0.95)$ and $\Omega_c = (0., 1.) \times (0.9, 1.)$. The spatial and temporal discretizations as well as the parameter $\alpha$ are the same as in Example 1. Newton's method takes 15 minutes cpu-time and its convergence statistics are presented in Tab. 2. The gradient algorithm needs 25 iterations and 26 minutes cpu to reduce the value of the cost functional from $\hat{J}(u^0) = 1.25 \times 10^{-3}$ to $\hat{J}(u^*) = 6.3 \times 10^{-4}$.

| Iteration | CG-steps | $\frac{\lvert \hat{J}'(u) \rvert}{\lvert \hat{J}'(u^0) \rvert}$ | $\frac{\lvert \delta u^k \rvert_U}{\lvert \delta u^{k-1} \rvert_U}$ | $\hat{J}(u^k)$ |
|---|---|---|---|---|
| 1 | - | 1.e0 | - | 1.254304e-3 |
| 2 | 7 | 9.266545e-3 | 1. | 6.438873e-4 |
| 3 | 8 | 9.958369e-4 | 0.16 | 6.270625e-4 |
| 4 | 8 | 5.919598e-5 | 0.054 | 6.269869e-4 |
| 5 | 5 | 2.027455e-5 | 0.015 | 6.269868e-4 |

TABLE 2. Performance of Newton's method for Example 2

**Example 3** Here $\Omega_o = (0., 1.) \times (0.2, 0.5)$ and $\Omega_c = (0., 1.) \times (0.4, 0.7)$. Again, the discretization of the spatial and the time domain as well as the parameter $\alpha$ are the

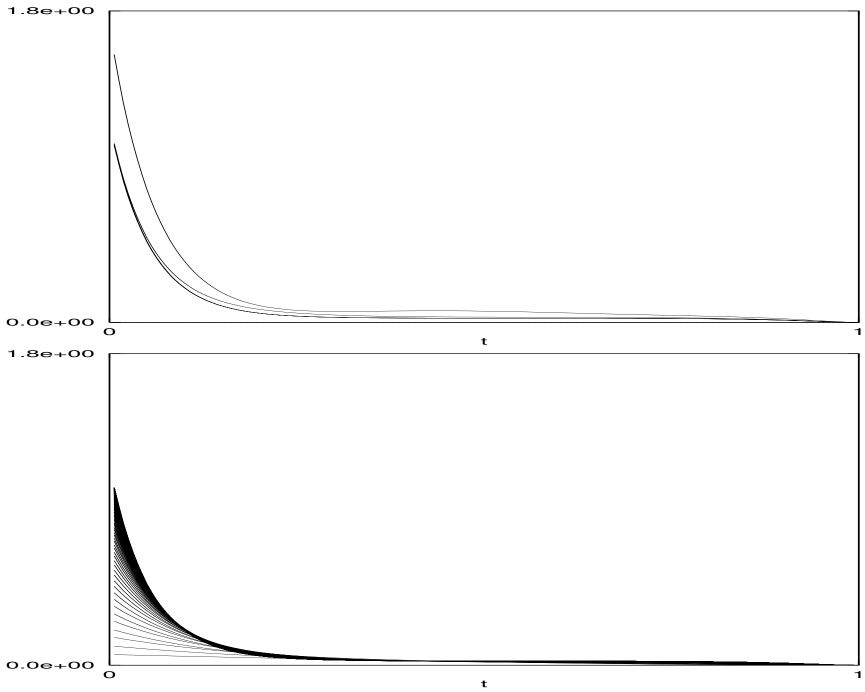


FIGURE 4. Newton's method (6 Iterations) (top) versus Gradient algorithm (96 Iterations), Re=400, $\alpha = 10^{-2}$: Evolution of control for relative accuracy = 1.d-5

same as in Example 2. The gradient algorithm needs 38 iterations to reduce the value of the cost functional from $\hat{J}(u^0) = 3.25 \times 10^{-3}$ to $\hat{J}(u^*) = 8.11 \times 10^{-4}$. It takes about 80 minutes cpu-time. We also implemented the Polak-Ribiére variant of the conjugate gradient algorithm. It converges after 37 iterations and yields a slightly better reduction of the residual. The amount of of cpu-time needed is nearly equal to that taken by the gradient algorithm. Newton's method is faster. It converges within 7 iterations to the approximate solution and needs 65 minutes cpu-time. The average cpu-time for the inner iteration loop is 7.5 minutes. As in the previous examples the average cost of a conjugate gradient iteration in the inner loop decreases with decreasing residual of the outer-iteration loop. The results are depicted in Tab. 3.

## Appendix A. Proof of Proposition 2.3

In the proof of Proposition 2.3 we make frequent use of the following Lemma.

**Lemma A.1.** For $u, v, w \in V$ let

$$b(u,v,w) \ := \int_\Omega (u \cdot \nabla) v \, w \, dx.$$

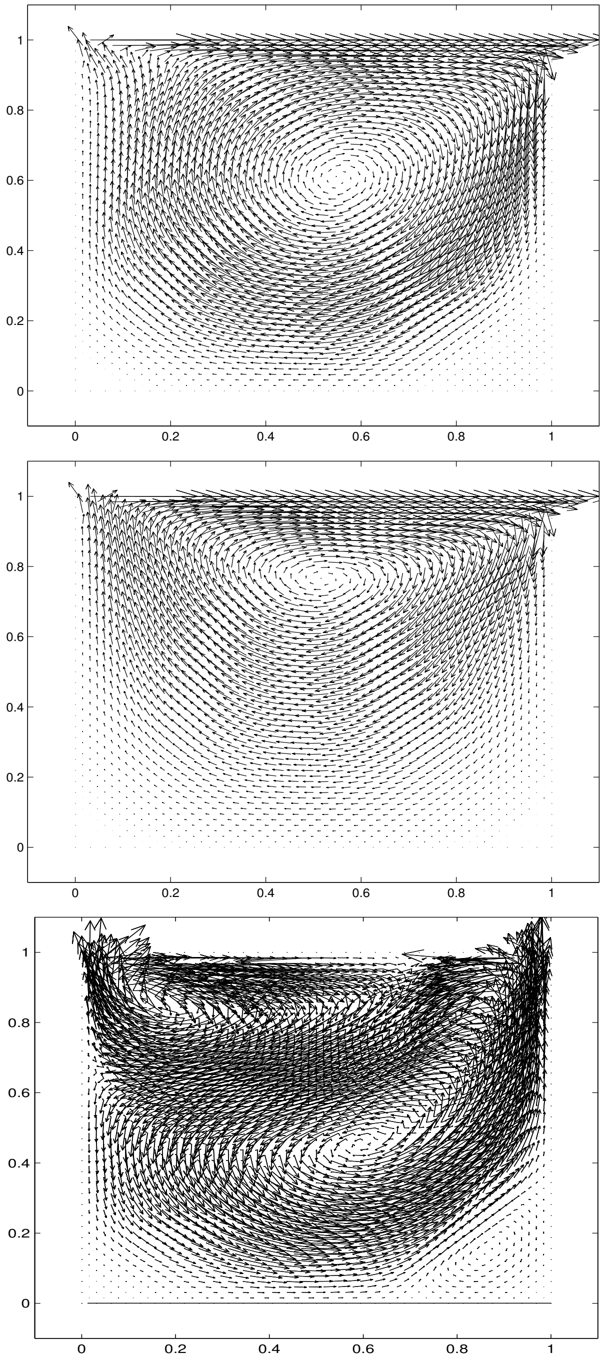

FIGURE 5. Results for $\alpha = 10^{-2}$, from top to bottom: uncontrolled flow, controlled flow at $t = 1$, and control force at $t = 0.75$

| Iteration | CG-steps | $\frac{\lvert\hat{J}'(u)\rvert}{\lvert\hat{J}'(u^0)\rvert}$ | $\frac{\lvert\delta u^k\rvert_U}{\lvert\delta u^{k-1}\rvert_U}$ | $\hat{J}(u^k)$ |
|---|---|---|---|---|
| 1 | - | 1.e0 | - | 3.253402e-3 |
| 2 | 5 | 3.824511e-1 | 1. | 1.198001e-3 |
| 3 | 8 | 2.270326e-1 | 0.600 | 9.829474e-4 |
| 4 | 8 | 5.762604e-2 | 0.619 | 8.333535e-4 |
| 5 | 12 | 9.527054e-3 | 0.404 | 8.108330e-4 |
| 6 | 11 | 1.920783e-4 | 0.083 | 8.105986e-4 |
| 7 | 18 | 9.283444e-6 | 0.029 | 8.106008e-4 |

TABLE 3. Performance of Newton's method for Example 3

Then $b(u,v,w) = -b(u,w,v)$ and there exists a positive constant $C$ such that
(A.1)

$$b(u,v,w) \;\le\; C \begin{cases} |u|_H^{\frac12}|u|_V^{\frac12}|v|_H^{\frac12}|v|_V^{\frac12}|w|_V & \text{for all}\, u,v,w\in V \\ |u|_H^{\frac12}|u|_V^{\frac12}|v|_V^{\frac12}|\Delta v|_H^{\frac12}|w|_H & \text{for all}\, u\in V, v\in V\cap H^2(\Omega)^2, w\in H \\ |u|_H|v|_V|w|_H^{\frac12}|\Delta w|_H^{\frac12} & \text{for all}\, u\in H, v\in V, w\in V\cap H^2(\Omega)^2 \\ |u|_H^{\frac12}|\Delta u|_H^{\frac12}|v|_V|w|_H & \text{for all}\, u\in V\cap H^2(\Omega)^2, v\in V, w\in H. \end{cases}$$

with a positive constant $C$.

**Proof.** In [T1]. □

**Lemma A.2.** There exists a positive constant $C$ such that

$$|(\nabla u)^t v|_{L^{4/3}(V^*)} + |(u\cdot\nabla)v|_{L^{4/3}(V^*)} \le C\,|u|_{L^2(V)}^{\frac12}\,|u|_{L^\infty(H)}^{\frac12}\,|v|_{L^2(V)} \quad \text{for all } u\in W, v\in L^2(V).$$

**Proof.** Since $b(u,v,\phi) = -b(u,\phi,v)$ the proof is identical to that of Lemma 3.1.□. Note that the power $4/3$ in the previous estimate cannot be improved by requiring $v\in W$. Sufficient conditions for $(\nabla u)^t v + (u\cdot\nabla)v \in L^2(V^*)$ are given by requiring in addition that $u$ or $v\in L^\infty(V)$.

**Proof of Proposition 2.3.** Existence and uniqueness of a solution to (2.5) can be shown following the lines of the existence and uniqueness proof for the instationary two-dimensional Navier-Stokes equations in [T, Chap.III]. In the following we sketch the derivation of the necessary a-priori estimates.

i. Test (2.5) with $v\in V$ pointwise with respect to time, use $b(u,v,v)=0$ and estimate using Young's inequality and the first estimate in (A.1). This results in

$$\text{(A.2)}\quad \frac{1}{2}\frac{d}{dt}|v|_H^2 \;+\; \nu|v|_V^2 \;\le\; C_\nu\left\{|f|_{V^*}^2 + |y|_V^2|v|_H^2\right\} \;+\; \frac{\nu}{2}|v|_V^2.$$

After integration from 0 to t Gronwall's inequality gives

$$\text{(A.3)}\quad |v|_{L^\infty(H)} \;\le\; \exp\left(2C_\nu|y|_{L^2(V)}\right)\left\{|f|_{L^2(V^*)} + |v_0|_H\right\}.$$

Using (A.3) in (A.2), the Cauchy-Schwarz inequality yields

$$\text{(A.4)}\quad |v|_{L^2(V)} \;\le\; C(|y|_{L^2(V)})\left\{|f|_{L^2(V^*)} + |v_0|_H\right\}.$$

Combining (A.3) and (A.4) yields the first claim.

ii. Test (2.5) with $\phi \in V$ pointwise in time and estimate using the Cauchy-Schwarz inequality and the first estimate in (A.1). This gives

$$\int_\Omega v_t \phi \, dx \leq \left\{ |f|_{V^*} + \nu |v|_V + C|y|_H^{\frac{1}{2}} |y|_V^{\frac{1}{2}} |v|_H^{\frac{1}{2}} |v|_V^{\frac{1}{2}} \right\} |\phi|_V, \tag{A.5}$$

which implies

$$|v_t|^2_{L^2(V^*)} \leq C \left\{ |f|^2_{V^*} + |v|^2_{L^2(V)} + |y|^2_{L^\infty(H)} |v|^2_{L^\infty(H)} |y|^2_{L^2(V)} |v|^2_{L^2(V)} \right\}. \tag{A.6}$$

This, together with $y \in W \subset L^\infty(H)$, and the estimates (A.3), (A.4) gives ii. Combining i. and ii. implies

$$|v|_W \leq C(|y|_{L^2(V)}, |y|_{L^\infty(H)}) \left\{ |f|_{L^2(V^*)} + |v_0|_H \right\}. \tag{A.7}$$

iii. For $y \in W$ we introduce the bounded linear operator $A(y) \in \mathcal{L}(W, Z^*)$ by

$$A(y)v = (v_t + (v \cdot \nabla)y + (y \cdot \nabla)v - \nu \Delta v, v(0)).$$

Note that $A(y)$ coincides with $e_y(x)$ of Section 3. Due to i. this operator admits a continuous inverse $A(y)^{-1} \in \mathcal{L}(Z^*, W)$. For the adjoint $A(y)^* \in \mathcal{L}(Z, W^*)$ we find by Lemma A.2

$$\begin{aligned} \langle A(y)v, (w, w_0) \rangle_{Z,Z^*} &= \langle v, A(y)^*(w, w_0) \rangle_{W,W^*} \\ &= \langle v_t, w \rangle + \langle v, (\nabla y)^t w - (y \cdot \nabla)w - \nu \Delta w \rangle_{W,W^*} + (v(0), w_0)_H \end{aligned}$$

for $v \in W$ and $w \in L^2(V)$. Since $A(y)^{-*} \in \mathcal{L}(W^*, Z)$ there exists for every $g \in W^*$ a unique solution $(w, w_0) \in Z$ to

$$\begin{aligned} \langle v_t, w \rangle + \langle v, (\nabla y)^t w - (y \cdot \nabla)w - \nu \Delta w \rangle_{W,W^*} + (v(0), w_0)_H \\ = \langle v, g \rangle_{W,W^*} \quad \text{for all } v \in W. \end{aligned} \tag{A.8}$$

From i. and ii. together with the fact that

$$\|A(y)^{-1}\|_{\mathcal{L}(Z^*,W)} = \|A(y)^{-*}\|_{\mathcal{L}(W^*,Z)}$$

we have

$$|w|_{L^2(V)} \leq C\left(|y|_{L^2(V)}, |y|_{L^\infty(H)}\right) |g|_{W^*}. \tag{A.9}$$

By Lemma A.2 and the assumption that $g \in L^{4/3}(V^*)$ the mapping

$$t \mapsto \left((y \cdot \nabla)w - (\nabla y)^t w + g\right)(t) \tag{A.10}$$

is an element of $L^\alpha(V^*)$, with $\alpha \in [1, 4/3]$. From (A.8) we therefore conclude that $w_t \in L^\alpha(V^*)$. Together with $w \in L^2(V)$ this implies $w \in C([0,T], H)$ [DL5]. From (A.8) we deduce that the first equation in (2.6) is well defined in $L^\alpha(V^*)$. Referring to (A.8) a third time and utilizing the fact that $w(T)$ is well defined in $H$ it follows that $w(T) = 0$ and $w_0 = w(0)$. By Lemma A.2 there exists a constant $C$ such that

$$|w_t|_{L^\alpha(V^*)} \leq C \left\{ |y|^{\frac{1}{2}}_{L^2(V)} |y|^{\frac{1}{2}}_{L^\infty(H)} + 1 \right\} |w|_{L^2(V)} + |g|_{L^\alpha(V^*)}.$$

Combining this estimate with (A.9) implies the estimate in iii.

iv. If $y \in L^\infty(V)$ and $g \in L^2(V^*)$ then $(y \cdot \nabla)w - (\nabla y)^t w \in L^2(V^*)$ and utilizing (A.8) we find that $w_t \in L^2(V^*)$. Moreover, by (A.1) we have

$$|w_t|_{L^2(V^*)} \le C\left(|y|_{L^\infty(V)}\right)\left\{|w|_{L^2(V)} + |g|_{L^2(V^*)}\right\}.$$

Together with (A.9) this gives the desired estimate in iv.

v. Test (2.5) with $\Delta v$ pointwise in time and utilize Young's inequality and the last estimate in (A.1) to obtain

$$\text{(A.11)} \quad \frac{1}{2}\frac{d}{dt}|v|_V^2 + \nu|\Delta v|_H^2 \le C_\nu\left\{|f|_H^2 + |y|_V^4|v|_H^2 + |y|_H|\Delta y|_H|v|_V^2\right\} + \frac{\nu}{2}|\Delta v|_H^2.$$

Integration from 0 to t together with (A.4) results in

$$\text{(A.12)} \quad |v|_V^2 \le C(|y|_{L^2(V)}, |y|_{L^\infty(V)})\left\{|f|_{L^2(H)}^2 + |v_0|_V^2\right\} + C_\nu \int_0^t |y|_H|\Delta y|_H|v|_V^2\,dt,$$

so that Gronwall's inequality gives

$$\text{(A.13)} \quad |v|_{L^\infty(V)} \le C(|y|_{L^2(V)}, |y|_{L^\infty(V)}, |y|_{L^\infty(H)}, |y|_{L^2(H^2(\Omega)^2)})\left\{|f|_{L^2(H)}^2 + |v_0|_V^2\right\}.$$

Using this in (A.11) yields

$$\text{(A.14)} \quad |v|_{L^2(H^2(\Omega)^2)} \le C(|y|_{L^2(V)}, |y|_{L^\infty(V)}, |y|_{L^\infty(H)}, |y|_{L^2(H^2(\Omega)^2)})\left\{|f|_{L^2(H)} + |v_0|_V\right\}.$$

To estimate $|v_t|_{L^2(H)}$ test (2.5) with $\phi \in V$ and use the last estimate in (A.1). This gives

$$\text{(A.15)} \quad \int_\Omega v_t\phi\,dx \le \left\{|f|_H + \nu|\Delta v|_H + C|y|_H^{\frac{1}{2}}|\Delta y|_H^{\frac{1}{2}}|v|_V + C|v|_H^{\frac{1}{2}}|\Delta v|_H^{\frac{1}{2}}|y|_V\right\}|\phi|_H,$$

so that $y \in L^\infty(V) \cap L^2(H^2(\Omega)^2)$ together with (A.13) and (A.14) implies

$$\text{(A.16)} \quad |v|_{L^2(H)} \le C(|y|_{L^2(V)}, |y|_{L^\infty(V)}, |y|_{L^\infty(H)}, |y|_{L^2(H^2(\Omega)^2)})\left\{|f|_{L^2(H)} + |v_0|_V\right\}.$$

Therefore,

$$\text{(A.17)} \quad |v|_{H^{2,1}(Q)} \le C(|y|_{L^2(V)}, |y|_{L^\infty(V)}, |y|_{L^\infty(H)}, |y|_{L^2(H^2(\Omega)^2)})\left\{|f|_{L^2(H)} + |v_0|_V\right\},$$

which is v. The estimation for $|w|_{L^\infty(V)\cap L^2(H^2(\Omega)^2)}$ is similar to that for $|v|_{L^\infty(V)\cap L^2(H^2(\Omega)^2)}$. In order to cope with $b(\phi, y, w)$ in the estimation of $|w_t|_{L^2(H)}$ one utilizes the third estimate in (A.1) to obtain the estimate vi. □

## References


[AM] W. Alt and K. Malanowski: The Lagrange–Newton method for state constrained optimal control problems, Comp. Optim. and Appl. **4** (1995), 217–239.

[AT] F. Abergel and R. Temam: On some control problems in fluid mechanics, Theoret. Comput. Fluid Dynamics **1** (1990), 303–325.

[BA] E. Bänsch: An adaptive Finite-Element-Strategy for the three-dimensional time-dependent Navier-Stokes-Equations, J. Comp. Math. **36**, 1991

[B] M. Berggren: Numerical solution of a flow-control problem: vorticity reduction by dynamic boundary action, SIAM J. Scientific Computing **19** (1998), 829–860.

[BMT] T. R. Bewley, P. Moin and R. Temam: DNS–based predictive control of turbulence: on optimal target for feedback algorithms, preprint 1998.

[CF] P. Constantin and C. Foias: Navier–Stokes Equations, The University of Chicago Press, Chicago, 1989.

[CHK] H. Choi, M. Hinze and K. Kunisch: Instantaneous control of backward-facing-step flows Preprint No. **571/97**, Technische Universitt Berlin, 1997, to appear in Applied Numerical Mathematics

[CTMK] H. Choi, R. Temam, P. Moin and J. Kim: Feedback control for unsteady flow and its application to the stochastic Burgers equation, J. Fluid Mech. **253**, 1993.

[DL5] R. Dautray, J.L. Lions: Mathematical Analysis and Numerical Methods for Science and Technology, Volume **5**, Springer Verlag (1992).

[DHV] J. E. Dennis, M. Heinkenschloss and L. N. Vincente: Trust–region interior–point SQP algorithms for a class of nonlinear programming problems, SIAM J. Control and Optimization **36** (1998), 1750–1794.

[FGH] A. V. Fursikov, M. D. Gunzburger and L. S. Hou: Boundary value problems and optimal boundary control for the Navier–Stokes system: the two–dimensional case, SIAM J. Control and Optimization **36** (1998), 852–894.

[G] R. Glowinski: Numerical Methods for Nonlinear Variational Problems, Springer, Berlin, 1984.

[GB] O. Ghattas and J. J. Bark: Optimal control of two–and three–dimensional incompressible Navier–Stokes Flows, Journal of Computational Physics **136** (1997), 231–244.

[GR] A. Griewank: The local convergence of Broyden-like methods in Lipschitzean problems in Hilbert spaces, SIAM J. Numer. Anal. **24** (1987), 684–705.

[GM] M. D. Gunzburger and S. Manservisi: The velocity tracking problem for Navier–Stokes flows with bounded distributed controls, preprint.

[GT] H. Goldberg and F. Tröltzsch: Second order sufficient optimality conditions for a class of nonlinear parabolic control problems, SIAM J. Control and Optimization **31** (1993), 1007–1025.

[H] M. Heinkenschloss: Formulation and analysis of a sequential quadratic programming method for the optimal Dirichlet boundary control of Navier–Stokes flow, Optimal Control: Theory, Algorithms, and Applications. Kluwer Academic Publishers B.V., 1998, pp. 178-203.

[HK] M. Hinze and A. Kauffmann: Reduced order Modelling and suboptimal control of a solid fuel ignition model Preprint No. **636/99**, Technische Universitt Berlin, 1999

[HKK] M. Hinze and K. Kunisch: Control strategies for fluid flows - optimal versus suboptimal control, ENUMATH 97 (Eds. H.G. Bock et al.), World Scientific, Singapore, 351-358.

[IK] K. Ito and K. Kunisch: Augmented Lagrangian–SQP–methods for nonlinear optimal control problems of tracking type, SIAM J. Control and Optimization **34** (1996), 874–891.

[IR] K. Ito and S. S. Ravindran: Optimal control of thermally convected fluid flow, SIAM J. Scientific Computing **19** (1998), 1767–1782.

[KA] A. Kauffmann: Optimal Control of the Solid Fuel Ignition Model PhD Thesis, Fachbereich Mathematik, Technische Universität Berlin, 1998

[KS] K. Kunisch and E.W. Sachs: Reduced SQP methods for parameter identification problems, SIAM J. Numer. Anal. **29**, 1992

[K] F.S. Kupfer:An infinite-dimensional convergence theory for reduced SQP-methods in Hilbert space, SIAM J. Optimization**6**, 1996

[KS1] C. T. Kelley and E. Sachs: Quasi-Newton methods and unconstrained optimal control problems, SIAM J. Control and Optimization **25** (1987), 1503–1516.

[KS2] C. T. Kelley and E. Sachs: Mesh independence of the gradient projection method for optimal control problems, SIAM J. Control and Optimization **30** (1992), 477–493.

[KV] K. Kunisch and S. Volkwein: Control of Burgers equation by a reduced order approach using Proper Orthogonal Decomposition Bericht Nr. **138**, Spezialforschungsbereich Optimierung und Kontrolle, Universität Graz, 1998

[L] P. L. Lions: Mathematical Topics in Fluid Mechanics I, Oxford Science Publications, Oxford, 1996.

[LT] H. V. Ly and H. T. Tran: Modelling and control of physical processes using proper orthogonal decomposition, CRSC–preprint, North Carolina State University.

[NT] P. Neittaanmäki and D. Tiba: Optimal Control of Nonlinear Parabolic Systems, Marcel Dekker, Basel, 1994.

[T] R. Temam: Navier-Stokes Equations, North Holland, Amsterdam, 1979.

[T1] R. Temam: Infinite-Dimensional Dynamical Systems in Mechanics and Physics, Second Edition, Springer, Berlin, 1997.

[V] S. Volkwein: Mesh-Independence of an Augmented Lagrangian-SQP Method in Hilbert Spaces and Control Problems for the Burgers Equation PhD Thesis, Fachbereich Mathematik, Technische Universität Berlin, 1997

Michael Hinze, Fachbereich Mathematik, Technische Universität Berlin, Strasse des 17. Juni 136, D-10623 Berlin and Institut für Mathematik, Karl-Franzens-Universität Graz, A-8010 Graz, Austria, research supported by a Marie-Curie Fellowship of the European Union.

Karl Kunisch, Institut für Mathematik, Karl-Franzens-Universität Graz, A-8010 Graz, Austria, research supported in part by the Fonds zur Förderung der wissenschaftlichen Forschung under SFB 03 ,,Optimierung und Kontrolle".