Numerical aspects of large-time optimal control of Burgers equation

Navid Allahverdi Alejandro Pozo Enrique Zuazua

1 Introduction

In this work, we analyze the numerical approximation of the inverse design problem for the Burgers equation, both in the viscous and in the inviscid case:

$$\min_{u_0} \mathcal{J}_\nu(u_0) = \min_{u_0} \frac{1}{2} \int_{\mathbb{R}} \left(u^\nu(x,T) - u^*(x)\right)^2 dx, \tag{1}$$

$$\text{subject to} \begin{cases} \partial_t u^\nu + \partial_x \left(\frac{(u^\nu)^2}{2}\right) = \nu \partial_{xx} u^\nu, & x \in \mathbb{R},\, t > 0, \\ u^\nu(x,0) = u_0(x), & x \in \mathbb{R}, \end{cases} \tag{2}$$

distinguishing the viscous, $\nu > 0$, and the inviscid case, $\nu = 0$. Given a time $T > 0$ and a target function u^* the aim is to identify the initial datum u_0 so that the solution, at time $t = T$, reaches the target u^* or gets as close as possible to it. In particular, we focus on problems with large values of T, for which convergence of the numerical schemes in the classical sense of numerical analysis might not suffice to obtain accurate results.

This issue is motivated by the challenging problem of sonic-boom minimization for supersonic aircrafts, which is governed by a Burgers-like equation. The travel time of the signal to the ground is larger than the time scale of the initial disturbance by orders of magnitude and this motivates our study of large time control of the sonic-boom propagation.

2 Description of the numerical algorithms

To implement the problem numerically, we opt for a discretization of (2) using classical conservative schemes. Let us denote spatial nodes $x_{j+1/2} = \Delta x(j + 1/2)$, $j \in \mathbb{Z}$, and time instants $t_n = n\Delta t$, $n \in \mathbb{N} \cup \{0\}$, where $\Delta x, \Delta t > 0$ are the mesh size and time-step respectively. We approximate the solution u of (2) by a piecewise constant function u_Δ such that

$$u_\Delta(x,t) = u_j^n, \quad x \in [x_{j-1/2}, x_{j+1/2}),\, t \in [t_n, t_{n+1}),$$

where

$$\begin{cases} u_j^{n+1} = u_j^n - \dfrac{\Delta t}{\Delta x}(g_{j+1/2}^n - g_{j-1/2}^n) \\ \qquad\qquad + \dfrac{\nu \Delta t}{\Delta x^2}(u_{j-1}^n - 2u_j^n + u_{j+1}^n),\ j \in \mathbb{Z}, n = 0, \dots, N, \\ u_j^0 = \dfrac{1}{\Delta x} \displaystyle\int_{x_{j-1/2}}^{x_{j+1/2}} u_0(x) dx,\ j \in \mathbb{Z}. \end{cases} \tag{3}$$

Here, $N = \lceil T/\Delta t \rceil$ is the number of time-steps needed to reach T. We denote $g^n_{j+1/2} = g(u^n_j, u^n_{j+1})$, where the function g is the numerical flux. In this paper we compare two fluxes:

$$\text{Engquist-Osher (EO): } g^{EO}(v,w) = \frac{v(v+|v|)}{4} + \frac{w(w-|w|)}{4},$$

$$\text{Modified Lax-Friedrichs (MLF): } g^{MLF}(v,w) = \frac{v^2+w^2}{4} - \frac{\Delta x}{\Delta t}\left(\frac{w-v}{4}\right).$$

In the simulations we follow the discrete approach to optimization. For the discretization of (1), we consider a simple quadrature rule:

$$\mathcal{J}_\Delta(u^0_\Delta) = \frac{\Delta x}{2}\sum_{\mathbb{Z}}(u^N_j - u^*_j)^2, \tag{4}$$

where the target function u^* has been discretized in the same manner as the initial data u_0 in (3).

Regarding the optimization technique, we use the classical gradient descent method based on the adjoint methodology. Following the discrete approach, it is easy to obtain the corresponding discrete adjoint system for (3):

$$\begin{cases} \rho^n_j = \rho^{n+1}_j + \frac{\Delta t}{\Delta x}\Big(\partial_v g(u^n_j, u^n_{j+1})\left(\rho^{n+1}_{j+1} - \rho^{n+1}_j\right) \\ \qquad\qquad +\partial_w g(u^n_{j-1}, u^n_j)\left(\rho^{n+1}_j - \rho^{n+1}_{j-1}\right)\Big) \\ \qquad +\frac{\nu\Delta t}{\Delta x^2}(\rho^{n+1}_{j-1} - 2\rho^{n+1}_j + \rho^{n+1}_{j+1}), \qquad j\in\mathbb{Z}, n = N-1\ldots,0, \\ \rho^N_j = u^N_j - u^*_j, \qquad\qquad j\in\mathbb{Z}, \end{cases} \tag{5}$$

To minimize (4), we will take the descent direction given by:

$$\delta u^0_j = -\rho^0_j, \quad j\in\mathbb{Z}.$$

This direction is straightforwardly obtained following the same arguments as for the continuous level. Thus, the new initial data $u^{0,\varepsilon}_\Delta$ will be given by

$$u^{0,\varepsilon}_j = u^0_j - \varepsilon\rho^0_j, \quad j\in\mathbb{Z}$$

for some $\varepsilon > 0$ small enough.

The pseudocode for the complete algorithm can be found in Algorithm 1. Note that it is valid both for the viscous and the inviscid case of the Burgers equation.

3 Numerical viscosity

It is well known that (3) can be rewritten in its viscous form in the following manner:

$$\frac{u^{n+1}_j - u^n_j}{\Delta t} + \frac{(u^n_{j+1})^2 - (u^n_{j-1})^2}{4\Delta x} = R(u^n_j, u^n_{j+1}) - R(u^n_{j-1}, u^n_j) + \frac{\nu}{\Delta x^2}(u^n_{j-1} - 2u^n_j + u^n_{j+1}),$$

where R is uniquely defined by

$$R(u,v) = \frac{1}{2\Delta x}\Big(\frac{u^2}{2} + \frac{v^2}{2} - 2g(u,v)\Big).$$

Algorithm 1 Solve discrete optimization problem

Input: Δx, Δt, N, $\{u_j^0\}_{j=0,\dots,M}$, $\{u_j^*\}_{j=0,\dots,M}$

for $j = 0$ **to** M **do**
  set $u_j^{0,new} = u_j^0$
end for
compute $\{u_j^n\}_{j=0,\dots,M}^{n=1,\dots,N}$ from $\{u_j^{new,0}\}_{j=0,\dots,M}$ using (3)
compute functional $\mathcal{J}(u_\Delta^{0,new})$ using (4)
while stopping criteria are not met **do**
  for $j = 0$ **to** M **do**
    set $u_j^{0,old} = u_j^{0,new}$
    set $\rho_j^N = u_j^N - u_j^*$
  end for
  compute $\{\rho_j^0\}_{j=0,\dots,M}$ from $\{\rho_j^N\}_{j=0,\dots,M}$ and $\{u_j^n\}_{j=0,\dots,M}^{n=1,\dots,N}$ using (5)
  compute descending step-size ε
  for $j = 0$ **to** M **do**
    set $u_j^{0,new} = u_j^{0,old} - \varepsilon\rho_j^0$
  end for
  compute $\{u_j^n\}_{j=0,\dots,M}^{n=1,\dots,N}$ from $\{u_j^{new,0}\}_{j=0,\dots,M}$ using (3)
  compute functional $\mathcal{J}(u_\Delta^{0,new})$ using (4)
end while
for $j = 0$ **to** M **do**
  set $u_j^{*,0} = u_j^{0,new}$
end for

Output: Optimal solution $\{u_j^{*,0}\}_{j=0,\dots,M}$

In the case of the numerical fluxes that we consider in this paper, we have:

$$R^{MLF}(u,v) = \frac{v-u}{4\Delta t},$$
$$R^{EO}(u,v) = \frac{1}{4\Delta x}(v|v| - u|u|).$$

Both schemes are convergent in the classical sense of the numerical analysis. However, the large-time behavior of u_Δ depends on the degree of homogeneity of the term R. In other words, let us assume that there exists $\alpha \in \mathbb{R}$ such that

$$R(\mu u, \mu v) = \mu^\alpha R(u,v), \qquad \forall u, v \in \mathbb{R} \text{ and } \forall \mu > 0.$$

It is clear that $\alpha = 2$ for Engquist-Osher and $\alpha = 1$ for modified Lax-Friedrichs. Thus, the numerical viscosity inherent in MLF drives the system into a diffusive wave too early and, consequently, continuous metastable states are not reproduced numerically.

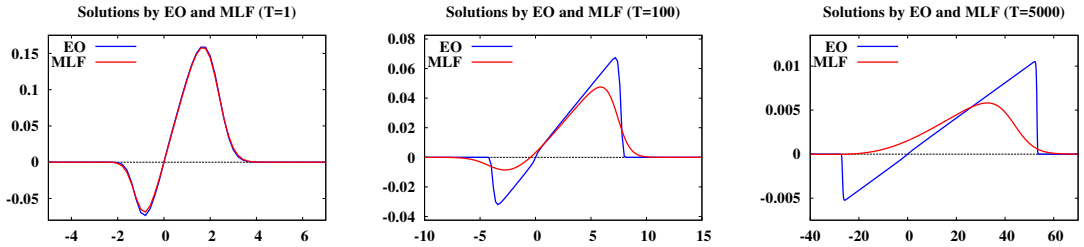

Figure 1: Solutions of (2) with $\nu = 10^{-6}$ at $t = 1$, $t = 100$ and $t = 5000$, using (3) with $\Delta x = 0.2$, $\Delta t = 0.5$ and numerical fluxes EO (blue) and MLF (red).

The main aim of our work is to emphasize that ignoring the dynamics of the continuous model at the numerical level can produce undesired results in optimal control problems in large time horizons. On one hand, we show that the gradient descent method performs successfully whenever the numerical flux and the mesh sizes are chosen appropriately. On the other hand, we present examples where excessive numerical viscosity dominates the physical one.