An alternating descent method for the optimal control of the inviscid Burgers equation in the presence of shocks [*]

Carlos CASTRO [†], Francisco PALACIOS [‡] and Enrique ZUAZUA [§]

June 20, 2007

Abstract

We introduce a new optimization strategy to compute numerical approximations of minimizers for optimal control problems governed by scalar conservation laws in the presence of shocks. We focus on the $1 - d$ inviscid Burgers equation. We first prove the existence of minimizers and, by a Γ-convergence argument, the convergence of discrete minima obtained by means of numerical approximation schemes satisfying the so called one-sided Lipschitz condition (OSLC). Then we address the problem of developing efficient descent algorithms. We first consider and compare the existing two possible approaches. The first one, the so-called discrete approach, based on a direct computation of gradients in the discrete problem and the so-called continuous one, where the discrete descent direction is obtained as a discrete copy of the continuous one. When optimal solutions have shock discontinuities, both approaches produce highly oscillating minimizing sequences and the effective descent rate is very weak. As a remedy we propose a new method that uses the recent developments of generalized tangent vectors and the linearization around discontinuous solutions. We develop a new descent stratagey, that we shall call *alternating descent method*, distinguishing descent directions that move the shock and those that perturb the profile of the solution away of it. As we shall see, a suitable alternating combination of these two classes of descent directions allows building very efficient and fast descent algorithms.

Contents

[†]Dep. Matemática e Informática, ETSI Caminos, Canales y Puertos, Univ. Politécnica de Madrid, 28040 Madrid, Spain. **carlos.castro@upm.es**

[‡]Instituto Nacional de Técnica Aeroespacial. Torrejón de Ardoz, 28850 Madrid, Spain **palaciosg@inta.es**

[§]Dep. Matemáticas & IMDEA-Matemáticas, Facultad de Ciencias, Univ. Autónoma de Madrid, 28049 Madrid, Spain. **enrique.zuazua@uam.es**.

1

1 Introduction

Optimal control for hyperbolic conservation laws is a difficult topic which requires a considerable computational effort. In the last years a number of methods have been proposed to reduce the computational cost and to render this type of problems affordable.

The aim of this paper is to present and discuss the main existing approaches to these problems in the context of $1 - d$ scalar conservation laws. We focus on the inviscid Burgers equation although most of our results extend to more general equations with convex fluxes. We show that the descent methods developed on the basis of the existing approaches produce highly oscillating minimizing sequences whose convergence is very slow. We then introduce a new optimization strategy that we shall refer to as *alternating descent method*, well adapted to the presence of discontinuities in the solutions, and that, as we shall see, exhibits very good descent properties. Indeed, as shown in a number of numerial experiments, the new method we propose is much more robust and efficient in a significantly smaller number of iterations of the descent method.

To be more precise, given a finite time horizon $T > 0$, we consider the following inviscid Burgers equation:

$$\begin{cases} \partial_t u + \partial_x(\frac{u^2}{2}) = 0, & \text{in } \mathbb{R} \times (0, T), \\ u(x, 0) = u^0(x), & x \in \mathbb{R} \end{cases} \qquad (1.1)$$

Given a target $u^d \in L^2(\mathbb{R})$) we consider the cost functional to be minimized $J : L^1(\mathbb{R}) \to \mathbb{R}$,

defined by

$$J(u^0) = \int_{\mathbb{R}} |u(x,T) - u^d(x)|^2 \, dx, \tag{1.2}$$

where $u(x,t)$ is the unique entropy solution of (1.1).

Although this paper is devoted to this particular choice of J, most of our analysis and numerical algorithms can be adapted to many other functionals and control problems (we refer for instance to [20] where the control variable is the nonlinearity of the scalar conservation law).

We also introduce the set of admissible initial data $\mathcal{U}_{ad} \subset L^1(\mathbb{R})$, that we define later in order to guarantee the existence of the following optimization problem: Find $u^{0,\min} \in \mathcal{U}_{ad}$ such that

$$J(u^{0,\min}) = \min_{u^0 \in \mathcal{U}_{ad}} J(u^0). \tag{1.3}$$

This is one of the model optimization problems that is often addressed in the context of optimal aerodynamic design, the so-called inverse design problem (see, for example, [12]).

As we will see, existence of minimizers is easily established under some natural assumptions on the class of admissible data $\mathcal{U}_{ad}$ using well known well-posedness and compactness properties of the inviscid Burgers equation. However, uniqueness is false, in general, due, in particular, to the possible presence of discontinuities in the solutions of (1.1).

In practical applications and in order to perform numerical computations and simulations one has to replace the continuous optimization problem above by a discrete approximation. It is then natural to consider a discretization of system (1.1) and the functional J. If this is done in an appropriate way, the discrete optimization problem has minimizers that are often taken, for small enough mesh-sizes, as approximations of the continuous minimizers. There are however few results in the context of hyperbolic conservation laws proving rigorously the convergence of the discrete optimal controls towards the continuous ones, as the mesh-size goes to zero.

One of the first goals of this paper is to provide such a convergence result based on a fine use of the known properties of monotone conservative schemes and more precisely of those satisfying the so called one-sided Lipschitz condition (OSLC).

In the following we will denote by u^Δ the approximation of u obtained by a suitable discretization of system (1.1) with mesh-sizes Δx and Δt for space-time discretizations. We also denote by J^Δ a discretization of J and by $\mathcal{U}_{ad,\Delta}$ a discrete version of the set of admissible controls $\mathcal{U}_{ad}$, and consider the approximate discrete minimization problem

$$J^\Delta(u_\Delta^{0,\min}) = \min_{u_\Delta^0 \in \mathcal{U}_{ad,\Delta}} J^\Delta(u_\Delta^0).$$

For fixed values of the mesh-size Δ, the existence of minimizers for this discrete problem is often easy to prove. But, even in that case, their convergence as $\Delta \to 0$ is harder to show. This will be done, as mentioned above, in the class of OSLC schemes that guarantee the needed compactness properties.

From a practical point of view it is however more important to be able to develop efficient algorithms for computing accurate approximations of the discrete minimizers. This is often not

an easy matter due to the high number of the parameters involved, the lack of convexity of the functional under consideration, etc.

The most efficient methods to approximate minimizers are the gradient methods (steepest descent, conjugate gradient, etc.) although they hardly distinguish local or global minimizers. This is an added difficulty in problems with many local minima, a fact that cannot be excluded in our optimization problem, due to the nonlinear dependence of the state on the initial datum. However we will not address this problem here. We shall rather focus on building efficient descent algorithms.

Descent algorithms are iterative processes. In each step of the iteration the descent direction is built by means of the gradient of the functional with respect to the controls, in this case the initial datum. Then the sensibility of the discrete cost functional J^Δ with respect to u_Δ^0 depends on the sensitivity of the solution of the numerical scheme, used to discretize (1.1), with respect to u_Δ^0, which in fact involves an infinite number of parameters, one for each mesh-point. Thus, in practice, computing the sensibility of the cost functional requires to differentiate this numerical scheme with respect to the initial datum.

When the numerical scheme under consideration is differentiable this is easy to do and the classical adjoint state method provides a significant shortcut when computing the derivatives with respect to all control parameteres (mesh-points in this case). We illustrate this in section 6.1 below.

But for large complex systems, as Euler equations in higher dimensions, the existing most efficient numerical schemes (upwind, Godunov, Roe, etc.) are not differentiable (see for example [17] or [19]). In this case, the gradient of the functional is not well defined and there is not a natural and systematic way to compute its variations.

In face of this difficulty, it would be natural to explore the possible use of non-smooth optimization techniques. But this does not seem to had been done and is out of the scope of this paper. By the contrary, the following two other approaches have been developped: The first one, is based on the use of *automatic differentiation*, which basically consists in differentiating the numerical method (even if it is not differentiable!), differentiating each line in the code (see for instance [24]). This approach often produces oscillations which are due precisely to the lack of differentiability. The second one, the so-called *continuous approach*, consists in proceeding in two steps as follows: One first linearizes the continuous system (1.1) to obtain a descent direction of the continuous functional J and then takes a numerical approximation of this descent direction with the discrete values provided by the numerical scheme. Of course the validity of this approximation as a descent direction for the discrete problem is not at all assured either.

But the continuous approach has to face another major drawback, when solutions develop shock discontinuities, as it is the case in the context of the hyperbolic conservation laws we are considering here. Indeed, the formal differentiation of the continuous state equation (1.1) yields

$$\partial_t \delta u + \partial_x (u \delta u) = 0.$$

But this is only justified when the state u on which the variations are being computed, is smooth enough. In particular, it is not justified when the solutions are discontinuous since

singular terms may appear on the linearization over the shock location. Accordingly in optimal control applications one also needs to take into account the sensitivity fo the shock location. This has been studied by different authors with different approaches (see [27], [14] or [8]). Roughly speaking, the main conclusion of that analysis is that the classical linearized system for the variations of the solutions must be complemented with some new equations for the sensitivity of the shock position.

These issues had been the object of intensive research, as indicated above, but there is not a systematic receipt about how to use this new notions of lineariations to implement efficient descent methods. This is due, to some extent, to the fact that two related but different issues have been treated simultaneously without sufficiently distinguishing one from another: a) The lack of regularity of the solutions of the continuous state equation that makes the formal linearization above difficult to justify and that adds some unexpected terms to the classical derivative of the functional, to take into account the possible contribution of jump discontinuities, and b) the numerical schemes being non-differentiable.

Therefore, the second goal of this paper is to build efficient descent algorithms taking into account (and advantage of) the possible presence of shock discontinuities on solutions. To be more precise, this paper is aimed to clarify these different aspects of the problem, proposing an overall new strategy to build descent algorithms. We thus use the existing results that allow deriving the correct linearization of the system in the presence of shocks. We then derive the adjoint system which contains an internal boundary condition along the shock which has been referred in the literature as an internal boundary condition for the adjoint system (see [12] where the quasi one dimensional stationary Euler equations are considered). From the numerical point of view, the use of this adjoint system makes methods more efficient, since it takes into account explicitly the sensibility of the solution with respect to shock variations. But if applied directly with the aid of the notion of generalized tangent vectors ([7], [8]) the descent method, in each step of the iteration, adds new discontinuities to the state, thus yielding solutions with increasing complexity.

To overcome this difficulty, in the present article, we propose a new *alternating descent method* that in the iterative process alternates descent directions altering the shock position and those that do not move it and only affect the shape of solutions away form the shock, in such a way that the number of shocks does not increase, thus keeping the overall complexity of solutions limited.

The detailed analysis of the continuous linearized equations in the presence of shocks is only well-understood in a certain number of situations: $1 - d$ scalar conservation laws ([3], [4]) with the aid of notions of duality and reversible measure valued solutions, multi-dimensional scalar conservation was subject to one-sided Lipschitz conditions ([5]), and also when the shock is a priori known to be located on a single regular curve, or a regular manifold in higher dimensions (see [7], [8] and [27] for $1 - d$ problems, and in the multi-dimensional case [22] and [23] where general systems of conservation laws in higher dimensions are considered). We also refer to [14] for an analysis of the linearization for multi-dimensional perturbations of $1 - d$ scalar conservation laws. But the general principles leading to the alternating descent method we propose here are of much widder application, as we shall develop in forthcoming articles

devoted to multi-dimensional scalar conservation laws and to the quasi1D-Euler equations.

The rest of this paper is divided as follows: in section 2 we study the existence of minimizers for the continuous problem (1.3). In section 3 we analyze the convergence of discrete minimizers obtained by discretizing the cost function and system (1.1) by mans of OSLC schemes. In section 4 we recall some known results on the sensitivity of the continuous functional by linearizing system (1.1) in the presence of a shock. In section 5 we propose the new *alternating descent method* based on the continuous approach. In section 6 we discuss more classical descent strategies based on both the continuous and discrete approach. In section 7 we present some numerical experiments which confirm the efficiency of the new strategy introduced in this paper. In section 8 we sketch the algorithms.

2 Existence of minimizers

In this section we prove that, under certain conditions on the set of admissible initial data $\mathcal{U}_{ad}$, there exists at least one minimizer of J given in (1.2).

To simplify the presentation we consider the class of admissible initial data $\mathcal{U}_{ad}$:

$$\mathcal{U}_{ad} = \{f \in L^\infty(\mathbb{R}),\ supp(f) \subset K,\ ||f||_\infty \leq C\},$$

where $K \subset \mathbb{R}$ be a bounded interval and $C > 0$ a constant. Note however that the same theoretical results and descent strategies can be applied in a much widder class of admissible sets.

Theorem 1. *Assume that $u^d \in L^2(\mathbb{R})$ and that $\mathcal{U}_{ad}$ is as above. Then the minimization problem*

$$\min_{u^0 \in \mathcal{U}_{ad}} J(u^0), \tag{2.1}$$

has at least one minimizer $u^{0,\min} \in \mathcal{U}_{ad}$.

Uniqueness is in general false for this optimization problem.

Proof. We first prove existence. Let $u_n^0 \in \mathcal{U}_{ad}$ be a minimizing sequence of J. Then u_n^0 is bounded in L^∞ and there exists a subsequence, still denoted by u_n^0, such that $u_n^0 \rightharpoonup u_*^0$ weakly-* in L^∞. Moreover, $u_*^0 \in \mathcal{U}_{ad}$.

Let $u_n(x,t)$ and $u_*(x,t)$ be the entropy solutions of (1.1) with initial data u_n^0 and u_*^0 respectively, and assume that

$$u_n(\cdot, T) \to u_*(\cdot, T), \quad \text{in } L^2(\mathbb{R}). \tag{2.2}$$

Then, clearly

$$\inf_{u^0 \in \mathcal{U}_{ad}} J(u^0) = \lim_{n\to\infty} J(u_n^0) = J(u_*^0),$$

and we deduce that u_*^0 is a minimizer of J.

Thus, the key point is to prove the strong convergence result (2.2). Two main steps are necessary to do it: a) The relative compactness of $u_n(\cdot, T)$ in L^2. Taking the structure of $\mathcal{U}_{ad}$ into account and using the maximum principle and the finite velocity of propagation that

entropy solutions fulfill, it is easy to see that the support of all solutions at time $t = T$, is uniformly included in the same compact set of $\mathbb{R}$. Thus, it is sufficient to prove compactness in L^2_{loc}. This is obtained from Oleinik's one-sided Lipschitz condition

$$\frac{u(x,t) - u(y,t)}{x - y} \leq \frac{1}{t}, \tag{2.3}$$

which guarantees in fact a uniform bound of the BV-norm of $u_n(\cdot, T)$, locally in space (see [6]). The needed compactness property is then a consequence of the compatcness of embedding $BV(I) \subset L^2(I)$, for all bounded interval I. b) The identification of the limit as the solution of (1.1) with initial datum u^0_*. This can be proved using similar compactness arguments passing to the limit in the variational formulation of (1.1). We refer to [10] for a detailed description of this imit process in the more delicate case where the initial data converge to a Dirac delta.

This completes the prove of the existence of minimizers.

We now prove that the uniqueness of the minimizer is in general false for this type of optimization problems. In fact, we prove that there are target functions u^d for which there exist two different minimizers u^0_1 and u^0_2 such that the corresponding solutions u_j, $j = 1, 2$ satisfy $u_j(T) = u^d$, $j = 1, 2$ in such a way that the minimal value of J vanishes. This is always possible as soon as we deal with solutions having shocks. For example,

$$u_1(x,t) = \left\{ \begin{array}{ll} 1 \text{ if } x \leq t/2, \\ 0 \text{ if } x > t/2, \end{array} \right. \qquad u_2(x,t) = \left\{ \begin{array}{ll} 1 \text{ if } x \leq t - 1/2, \\ x + (1/2 - x)t \text{ if } t - 1/2 \leq x \leq 1/2, \\ 0 \text{ if } x > 1/2, \end{array} \right.$$

are two different entropy solutions for which $u_1(x,T) = u_2(x,T)$ at $T = 1$. Thus if we take $T = 1$ and $u^d(x) = u_1(x,1)$ then there exist two different initial data $u^0_1(x,0)$ and $u^0_2(x,0)$ for which J attains its minimum. Note that this is impossible within the class of smooth solutions by backwards uniqueness.

Note that u^d as above does not belong to $L^2(\mathbb{R})$ but, the same argument is valid is u^d is truncated to take zero values at infinity. $\qquad \square$

Remark 1. *The above proof is in fact quite general and it can be adapted to other optimization problems with different functionals and admissible sets. In particular, using Oleinik's one-sided Lipschitz condition (2.3), one can also consider admissible sets of the form $\mathcal{U}_{ad} = \{f \in L^1(\mathbb{R}), \; supp(f) \subset K, ||f||_1 \leq C\}$.*

3 The discrete minimization problem

The purpose of this section is to show that discrete minimizers obtained through a numerical scheme to approximate (1.1) satisfying the so-called OSLC property, p converge to a minimizer of the continuous problem as the mesh-size tends to zero. This justifies the usual engineering practice of replacing the continuous functional and model by discrete ones to compute an approximation of the continuous minimizer.

Let us introduce a mesh in $\mathbb{R} \times [0, T]$ given by $(x_j, t^n) = (j\Delta x, n\Delta t)$ $(j = -\infty, ..., \infty;$ $n = 0, ..., N+1$ so that $(N+1)\Delta t = T)$, and let u_j^n be a numerical approximation of $u(x_j, t^n)$ obtained as solution of a suitable discretization of the equation (1.1).

Let us consider the following aproximation of the functional J in (1.2):

$$J^\Delta(u_\Delta^0) = \frac{\Delta x}{2} \sum_{j=-\infty}^{\infty} (u_j^{N+1} - u_j^d)^2, \tag{3.1}$$

where $u_\Delta^0 = \{u_j^0\}$ is the discrete initial datum and $u_\Delta^d = \{u_j^d\}$ is the discretization of the target u^d at x_j, respectively. A common choice is to take,

$$u_j^d = \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_{j+1/2}} u^d(x) dx, \tag{3.2}$$

where $x_{j\pm1/2} = x_j \pm \Delta x/2$.

We also introduce an approximation of the class of admissible initial data $\mathcal{U}_{ad}$ denoted by $\mathcal{U}_{ad}^\Delta$ and constituted by sequences $\varphi_\Delta = \{\varphi_j\}_{j \in \mathbb{Z}}$ for which the associated piecewise constant interpolation function, that we still denote by φ_Δ, defined by

$$\varphi_\Delta(x) = \varphi_j, \quad x_{j-1/2} < x < x_{j+1/2},$$

satisfies $\varphi_\Delta \in \mathcal{U}_{ad}$. Obviously, $\mathcal{U}_{ad}^\Delta$ coincides with the class of discrete vectors with support on those indices j such that $x_j \in K$ and for which the discrete L^∞-norm is bounded above by the same constant C.

Finally we introduce a 3-point conservative numerical approximation scheme for (1.1):

$$u_j^{n+1} = u_j^n - \lambda \left(g_{j+1/2}^n - g_{j-1/2}^n \right) = 0, \quad \lambda = \frac{\Delta t}{\Delta x}, \qquad j \in \mathbb{Z}, \ n = 0, ..., N, \tag{3.3}$$

where,

$$g_{j+1/2}^n = g(u_j^n, u_{j+1}^n),$$

and g is the numerical flux. These schemes are consistent with (1.1) when $g(u, u) = u^2/2$.

When the function $H(u, v, w) = v - \lambda(g(u, v) - g(v, w))$ is a monotone increasing function in each argument the scheme is said to be monotone. These are particularly interesting schemes since the discrete solutions obtained with them converge to weak entropy solutions of the continuous conservation law, as the discretization parameters tend to zero, under a suitable CFL condition (see [15], Chp.3, Th. 4.2).

For each $h > 0$ it is easy to see that the discrete analogue of Theorem 1 holds. In fact this is automatic in the present setting since $\mathcal{U}_{ad}^\Delta$ only involves a finite number of mesh-points. But passing to the limit as $h \to 0$ requires a more careful treatment. In fact, for that to be done, one needs to assume that the scheme under consideration satisfies the so-called one-sided Lipszhitz condition (OSLC), which is a discrete version of Oleinik's condition above:

$$\frac{u_{j+1}^n - u_j^n}{\Delta x} \leq \frac{1}{n\Delta t}. \tag{3.4}$$

We then consider the following discrete minimization problem: Find $u_\Delta^{0,\min}$ such that

$$J^\Delta(u_\Delta^{0,\min}) = \min_{u_\Delta^0 \in \mathcal{U}_{ad}^\Delta} J^\Delta(u_\Delta^0). \tag{3.5}$$

It is by now well known that Godunov's, Lax-Friedrichs and Engquits-Osher schemes satisfy this OSLC condition. We refer to [6] for a discussion of this issue and also for the construction of a second order MUSCL method satisfying OSLC.

Theorem 2. *Assume that u_Δ^n is obtained by a conservative monotone numerical scheme consistent with (1.1) and satisfying the OSLC condition.*
   Then:

- *For all $\Delta x, \Delta t > 0$, the discrete minimization problem (3.5) has at least one solution $u_\Delta^{0,min} \in \mathcal{U}_{ad}^\Delta$.*

- *Any accumulation point of $u_\Delta^{0,\min}$ with respect to the weak$-*$ topology in L^∞, as $\Delta x, \Delta t \to 0$ (with $\Delta t / \Delta x = \lambda$ fixed and under a suitable CFL condition), is a minimizer of the continuous problem (2.1).*

The existence of discrete minimizers in the first statement of the Theorem is obvious in this case since we are dealing with a finite dimensional problem. Actually, at this point the OSLC property is not necessary. However, in more general situations (for other classes of discrete admissible controls) we could apply the same argument as in the proof of Theorem 1 based on the OSLC property and the BV estimates this yields (see [6]).

The second statement is less trivial. It requires, definitely, of the OSLC property to guarantee the compactness of numerical solutions as $\Delta x, \Delta t \to 0$.

Remark 2. *The most frequently used conservative 3-point numerical schemes derived to approximate (1.1) satisfy the hypotheses of Theorem 2. This is in particular the case of the Lax-Friedrichs, Engquist-Osher or Godunov schemes whose numerical fluxes for Burgers equation are given, respectively, by*

$$g^{LF}(u,v) = \frac{u^2 + v^2}{4} - \frac{v - u}{2\lambda}, \tag{3.6}$$

$$g^{EO}(u,v) = \frac{u(u + |u|)}{4} + \frac{v(v - |v|)}{4}, \tag{3.7}$$

$$g^G(u,v) = \begin{cases} \min_{w \in [u,v]} w^2/2, & \text{if } u \leq v, \\ \max_{w \in [u,v]} w^2/2, & \text{if } u \geq v, \end{cases} \tag{3.8}$$

Proof of Theorem 2. The case where Δx and Δt are fixed being trivial, we address the limit problem $\Delta \to 0$.

We follow an standard Γ-convergence argument. The key ingredient is the following continuity property: Assume that $u_\Delta^0 \in \mathcal{U}_{ad}^{\Delta x}$ satisfies $u_\Delta^0 \rightharpoonup u^0$ in $L^\infty(\mathbb{R})$ with respect to the weak$-*$ topology, then

$$J^\Delta(u_\Delta^0) \to J(u^0). \tag{3.9}$$

This is due to the fact that the OSLC condition guarantees uniform local BV bounds on the discrete solutions and the compactness of the embedding and the finite velocity of propagation of solutions in view of the class of initial data $\mathcal{U}_{ad}^{\Delta}$ we are considering (see [6]) which guarantees that $u_{\Delta}^{N+1} \to u(\cdot, T)$ in $L^2(\mathbb{R})$, under the CFL condition that guarantees the convergence of the numerical schemes, and its monotonicity.

Now, let $\hat{u}^0 \in \mathcal{U}_{ad}$ be an accumulation point of $u_{\Delta}^{0,\min}$ with respect to the weak$-*$ topology of L^{∞}. To simplify the notation we still denote by $u_{\Delta}^{0,\min}$ the subsequence for which $u_{\Delta}^{0,\min} \rightharpoonup \hat{u}^0$, weakly$-*$ in $L^{\infty}(\mathbb{R})$, as $\Delta x \to 0$. Let $v^0 \in \mathcal{U}_{ad}$ be any other function. We are going to prove that

$$J(\hat{u}^0) \leq J(v^0). \tag{3.10}$$

To do this we construct a sequence $v_{\Delta}^0 \in \mathcal{U}_{ad}^{\Delta}$ such that $v_{\Delta}^0 \to v^0$, in $L^1(\mathbb{R})$, as $\Delta x, \Delta t \to 0$ (we can consider in particular the approximation in (3.2)).

Taking into account the continuity property (3.9), we have

$$J(v^0) = \lim_{\Delta \to 0} J^{\Delta}(v_{\Delta}^0) \geq \lim_{\Delta \to 0} J^{\Delta}(u_{\Delta}^{0,\min}) = J(\hat{u}^0),$$

which proves (3.10).

Remark 3. *Theorem 2 concerns global minima. However, both the continuous and discrete functionals may possibly have local minima as well. Extending this kind of Γ-convergence result for local minima requires important further developments.*

4 Sensitivity analysis: the continuous approach

In this section we analyze the *continuous approach* to obtain descent directions for the discrete functional J^{Δ}.

We divide this section in four more subsections: More precisely, in the first one we consider the case where the solution u of (1.1) has no shocks, in the second and third subsections we analyze the sensitivity of the solution and the functional in the presence of a single shock located on a regular curve, and finally, in the last subsection we discuss various possibilities to obtain discrete descent directions.

4.1 Sensitivity without shocks

In this subsection we give an expression for the sensitivity of the functional J with respect to the initial datum based on a classical adjoint calculus for smooth solutions. First we present a formal calculus and then we show how to justify it when dealing with a classical solution for (1.1), i.e. when there are no discontinuities.

Let $C_0^1(\mathbb{R})$ be the set of C^1 functions with compact support and let $u^0 \in C_0^1(\mathbb{R})$ be a given datum for which there exists a classical solution $u(x, t)$ of (1.1) in $(x, t) \in \mathbb{R} \times [0, T]$, which can be extended to a classical solution in $t \in [0, T + \tau]$ for some $\tau > 0$. Note that this imposes some restrictions on u^0 other than being smooth. More precisely, we must have

$T + \tau > \max_x[1/u'_0(x)]$ to guarantee that two different characteristics do not meet in the time interval $[0, T + \tau]$. Let $\delta u^0 \in C_0^1(\mathbb{R})$ be any possible variation of the initial datum u^0. Due to the finite speed of propagation, this perturbation will only affect the solution in a bounded set of $(x, t) \in \mathbb{R} \times [0, T]$. This simplifies the argument below that applies in a much more general setting provided solutions are smooth enough.

Then for $\varepsilon > 0$ sufficiently small, the solution $u^\varepsilon(x, t)$ corresponding to the initial datum

$$u^{\varepsilon,0}(x) = u^0(x) + \varepsilon \delta u^0(x), \tag{4.1}$$

is also a classical solution in $(x, t) \in \mathbb{R} \times (0, T)$ and $u^\varepsilon \in C^1(\mathbb{R} \times [0, T])$ can be written as

$$u^\varepsilon = u + \varepsilon \delta u + o(\varepsilon), \quad \text{with respect to the } C^1 \text{ topology}, \tag{4.2}$$

where δu is the solution of the linearized equation,

$$\begin{cases} \partial_t \delta u + \partial_x (u \delta u) = 0, \\ \delta u(x, 0) = \delta u^0(x). \end{cases} \tag{4.3}$$

Let δJ be the Gateaux derivative of J at u^0 in the direction δu^0. We have

$$\delta J = \int_{\mathbb{R}} (u(x, T) - u^d(x)) \delta u(x, T) \, dx, \tag{4.4}$$

where δu solves the linearized system above (4.3). Now, we introduce the adjoint system

$$\begin{cases} -\partial_t p - u \partial_x p = 0, \\ p(x, T) = p^T(x), \end{cases} \tag{4.5}$$

where $p^T = u(x, T) - u^d(x)$. Multiplying the equations satisfied by δu by p, integrating by parts, and taking into account that p satisfies (4.5), we easily obtain

$$\int_{\mathbb{R}} (u(x, T) - u^d(x)) \delta u(x, T) \, dx = \int_{\mathbb{R}} p(x, 0) \delta u^0 \, dx. \tag{4.6}$$

Thus, δJ in (4.4) can be written as,

$$\delta J = \int_{\mathbb{R}} p(x, 0) \delta u^0(x) \, dx. \tag{4.7}$$

This expression provides an easy way to compute a descent direction for the continuous functional J, once we have computed the adjoint state. We just take

$$\delta u^0 = -p(x, 0). \tag{4.8}$$

Under the assumptions above on u^0, u, δu, and p can be obtained from their data $u^0(x)$, $\delta u^0(x)$ and $p^T(x)$ by using the characteristic curves associated to (1.1). For the sake of completeness we briefly explain this below.

11

The characteristic curves associated to (1.1) are defined by

$$x'(t) \;=\; u(t, x(t)), \qquad t \in (0, T); \qquad x(0) = x_0. \tag{4.9}$$

They are straight lines whose slopes depend on the initial data:

$$x(t) = x_0 + t u^0(x_0), \qquad t \in (0, T).$$

As we are dealing with classical solutions, u is constant along such curves and, by assumption, two different characteristic curves do not meet each other in $\mathbb{R} \times [0, T+\tau]$. This allows to define u in $\mathbb{R} \times [0, T + \tau]$ in a unique way from the initial data.

For $\varepsilon > 0$ sufficiently small, the solution $u^\varepsilon(x, t)$ corresponding to the initial datum (4.1) has similar characteristics to those of u. This allows guaranteeing that two different characteristic lines do not intersect for $0 \le t \le T$ if $\varepsilon > 0$ is small enough. Note that u^ε may possibly be discontinuous for $t \in (T, T+\tau]$ if u^0 generates a discontinuity at $t = T+\tau$ but this is irrelevant for the analysis in $[0, T]$ we are carrying out. Therefore $u^\varepsilon(x, t)$ is also a classical solution in $(x, t) \in \mathbb{R} \times [0, T]$ and it is easy to see that the solution u^ε can be written as (4.2) where δu satisfies (4.3).

System (4.3) can be solved again by the method of characteristics. In fact, as u is a regular function, the first equation in (4.3) can be written as

$$\partial_t \delta u + u \partial_x \delta u = -\partial_x (u) \, \delta u, \tag{4.10}$$

i.e.

$$\frac{d}{dt} \delta u(x(t), t) = -\partial_x(u) \delta u, \tag{4.11}$$

where $x(t)$ are the characteristics curves defined by (4.9). Thus, the solution δu along a characteristic line can be obtained from δu^0 by solving this differential equation, i.e.

$$\delta u(x(t), t) = \delta u^0(x_0) \exp \left(- \int_0^t \partial_x u(x(s), s) ds \right).$$

Finally, the adjoint system (4.5) is also solved by characteristics, i.e.

$$p(x(t), t) = p^T(x(T)).$$

This yields the steepest descent direction in (4.8) for the continuous functional.

Remark 4. *Note that for classical solutions the Gateaux derivative of J at u^0 is given by (4.7) and this provides an obvious descent direction for J at u^0, given by $\delta u^0 = -p(x, 0) \in C_0^1(\mathbb{R})$. However this is not very useful in practice since, even when we initialize the iterative descent algorithm with a smoth u^0 we cannot guarantee that the solution will remain classical along the iteration.*

4.2 Sensitivity of the state in the presence of shocks

In this section we collect some existing results on the sensitivity of the solution of conservation laws in the presence of shocks. We follow the analysis in [7] but similar results in different forms and degrees of generality can be found in [1], [3], [4], [27] or [14], for example.

We focus on the particular case of solutions having a single shock, but the analysis can be extended to consider more general one-dimensional systems of conservation laws with a finite number of noninteracting shocks (see [7]). The theory of duality and reversible solutions developed in [3], [4] is the one leading to more general results.

We introduce the following hypothesis:

(H) Assume that $u(x,t)$ is a weak entropy solution of (1.1) with a discontinuity along a regular curve $\Sigma = \{(t, \varphi(t)), t \in [0, T]\}$, which is Lipschitz continuous outside Σ. In particular, it satisfies the Rankine-Hugoniot condition on Σ

$$\varphi'(t)[u]_{\varphi(t)} = \left[u^2/2\right]_{\varphi(t)}. \tag{4.12}$$

Here we have used the notation: $[v]_{x_0} = v(x_0^+) - v(x_0^-)$ for the jump at x_0 of any piecewise continuous function v with a discontinuity at $x = x_0$.

Note that Σ divides $\mathbb{R} \times (0, T)$ in two parts: Q^- and Q^+, the subdomains of $\mathbb{R} \times (0, T)$ to the left and to the right of Σ respectively.
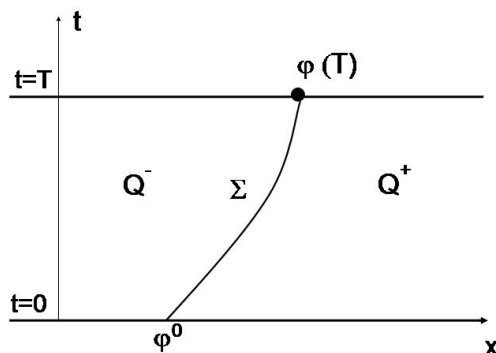

Figure 1: Subdomains Q^- and Q^+.

As we will see, in the presence of shocks, for correctly dealing with optimal control and design problems, the state of the system has to be viewed as being a pair (u, φ) combining the solution of (1.1) and the shock location φ. This is relevant in the analysis of sensitivity of functions below and when applying descent algorithms.

Then the pair (u, φ) satisfies the system

$$\begin{cases} \partial_t u + \partial_x(\frac{u^2}{2}) = 0, & \text{in } Q^- \cup Q^+, \\ \varphi'(t)[u]_{\varphi(t)} = [u^2/2]_{\varphi(t)}, & t \in (0, T), \\ \varphi(0) = \varphi^0, \\ u(x,0) = u^0(x), & \text{in } \{x < \varphi^0\} \cup \{x > \varphi^0\}. \end{cases} \tag{4.13}$$

We now analyze the sensitivity of (u, φ) with respect to perturbations of the initial datum, in particular, with respect to variations δu^0 of the initial profile u^0 and $\delta \varphi^0$ of the shock location φ^0. To be precise, we adopt the functional framework based on the generalized tangent vectors introduced in [7].

Definition 1. *([7]) Let $v : \mathbb{R} \to \mathbb{R}$ be a piecewise Lipschitz continuous function with a single discontinuity at $y \in \mathbb{R}$. We define Σ_v as the family of all continuous paths $\gamma : [0, \varepsilon_0] \to L^1(\mathbb{R})$ with*

1. $\gamma(0) = v$ and $\varepsilon_0 > 0$ possibly depending on γ.

2. For any $\varepsilon \in [0, \varepsilon_0]$ the functions $u^\varepsilon = \gamma(\varepsilon)$ are piecewise Lipschitz with a single discontinuity at $x = y^\varepsilon$ depending continuously on ε and there exists a constant L independent of $\varepsilon \in [0, \varepsilon_0]$ such that

$$|v^\varepsilon(x) - v^\varepsilon(x')| \le L|x - x'|,$$

whenever $y^\varepsilon \notin [x, x']$.

Furthermore, we define the set T_v of generalized tangent vectors of v as the the space of $(\delta v, \delta y) \in L^1 \times \mathbb{R}$ for which the path $\gamma_{(\delta v, \delta y)}$ given by

$$\gamma_{(\delta v, \delta y)}(\varepsilon) = \begin{cases} v + \varepsilon \delta v + [v]_y \ \chi_{[y + \varepsilon \delta y, y]} & \text{if } \delta y < 0, \\ v + \varepsilon \delta v - [v]_y \ \chi_{[y, y + \varepsilon \delta y]} & \text{if } \delta y > 0, \end{cases} \tag{4.14}$$

satisfies $\gamma_{(\delta v, \delta y)} \in \Sigma_v$.

Finally, we define the equivalence relation $\sim$ defined on Σ_v by

$$\gamma \sim \gamma' \text{ if and only if } \lim_{\varepsilon \to 0} \frac{\|\gamma(\varepsilon) - \gamma'(\varepsilon)\|_{L^1}}{\varepsilon} = 0,$$

and we say that a path $\gamma \in \Sigma_v$ generates the generalized tangent vector $(\delta v, \delta y) \in T_v$ if γ is equivalent to $\gamma_{(\delta v, \delta y)}$ as in (4.14).

Remark 5. *The path $\gamma_{(\delta v, \delta y)} \in \Sigma_v$ in (4.14) represents, at first order, the variation of a function v by adding a perturbation function $\varepsilon \delta v$ and by shifting the discontinuity by $\varepsilon \delta y$.*

Note that, for a given v (piecewise Lipschitz continuous function with a single discontinuity at $y \in \mathbb{R}$) the associated generalized tangent vectors $(\delta v, \delta y) \in T_v$ are those pairs for which δv is Lipschitz continuous with a single discontinuity at $x = y$.

Let u^0 be the initial datum in (4.13) that we assume to be Lipschitz continuous to both sides of a single discontinuity located at $x = \varphi^0$, and consider a generalized tangent vector $(\delta u^0, \delta \varphi^0) \in L^1(\mathbb{R}) \times \mathbb{R}$ for all $0 \le \le T$. Let $u^{0,\varepsilon} \in \Sigma_{u^0}$ be a path which generates $(\delta u^0, \delta \varphi^0)$. For ε sufficiently small the solution $u^\varepsilon(\cdot, t)$ of (4.13) is Lipschitz continuous with a single discontinuity at $x = \varphi^\varepsilon(t)$, for all $t \in [0, T]$. Thus $u^\varepsilon(\cdot, t)$ generates a generalized tangent vector $(\delta u(\cdot, t), \delta \varphi(t)) \in L^1 \times \mathbb{R}$. Moreover, in [8] it is proved that it satisfies the following linearized system:

$$\begin{cases} \partial_t \delta u + \partial_x(u \delta u) = 0, & \text{in } Q^- \cup Q^+, \\ \delta \varphi'(t)[u]_{\varphi(t)} + \delta \varphi(t) \left(\varphi'(t)[u_x]_{\varphi(t)} - [u_x u]_{\varphi(t)} \right) \\ \quad + \varphi'(t)[\delta u]_{\varphi(t)} - [u \delta u]_{\varphi(t)} = 0, & \text{in } (0, T), \\ \delta u(x, 0) = \delta u^0, & \text{in } \{x < \varphi^0\} \cup \{x > \varphi^0\}, \\ \delta \varphi(0) = \delta \varphi^0, \end{cases} \tag{4.15}$$

with the initial data $(\delta u^0, \delta\varphi^0)$.

Remark 6. *The linearized system (4.15) can be obtained, at least formally, by a perturbation argument in two steps: first we make the change of variables $\widehat{x} = x - \varphi(t)$ which transforms system (4.15) in a new coupled system but in a fixed domain $\{\widehat{x} < 0\} \cup \{\widehat{x} > 0\}$, and where the variable φ enters in the equations satisfied by u to both sides of $\widehat{x} = 0$. Then, we introduce a perturbation of the data $(u_\varepsilon^0, \varphi_\varepsilon^0) = (u^0, \varphi^0) + \varepsilon(\delta u^0, \delta\varphi^0)$ and compute the equations of the first order perturbation of the solution. This is in fact the usual approach in the study of the linearized stability of shocks. We refer to [14] for a detailed description of this method in the scalar case and [22], [23] for more general systems of conservation laws in higher dimensions.*
In this way, we can obtain formally the expansion

$$(u_\varepsilon, \varphi_\varepsilon) = (u, \varphi) + \varepsilon(\delta u, \delta\varphi) + \mathcal{O}(\varepsilon^2).$$

However, this expansion is only justified for general scalar one-dimensional conservation laws of the form

$$\partial_t u + \partial_x(f(u)) = 0,$$

when the function $f \in C^1$ is convex. In this case, it is possible to establish a differentiability result of the solution u with respect to small perturbations of the initial data u^0 and the discontinuity position φ^0 (see [4]). For more general situations this differentiability has been proved only in a weak sense, as in [7] for systems of conservation laws, or [27], for scalar equations in several space dimensions.

Remark 7. *The linearized system (4.15) has a unique solution which can be computed in two steps. The method of characteristics determines δu in $Q^- \cup Q^+$, i.e. outside Σ, from the initial data δu^0, by the method of characteristics (note that system (4.15) has the same characteristics as (4.13)). This yields the value of u and u_x to both sides of the shock Σ and allows determining the coefficients of the ODE that $\delta\varphi$ satisfies. Then, we solve the ordinary differential equation to obtain $\delta\varphi$.*

In this section we have assumed that the discontinuity of the solution of the Burgers equation u is present in the whole time interval $t \in [0, T]$. It is interesting to note that discontinuities may appear at time $\tau \in (0, T)$ for some regular initial data. In this case we can, at least formally, obtain a generalization of (4.15). Let us show this in a particular situation. Assume that u^0 is a regular initial datum for which the weak entropy solution u of the Burgers equation has a discontinuity at $x = \varphi(t)$ with $t \in [\tau, T]$. Assume that we consider variations δu^0 for which the corresponding solution of (1.1) has also a discontinuity in the same time interval $t \in [\tau, T]$. Then, the linearization can be done separately in $t \in [0, \tau)$ and in $t \in [\tau, T]$. The linearized equations in the last interval are similar to the ones obtained in (4.15). Concerning the interval $[0, \tau)$ the solution is regular and the linearization is obviously given by (4.3). The only question is then how to compute the value of $\delta\varphi(\tau)$ from the initial datum δu^0. This can be obtained by linearizing the weak formulation of the Burgers equation in $t \in (0, \tau)$.

The weak solutions of the Burgers equation satisfy

$$-\int_0^\tau \int_{\mathbb{R}} u\psi_t \, dx \, dt - \int_0^\tau \int_{\mathbb{R}} \frac{u^2}{2}\psi_x \, dx \, dt + \int_{\mathbb{R}} u(x,\tau)\psi(x,\tau) \, dx$$
$$- \int_{\mathbb{R}} u^0(x)\psi(x,0) \, dx = 0, \qquad \forall \psi \in C_0^1(\mathbb{R} \times [0,\tau]).$$

The linearized weak formulation is given by

$$-\int_0^\tau \int_{\mathbb{R}} \delta u\psi_t \, dx \, dt - \int_0^\tau \int_{\mathbb{R}} u\delta u\psi_x \, dx \, dt + \int_{\mathbb{R}} \delta u(x,\tau)\psi(x,\tau) \, dx - [u^0]_{\varphi(\tau)}\delta\varphi(\tau)\psi(\varphi(\tau),\tau)$$
$$- \int_{\mathbb{R}} \delta u^0(x)\psi(x,0) \, dx = 0, \qquad \forall \psi \in C_0^1(\mathbb{R} \times [0,\tau]).$$

Taking into account that δu is constant along the characteristic lines of the Burgers equation we easily obtain

$$-\int_D \delta u\psi_t \, dx \, dt - \int_D u\delta u\psi_x \, dx \, dt - [u^0]_{\varphi(\tau)}\delta\varphi(\tau)\psi(\varphi(\tau),\tau) - \int_{D_0} \delta u^0(x)\psi(x,0) \, dx = 0,$$
$$\forall \psi \in C_0^1(\mathbb{R} \times [0,\tau]).$$

where D is the triangular region $D \in \mathbb{R} \times [0,\tau]$ occupied by the characteristics that meet at $(x,t) = (\varphi(\tau),\tau)$ and D_0 is $D \cap \{t = 0\}$. Taking, in particular, $\psi(x,t) = 1$ in $(x,t) \in D$ we obtain

$$\int_{D_0} \delta u^0(x) \, dx = -[u^0]_{\varphi(\tau)}\delta\varphi(\tau).$$

Thus, the linearized system in this case reads,

$$\begin{cases} \partial_t \delta u + \partial_x(u\delta u) = 0, \quad \text{in } Q^- \cup Q^+, \\ \delta\varphi'(t)[u]_{\varphi(t)} + \delta\varphi(t)\left(\varphi'(t)[u_x]_{\varphi(t)} - [u_x u]_{\varphi(t)}\right) \\ \quad + \varphi'(t)[\delta u]_{\varphi(t)} - [u\delta u]_{\varphi(t)} = 0, \quad \text{in } (\tau, T), \\ \delta\varphi(\tau) = -\frac{1}{[u^0]_{\varphi(\tau)}}\int_{D_0} \delta u^0, \\ \delta u(x,0) = \delta u^0, \quad \text{in } x \in \mathbb{R}. \end{cases} \qquad (4.16)$$

4.3   Sensitivity of J in the presence of shocks

In this section we study the sensitivity of the functional J with respect to variations associated with the generalized tangent vectors defined in the previous section. We first define an appropriate generalization of the Gateaux derivative.

Definition 2. *([7]) Let $J : L^1(\mathbb{R}) \to \mathbb{R}$ be a functional and $u^0 \in L^1(\mathbb{R})$ be Lipschitz continuous with a discontinuity at $x = \varphi^0$, an initial datum for which the solution of (1.1) satisfies hypothesis (H). We say that J is Gateaux differentiable at u^0 in a generalized sense if for any*

generalized tangent vector $(\delta u^0, \delta \varphi^0)$ and any family $u^{0,\epsilon} \in \Sigma_{u^0}$ associated to $(\delta u^0, \delta \varphi^0)$ the following limit exists

$$\delta J = \lim_{\epsilon \to 0} \frac{J(u^{0,\epsilon}) - J(u^0)}{\epsilon},$$

and it depends only on (u^0, φ^0) and $(\delta u^0, \delta \varphi^0)$, i.e. it does not depend on the particular family $u^{0,\epsilon}$ which generates $(\delta u^0, \delta \varphi^0)$.

The limit δJ is the generalized Gateux derivative of J in the direction $(\delta u^0, \delta \varphi^0)$.

The following result provides an easy characterization of the generalized Gateaux derivative of J in terms of the solution of the associated adjoint system. A similar result is formally obtained in [12] in the context of the one-dimensional Euler system. In [8] it is shown how this generalization of the Gateaux derivative can be used to obtain some optimality conditions in a similar optimization problem but, as far as we know, it has not been used to develop a complete descent algorithm as we do here.

Proposition 1. *The Gateaux derivative of J can be written as*

$$\delta J = \int_{\{x < \varphi^0\} \cup \{x > \varphi^0\}} p(x,0) \delta u^0(x) \, dx + q(0)[u^0]_{\varphi^0} \delta \varphi^0, \tag{4.17}$$

where the adjoint state pair (p, q) satisfies the system

$$
\begin{cases}
-\partial_t p - u \partial_x p = 0, & in \ Q^- \cup Q^+, \\
[p]_\Sigma = 0, \\
q(t) = p(\varphi(t), t), & in \ t \in (0, T) \\
q'(t) = 0, & in \ t \in (0, T) \\
p(x, T) = u(x, T) - u^d, & in \ \{x < \varphi(T)\} \cup \{x > \varphi(T)\} \\
q(T) = \dfrac{\frac{1}{2}\left[(u(x,T) - u^d)^2\right]_{\varphi(T)}}{[u]_{\varphi(T)}}.
\end{cases}
\tag{4.18}
$$

Remark 8. *System (4.20) has a unique solution. In fact, to solve the backwards system (4.20) we first define the solution q on the shock Σ from the condition $q' = 0$, with the final value $q(T)$ given in (4.20). This determines the value of p along the shock. We then propagate this information, together with the datum of p at time $t = T$ to both sides of $\varphi(T)$, by characteristics. As both systems (1.1) and (4.20) have the same characteristics, any point $(x,t) \in \mathbb{R} \times (0, T)$ is reached backwards in time by an unique characteristic coming either from the shock Σ or the final data at (x, T) (see Figure 2 where we illustrate this construction in the case of a shock located along a stright line, as it happens to the Riemann problem). The solution obtained this way coincies with the reversible solutios introduced in [3] and [4].*

Remark 9. *Note that the second third and fourth equations in (4.20) come, by duality, from the linearization of the Rankine-Hugoniot condition (4.12). Besides, they are in fact the conditions that allow us to obtain a unique solution in (4.20). They are needed to determine the value of p on Σ.*
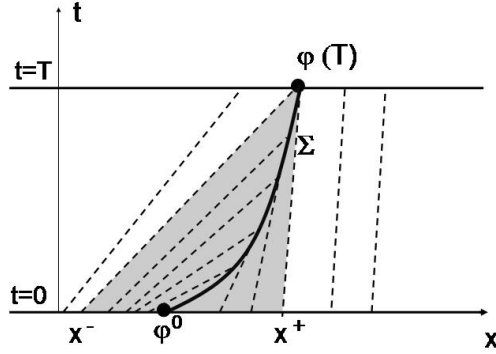
Figure 2: Characteristic lines entering on a shock and how they may be used to build the solution of the adjoint system both away form the shock and on its region of influence.

Solutions of (4.20) can be also obtained as limit of solutions of the transport equation with artificial viscosity depending of a small parameter $\varepsilon \to 0$,

$$\begin{cases} -\partial_t p - u\partial_x p = \varepsilon\partial_{xx}p, & in \ x \in \mathbb{R}, \ t \in (0,T), \\ p(x,T) = p_n^T(x), & in \ x \in \mathbb{R}, \end{cases} \tag{4.19}$$

and a suitable choice of the initial data $p_n^T(x)$, depending on $n \to \infty$. To be more precise, let $p_n^T(x)$ be any sequence of Lipschitz continuous functions, uniformly bounded in $BV_{loc}(\mathbb{R})$, such that

$$p_n^T(x,T) \to p^T(x) = u(x,T) - u^d(x), \quad in \ L_{loc}^1(\mathbb{R}),$$

and

$$p_n^T(\varphi(T),T) = \frac{\frac{1}{2}\left[(u(x,T) - u^d)^2\right]_{\varphi(T)}}{[u]_{\varphi(T)}}.$$

We first take the limit of the solutions $p_{\varepsilon,n}$ of (4.19) as $\varepsilon \to 0$, to obtain the solution p_n of

$$\begin{cases} -\partial_t p - u\partial_x p = 0, & in \ x \in \mathbb{R}, \ t \in (0,T), \\ p(x,T) = p_n^T(x), & in \ x \in \mathbb{R}, \end{cases}$$

the so called reversible solution (see [3]). These solutions can be characterized by the fact that they take the value $p_n(\varphi(T),T)$ in the whole region occupied by the characteristics that meet the shock (see [3], Th. 4.1.12). Thus, in particular they satisfy the 2nd, 3rd, 4th and 6th equations in (4.20). Moreover, $p_n \to p$ as $n \to \infty$, and p takes a constant value in the region occupied by the characteristics that meet the shock. Note that, by construction, this constant is the same value for all p_n in this region. Thus, this limit solution p coincides with the solution of (4.20) constructed above. This allows in fact extending the notion of reversible solutions in [3] to data that, on the point $x(T)$ are completely disconnected with the values of p to both sides of

18

it. This is precisely due to the point of view we have adopted in which the linearized state has two different components $(\delta u, \delta \varphi)$ so that the adjoint state has also two components (p, q) with different initial data at time $t = T$.

Remark 10. *In the expression (4.17) for the derivative of J the shock of the initial datum u^0 appears. When u^0 does not present a shock, obviously, this term cancels in δJ. It is however important to note that, this is compatible with the possible appearence of shocks for times $\tau \in (0, T)$. In that case this singular term does not affect the value of δJ apparently but in practice it does. Indeed, in this case the adjoint system has to be written in the form (4.20) for $\tau < t < T$ and later extended to the time interval $(0, \tau)$ as the classical adjoint system (4.5). Thus, the presence of the shock does affect the value of the adjoint state p at the initial time $t = 0$ and consequently, also, the value of δJ.*

The adjoint system in this case is obtained from (4.16), as in the proof of Proposition 1 below, and it is given by

$$\begin{cases} -\partial_t p - u \partial_x p = 0, & in \ (x, t) \in \mathbb{R} \times (0, T) \backslash \Sigma, \\ [p]_\Sigma = 0, \\ q(t) = p(\varphi(t), t), \ in \ t \in (\tau, T) \\ q'(t) = 0, \ in \ t \in (\tau, T) \\ p(x, T) = u(x, T) - u^d, & in \ \{x < \varphi(T)\} \cup \{x > \varphi(T)\} \\ q(T) = \dfrac{\frac{1}{2}\left[(u(x,T) - u^d)^2\right]_{\varphi(T)}}{[u]_{\varphi(T)}}. \end{cases} \quad (4.20)$$

Let us briefly comment the result of Proposition 1 before giving its proof.

Formula (4.17) provides an obvious way to compute a first descent direction of J at u^0. We just take

$$(\delta u^0, \delta \varphi^0) = (-p(x, 0), -q(0)[u]_{\varphi^0}). \quad (4.21)$$

Here, the value of $\delta \varphi^0$ must be interpreted as the optimal infinitesimal displacement of the discontinuity of u^0.

However, it is important to underline that this $(\delta u^0, \delta \varphi^0)$ is not a generalized tangent vector in T_{u^0} since $p(x, 0)$ is not continuous away from $x \neq \varphi^0$. In fact, $p(x, t)$ takes the constant value $q(T)$ in the whole triangular region occupied by the characteristics of (1.1) which meet the shock Σ. Thus, p has, in general, two discontinuities at the boundary of this region and so will have $p(x, 0)$ (see Figure 3).

This is an important drawback in developing a descent algorithm for J. Indeed, according to the Definition 1, if $(\delta u^0, \delta \varphi^0)$ is a descent direction belonging to T_{u^0}, the new datum $u^{0,new}$ should be obtained from u^0 following a path associated to this descent direction

$$u^{0,new} = \begin{cases} u^0 + \varepsilon \delta u^0 + [u^0]_{\varphi^0} \ \chi_{[\varphi^0 + \varepsilon \delta \varphi^0, \varphi^0]} \ \text{if } \delta \varphi^0 < 0, \\ u^0 + \varepsilon \delta u^0 - [u^0]_{\varphi^0} \ \chi_{[\varphi^0, \varphi^0 + \varepsilon \delta \varphi^0]} \ \text{if } \delta \varphi^0 > 0, \end{cases} \quad (4.22)$$

for some $\varepsilon > 0$ small enough, correctly chosen. Note that, if we take (4.21) as descent direction $(\delta u^0, \delta \varphi^0)$, which is not a generalized tangent vector as explained above, the new datum $u^{0,new}$ will have three discontinuities; the one coming from the displacement of the discontinuity of u^0
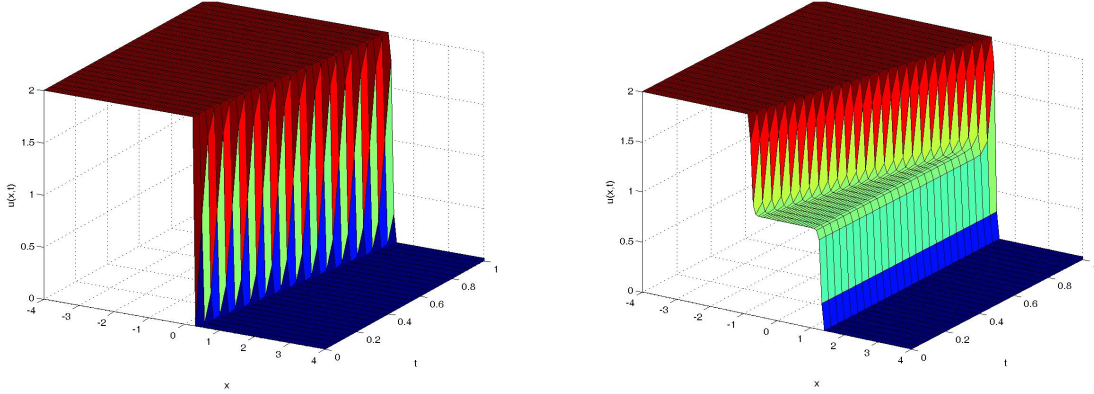
Figure 3: Solution $u(x,t)$ of the Burgers equation with an initial datum having a discontinuity (left) and adjoint solution which takes a constant value in the region occupied by the characteristics that meet the shock (right).

at φ^0 and two more produced by the discontinuities of $p(x,0)$. Thus, in an iterative process, the descent algorithm will create more and more discontinuities increasing artificially the complexity of solutions. This motivates the alternating descent method we propose that, based on this notion of generalized gradients, develops a descent algorithm that keeps the complexity of solutions bounded. This will be done in the following Section.

We finish this section with the proof of the Proposition 1.

Proof. (Proposition 1) A straightforward computation shows that J is Gateaux differentiable in the sense of Definition 2 and that, the generalized Gateaux derivative of J in the direction of the generalized tangent vector $(\delta u^0, \delta \varphi^0)$ is given by

$$\delta J = \int_{\{x<\varphi(T)\}\cup\{x>\varphi(T)\}} (u(x,T) - u^d(x))\delta u(x,T) - \left[\frac{(u(x,T) - u^d(x))^2}{2}\right]_{\varphi(T)} \delta\varphi(T), \quad (4.23)$$

where the pair $(\delta u, \delta \varphi)$ solves the linearized problem (4.15) with initial data $(\delta u^0, \delta \varphi^0)$.

We now introduce the adjoint system (4.20). Multiplying the equations of δu by p and integrating we obtain

$$
\begin{aligned}
0 &= \int_{Q^-\cup Q^+} (\partial_t \delta u + \partial_x(u\delta u))p \; dxdt = -\int_{Q^-\cup Q^+} (\partial_t p + u\partial_x p)\delta u \; dxdt \\
&+ \int_{\{x<\varphi(T)\}\cup\{x>\varphi(T)\}} \delta u(x,T)p(x,T) \; dx - \int_{\{x<\varphi^0\}\cup\{x>\varphi^0\}} \delta u^0(x)p(x,0) \; dx \\
&- \int_{\Sigma} \left([\delta u p]_\Sigma n_t + [u\delta u p]_\Sigma n_x\right) \; d\Sigma,
\end{aligned}
\quad (4.24)
$$

where (n_x, n_t) are the cartesian components of the normal vector to the curve Σ.

Therefore,

$$
\delta J = \int_{\{x<\varphi(T)\}\cup\{x>\varphi(T)\}} \delta u(x,T)(u(x,T)-u^d(x)) \ dx - \left[\frac{(u(x,T)-u^d(x))^2}{2}\right]_{\varphi(T)} \delta\varphi(T)
$$

$$
= \int_{\{x<\varphi^0\}\cup\{x>\varphi^0\}} \delta u^0(x)p(x,0) \ dx + \int_\Sigma \left([\delta up]_\Sigma n_t + [u\delta up]_\Sigma n_x\right) \ d\Sigma
$$

$$
- \left[\frac{(u(x,T)-u^d(x))^2}{2}\right]_{\varphi(T)} \delta\varphi(T), \tag{4.25}
$$

Assume, for the moment, that the following identity holds:

$$
\int_\Sigma \left([\delta up]_\Sigma n_t + [u\delta up]_\Sigma n_x\right) \ d\Sigma = \int_\Sigma [p]_\Sigma \left(\overline{\delta u}n_t + \overline{u\delta u}n_x\right) \ d\Sigma
$$

$$
- \int_\Sigma \partial_{tg}\overline{p}\left(\delta\varphi(t)[u]_{\varphi(t)}\right) \ d\Sigma + \overline{p}^T(\varphi(T))\delta\varphi(T)[u]_{\varphi(T)} - \overline{p}(\varphi(0),0)\delta\varphi^0[u]_{\varphi(0)}. \tag{4.26}
$$

Here $\overline{g}$ represents the average of g to both sides of the shock Σ, i.e.

$$
\overline{g}(x) = \frac{1}{2}\left(\lim_{\varepsilon\to0}g(x+\varepsilon n_\Sigma) + \lim_{\varepsilon\to0}g(x-\varepsilon n_\Sigma)\right), \quad x \in \Sigma.
$$

Then, substituting (4.26) in (4.25) and taken into account the final condition on (p,q) at $t = T$ in (4.20), we obtain

$$
\delta J = \int_\mathbb{R} \delta u^0(x)p(x,0) \ dx + \int_\Sigma [p]_\Sigma \left(\overline{\delta u}n_t + \overline{u\delta u}n_x\right) \ d\Sigma
$$

$$
- \int_\Sigma \partial_{tg}\overline{p}\left(\delta\varphi(t)[u]_{\varphi(t)}\right) \ d\Sigma - \overline{p}(\varphi(0),0)\delta\varphi^0[u]_{\varphi(0)}.
$$

To obtain formula (4.17), the second and third terms in this expression must vanish. This is the case if (p,q) satisfies (4.20). This concludes the proof of Proposition 1.

Let us now prove formula (4.26). Using the elementary identity

$$
[fg]_\Sigma = [f]_\Sigma\overline{g} + [g]_\Sigma\overline{f}, \tag{4.27}
$$

we get

$$
\int_\Sigma \left([\delta up]_\Sigma n_t + [u\delta up]_\Sigma n_x\right) \ d\Sigma = \int_\Sigma [p]_\Sigma \left(\overline{\delta u}n_t + \overline{u\delta u}n_x\right) \ d\Sigma + \int_\Sigma \overline{p}\left([\delta u]_\Sigma n_t + [u\delta u]_\Sigma n_x\right) \ d\Sigma,
$$

and we obtain the first term in the identity (4.26). We now simplify the second term:

$$
\int_\Sigma \overline{p}\left([\delta u]_\Sigma n_t + [u\delta u]_\Sigma n_x\right). \tag{4.28}
$$

21

The cartesian components of the normal vector to Σ are given by

$$n_t = \frac{-\varphi'(t)}{\sqrt{1 + (\varphi'(t))^2}}, \quad n_x = \frac{1}{\sqrt{1 + (\varphi'(t))^2}}.$$

Therefore, taking into account the second equation in system (4.15),

$$
\begin{aligned}
[\delta u]_\Sigma n_t + [u\delta u]_\Sigma n_x &= \frac{-\varphi'(t)[\delta u]_\Sigma + [u\delta u]_\Sigma}{\sqrt{1 + (\varphi'(t))^2}} \\
&= \frac{\delta\varphi'(t)[u]_{\varphi(t)} + \delta\varphi(t)\left(\varphi'(t)[u_x]_{\varphi(t)} - [\partial_x(u^2/2)]_{\varphi(t)}\right)}{\sqrt{1 + (\varphi'(t))^2}} \\
&= \frac{\delta\varphi'(t)[u]_{\varphi(t)} + \delta\varphi(t)[\frac{d}{dt}u(\varphi(t), t)]_{\varphi(t)}}{\sqrt{1 + (\varphi'(t))^2}} = \partial_{tg}\left(\delta\varphi(t)[u]_{\varphi(t)}\right).
\end{aligned}
$$

Finally,

$$
\begin{aligned}
\int_\Sigma \bar{p}\left([\delta u]_\Sigma n_t + [u\delta u]_\Sigma n_x\right) \, d\Sigma &= \int_\Sigma \bar{p}\partial_{tg}\left(\delta\varphi(t)[u]_{\varphi(t)}\right) \, d\Sigma \\
&= -\int_\Sigma \partial_{tg}\bar{p}\left(\delta\varphi(t)[u]_{\varphi(t)}\right) + \bar{p}^T(\varphi(T))\delta\varphi(T)[u]_{\varphi(T)} \, d\Sigma - \bar{p}(\varphi(0), 0)\delta\varphi^0[u]_{\varphi(0)}.
\end{aligned}
$$

$\square$

5 Alternating descent directions

As we said in the end of the previous section, one of the main drawbacks of the continuous approach in the presence of discontinuities is that, in general, the descent algorithm that uses the optimal descent directions based on the generalized tangent vector calculus, produces minimizing sequences with increasing complexity. The remedy is to use true generalized tangent vectors in T_{u^0} as descent directions for J.

Motivated by the above discussion we introduce a decomposition of the generalized tangent vectors. This requires first to introduce some notation. Let

$$x^- = \varphi(T) - u^-(\varphi(T))T, \qquad x^+ = \varphi(T) - u^+(\varphi(T))T,$$

and consider the following subsets (see Figure 4),

$$\hat{Q}^- = \{(x,t) \in \mathbb{R} \times (0,T) \text{ such that } x < \varphi(T) - u^-(\varphi(T))t\},$$

$$\hat{Q}^+ = \{(x,t) \in \mathbb{R} \times (0,T) \text{ such that } x > \varphi(T) - u^+(\varphi(T))t\}.$$
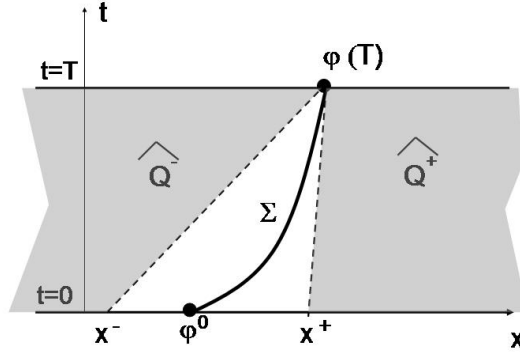
Figure 4: Subdomains $\hat{Q}^-$ and $\hat{Q}^+$

Proposition 2. *Assume that we restrict the set of paths in Σ_{u^0} to those for which the associated generalized tangent vectors $(\delta u^0, \delta\varphi^0) \in T_{u^0}$ satisfy,*

$$\delta\varphi^0 = \frac{\int_{x^-}^{\varphi^0} \delta u^0 + \int_{\varphi^0}^{x^+} \delta u^0}{[u]_{\varphi^0}}. \tag{5.1}$$

Then, the solution $(\delta u, \delta\varphi)$ of system (4.15) satisfies $\delta\varphi(T) = 0$ and the generalized Gateaux derivative of J in the direction $(\delta u^0, \delta\varphi^0)$ can be written as

$$\delta J = \int_{\{x<x^-\}\cup\{x>x^+\}} p(x,0)\delta u^0(x) \; dx, \tag{5.2}$$

where p satisfies the system

$$\begin{cases} -\partial_t p - u\partial_x p = 0, & \text{in } \hat{Q}^- \cup \hat{Q}^+, \\ p(x,T) = u(x,T) - u^d, & \text{in } \{x < \varphi(T)\} \cup \{x > \varphi(T)\}. \end{cases} \tag{5.3}$$

Analogously, if we restrict the set of paths in Σ_{u^0} to those for which the associated generalized tangent vectors $(\delta u^0, \delta\varphi^0) \in T_{u^0}$ satisfy $\delta u^0 = 0$, then $\delta u(x,T) = 0$ and the generalized Gateaux derivative of J in the direction $(\delta u^0, \delta\varphi^0)$ can be written as

$$\delta J = - \left[\frac{(u(x,T) - u^d(x))^2}{2} \right]_{\varphi(T)} \frac{[u^0]_{\varphi^0}}{[u(\cdot,T)]_{\varphi(T)}} \delta\varphi^0. \tag{5.4}$$

Remark 11. *Formula (5.2) establishes a simplified expression for the generalized Gateaux derivative of J when considering directions $(\delta u^0, \delta\varphi^0)$ that do not move the shock position at $t = T$. These directions are characterized by formula (5.1) which determine the infinitesimal displacement of the shock position $\delta\varphi^0$ in terms of the variation of u^0 to both sides of $x = \varphi^0$. Note, in particular, that to any value δu^0 to both sides of the jump φ^0 it corresponds an unique infinitesimal translation $\delta\varphi^0$ of the initial shock position that does not move the shock at $t = T$.*

Note also that system (5.3) does not allow to determine the function p outside the region
$\hat{Q}^- \cup \hat{Q}^+$, i.e. in the region under the influence of the shock by the characteristic lines emanating
from it. However the value of p in this region is not required to evaluate the generalized Gateaux
derivative in (5.2).

Analogously, formula (5.4) provides a simplified expression of the generalized Gateaux deriva-
tive of J when considering directions $(\delta u^0, \delta\varphi^0)$ that uniquely move the shock position at $t = T$
and which correspond to purely translating the shock.

Let us briefly explain the main interest of Proposition 2 before giving its proof. The results
in Proposition 2 suggest the following decomposition of the set of generalized tangent vectors:

$$T_{u^0} = T^1_{u^0} \oplus T^1_{u^0} \tag{5.5}$$

where $T^1_{u^0}$ contains those $(\delta u^0, \delta\varphi^0)$ for which identity (5.1) holds, and $T^2_{u^0}$ the ones for which
$\delta u^0 = 0$. This provides two classes descent directions for J at u^0. In principle they are not
optimal in the sense that they are not the steepest descent directions but they both have three
important properties:

1. They both are descent directions.

2. They allow to split the design of the profile and the shock location.

3. They are true generalized gradients and therefore keep the structure of the data without
   increasing its complexity.

When considering generalized tangent vectors belonging to $T^1_{u^0}$ we can choose as descent
direction,

$$\delta u^0 = \begin{cases} -p(x,0) & \text{if } x < x^-, \\ -\lim_{\substack{x \to x^- \\ x < x^-}} p(x,0) & \text{if } x^- < x < \varphi^0, \\ -\lim_{\substack{x \to x^+ \\ x > x^+}} p(x,0) & \text{if } \varphi^0 < x < x^+, \\ -p(x,0) & \text{if } x^+ < x, \end{cases} \qquad \delta\varphi^0 = -\frac{\int_{x^-}^{\varphi^0} p(x,0) + \int_{\varphi^0}^{x^+} p(x,0)}{[u]_{\varphi^0}}, \tag{5.6}$$

while for $T^2_{u^0}$ a good choice is:

$$\delta u^0 = 0, \qquad \delta\varphi^0 = \left[\frac{(u(x,T) - u^d(x))^2}{2}\right]_{\varphi(T)} \frac{[u(\cdot, T)]_{\varphi(T)}}{[u^0]_{\varphi^0}}. \tag{5.7}$$

In (5.6) the value of δu^0 in the interval (x^-, x^+) does not affect the generalized Gateaux deriva-
tive in (5.2) under the condition that $\delta\varphi^0$ is chosen exactly as indicated (otherwise the shock
would move and this would produce an extra term on the derivative of the functional J). We
have chosen the simplest constant value that preserves the Lipschitz continuity of δu^0 at $x = x^-$
and $x = x^+$, but not necessarily at $x = \varphi^0$. Other choices would also provide descent directions
for J at u^0, but would yield the same Gateaux derivative according to (5.2).

This allows us to define a strategy to obtain descent directions for J at u^0 in T_{u^0}.

To illustrate this we consider the simplest case in which u^d

$$u^d \text{ is Lipschitz continuous with a discontinuity at } x = x^d. \tag{5.8}$$

To initialize the descent algorithm, in view of the structure of u^d we choose u^0 with a similar structure, with a single discontinuity located at φ^0. Typically this produces a solution u with a shock disciontinuity that at the final time $t = T$ is located at $\varphi(T)$. Then, there are two possibilities depending on $\varphi(T)$ before applying the descent method:

1. $\varphi(T) \neq x^d$. Then, we consider a descent direction of the form (5.7) that will move the discontinuity of u^0 until we have $x^d = \varphi(T)$.

2. We already have $x^d = \varphi(T)$ and we consider descent directions of the form (5.6). To first order, these directions will not move the value of φ at $t = T$.

In practice, the deformations of the second step will slightly move the position of the shock because of its nonlinear dependence on the parameter ε. Thus, one has to iterate this procedure to assure a simultaneous better placement of the shock and a better fitting of the value of the solution away from it.

In the next section we explain how to implement a descent algorithm following these ideas that, of course, can also be used in the case where the number of shocks of u^0 and u^d is not necessarily one, or the same.

Proof. (Proposition 2) Assume that $(\delta u^0, \delta \varphi^0)$ is a generalized tangent vector for which the solution of (4.20) satisfies $\delta \varphi(T) = 0$. The first equation in (4.15) can be written as

$$\text{div}_{t,x}(\delta u, u \delta u) = 0.$$

Thus, integrating this equation over the triangle $Q^- \backslash \hat{Q}^-$ and using the divergence theorem we obtain

$$0 = - \int_{x^-}^{\varphi^0} \delta u^0 \, dx + \int_{\Sigma} (\delta u, u \delta u) \cdot \mathbf{n} \, ds,$$

where $\mathbf{n}$ is the normal vector to Σ. Of course we obtain an analogous formula if we integrate over $Q^+ \backslash \hat{Q}^+$. Combining these two identities, and (4.29), we have

$$\int_{\varphi^0}^{x^+} \delta u^0 \, dx + \int_{x^-}^{\varphi^0} \delta u^0 \, dx = - \int_{\Sigma} ([\delta u], [u \delta u]) \cdot \mathbf{n} \, ds = - \int_{\Sigma} \partial_{tg}(\delta \varphi [u]_{\Sigma}) \, ds = \delta \varphi(0) [u]_{\varphi(0)},$$

and therefore we obtain the characterization (5.1).

Now we prove formula (5.3). We follow the argument in the proof of Proposition 1. Since $\delta \varphi(T) = 0$, in this case, formula (4.23) is reduced to

$$\delta J \;=\; \int_{\{x < \varphi(T)\} \cup \{x > \varphi(T)\}} (u(x,T) - u^d(x)) \delta u(x,T). \tag{5.9}$$

When multiplying the equations of δu by the solution p of (5.3) and integrating, this time over $\bar{Q}^- \cup \hat{Q}^+$, we easily obtain (5.3). $\qquad \square$

6 Numerical approximations of the descent direction

We have computed the gradient of the continuous functional J in several cases (u smooth and having shock discontinuities) but, in practice, one has to look for descent directions for the discrete functional J^Δ. In this section we discuss the various possibilities for searching them. There are several possibilities which are based on the approach chosen (continuous versus discrete) and the degree of sophistication adopted.

We consider the following possibilities:

- The discrete approach: differentiable schemes.

- The discrete approach: non-differentiable schemes.

- The continuous approach: Internal boundary conditions on the shock.

- The continuous approach: The alternating descent method.

The last one is the new method we propose in this article.

In the following Section we present some numerical experiments that allow us to easily compare the efficiency of each method. As we shall see, the alternating descent method we propose, alternating the generalized tangent vectors to sometimes move the shock and some others correct the profile to both sides of it, is superior in several ways. It avoids the drawbacks of the other methods related either to the inefficiency of the differentiable methods to capture shocks, the difficulty to dealing with non-differentiable schemes and the uncertainty of using "pseudo-linearizations", or the difficulty to efficiently impose internal boundary conditions in practice. As a consequence of this, the method we propose is much more robust and the functional decreases in a much more efficient way in a significantly smaller number of iterations.

The rest of this section is divided as follows: we first compute the gradient of the discrete cost functional when the numerical scheme chosen to approximate the Burgers equation is differentiable. When the numerical scheme is not differentiable the gradient of the cost functional is not well-defined and a descent direction must be computed in a different way. In the second subsection we present an alternative method which consists roughly in computing a subgradient of the discrete functional. The last two subsections contain methods based on the continuous approach. More precisely, the third one describes the *a priori* more natural method based on the discretization of the continuous gradient while the fourth subsection is devoted to the new method introduced in this work in which we consider a suitable decomposition of the generalized tangent vectors.

We do not address here the convergence of these algorithms, but, in the present case, and taking into account that when dealing with the discrete functional J^Δ the number of control parameters is finite, one could prove convergence by using LaSalle's invariance principle and using the cost functional as Lyapunov functional.

6.1 The discrete approach: Differentiable numerical schemes

Computing the gradient of the discrete functional J^Δ requires computing one derivative of J^Δ with respect to each node of the mesh. This can be done in a cheaper way using the adjoint state. We illustrate it on two different numerical schemes: Lax-Friedrichs and Engquist-Osher. Note that both schemes satisfy the hypothesis of Theorem 2 and therefore the numerical minimizers are good approximations of minimizers of the continuous problem. However, as the discrete functionals J^Δ are not necessarily convex the gradient methods could possibly provide sequences that do not converge to a global minimizer of J^Δ. But this drawback and difficulty appears in most applications of descent methods in optimal design and control problems. As we will see, in the present context, the approximations obtained by gradient methods are satisfactory, although convergence is slow due to unnecessary oscillations that the descent method introduces.

Computing the gradient of J^Δ, rigoroulsy speaking, requires the numerical scheme (3.3) under consideration to be differentiable and, often, this is not the case. To be more precise, for the Burgers equation (1.1) we can choose several efficient methods which are differentiable (as the Lax-Friedrichs and the Engquist-Osher one) but this is not the situation for general systems of conservation laws in higher dimensions, as Euler equations. For such complex systems the efficient methods, as Godunov, Roe, etc., are not differentiable (see, for example [17] or [21]) thus making the approach in this section useless.

We observe that when the 3-point conservative numerical approximation scheme (3.3) used to approximate the Burgers equation (1.1) has a differentiable numerical flux function g, then the linearization is easy to compute. We obtain

$$\begin{cases} \delta u_j^{n+1} = \delta u_j^n - \lambda \left(\partial_1 g_{j+1/2}^n \delta u_j^n + \partial_2 g_{j+1/2}^n \delta u_{j+1}^n - \partial_1 g_{j-1/2}^n \delta u_{j-1}^n - \partial_2 g_{j-1/2}^n \delta u_j^n \right) = 0, \\ j \in \mathbb{Z}, \quad n = 0, ..., N. \end{cases} \quad (6.1)$$

In view of this, the discrete adjoint system can also be written for any differentiable flux function g:

$$\begin{cases} p_j^n = p_j^{n+1} + \lambda \left(\partial_1 g_{j+1/2}^n (p_{j+1}^{n+1} - p_j^{n+1}) + \partial_2 g_{j-1/2}^n (p_j^{n+1} - p_{j-1}^{n+1}) \right), \\ p_j^{N+1} = p_j^T, \quad j \in \mathbb{Z}, \quad n = 0, ..., N. \end{cases} \quad (6.2)$$

In fact, when multiplying the equations in (6.1) by p_j^{n+1} and adding in $j \in \mathbb{Z}$ and $n = 0, ..., N$, the following identity is easily obtained,

$$\Delta x \sum_{j \in \mathbb{Z}} p_j^T \delta u_j^{N+1} = \Delta x \sum_{j \in \mathbb{Z}} p_j^0 \delta u_j^0. \quad (6.3)$$

This is the discrete version of formula (4.6) which allows us to simplify the derivative of the discrete cost functional.

Thus, for any variation $\delta u_\Delta^0 \in \mathcal{U}^\Delta$ of u_Δ^0, the Gateaux derivative of the cost functional defined in (3.1) is given by

$$\delta J^\Delta = \Delta x \sum_{j \in \mathbb{Z}} (u_j^{N+1} - u_j^d) \delta u_j^{N+1}, \quad (6.4)$$

where δu_j^n solves the linearized system (6.1). If we consider p_j^n the solution of (6.2) with final datum

$$p_j^T = u_j^{N+1} - u_j^d, \quad j \in \mathbb{Z}, \tag{6.5}$$

then δJ^Δ in (6.4) can be written as,

$$\delta J^\Delta = \Delta x \sum_{j \in \mathbb{Z}} p_j^0 \delta u_j^0, \tag{6.6}$$

and this allows to obtain easily the steepest descent direction for J^Δ by considering

$$\delta u_\Delta^0 = -p_\Delta^0. \tag{6.7}$$

We now present two particular examples. Let us consider first the Lax-Friedrichs scheme:

$$\begin{cases} \frac{u_j^{n+1} - \frac{u_{j-1}^n + u_{j+1}^n}{2}}{\Delta t} + \frac{f(u_{j+1}^n) - f(u_{j-1}^n)}{2\Delta x} = 0, & n = 0, ..., N, \\ u_j^0 = u_{0,j}, & j \in \mathbb{Z}, \end{cases} \tag{6.8}$$

where $f(s) = s^2/2$. The numerical scheme (6.8) can be written in conservation form with the numerical flux given in (3.6). Moreover, it satisfies the hypotheses of Theorem 2, under the Courant-Friedrichs-Levy (CFL) condition $\lambda|\max u^0| \leq 1$, and it is differentiable.

For any variation $\delta u_\Delta^0 \in \mathcal{U}^\Delta$ of u_Δ^0, the Gateaux derivative of the cost functional is given by (6.6) where the values p_j^n satisfy the adjoint system,

$$\begin{cases} \frac{p_j^n - \frac{p_{j+1}^{n+1} + p_{j-1}^{n+1}}{2}}{\Delta t} + u_j^n \frac{p_{j-1}^{n+1} - p_{j+1}^{n+1}}{2\Delta x} = 0, & n = 0, ..., N \\ p_j^{N+1} = p_j^T, & j \in \mathbb{Z}, \end{cases} \tag{6.9}$$

with $p_j^T = (u_j^{N+1} - u_j^d) \in \mathcal{U}_{ad}^\Delta$.

Note that, formally, (6.9) is in fact the Lax-Friedrichs numerical scheme applied to the continuous adjoint system (4.5).

The Engquist-Osher scheme can be treated similarly. In this case the numerical flux is given by (3.7) and we get the adjoint system

$$\begin{cases} p_j^n = p_j^{n+1} + \lambda \left(\frac{u_j^n + |u_j^n|}{2}(p_{j+1}^{n+1} - p_j^{n+1}) + \frac{u_j^n - |u_j^n|}{2}(p_j^{n+1} - p_{j-1}^{n+1}) \right) = 0, & n = 0, ..., N, \\ p_j^{N+1} = p_j^T, & j \in \mathbb{Z}. \end{cases} \tag{6.10}$$

The derivative δJ^Δ is given again by (6.6) and the steepest descent direction is (6.7) where, now, p solves (6.10).

We observe that (6.10) is the upwind method for the continuous adjoint system. Thus, this is another case in which the adjoint of the discretization corresponds to a well-known discretization of the adjoint problem.

Remark 12. *We do not address here the problem of the convergence of these adjoint schemes towards the solutions of the continuous adjoint system. Of course, this is an easy matter when u is smooth but is is far from being trivial when u has shock discontinuities. Whether or not these discrete adjoint systems, as $\Delta \to 0$, allow reconstructing the complete adjoint system, with the inner Dirichlet condition along the shock, constitutes an interesting problem for future research. We refer to [18] for preliminary work on this direction.*

6.2 The discrete approach: Non-differentiable numerical schemes

We describe here the most common method to compute "gradients" of functionals when the underlying numerical scheme used to approximate the flow equations is non-differentiable (see for example [14] where this method is used in the context of linearized stability). To illustrate this method we focus on the Roe scheme which is one of the most popular ones to approximate solutions of conservation laws.

In the particular case of the Burgers equation under consideration Roe's scheme coincides with Godunov's one and therefore Theorem 2 applies.

However, Roe's scheme is not monotone for general fluxes f and it is well-known that this scheme admits entropy violating discontinuities (see [15]). Therefore, convergence of discrete minimizers towards continuous ones can not be guaranteed for more general fluxes.

The scheme can be modified to obtain the conservation of entropy (see [15] for the Harten and Hyman modification) but we will not consider this modification here (which is still non-differentiable) since we are mainly interested in the issue of "linearizing" non-differentiable schemes.

The Roe scheme for a general conservation law

$$\partial_t u + \partial_x f(u) = 0,$$

is a 3-point conservative scheme of the form (3.3) with numerical flux

$$g^R(u,v) \;\; = \;\; \frac{1}{2}(f(u) + f(v) - |A(u,v)|(v - u)),$$

where the matrix $A(u,v)$ is a *Roe linearization* which is an approximation of f' (see, for example, [14]). In the scalar case under consideration $f(u) = u^2/2$ and

$$A(u,v) = \begin{cases} \frac{f(u)-f(v)}{u-v} = \frac{u+v}{2}, & \text{if } u \neq v, \\ f'(u) = u, & \text{if } u = v \,. \end{cases}$$

Note that the previous scheme is not differentiable, in general, due to the presence of the absolute value of A in g^R. Thus, we cannot linearize this system and obtain its adjoint, in a rigurous sense.

In [14] the following scheme is proposed for the linearization

$$\delta u_j^{n+1} = \delta u_j^n - \lambda(h_{j+1/2}^n - h_{j-1/2}^n), \quad j \in \mathbb{Z}, \quad n = 0,...,N, \qquad (6.11)$$

where

$$h_{j+1/2}^n \;\; = \;\; h(u_j^n, u_{j+1}^n; \delta u_j^n, \delta u_{j+1}^n),$$

$$h(u,v;w,z) \;\; = \;\; \frac{1}{2}\left(A(u,v)(w + z) - |A(u,v)|(z - w)\right). \qquad (6.12)$$

Equation (6.12) is in fact an approximation of the natural choice $h(u,v;w,z) = \frac{\partial g^R}{\partial u}w + \frac{\partial g^R}{\partial v}z$, where $\frac{\partial f}{\partial u}$ is approximated by the Roe linearization $A(u,v)$, and the non-differentiable term

$|A(u,v)|$ in (6.11) is assumed to have zero derivative. This last assumption could be formally interpreted as a particular choice of the subgradient of the absolute value function $a(x) = |x|$ at $x = 0$.

In this way

$$
\begin{aligned}
h^n_{j+1/2} &= \frac{1}{2}\left(A_{j+1/2}(\delta u^n_j + \delta u^n_{j+1}) - |A_{j+1/2}|(\delta u^n_{j+1} - \delta u^n_j)\right), \\
A_{j+1/2} &= A(u^n_j, u^n_{j+1}).
\end{aligned}
$$

The corresponding adjoint system to the linearized equations (6.11) is given by

$$
\begin{cases}
p^n_j = p^{n+1}_j + \lambda\left(\alpha^n_j(p^{n+1}_{j+1} - p^{n+1}_j) + \beta^n_j(p^{n+1}_j - p^{n+1}_{j-1})\right), & n = 0, ..., N, \\
p^{N+1}_j = p^T_j, & j \in \mathbb{Z},
\end{cases}
\tag{6.13}
$$

where

$$
\alpha^n_j = \frac{1}{2}(A_{j+1/2} + |A_{j+1/2}|), \qquad \beta^n_j = \frac{1}{2}(A_{j+1/2} - |A_{j+1/2}|).
$$

In fact, multiplying the equations in (6.11) by p^{n+1}_j and adding in $j \in \mathbb{Z}$ and $n = 0, ..., N$ we obtain:

$$
\begin{aligned}
0 &= \sum_{j\in\mathbb{Z}}\sum_{n=0}^{N}\left(\delta u^{n+1}_j - \delta u^n_j + \lambda(h^n_{j+1/2} - h^n_{j-1/2})\right) p^{n+1}_j \\
&= \sum_{j\in\mathbb{Z}}\sum_{n=0}^{N}\left(p^n_j - p^{n+1}_j - \lambda\left[\alpha^n_j(p^{n+1}_{j+1} - p^{n+1}_j) + \beta^n_j(p^{n+1}_j - p^{n+1}_{j-1})\right]\right)\delta u^n_j \\
&\quad + \sum_{j\in\mathbb{Z}}\delta u^{N+1}_j p^{N+1}_j - \sum_{j\in\mathbb{Z}}\delta u^0_j p^0_j = \sum_{j\in\mathbb{Z}}\delta u^{N+1}_j p^{N+1}_j - \sum_{j\in\mathbb{Z}}\delta u^0_j p^0_j.
\end{aligned}
\tag{6.14}
$$

To obtain (6.14) we have used the following identity:

$$
\begin{aligned}
\sum_{j\in\mathbb{Z}} h^n_{j+1/2} p^{n+1}_j &= \sum_{j\in\mathbb{Z}}\frac{1}{2}A_{j+1/2}(\delta u^n_j + \delta u^n_{j+1})p^{n+1}_j \\
&\quad - \sum_{j\in\mathbb{Z}}\frac{1}{2}|A_{j+1/2}|(\delta u^n_{j+1} - \delta u^n_j)p^{n+1}_j \\
&= \sum_{j\in\mathbb{Z}}\frac{1}{2}(A_{j+1/2}p^{n+1}_j + A_{j-1/2}p^{n+1}_{j-1})\delta u^n_j \\
&\quad - \sum_{j\in\mathbb{Z}}\frac{1}{2}(|A_{j-1/2}|p^{n+1}_{j-1} - |A_{j+1/2}|p^{n+1}_j)\delta u^n_j,
\end{aligned}
$$

and an analogous one for the term $\sum_{j\in\mathbb{Z}} h^n_{j-1/2} p^{n+1}_j$.

Then, as for differentiable schemes, formula (6.14) allows to simplify the "derivative" δJ^Δ which is formally written as (6.6). Thus, a tentative descent direction for J^Δ is given by (6.7), where p_j^n is the solution of the adjoint system (6.13) with final datum $p_j^{N+1} = u_j^{N+1} - u_j^d$.

The above computation does not provide the gradient of the discrete functional, which is non-differentiable in this case. But the value obtained through this computation could be used as an alternative "descent direction" in a gradient-type algorithm.

Note that the approach of using "pseudogradients" we have presented here is a common practice in optimal design in aeronautics where efficient solvers are often non differentiable (see [25]).

6.3 The continuous approach: Internal boundary condition on the shock

This method is based on the result stated in Proposition 1 indicating that the sensitivity of the functional is obtained by approximating $(-p(x,0), -q(0)[u]_{\varphi^0})$. We recall that the continuous adjoint system is well-posed and its solution can be obtained in two steps. We first obtain the value of p on the shock φ of u from the differential equation $q'(t) = 0$ and the end condition on $q(T)$. Note that, in our case, p takes the constant value $q(T)$ along the shock Σ. Then we solve backwards the adjoint equation taking into account both the value of p at $t = T$ and the value of p on the shock.

At the numerical level we can proceed similarly distinguishing the computation of the discrete adjoint state in the region of influence of the shock and away of it. We first introduce a suitable discretization of the adjoint equation in the whole domain (for instance by taking the adjoint of a linearizable numerical scheme), that we solve. This gives an approximation of the adjoint state away from the influence region of the shock. We then determine the value of j_n which corresponds to the nearest grid point $x = x_{j_n}$ to the shock position at $t = t^n$, and impose $p_{j_n}^n = q(T)$ for this particular j_n. Finally, we take p_j^n to coincide with $p_{j_n}^n$ in all the influence region of the shock.

In this way we get a descent direction of the form

$$(\delta u_j^0, \delta\varphi^0) = (-p_j^0, -q^0[u_j^0]_{\varphi^0}). \tag{6.15}$$

In particular, the second value must be interpreted as a displacement of the position of the discontinuity of u^0. Note that this interpretation is formal at the continuous level since formula (4.22) was derived for generalized tangent vectors, which is not the case here, as discussed after the statement of Proposition 1.

To be more precise, we now describe how to obtain a new initial datum out of the previous one within the descent iteration, in view of the approximation above of the gradients.

For example, if $\delta\varphi^0 > 0$, one can choose

$$u_j^{0,new} = \begin{cases} u_j^0 + \varepsilon\delta u_j^0, & \text{if } j < \varphi^0 \text{ or } j > \varphi^0 + \varepsilon\delta\varphi^0/\Delta x, \\ u_j^0 + \varepsilon\delta u_j^0 + [u_j^0]_{\varphi^0}, & \text{if } \varphi^0 \le j \le \varphi^0 + \varepsilon\delta\varphi^0/\Delta x. \end{cases}$$

The main drawbacks of this approach are the following:

1. At any step of the descent algorithm, a numerical approximation of the position of the shock is required.

2. The pair $(p(x,0), q(0))$ is not a generalized tangent vector and, as discussed after the statement of Proposition 1, an iterative gradient method based on these ideas generates increasingly complex initial data. Numerical experiments confirm that this actually occurs.

3. A pure displacement of the discontinuity will never be a descent direction computed by this method. Indeed, a generalized vector of the form $(0, \alpha)$ which only moves the shock, i.e. with vanishing first component, cannot be obtained as $(p(x,0), q(0))$ for any solution of (4.20). In fact, if $p(x,0) = 0$ then $q(0) = 0$, since $p(x,T) = q(T) = q(0)$ in the whole region occupied by the characteristics of u that meet Σ.

6.4   The alternating descent method

Here we propose a new method suggested by the results in Proposition 2 and the discussion thereafter. We shall refer to this new method as the *alternating descent method.*

In order to illustrate how the method can be implemented, we assume that we have a final target u^d which is Lipschitz continuous function with a single discontinuity at $x = x_T$ with negative jump, i.e. $[u^d]_{x_T} < 0$, to guarantee that this discontinuity can be generated by the solution at $t = T$ of (1.1) for some solution u having a shock. To initialize the iterative descent method we choose an initial datum u^0 in such a way that the solution at time $t = T$ has a profile similar to u^d, i.e., it is a Lipschitz continuous function with a single continuity of negative jump, located on an arbitrary point $x \in \mathbb{R}$. The main idea now is to approximate a minimizer of J alternating the following two steps: first we perturb the initial datum u^0 by simply moving the discontinuity of the solution u of (1.1) at time $t = T$, regardless of its value to both sides of the discontinuity. Once this is done we perturb the resulting u^0 without altering the position of the discontinuity of $u(x, T)$. This is done by decomposing the set of generalized tangent vectors associated to u^0 into the two subsets introduced in (5.5) considering alternatively (5.6) and (5.7) as descent directions.

More precisely, for a given initialization of u^0 as above, in each step of the descent iteration process we proceed in the following two substeps:

1. Compute (5.7) and find the optimal step size ε for which this datum must be modified in the direction given by (5.7). This involves a one-dimensional optimization problem that we can solve with a classical method (bisection, Armijo's rule, etc.). In this way we obtain the best location of the discontinuity for this u^0.

2. We then use the descent direction (5.6) to modify the value of the solution at time $t = T$ to both sides of the discontinuity. Here, we can again estimate the step size by solving a one-dimensional optimization problem or simply take a constant step size.

The main advantage of this method is that for an initial datum u^0 with a single discontinuity, the assumption (5.8) holds and the descent directions are generalized tangent vectors, i.e.

they introduce Lipschitz continuous variations of u^0 to both sides of the discontinuity and a displacement of the shock position. In this way, the new datum obtained modifying the old one, in the direction of this generalized tangent vector, will have again a single discontinuity. Therefore, the iterative optimization process will not introduce new discontinuities in u^0, as in the previous method.

We have presented here the method in the particular case in which both the target u^d and the initial datum u^0 that initializes the process have one single shock discontinuity. But these ideas can be applied in a much more general context in which the number of shocks does not necessarily coincide. Indeed, as we shall see in various numerical experiments, this method is able both to generate shock and to distroy them, if any of these facts contributes to the decrease of the functional.

This method is in some sense close the the present methods emploeyd in shape design in elasticity in which topological derivatives (that in the present setting would correspond to controlling the location of the shock) are combined with classical shape deformations (that would correspond to simply shaping the solution away form the shock in the present setting) ([11]).

7 Numerical experiments

In this section we present some numerical experiments which illustrate the results obtained in an optimization model problem with each one of the numerical methods described in the previous section.

We have chosen as computational domain the interval $(-4, 4)$ and we have taken as boundary conditions in (1.1), at each time step $t = t^n$, the value of the initial data at the boundary. This can be justified if we assume that the initial datum u^0 is constant in a sufficiently large inner neighborhood of the boundary $x = \pm 4$ (which depends on the size of the L^∞-norm of the data under consideration and the time horizon T), due to the finite speed of propagation. A similar procedure is employed for the adjoint equation.

We underline once more that the solutions obtained with each method may correspond to global minima or local ones since the gradient algorithm does not distinguish them.

Experiment 1. We first consider a piecewise constant target profile u^d given by

$$u^d = \begin{cases} 1 \text{ if } x < 0, \\ 0 \text{ if } x \geq 0, \end{cases} \tag{7.1}$$

and the time $T = 1$. Note that in this case one solution of the optimization problem is obviously given by

$$u^{0,min} = \begin{cases} 1 \text{ if } x < -1/2, \\ 0 \text{ if } x \geq 0. \end{cases} \tag{7.2}$$

This means that the optimal value $u^{0,min}$ can be attained and the minimum value of J in this case is zero.

We solve the optimization problem (3.5) with the above described different methods starting from the following initialization for u^0:

$$u^0 = \begin{cases} 2 \text{ if } x < 1/4, \\ 0 \text{ if } x \geq 1/4. \end{cases} \tag{7.3}$$

which also has a discontinuity but located on a different point.

To compare the efficiency of the different methods we consider a fixed $\Delta x > 0$, $\lambda = \Delta t/\Delta x = 1/2$ and we focus on the number of iterations for each method to attain a prescribed value of the functional. In Table 1 we give these values when the spatial discretization parameter is $\Delta x = 1/20$ and $\Delta x = 1/80$ respectively.

$\log(J^\Delta)$	-3	-4	-5	-6	-7
Lax-Friedrichs	14	39	> 1000		
Engquist-Osher	26	85	288	> 1000	
Roe	18	33	54	114	> 1000
Imposing b.c.	5	6	9	21	> 1000
Alternating descent	3	3	3	Not attained	

$\log(J^\Delta)$	-3	-4	-5	-6	-7
Lax-Friedrichs	15	49	> 1000		
Engquist-Osher	115	673	> 1000		
Roe	185	> 1000			
Imposing b.c.	5	6	52	440	> 1000
Alternating descent	3	3	3	3	Not attained

Table 1: Experiment 1. Number of iterations needed for a descent algorithm to obtain the value of $\log(J)$ indicated in the upper row, by the different methods presented above. The upper table corresponds to $\Delta x = 1/20$ and the lower one to $\Delta x = 1/80$. In both cases $\lambda = \Delta t/\Delta x = 1/2$.

In Figure 5 we show the initial data u^0 obtained with the different methods after 30 iterations for $\Delta x = 1/20$ and in Figure 6 the value of the functional one achieves, with $\Delta x = 1/20$ and $\Delta x = 1/80$. In both cases $\lambda = \Delta t/\Delta x = 1/2$.

We observe the following:

1. In Figure 5, we see that the different numerical approximation and descent methods lead to different solutions.

   Obviously, the final output of the descent algorithm may also depend on the initialization u^0. This will be discussed in another experiment later. In this one the initialization is the same for all the five methods under consideration.

2. For the first four methods the initial datum u^0 we obtain after the iteration process presents strong oscillations. That is not the case for the method we have developped in this

article based on alternating descent directions. Note that, actually, the highest oscillations are produced when using the Lax-Friedrichs scheme, which is the most dissipative one.

3. In Figure 6 and Table 1 we see that the numerical methods that ignore the presence of the shock (Lax-Friedrichs, Engquist-Osher and Roe) descend more slowly than those that take into account the sensitivity with respect to the shock position (by imposing the boundary condition on the shock or the alternating descent method).

4. For fixed Δx the alternating descent method stabilizes quickly in a few iterations. This is due to the fact that the descent direction is computed for the continuos system and not for the discrete one, and therefore Δx needs to be small for that computation to be valid at the discrete level as well.

5. For smaller values of Δx the only method that remains effective is the alternating descent method. The other methods descent more slowly.

Experiment 2. We consider the same target u^d as in the previous experiment but with different initial data. We see that different initialization functions u^0, with more or less discontinuities, do not alter the efficiency of the alternating descent method. The numerical results are presented in Figure 7.

Experiment 3. We now consider a piecewise constant target profile u^d with two discontinuities:

$$u^d = \begin{cases} 1 \text{ if } x < -1/4, \\ 1/2 \text{ if } -1/4 \le x < 3/2, \\ 0 \text{ if } x \ge 3/2, \end{cases} \tag{7.4}$$

and the time $T = 1$.

We solve the optimization problem (3.5) with the above described methods starting from the following initial datum

$$u^0 = \begin{cases} 2 \text{ if } x < 0, \\ 1/2 \text{ if } 0 \le x < 2, \\ 0 \text{ if } x \ge 0. \end{cases} \tag{7.5}$$

which also has two discontinuities, as the target.

The conclusions are similar to those of the first experiment.

In Table 2 we give the number of iterations for each method to attain a prescribed value of the functional when the spatial discretization parameter is $\Delta x = 1/20$ and $\lambda = \Delta t/\Delta x = 1/2$.. The solutions obtained after 30 iterations of each method are given in Figure 8. Of course, as in the first experiment, the alternating descent method becomes much more efficient for lower values of Δx.

Experiment 4. We now consider a piecewise constant target profile u^d with a discontinuity with positive jump:

$$u^d = \begin{cases} 1/2 \text{ if } x < 1/4, \\ 1 \text{ if } x \ge 1/4, \end{cases} \tag{7.6}$$

$\log(J^\Delta)$	-3	-4	-5	-6	-7
Lax-Friedrichs	5	8	30	> 1000	
Engquist-Osher	5	17	54	> 1000	
Roe	4	6	13	34	101
Imposing b.c.	3	5	16	55	> 1000
Alternating descent method	3	4	4	4	Not attained

$\log(J^\Delta)$	-3	-4	-5	-6	-7
Lax-Friedrichs	6	10	270	> 1000	
Engquist-Osher	10	235	> 1000		
Roe	20	Not attained			
Imposing b.c.	8	Not attained			
Alternating descent method	3	4	4	5	Not attained

Table 2: Experiment 3. Number of iterations needed for a descent algorithm to obtain the value of $\log(J)$ indicated in the upper row, when considering the different descent strategies. Here $\Delta x = 1/20$ in the upper table and $\Delta x = 1/80$ in the lower one. In both cases $\lambda = \Delta t/\Delta x = 1/2$.

and the time $T = 1$.

We observe that the alternating descent method obtain the same values as the other methods but in less iterations (see Figures 9 and 10).

8   Numerical algorithms

In this section we briefly describe the algorithms we have used to implement the various numerical methods.

We first consider the *discrete approach*. The algorithm is the same for both differentiable and non-differentiable schemes and uses a constant descent step. Of course, when the numerical scheme is not differentiable one has to choose a suitable pseudolinearization of the numerical flux for the algorithm to make sense as we have described in the context of Roe's scheme.

Algorithm 1: **solve** Burgers equation with initial datum $\{u_j^0\}_{j=1,\ldots,N}^k \to \{u_j^n\}_{j=1,\ldots,N}^{n=1,\ldots,M}$

```
1   input Δx, Δt, {u_j^0}_{j=1,...,N}
2   set λ = Δt/Δx
3   for n = 0(1)M − 1 repeat
4       set u_1^{n+1} = u_1^0, u_N^{n+1} = u_N^0
5       for j = 2(1)N − 1 repeat
6           set u_j^{n+1} = u_j^n + λ(g(u_j^n, u_{j+1}^n) − g(u_{j-1}^n, u_j^n))
7       end
8   end
```

Line	Comments
2	λ satisfies the CFL condition.
6	g is the numerical convective flux.

Algorithm 2: **solve** adjoint equation with final datum $\{p_j^T\}_{j=1,\dots,N} \rightarrow \{p_j^0\}_{j=1,\dots,N}$

1 **input** Δx, Δt, $\{u_j^n\}_{j=1,\dots,N}$
2 **set** $\lambda = \Delta t/\Delta x$
2 **for** $n = 0(1)M$ **repeat**
3   **set** $p_1^{n-1} = p_1^M$, $p_N^{n-1} = p_N^M$,
4   **for** $j = 2(1)N - 1$ **repeat**
5   **set** $p_j^{n-1} = p_j^n + \lambda(\partial_1 g(u_j^{n-1}, u_{j+1}^{n-1}) * (p_j^n - p_{j+1}^n)$
6        $+\partial_2 g(u_{j-1}^{n-1}, u_j^{n-1}) * (p_{j-1}^n - p_j^n))$
7   **end**
8 **end**

Line	Comments
2	λ satisfies the CFL condition.
6	g is the numerical convective flux.

Algorithm 3: Discrete approach

STEP 0:   initialization
1   **input** Δx, Δt, $\{u_j^0\}_{j=1,\dots,N}$, $\{u^d\}_{j=1,\dots,N}$
2   **set** $\lambda = \Delta t/\Delta x$
3   **solve** Burgers equation with initial datum $\{u_j^0\}_{j=1,\dots,N}^k \rightarrow \{u_j^n\}_{j=1,\dots,N}^{n=1,\dots,M}$
4   **for** $j = 1(1)N$ **repeat**
5   **set** $p_j^T = u_j^M - u_j^d$, $p_j^M = p_j^T$
6   **end**
7   **solve** adjoint equation with final datum $\{p_j^T\}_{j=1,\dots,N} \rightarrow \{p_j^0\}_{j=1,\dots,N}$

STEP 1:   optimization loop
1   **input** ε
2   **for** $k = 0, 1, \dots$**repeat**
3   **solve** Burgers equation with initial datum $\{u_j^0\}_{j=1,\dots,N}^k \rightarrow \{u_j^n\}_{j=1,\dots,N}^{n=1,\dots,M}$
4   **for** $j = 1(1)N$ **repeat**
5   **set** $p_j^T = u_j^M - u_j^d$
6   **end**
7   **solve** adjoint equation with final datum $\{p_j^T\}_{j=1,\dots,N} \rightarrow \{p_j^0\}_{j=1,\dots,N}$
8   **set** $g_k = \{p_j^0\}_{j=1,\dots,N}$,
9   **compute** α_k
10   **set** $\{u_j^0\}_{j=1,\dots,N}^{k+1} = \{u_j^0\}_{j=1,\dots,N}^k - \alpha_k * g_k$
11   **end until** $||g_{k+1}|| < \varepsilon$

Line	Comments
1	ε is the tolerance.
9	Compute the descent step α_k arg min. $J(\{u_j^0\}_{j=1,...,N}^k - \alpha_k * g_k)$.
11	$\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^N$.

We now consider the *continuous approach, imposing the internal conditions along the shock.* In this case he algorithm must be slightly modified in order to take into account the presence of discontinuities. Thus we first describe a subroutine to find the "discontinuities" of a vector $\{u_j\}_{j=1,...,N}$ based on a shift condition. Roughly, we introduce two parameters α and ρ and we look for the indexes j where

$$\frac{u_{j-\alpha} - u_{j+\alpha}}{|u_{j-\alpha}|} > \rho.$$

To simplify the presentation we consider the case in which only one discontinuity is relevant in the numerical experiment, and arises in the discrete vector.

Algorithm 4: **solve** jump$(\{u_j\}_{j=1,...,N}, \Delta x) \rightarrow (index, u_{left}, u_{right})$

1 **input** sh, js, $\{u_j\}_{j=1,...,N}$, Δx
2 **set** $\alpha = \text{INT}(sh/\Delta x)$
3 **set** $max = \arg\max_j(u_{j-\alpha} - u_{j+\alpha})/\text{ABS}(u_{j-\alpha})$
4 **if** $max > js$ **then**
5   **set** $index = max$
6   **set** $u_{left} = u_{j-index}$, $u_{right} = u_{j+index}$
7 **end**

Line	Comments
1	sh is the shift parameter and the js the jump sensibility parameter

The complete algorithm is now as follows:

Algorithm 5: Continuous approach: interior conditions on the shock

STEP 0: initialization.
1   **input** Δx, Δt, $\{u^d\}_{j=1,\dots,N}$, $\{u_j^0\}_{j=1,\dots,N}$
2   **solve** jump$(\{u_j^{obj}\}_{j=1,\dots,N}, \Delta x) \to (index^{obj}, ul^{obj}, ur^{obj})$
3   **solve** jump$(\{u_j^0\}_{j=1,\dots,N}, \Delta x) \to (index^0, ul^0, ur^0)$
4   **solve** Burgers equation with initial datum $\{u_j^0\}_{j=1,\dots,N}^k \to \{u_j^n\}_{j=1,\dots,N}^{n=1,\dots,M}$
5   **solve** jump$(\{u_j^M\}_{j=1,\dots,N}, \Delta x) \to (index^T, ul^T, ur^T)$! Shock position at $t = T$
6   **for** $j = 1(1)N$ **repeat**
7   **set** $p_j^T = u_j^M - u_j^d$, $p_j^M = p_j^T$
8   **end**
9   **set** $q^T = ((ur^T - ur^{obj})^2/2 - (ul^T - ul^{obj})^2/2)/(ur^T - ul^T)$
10   **solve** adjoint equation with final datum $\{p_j^T\}_{j=1,\dots,N} \to \{p_j^0\}_{j=1,\dots,N}$
11   **for** $j = index^0 - \mathrm{INT}(ul^T * T)(1)index^0 - \mathrm{INT}(ur^T * T)$ **repeat**
12           $p_j^0 = q^T$
13   **end**
14   **set** $(g_1, g_2) = (\{p_j^0\}_{j=1,\dots,N}, q^T/(ur^0 - ul^0))$

Line   Comments
 11   Impose the internal condition on the shock.
 14   (g_1, g_2) is the gradient.

STEP 1:   optimization loop
1   **input** ε
2   **for** $k = 0, 1, ...$**repeat**
3   **compute** α_k
4   **set** $\{u_j^0\}_{j=1,...,N}^{k+1} = \{u_j^0\}_{j=1,...,N}^{k} - \alpha_k * g_{1,k}$
5   **if** $g_2 > 0$ **then**
6   **for** $j = index^0(1)index^0 + \text{INT}(g_2 * \alpha_k / \Delta x)$
7   **set** $u_j^0 = ul^0$
8   **end**
9   **else**
10   **for** $j = index^0 + \text{INT}(g_2 * \alpha_k / \Delta x)(1)index^0$
11   **set** $u_j^0 = ur^0$
12   **end**
13   **end**
14   **solve** Burgers equation with initial datum $\{u_j^0\}_{j=1,...,N}^{k} \to \{u_j^n\}_{j=1,...,N}^{n=1,...,M}$
15   **solve** jump$(\{u_j^M\}_{j=1,...,N}, \Delta x) \to (index^T, ul^T, ur^T)$! Shock position at $t = T$
16   **for** $j = 1(1)N$ **repeat**
17   **set** $p_j^T = u_j^M - u_j^d$
18   **end**
19   **set** $q^T = ((ur^T - ur^{obj})^2/2 - (ul^T - ul^{obj})^2/2)/(ur^T - ul^T)$
20   **solve** adjoint equation with final datum $\{p_j^T\}_{j=1,...,N} \to \{p_j^0\}_{j=1,...,N}$
21   **set** internal boundary condition on the shock
22   **set** $(g_1, g_2)_k = (\{p_j^0\}_{j=1,...,N}, q^T/(ur^0 - ul^0))$
23   **end until** $||(g_1, g_2)_{1,k+1}|| < \varepsilon$

Line	Comments
1	ε is the tolerance.
3	Compute the descent step α_k arg min $J(\{u_j^0\}_{j=1,...,N}^{k} - \alpha_k * g_{1,k})$
5:13	Move the discontinuity.
22	(g_1, g_2) is the gradient.
23	$||\cdot||$ is the Euclidean norm in $\mathbb{R}^{N+1}$.

Finally we describe the algorithm for the *alternating descent method* we propose.

Algorithm 6: Alternating descent method

STEP 0:   initialization.
1   **input** Δx, Δt, $\{u^d\}_{j=1,\ldots,N}$, $\{u^0_j\}_{j=1,\ldots,N}$
2   **solve** jump$(\{u^{obj}_j\}_{j=1,\ldots,N}, \Delta x) \rightarrow (index^{obj}, ul^{obj}, ur^{obj})$
3   **solve** jump$(\{u^0_j\}_{j=1,\ldots,N}, \Delta x) \rightarrow (index^0, ul^0, ur^0)$
4   **solve** Burgers equation with initial datum $\{u^0_j\}^k_{j=1,\ldots,N} \rightarrow \{u^n_j\}^{n=1,\ldots,M}_{j=1,\ldots,N}$
5   **solve** jump$(\{u^M_j\}_{j=1,\ldots,N}, \Delta x) \rightarrow (index^T, ul^T, ur^T)$
6   **set** $q^T = ((ur^T - ur^{obj})^2/2 - (ul^T - ul^{obj})^2/2)/(ur^T - ul^T)$
7   **set** $(g_1, g_2) = (0, q^T)$! first generalized tangent vector
8   **solve** discontinuity position (using optimal step) $\rightarrow \{u^0_j\}_{j=1,\ldots,N}$
9   **solve** jump$(\{u^0_j\}_{j=1,\ldots,N}, \Delta x) \rightarrow (index^0, ul^0, ur^0)$
10   **solve** Burgers equation with initial datum $\{u^0_j\}^k_{j=1,\ldots,N} \rightarrow \{u^n_j\}^{n=1,\ldots,M}_{j=1,\ldots,N}$
11   **solve** jump$(\{u^M_j\}_{j=1,\ldots,N}, \Delta x) \rightarrow (index^T, ul^T, ur^T)$
12   **for** $j = 1(1)N$ **repeat**
13   **set** $p^T_j = u^M_j - u^d_j$, $p^M_j = p^T_j$
14   **end**
15   **set** $q^T = ((ur^T - ur^{obj})^2/2 - (ul^T - ul^{obj})^2/2)/(ur^T - ul^T)$
16   **solve** adjoint equation with final datum $\{p^T_j\}_{j=1,\ldots,N} \rightarrow \{p^0_j\}_{j=1,\ldots,N}$
17   **solve** jump$(\{u^M_j - u^{obj}_j\}_{j=1,\ldots,N}, \Delta x) \rightarrow (index^T, pl^T, pr^T)$
18   **for** $j = index^0 - \text{INT}(ul^T * T)(1)index^0$ **repeat**
19        $p^0_j = pl^T$
20   **end**
21   **for** $j = index^0(1)index^0 - \text{INT}(ur^T * T)$
22        $p^0_j = pr^T$
23   **end**
24   **set** $(g_1, g_2) = (\{p^0_j\}_{j=1,\ldots,N}, q^T/(ur^0 - ul^0))$, ! second tangent vector

Line	Comments
4:9	Optimize location of the discontinuity.
8	See lines 5:13 of *Algorithm 4*, STEP 1.
17:23	Impose the condition in (5.6).
24	(g_1, g_2) is the gradient.

STEP 1:	optimization loop

```
1          input ε
2          for k = 0, 1, ...repeat
3              compute α_k
```

1. **input** ε
2. **for** $k = 0, 1, ...$**repeat**
3.   **compute** α_k
4.   **set** $\{u_j^0\}_{j=1,...,N}^{k+1} = \{u_j^0\}_{j=1,...,N}^k - \alpha_k * g_{1,k}$
5.   **solve** Burgers equation with initial datum $\{u_j^0\}_{j=1,...,N}^k \to \{u_j^n\}_{j=1,...,N}^{n=1,...,M}$
6.   **solve** jump($\{u_j^M\}_{j=1,...,N}, \Delta x) \to (index^T, ul^T, ur^T)$! Shock position at $t = T$
7.   **for** $j = 1(1)N$ **repeat**
8.   **set** $p_j^T = u_j^M - u_j^d$
9.   **end**
10.   **set** $q^T = ((ur^T - ur^{obj})^2/2 - (ul^T - ul^{obj})^2/2)/(ur^T - ul^T)$
11.   **solve** adjoint equation with final datum $\{p_j^T\}_{j=1,...,N} \to \{p_j^0\}_{j=1,...,N}$
12.   **set** impose the condition in (5.6)
13.   **set** $(g_1, g_2)_k = (\{p_j^0\}_{j=1,...,N}, q^T/(ur^0 - ul^0))$
14.   **end until** $||(g_1, g_2)_{1,k+1}|| < \varepsilon$

Line	Comments
1	ε is the tolerance.
3	Compute the descent step α_k arg min $J(\{u_j^0\}_{j=1,...,N}^k - \alpha_k * g_{1,k})$

References

[1] Bardos C. and Pironneau O., A formalism for the differentiation of conservation laws, C. R. Acad. Sci., Paris, Ser I 335 (2002) 839-845.

[2] Bardos C. and Pironneau O., Derivatives and control in presence of shocks, Computational Fluid Dynamics Journal, 11 (4) (2003), 383-392.

[3] Bouchut F. and James F., One-dimensional transport equations with discontinuous coefficients, Nonlinear Analysis, Theory and Applications, **32** (7) (1998), 891-933.

[4] Bouchut F. and James F., Differentiability with respect to inital data for a scalar conservation law, Proceedings of the 7th Int. Conf. on Hyperbolic Problems, Zurich 98, International Series Num. Math. 129, pages 113-118. Birkäuser, 1999.

[5] Bouchut, F., James, F. and Mancini, S. Uniqueness and weak stability for multi-dimensional transport equations with one-sided Lipschitz coefficient. Ann. Sc. Norm. Super. Pisa Cl. Sci. **4**(5) (2005), no. 1, 1–25.

[6] Brenier, Y. and Osher, S. The Discrete One-Sided Lipschitz Condition for Convex Scalar Conservation Laws, SIAM Journal on Numerical Analysis, **25** (1) (1988), 8-23.

[7] Bressan A. and Marson A., A variational calculus for discontinuous solutions of systems of conservation laws, Commun. in Partial Differential Equations, **20** (9 10), (1995) 1491-1552.

[8] Bressan A. and Marson A., A maximum principle for optimally controlled systems of conservation laws, Rend. Sem. Mat. Univ. Padova, **94** (1995), 79-94

[9] Dal Maso G., Le Floch Ph. and Murat F., Definition and weak stability of nonconservative products, J. Math. Pures Appl. **74** (1995), 458-483.

[10] Escobedo M., Vázquez J.L. and Zuazua E., Asymptotic behaviour and source-type solutions for a diffusion-convection equation, Arch. Rat. Mech. Anal., **124** (1),(1993), 43-65.

[11] Garreau, S., Guillaume, Ph. and Masmoudi, M. The topological asymptotic for PDE systems: the elasticity case. SIAM J. Control Optim. **39** (6) (2001), 1756–177

[12] Giles M.B. and Pierce N.A., Analytic adjoint solutions for the quasi one-dimensional Euler equations, J. Fluid Mechanics, **426** (2001), 327-345.

[13] Glowinski, *Numerical Methods for Fluids (Part 3)*, R. Handbook of Numerical analysis, vol. IX, Ph. Ciarlet and J. L. Lions, eds., Elsevier, 2003

[14] Godlewski E. and Raviart P.A., The linearized stability of solutions of nonlinear hyperbolic systems of conservation laws. A general numerical approach, Mathematics and Computer in Simulations, **50** (1999), 77-95.

[15] Godlewski E. and Raviart P.A., *Hyperbolic systems of conservation laws*, Mathématiques and Applications 3/4, Ellipses, Paris, 1991.

[16] Godlewski E., Olazabal M., Raviart P.A., On the linearization of hyperbolic systems of conservation laws. Application to stability, in: Equations différentielles et Applications, articles dédiés à Jacques-Louis Lions, Gauthier-Villars, Paris, 1998, pp. 549-570.

[17] Godlewski E. and Raviart P.A., Numerical approximation of hyperbolic systems of conservation laws, Springer Verlag, 1996.

[18] Gosse, L. and JAmes, F. Numerical approximations of one-dimensional linear conservation equations with discontinuious coefficients, Mathematics of Computation, **69**, 231 (2000), 987-1015 .

[19] Hirsch C., *Numerical computation of internal and external flows*, Vol. 1 and 2, John Wiley and Sons, 1988.

[20] James, F. and Sepúlveda, M. Convergence results for the flux identification in a scalar conservation law. SIAM J. Control Optim. **37** (3) (1999), 869–891

[21] LeVeque R., *Finite Volume Methods for Hyperbolic Problems*, Cambridge Univesity Press, 2002.

[22] Majda A., The stability of multidimensional shock fronts, AMS 1983.

[23] Métivier G., Stability of multidimensional shocks, course notes at http://www.math.u-bordeaux.fr/ metivier/cours.html (2003).

[24] Mohammadi B. and Pironneau O., Shape optimization in fluid mechanics, Annual Rev Fluids Mechanics, **36** (2004), 255-279.

[25] Nadarajah S., and Jameson A., A Comparison of the Continuous and Discrete Adjoint Approach to Automatic Aerodynamic Optimization, AIAA Paper 2000-0667, 38th Aerospace Sciences Meeting and Exhibit, January 2000, Reno, NV.

[26] O. Oleinik, Discontinuous solutions of nonlinear differential equations, Uspekhi Mat. Nauk., **12** (1957), pp. 3-73. (In Russian.) Amer. Math. Soc. Trans]., **26**, pp. 95-172. (In English.)

[27] Ubrich S., Adjoint-based derivative computations for the optimal control of discontinuous solutions of hyperbolic conservation laws, Systems and Control Letters 48 (2003), 313-328.
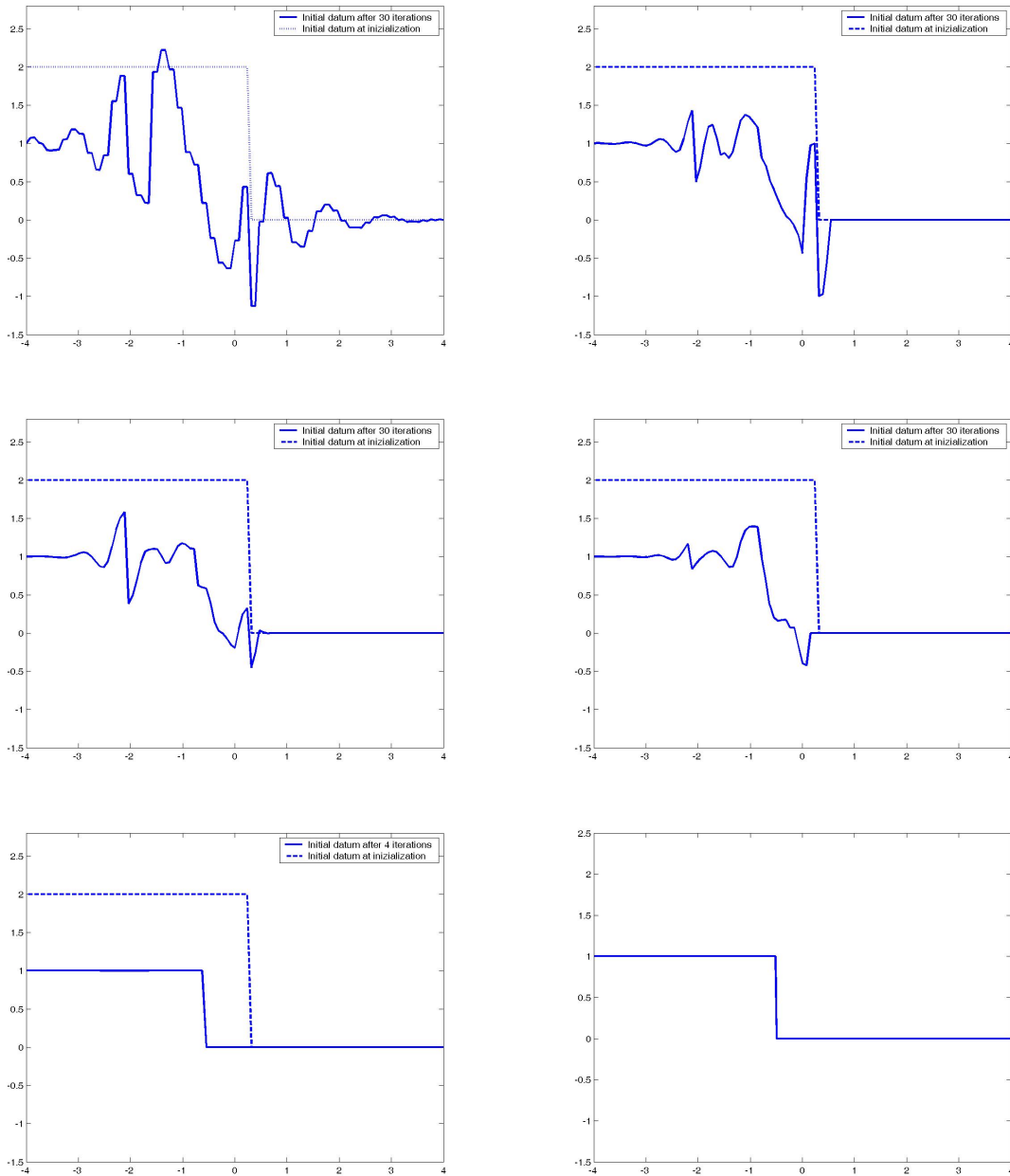
Figure 5: Experiment 1. Initialization (dashed line) and initial data obtained after 30 iterations (solid line) with Lax-Friedrichs (upper left) , Engquist-Osher (upper right), Roe (middle left), the continuous approach imposing a boundary condition on the shock (middle right) and the alternating descent method (lower left). A minimizer u^0 of the continuous functional is given in the lower right figure.
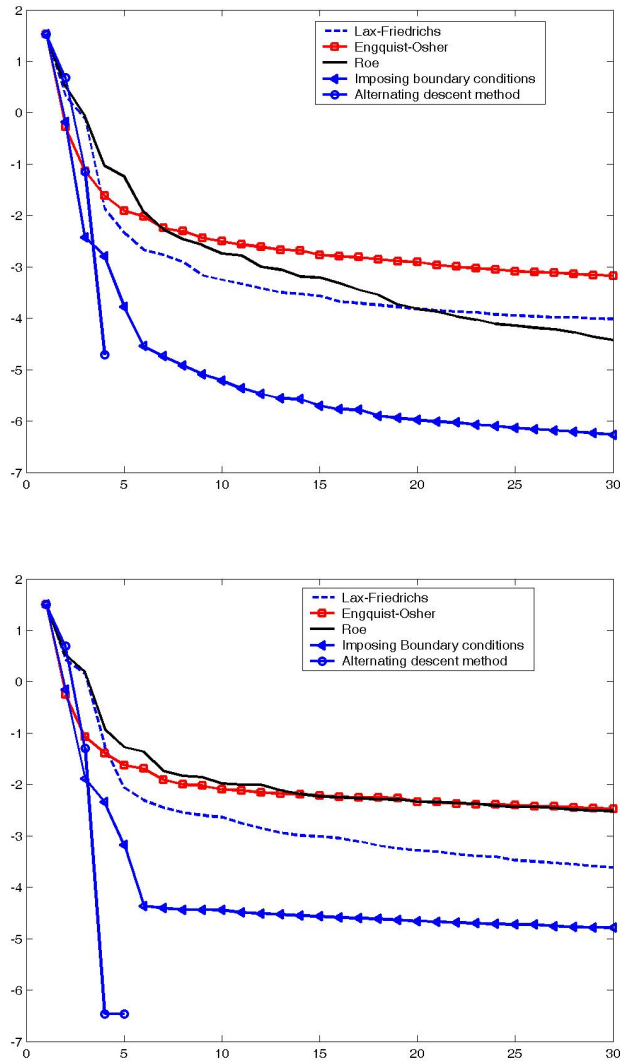
45

Figure 6: Experiment 1. Log of the value of the functional versus the number of iterations in the descent algorithm for the Lax-Friedrichs, Engquist-Osher and Roe schemes, the continuous approach imposing the internal boundary condition on the shock and the alternating descent method proposed in this article. The upper figure corresponds to $\Delta x = 1/20$ and the lower one to $\Delta x = 1/80$. We see that the last method stabilizes in a few iterations and it is much more efficient when consider small enough values of Δx in order to be able to resolve the shock sufficiently well.
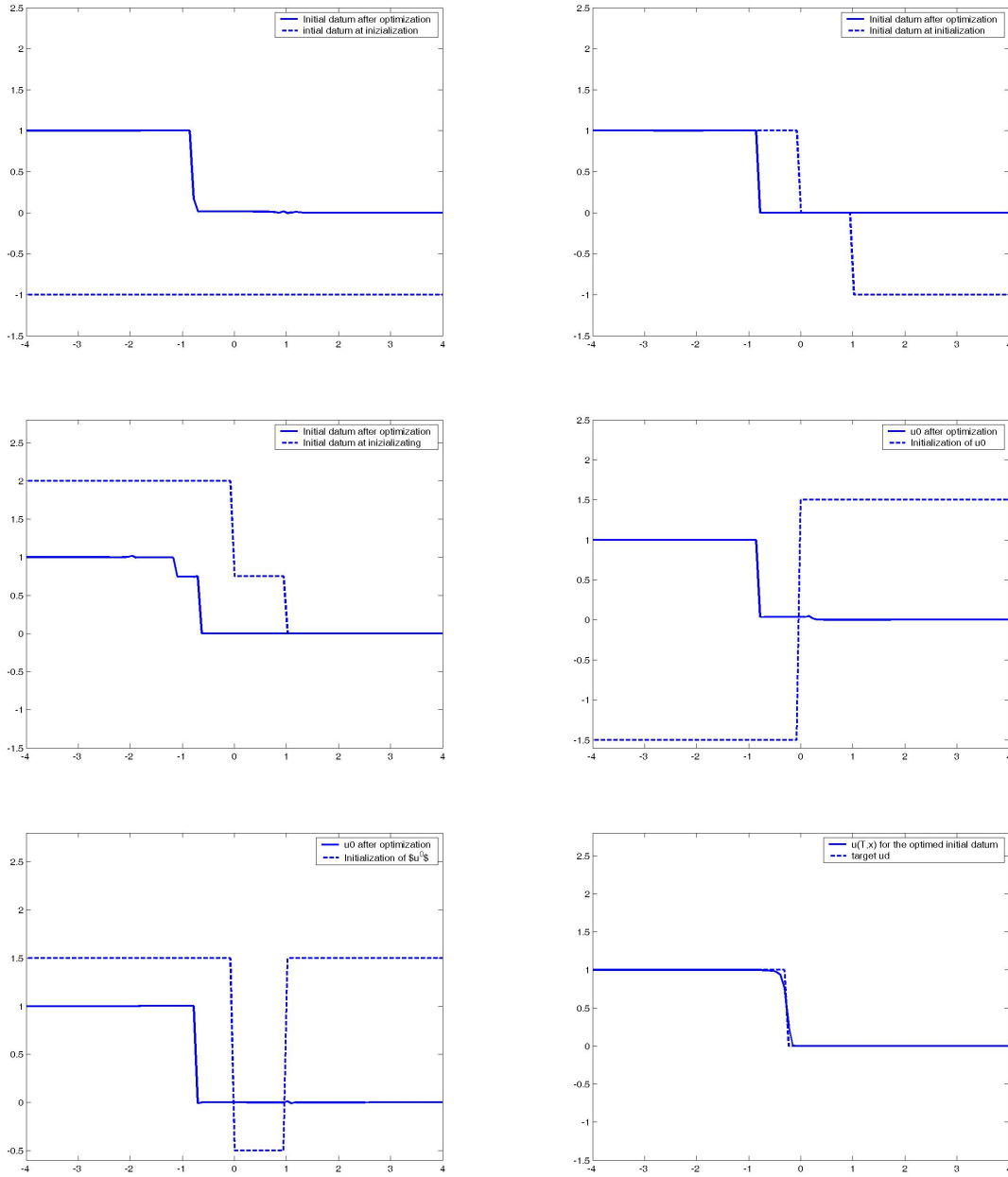
Figure 7: Experiment 2. The four upper figures and the lower left one show the initial data obtained once the descent iteration stops (solid) with different initialization functions u^0 (dashed) with the alternating descent method proposed in this article. In the lower right figure, the target $u^d(x)$ and the solution $u(x, T)$ (here $T = 1$) corresponding to the obtained u^0 are drawn for the last initialization. The function $u(x, T)$ one obtains for the other initializations is very similar to this one. In this experiment $\Delta x = 1/20$ and $\lambda = \Delta t / \Delta x = 1/2$.
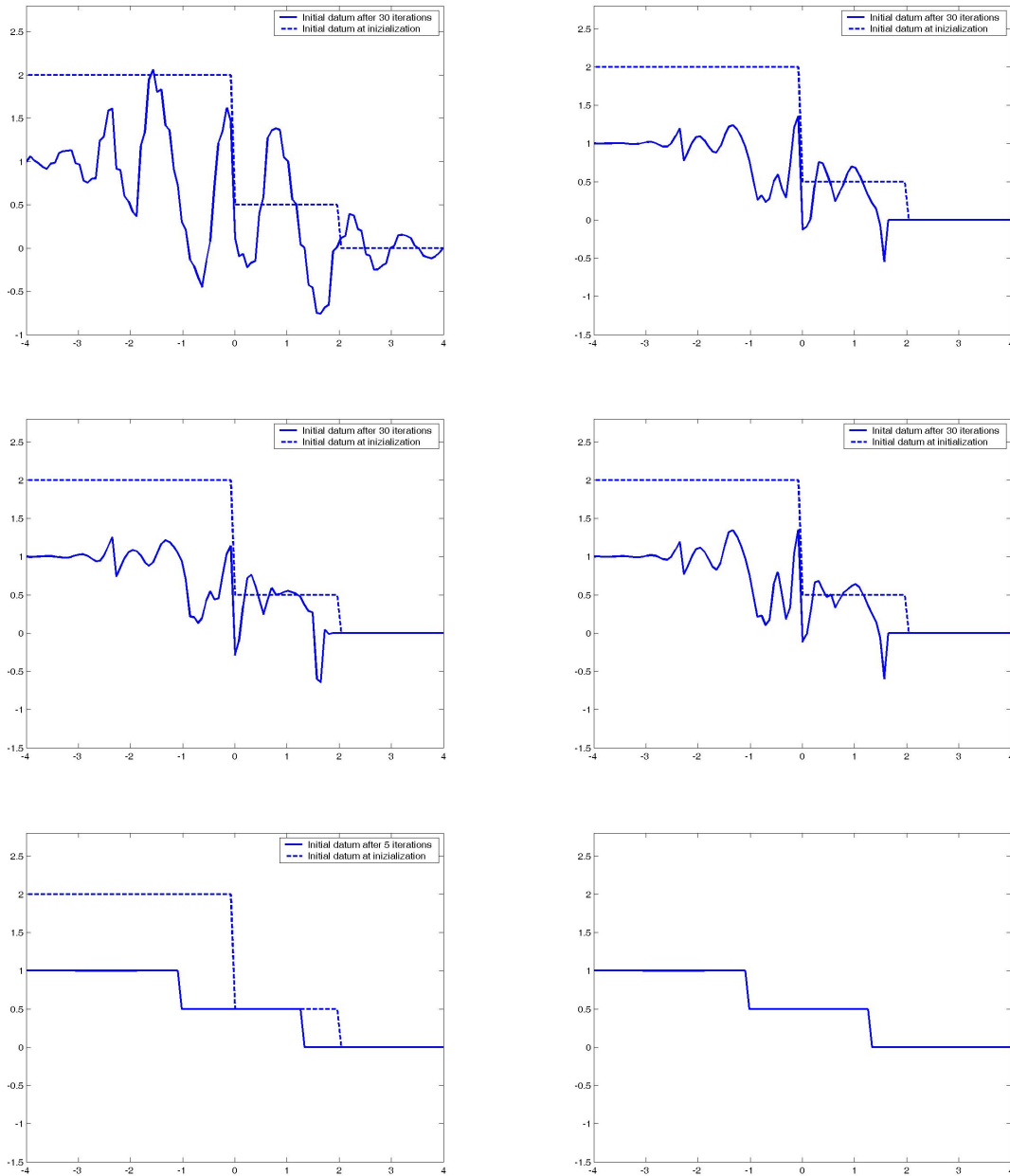
Figure 8: Experiment 3. Initialization (dashed line) and initial data obtained after 30 iterations (solid line) with Lax-Friedrichs (upper left) , Engquist-Osher (upper right), Roe (middle left), the continuous approach imposing a boundary condition on the shock (middle right) and the alternating descent method (lower left). A minimizer u^0 of the continuous functional is given in the lower right figure.
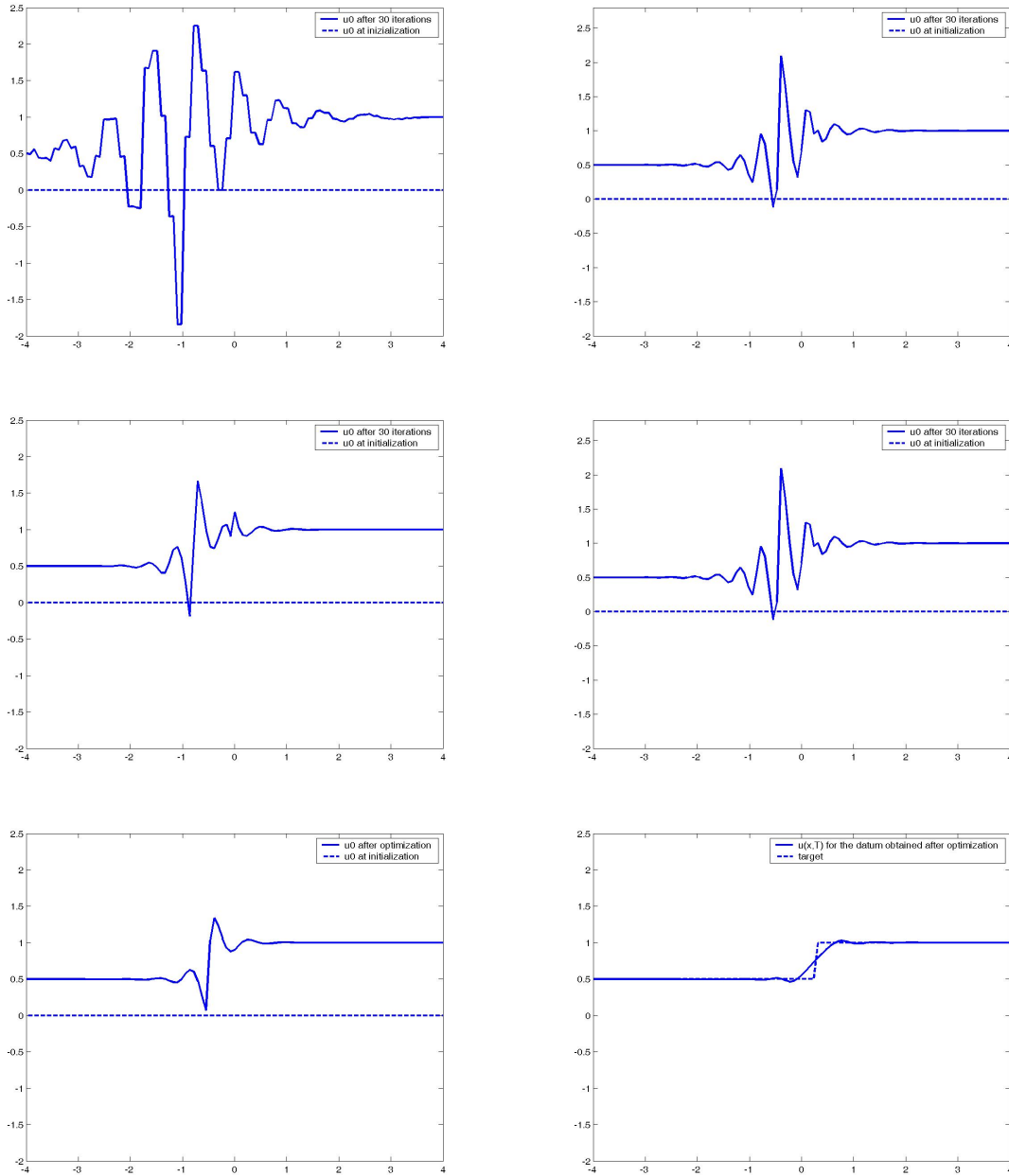
Figure 9: Experiment 4. Initial datum obtained after 30 iterations with the Lax-Friedrichs (upper left) , Engquist-Osher (upper right), Roe schemes (medium left), imposing internal boundary conditions (medium right) and the alternating descent method (lower left). In the lower right figure there are both the target u^d given in (7.6) (dashed) and the solution of the Burgers equation u at time $t = T = 1$ with the initial datum obtained after optimization with the alternating descent method (solid).
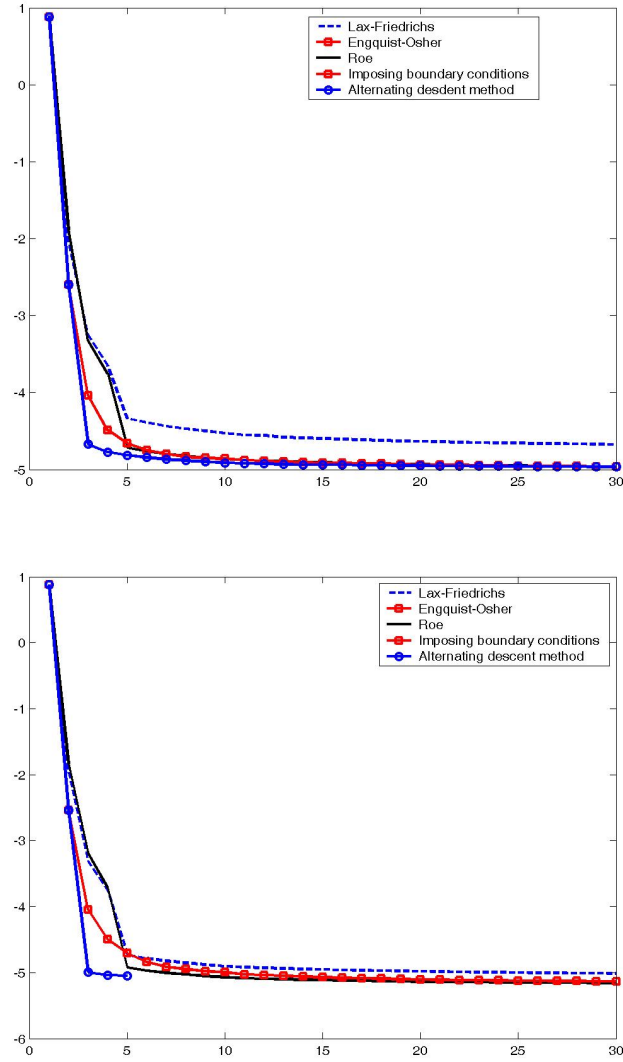
Figure 10: Experiment 4. Log of the value of the functional versus the number of iteration in the descent algorithm for the Lax-Friedrichs, Engquist-Osher, Roe schemes, imposing internal boundary conditions and the alternating descent method we propose. Note that, in this case, no shocks are involved and therefore the Engquist-Osher scheme and the continuous method imposing the boundary conditions coincide. Here the upper figure corresponds to $\Delta x = 1/20$ and the lower one to $\Delta x = 1/40$. In both cases $\lambda = \Delta t/\Delta x = 1/2$.