Boundary Control of Burgers' Equation—
A Numerical Approach

J.-M. LELLOUCHE
Centre d'Océanologie de Marseille, Campus de Luminy
Case 901-F 13288 Marseille Cedex 1, France

J.-L. DEVENON
LSEET, Université de Toulon et du Var
BP 132-F 83957 La Garde Cedex, France

I. DEKEYSER
Centre d'Océanologie de Marseille, Campus de Luminy
Case 901-F 13288 Marseille Cedex 1, France

(Received and accepted February 1994)

Abstract—We describe a methodology for solving boundary control problems for the viscous Burgers' equation. The aim is to identify boundary forcing in order to ensure the "best" fit between data and model results, by minimizing a functional which measures model and data discrepancies. A continuous variational formulation involving the adjoint technique is used, and its counterpart discretized version is obtained with a matrix approach as a guideline. A particular discretization of the nonlinear term of the equation is performed in order to insure, for the gradient of the functional to be minimized, a discretized expression which can be directly deduced from its continuous counterpart. Numerical experiments validate the proposed optimization algorithm.

Keywords—Optimal control, Adjoint model, Burgers' equation, Boundary control, Data assimilation.

1. INTRODUCTION

In recent years, mathematical techniques based on optimal control methods have been extensively developed, particularly in the field of physical meteorology [1] and, more recently, in the field of coastal physical oceanography [2]. Indeed, these approaches have been felt to be particularly relevant to enforce model results to fit datasets and to identify either physical parameters or boundary or initial conditions. These are particular cases of the so-called "data assimilation" methods which are presently among the best suited to make the most efficient use of very large quantity of observational information. One is often led, in solving coastal dynamic problems, to consider a special degenerated form of the Navier-Stokes equations, usually called the "Shallow Water" equations. For a one-dimensional evolution problem, these take the form of a nonlinear advection-diffusion model given by the viscous Burgers' equation, on which we have decided to focus the present study.

The variational method involves minimizing a certain functional which is a norm of the difference between the computed and measured solution model values. An algorithm is obtained, via

---

We should like to thank J. Ch. Gilbert and C. Lemaréchal for kindly helping us with the implementation of their program.

Typeset by A\LaTeX\-TeX
the so-called adjoint equations, for construction of the gradient of this functional with respect to the parameters. Once the gradient has been determined, the minimization can be performed by using any numerical optimization algorithm. Here, we have used the variable storage (or limited memory) quasi-Newton method whose computer code has been given by J. Ch. Gilbert and C. Lemaréchal [3].

It has been identified, since the first studies, that it is difficult to pass from the continuous to the discrete formulation, especially for nonlinear advection terms. This has led many researchers to use methods working exclusively from the discrete equations of the model. As a consequence, studies have been undertaken in order to perfect automatic generators of adjoint models [4]. The other way is to deduce the discrete counterpart from the continuous formulation, by using particular discretization. That is what is attempted in this paper, by solving a problem of nonlinear boundary control on which no general rules have been drawn.

Section 2 gives the mathematical formalism and the exact model that we have used. In Section 3, the optimal control theory is applied to supply theoretical solution to the control problem. Section 4 gives the corresponding discrete problem. Section 5 points out a series of numerical tests in order to check the model-solving algorithm. Section 6 discusses the obtained results. Lastly, Section 7 summarizes the results and conclusions.

2. THE CONTROL PROBLEM

According to the preceding section, the numerical model chosen is the viscous Burgers' equation in one space dimension. If we interpret the scalar function \( y(x, t) \) as modelling the velocity at a point \( x \) and a time \( t \), then the governing equation listed with suitable boundary and initial conditions is given by

\[
\begin{align*}
\frac{\partial y}{\partial t} + \frac{1}{2} \frac{\partial (y^2)}{\partial x} - \nu \frac{\partial^2 y}{\partial x^2} &= 0, \quad \text{in } Q, \\
y(x, 0) &= u(x), \quad \text{for } x \in (0, L), \\
y(0, t) &= \Psi_1(t), \quad \text{for } t \in (0, T), \\
y(L, t) &= \Psi_2(t), \quad \text{for } t \in (0, T),
\end{align*}
\]

where \( Q = (0, L) \times (0, T) \), and \( \nu \) plays the role of a viscosity coefficient.

Our problem is to control this system to produce a desired state concerning the modelized quantity \( y \), denoted by \( \bar{y} \), in \( Q \) or a subset of \( Q \) as E. J. Dean and P. Gubernatis [5] made by introducing pointwise control. This problem could also be solved by adjusting parameters of the equations (see [6] or [7], for example) or by performing a boundary control [8-11], as will be the case here. The control procedure consists in finding the optimal control \( \Psi_{\text{opt}} = (\Psi_1, \Psi_2)_{\text{opt}} \) (and the corresponding optimal solution \( y_{\text{opt}} \)) which minimizes a cost criterion measuring the Euclidean norm of the difference between \( y \) and \( \bar{y} \).

Inner products are defined on Hilbert-space \( L^2(Q) \) and on \( L^2(0, T) \) for a pair of continuous functions \( f_1 \) and \( f_2 \) belonging to these spaces, by

\[
(f_1, f_2) = \frac{1}{LT} \int_0^T \int_0^L f_1 f_2 \, dx \, dt,
\]

\[
\langle f_1, f_2 \rangle = \frac{1}{T} \int_0^T f_1 f_2 \, dt.
\]

Considering the cost function

\[
J(\Psi) = \frac{1}{2LT} \| y - \bar{y} \|^2_{L^2(Q)},
\]

thus, the problem is to find

\[
\min \{ J(\Psi) : \Psi \in L^2(0, T) \}
\]
under the constraint

\[ y = y(\Psi). \]  

As the optimal control cannot be obtained as a solution of necessary and sufficient conditions, there is an attempt to reach it in a progressive way, by minimizing \( J \) through a descent algorithm. We are, thus, faced with the computation of the gradient of the cost function.

### 3. Evaluation of the Gradient of the Cost Function: The Adjoint Model

Let us introduce the directional derivative of \( J \) at \( \Psi \in L^2(0,T) \) along \( \phi \in L^2(0,T) \), given by

\[
J'(\Psi; \phi) = \lim_{\epsilon \to 0} \frac{J(\Psi + \epsilon \phi) - J(\Psi)}{\epsilon}.
\]  

Assuming \( J'(\Psi; \phi) \) exists for all \( \phi \) and \( J'(\Psi; \cdot) \) is linear and continuous in \( \phi \), the gradient \( \nabla J(\Psi) \) will be defined as the linear form satisfying

\[
J'(\Psi; \phi) = \langle \nabla J(\Psi), \phi \rangle, \quad \forall \phi \in L^2(0,T).
\]  

From (2) and (5), it appears that \( J'(\Psi; \cdot) \) can be put under the form

\[
J'(\Psi; \cdot) = (y - \hat{y}, y'(\Psi; \cdot)), \quad \forall \phi \in L^2(0,T),
\]  

which gives, with (6),

\[
\langle \nabla J(\Psi), \phi \rangle = (y - \hat{y}, y'(\Psi; \phi)), \quad \forall \phi \in L^2(0,T),
\]  

where \( y'(\Psi; \phi) \) verify the following linearized Burgers’ tangent model:

\[
\begin{aligned}
\frac{\partial y'}{\partial t} + y \frac{\partial y'}{\partial x} + \nu \frac{\partial^2 y'}{\partial x^2} &= 0, \quad \text{in } Q, \\
y'(\Psi, \cdot; x, 0) &= 0, \quad \text{for } x \in (0, L), \\
y'(\Psi, \cdot; 0, t) &= \phi_1(t), \quad \text{for } t \in (0, T), \\
y'(\Psi, \cdot; L, t) &= \phi_2(t), \quad \text{for } t \in (0, T).
\end{aligned}
\]  

Let \( p = p(x, t) \) the adjoint state corresponding to the Burgers’ solution \( y \). By requiring the adjoint state function \( p \), which belongs to \( L^2(Q) \), to satisfy the adjoint model

\[
\begin{aligned}
\frac{\partial p}{\partial t} + y \frac{\partial p}{\partial x} + \nu \frac{\partial^2 p}{\partial x^2} &= y - \hat{y}, \quad \text{in } Q, \\
p(x, T) &= 0, \quad \text{for } x \in (0, L), \\
p(0, t) &= p(L, t) = 0, \quad \text{for } t \in (0, T),
\end{aligned}
\]  

and multiplying both sides of the first equation of the problem (9) by \( p \), and integrating in both space and time, after suitable integrations by parts, we obtain

\[
\int_0^T \int_0^L (y - \hat{y}) y'(\Psi; \phi) \, dx \, dt = \nu \int_0^T [\phi_2(t) \left( \frac{\partial p}{\partial x}(L, t) \right) - \phi_1(t) \left( \frac{\partial p}{\partial x}(0, t) \right)] \, dt.
\]  

With (8), we can write, for all \( \phi = (\phi_1, \phi_2) \in L^2(0,T) \),

\[
\langle \nabla J(\Psi), \phi \rangle = \frac{\nu}{TL} \int_0^T [\phi_2(t) \left( \frac{\partial p}{\partial x}(L, t) \right) - \phi_1(t) \left( \frac{\partial p}{\partial x}(0, t) \right)] \, dt,
\]  

which allows us to identify the gradient’s components as

\[
\nabla J(\Psi) = \frac{\nu}{L} \left[ \left( -\left( \frac{\partial p}{\partial x}(0, t) \right)_{t \in (0,T)} \right), \left( \frac{\partial p}{\partial x}(L, t) \right)_{t \in (0,T)} \right]^*,
\]  

where the subscript * denotes the transposition operation.
4. THE DISCRETE CONTROL PROBLEM

In Section 3, we have expressed the gradient of the cost function of a continuous problem in time and space. In practice, we have a discrete numerical model, which is only an approximation of the continuous equations. Then, the main difficulty is to discretize the problems (9) and (10) in such a way that the two discretized models will remain adjoint to each other. Furthermore, the gradient, that will stem from an “equivalent” of the integration by parts operation directly applied on the discretized version of the problem equations (9), must coincide with the discretized version of the formula (13).

Let the time interval \((0, T)\) (respectively, the space interval \((0, L)\)) be divided into \(N\) subintervals (respectively, into \(I+1\) subintervals), each of length \(\Delta t = T/N\) (respectively, \(\Delta x = L/(I+1)\)). The discrete version of the problems (3), (4) is then to find

$$
\min \left\{ J(\Psi) : \Psi \in \mathbb{R}^{2N} \right\},
$$

where

$$
J(\Psi) = \frac{1}{2NT} \sum_{i=1}^{I} \sum_{n=1}^{N} \left| y_{i}^{n} - \hat{y}_{i}^{n} \right|^2,
$$

and \(y_{i}^{n}\) (respectively, \(\hat{y}_{i}^{n}\)) is the approximation to \(y(i\Delta x, n\Delta t)\) (respectively, is the observed value at point \((i\Delta x, n\Delta t)\)).

The problem is now to choose the discretization of the two models (1) and (10) in order to obtain a discrete gradient in a good adequacy with the formula (13). As a guideline, the matrix approach can be helpful. After discretization, the system of equations (9) can be put under the form of the following great algebraic equation system:

$$
AY' = B_{ic}U_{ic}' + B_{bc}U_{bc}',
$$

where \(A, B_{ic},\) and \(B_{bc}\) are matrix operators, \(Y'\) the discretized linear tangent model solution vector, \(U_{ic}'\) and \(U_{bc}'\) represent, respectively, the values of \(Y'\) at the initial time and the values of \(Y'\) on the physical boundary. If we note \(P\) the discretized solution of the adjoint model, it is easy to see that the discretized adjoint system is given as the solution of the equation

$$
A^*P = \hat{Y} - Y.
$$

Then, the gradient can be expressed as

$$
\nabla J(\Psi) = -B_{bc}^*P.
$$

If we discretize the models (1), (10) and the gradient (13) without a lot of care, we may get a bad discretized gradient (cf., Figure 1). For example, let us discretize the first equation of the model (1) with an explicit forward Euler scheme in time and a centered scheme in space as follows:

For \(n = 0, \ldots, N - 1:\)

For \(i = 1, \ldots, I:\)

$$
\frac{1}{\Delta t} (y_{i}^{n+1} - y_{i}^{n}) - \frac{\nu}{\Delta x^2} (y_{i+1}^{n} - 2y_{i}^{n} + y_{i-1}^{n}) + \frac{1}{4\Delta x} (y_{i+1}^{n+1}y_{i+1}^{n} - y_{i-1}^{n+1}y_{i-1}^{n}) = 0.
$$

The discretization of the linear tangent model (9) that ensues, then takes the form of the equation (16). So, the matrix \(B_{bc}\) is a function of terms \(y_{i}^{n}\) that we don’t have a chance to obtain in discretizing the continuous gradient (13). These coefficients \(y_{i}^{n}\) come from the advection’s term of the Burgers’ equation. Then, the idea is to avoid the discretization of the nonlinear term of the equation on the boundaries, using upwind discretization in the vicinity of the boundaries.
4.1. "Direct Model" Discretization

- Initial conditions: \( y^n_i = u_i, \quad 1 \leq i \leq I \)
- Boundary conditions: \( y^n_0 = \Psi^n_0 \quad \text{and} \quad y^n_{I+1} = \Psi^n_2, \quad 0 \leq n \leq N - 1 \)
- Discretization scheme:

For \( n = 0, \ldots, N - 1 \):

\( i = 1: \)
\[
\frac{1}{\Delta t} (y^{n+1}_i - y^n_i) - \frac{\nu}{\Delta x^2} (y^{n+1}_{i+1} + y^n_{i-1} - 2y^n_i) + \frac{1}{2\Delta x} (y^n_{i+1}y^{n+1}_i - y^n_iy^n_{i+1}) = 0,
\]

\( 2 \leq i \leq I - 1: \)
\[
\frac{1}{\Delta t} (y^{n+1}_i - y^n_i) - \frac{\nu}{\Delta x^2} (y^n_{i+1} - 2y^n_i + y^n_{i-1}) + \frac{1}{2\Delta x} (y^n_{i+1}y^n_i - y^n_{i-1}y^n_{i+1}) = 0,
\]

\( i = I: \)
\[
\frac{1}{\Delta t} (y^{n+1}_i - y^n_i) - \frac{\nu}{\Delta x^2} (y^n_{i+1} - 2y^n_i + y^n_{i-1}) + \frac{1}{2\Delta x} (y^n_{i+1}y^n_i - y^n_{i-1}y^n_{i+1}) = 0.
\]

Here, the matrix \( A \) takes the following form:

\[
A = \begin{pmatrix}
A_d & O & \cdots & \cdots & O \\
A_{bd}(1) & A_d & O & \cdots & \cdots \\
O & A_{bd}(2) & \ddots & \cdots & \cdots \\
\vdots & \vdots & \ddots & \ddots & \cdots \\
O & \cdots & O & A_{bd}(N-1) & A_d
\end{pmatrix},
\]

where

\[
A_d = \begin{pmatrix}
\frac{1}{\Delta t} & 0 & \cdots & 0 \\
0 & \frac{2\nu}{\Delta x^2} - \frac{1}{\Delta t} & \cdots & 0 \\
\vdots & \ddots & \ddots & \cdots \\
0 & \cdots & 0 & \frac{2\nu}{\Delta x^2} - \frac{1}{\Delta t}
\end{pmatrix}, \quad A_{bd}(n) = \begin{pmatrix}
\frac{1}{\Delta t} & c^n_1 & c^n_2 & \cdots & c^n_I \\
0 & a^n_1 & b^n_2 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \cdots \\
0 & \cdots & 0 & a^n_{I-2} & b^n_{I-1} & c^n_I
\end{pmatrix},
\]

with

\[
b^n_i = \frac{2\nu}{\Delta x^2} - \frac{1}{\Delta t} - \frac{1}{\Delta x} y^n_i, \quad 1 \leq i \leq I,
\]

\[
b^n_i = \frac{2\nu}{\Delta x^2} - \frac{1}{\Delta t}, \quad 2 \leq i \leq I - 1,
\]

\[
b^n_i = \frac{2\nu}{\Delta x^2} - \frac{1}{\Delta t} + \frac{1}{\Delta x} y^n_i, \quad 1 \leq i \leq I - 2,
\]

\[
a^n_i = -\frac{\nu}{\Delta x^2} - \frac{2}{\Delta x} y^n_i, \quad 1 \leq i \leq I - 2,
\]

\[
a^n_{I-1} = -\frac{\nu}{\Delta x^2} - \frac{1}{\Delta x} y^n_{I-1},
\]

\[
c^n_2 = \frac{1}{\Delta x} y^n_2 - \frac{\nu}{\Delta x^2}, \quad 3 \leq i \leq I.
\]
4.2. Adjoint Model Discretization

The adjoint model discretization must verify the equality $A^* P = \hat{Y} - Y$. Moreover, the adjoint model is integrated backward in time; thus, initial conditions for this model become "final" conditions.

If we note $p_i^n$ the approximation to $p(i\Delta x, n\Delta t)$, a general discretization can be established for $I \geq 5$ as follows:

- **Final conditions:** $p_i^N = 0, \quad 1 \leq i \leq I$
- **Boundary conditions:** $p_0^0 = p_I^1 = 0, \quad 0 \leq n \leq N$
- **Discretization scheme:**

  - **$i = 1$:**
    \[
    \frac{1}{\Delta t} (p_i^{n-1} - p_i^n) - \frac{\nu}{\Delta x^2} (p_{i+1}^n - 2p_i^n + p_{i-1}^n) - \frac{1}{2\Delta x} y_i^n (2p_i^n + p_{i+1}^n) = \dot{y}_i^n - y_i^n,
    \]
  - **$i = 2$:**
    \[
    \frac{1}{\Delta t} (p_i^{n-1} - p_i^n) - \frac{\nu}{\Delta x^2} (p_{i+1}^n - 2p_i^n + p_{i-1}^n) + \frac{1}{2\Delta x} y_i^n (2p_i^n - p_{i+1}^n) = \dot{y}_i^n - y_i^n,
    \]
  - **$3 \leq i \leq I - 2$:**
    \[
    \frac{1}{\Delta t} (p_i^{n-1} - p_i^n) - \frac{\nu}{\Delta x^2} (p_{i+1}^n - 2p_i^n + p_{i-1}^n) + \frac{1}{2\Delta x} y_i^n (p_{i-1}^n - p_{i+1}^n) = \dot{y}_i^n - y_i^n,
    \]
  - **$i = I - 1$:**
    \[
    \frac{1}{\Delta t} (p_i^{n-1} - p_i^n) - \frac{\nu}{\Delta x^2} (p_{i+1}^n - 2p_i^n + p_{i-1}^n) + \frac{1}{2\Delta x} y_i^n (p_{i-1}^n - 2p_{i+1}^n) = \dot{y}_i^n - y_i^n,
    \]
  - **$i = I$:**
    \[
    \frac{1}{\Delta t} (p_i^{n-1} - p_i^n) - \frac{\nu}{\Delta x^2} (p_{i+1}^n - 2p_i^n + p_{i-1}^n) + \frac{1}{2\Delta x} y_i^n (p_{i-1}^n + 2p_{i+1}^n) = \dot{y}_i^n - y_i^n.
    \]

Notice that this discretization is classical for all terms except for the "adjoint advection" term. This comes from the adopted discretization of the advection term in the direct model.

This way of discretizing the two models (1) and (10) gives, with the formula (18), the following discretized gradient

\[ \nabla \hat{J}(\Psi) = -\frac{\nu}{\Delta x^2} (p_1^0, p_1^1, \ldots, p_1^{N-1}, p_I^0, p_I^1, \ldots, p_I^{N-1})^*, \quad (20) \]

which corresponds to a discretization of the formula (13) given by:

- $-\nu \left( \frac{\partial p}{\partial x} \right)_n^0 = -\frac{\nu}{\Delta x} (p_1^0 - p_1^n) = -\frac{\nu}{\Delta x} p_1^n, \quad 0 \leq n \leq N - 1,$

and

- $\nu \left( \frac{\partial p}{\partial x} \right)_{I+1}^n = \frac{\nu}{\Delta x} (p_{I+1}^n - p_{I+1}^0) = -\frac{\nu}{\Delta x} p_1^n, \quad 0 \leq n \leq N - 1.$

5. NUMERICAL RESULTS

Before beginning the assimilation experiments, we have tested the validity of the obtained gradient. This test consists of checking the Taylor formula in the direction of the gradient computed by the adjoint model. More precisely, it consists in verifying the equality:

\[ \lim_{h \to 0} \frac{J(\Psi + h\nabla J(\Psi)) - J(\Psi)}{(\nabla J(\Psi), h\nabla J(\Psi))} = 1. \quad (21) \]

The variations of this ratio, noted $R(h)$, have been plotted versus $\log(h)$ in Figure 1, $h$ varying between $10^{-1}$ and $10^{-15}$. For example, using the discretization (19) for the "direct model,"
Figure 1 (Case 1) shows that a discretization made without care on the continuous formulation described in Sections 2 and 3, gives a bad approximation of the gradient. On the other hand, with the "direct model" discretization described in Section 4, the limit defined by the formula (21) converges effectively to 1 (Case 2). However, for \( h \approx 10^{-13} \), the ratio \( R(h) \) suddenly increases because the rounding off errors are no further negligible and the limit of \( R(h) \) is no more significant for so small values of \( h \).

To verify the efficiency of the method described in the last sections, the optimization process has been checked through three numerical experiments with, for each of them, the following coefficient's values:

\[
\nu = 10^{-2} \frac{m^2}{s}, \quad L = 1 \text{ m}, \quad T = 1 \text{ s}, \quad N = 100, \quad I = 69.
\]

Initial conditions have been set to known values throughout these numerical experiments:

\[ y(x, 0) = u(x) = 0, \quad \text{for } x \in (0, L). \]

In the three experiments, the used values of a "true control" denoted by \( \Psi_{\text{true}} = (\Psi_1, \Psi_2) \), will be arbitrary defined as

\[ \Psi_1(t) = 1 - t, \quad \text{for } t \in (0, T), \quad \Psi_2(t) = t - 1, \quad \text{for } t \in (0, T). \]

The goal of the following experiments is to find a control, and thus, the corresponding solution of the "direct model," in order to minimize the difference between this solution and a state of information (dataset) concerning it.

**EXPERIMENT 1.** Using the "direct model" discretization described in Section 4, a dataset is simulated by computing a solution of the model (1) corresponding to the given control \( \Psi_{\text{true}} \).

The aim of the experiment is to identify the "optimal control" denoted by \( \Psi_{\text{opt}} \) when the optimization procedure is started with an "initial control" denoted by \( \Psi_{\text{init}} \) different from \( \Psi_{\text{true}} \). Here, \( \Psi_{\text{init}} \) is obtained by adding a Gaussian noise to \( \Psi_{\text{true}} \).

Figure 2a shows the controls \( \Psi_{\text{true}}, \Psi_{\text{init}} \) and the computed control \( \Psi_{\text{opt}} \). Variations of \( J \) and of its gradient norm during this experiment are shown in Figure 2b where a decimal logarithmic representation has been adopted.

**EXPERIMENT 2.** Using this time the discretization (19), a dataset is simulated by computing a solution of the model (1). Then, the same protocol as for Experiment 1 is used. Figure 3a shows the controls \( \Psi_{\text{true}}, \Psi_{\text{init}} \) and the computed control \( \Psi_{\text{opt}} \). Variations of \( J \) and of its gradient norm during this experiment are shown in Figure 3b.
(a) Comparison between the controls $\Psi_{\text{true}}$ and $\Psi_{\text{opt}}$ in the case of Experiment 1.

(b) Variations of the scaled cost function logarithm $\log_{10} J$ (left gradations) and of the scaled function $\log_{10}(\text{norm}(\nabla J))$ (right gradations) with the number of iterations, in the case of Experiment 1.

Figure 2.
Burgers' Equation

(a) Comparison between the controls \( \Psi_{\text{true}} \) and \( \Psi_{\text{opt}} \) in the case of Experiment 2.

(b) Variations of the scaled cost function logarithm \( \log_{10} J \) (left gradations) and of the scaled function \( \log_{10}(\text{norm}(\nabla J)) \) (right gradations) with the number of iterations, in the case of Experiment 2.

Figure 3.
(a) Variations of the founded optimal control $\Psi_{\text{opt}}$ starting the optimization procedure from $\Psi_{\text{init}}$, in the case of Experiment 3.

(b) Variations of the scaled cost function logarithm $\log_{10} J$ (left gradations) and of the scaled function $\log_{10}(\text{norm}(\nabla J))$ (right gradations) with the number of iterations, in the case of Experiment 3.

Figure 4.
Experiment 3. In this case, the dataset corresponds to a noisy computed solution of the model (1). It can be noticed that the choice of the model (1) discretization doesn’t matter here. The dataset becomes, thus, obviously not a solution of the model and the aim of this experiment is then to identify the “optimal control” $\Psi_{\text{opt}}$ and so the “optimal solution” that corresponds to it. The optimization procedure is started here with a simple “initial control” $\Psi_{\text{init}}$, for example equal to $\Psi_{\text{true}}$. Figure 4a shows the control $\Psi_{\text{init}}$ and the computed control $\Psi_{\text{opt}}$. Variations of $J$ and of its gradient norm during this experiment are shown in Figure 4b.

6. DISCUSSION

Concerning Experiment 1, Figure 2a shows that the true and the computed optimal controls are virtually indistinguishable. The average value of their relative difference is of order of $10^{-3}$. Moreover, during the minimization procedure, the numerical value of the cost function is divided by about $10^6$. It can be noticed that the functional $J$ decreases continuously from one iteration step to another, although the main decrease of the functional occurs in the first iterations. To illustrate it in a visible manner (Figure 2b), at the beginning as well as at the end of the procedure, a decimal logarithm representation has been chosen. The decrease of $J$ is consistent with the global decrease of the norm of the gradient of the cost function by a factor of order four. However, it can be seen that the decrease of the norm of the gradient of the cost function is not regular. This is due to the intrinsic properties of the minimization algorithm used by Gilbert and Nocedal [12] and applied here. So, in the case where the model (1) discretization used to simulate the dataset identifies with the “direct model” discretization, a good accuracy is obtained by boundary control.

About Experiment 2, the same comments concerning the decrease of $J$ and its gradient norm can be made (Figure 3b). On the other hand, Figure 3a shows that the two controls $\Psi_{\text{true}}$ and $\Psi_{\text{opt}}$ differ notably in the first time steps only. This can be explained as follows. The discretization scheme used to simulate the dataset in Experiment 2 is convergent, and its truncature error is of order of $O(\Delta t, \Delta x^2)$. The “direct model” discretization used during the optimization procedure doesn’t take into account the influence of the boundary conditions in the advection term. The boundary information is only introduced in the domain by the diffusion term. This discretization scheme is also convergent, but its truncature error is of order of $O(\Delta t, \Delta x)$. To compensate this lack of accuracy, the optimization procedure produces an optimal control different to the “true control” in the first time steps as can be seen in Figure 3a.

Then, the first two experiments show that when the dataset identifies with a solution of Burgers’ equation, it is possible to obtain a fairly good decrease of the cost function (Figures 2b and 3b). In this way, the optimal solution corresponding to the founded optimal control, agrees perfectly with the dataset.

A more constraining situation arises when the dataset is not coherent with a solution of Burgers’ equation. This is exemplified in Experiment 3, using a zero mean noise with a standard deviation of order of the higher value of the discrete solutions. In this case, Figure 4b shows that the cost function does not have a similar decrease as in Experiments 1 and 2. During the optimization procedure, it seems that the time-space distribution of the values of the solutions has only changed in order to approach as much as possible the fixed dataset. Concerning the founded “optimal control,” a very chaotic profile (Figure 4a) can be observed according to observations (dataset) issued from a noisy solution of the “direct model.”

7. SUMMARY AND CONCLUSION

A method has been described to optimize boundary conditions in a numerical model of nonlinear advection-diffusion. The method is based on an optimal control approach whereby a norm of the discrepancies between computed and measured values is minimized. The numerical procedure we have used for the optimization is a variable storage (or limited memory) quasi-Newton
method. Concentrating on numerical aspects, we have shown how the method applies in the simple case of a viscous Burgers' equation. It is found, using a matrix approach as a guideline, that boundary values must not be involved in the discretized version of "direct model" nonlinear advection terms. So, a particular discretization of the equation is performed in order to insure a discretized gradient expression which can be directly deduced from its continuous counterpart. This requires slight modification of the numerical scheme of the model, and it is thought these rules are quite general in the case of boundary control problems.

This study about the viscous Burgers' equation constitutes a preliminary stage for the development of an adjoint model of a three-dimensional model of circulation and dispersion in coastal physical oceanography. We can then wonder about the general nature of the method exposed in this paper. It can be noticed that the problems to find the "good" gradient arise from some of the nonlinear terms of the equation. More precisely, it is only the advection terms that prevent a classical discretization from giving a good gradient. More generally, the problems come from nonlinear terms including partial derivatives. In this sense, the method exposed in this paper seems to be applicable to other partial differential equations. Motivated by applications in engineering and environmental sciences, the method should be easily applicable to the particular case of the Navier-Stokes equations. More particularly, the implementation of the method to a three-dimensional model of circulation and dispersion in coastal physical oceanography will permit proper management of the open boundary conditions which are often poorly taken into account in the numerical models.

REFERENCES