Sparse source identification of linear diffusion–advection equations by adjoint methods

Azahar Monge [*] [†]

Enrique Zuazua [*] [†] [‡] [§]

July 7, 2019

Abstract

We present an algorithm for the time-inversion of diffusion–advection equations, based on the adjoint methodology. Given a final state distribution our main aim is to recover sparse initial conditions, constituted by a finite combination of Dirac deltas, identifying their location and mass. We discuss the strengths of the adjoint machinery and the difficulties that are to be faced, in particular when the diffusivity coefficient or the time horizon are large.

Keywords: Inverse Problems, Adjoint Problem, Optimal Control, Optimization, Diffusion–Advection Equation

1 Introduction

Inverse problems are widely found in many applications in science and engineering: in acoustics [13, 27], medical imaging [4], geophysics [26], oceanography [28], machine learning [25], nondestructive testing [10], and many other fields.

The identification of moving pollution sources in either compressible or incompressible fluids is a prototypical and relevant example, [16]. In that framework it is natural to assume that the initial condition to be found is a linear combination of Dirac deltas, whose locations and their corresponding weights or intensities are to be identified.

[*] DeustoTech, University of Deusto, 48007 Bilbao, Basque Country, Spain. Emails: azahar.monge@deusto.es, enrique.zuazua@deusto.es

[†] Facultad Ingeniería, Universidad de Deusto, Avda. Universidades, 24, 48007, Bilbao, Basque Country, Spain.

[‡] Departamento de Matemáticas, Universidad Autónoma de Madrid, 28049 Madrid, Spain.

[§] Sorbonne Universités, UPMC Univ Paris 06, CNRS UMR 7598, Laboratoire Jacques-Louis Lions, F-75005, Paris, France.

Some work has already been done in this direction. A straight forward idea is to exploit the sparse nature of the initial condition. In [19], an heuristic algorithm is introduced to identify the initial sources of the diffusion-advection-reaction equation. The algorithm is mainly based on a smart area search through two different procedures: refinement of the mesh in the areas where sources have been located, while discarding the areas where sources are likely not to exist. A different approach is taken in [18] where the sparsity of the initial sources is exploited by performing a ℓ_1 minimization through the classical Bregman iteration. This technique was taken from image restoration where noisy and blurry images are cleaned up [23]. Another result in this direction can be found in [5] where the recovery of the initial measures is achieved by minimizing the Tikhonov regularization with a total variation penalty.

A different approach is presented in [14] where the state constraints and the optimal control problem are decoupled and optimized separately. In order to coordinate the two components of the problem, splitting methods are used. In [15], the efficiency of the alternating direction method of multipliers (ADMM) is tested for some control problems in two dimensions. Furthermore, in [3] and alternating optimization using proximal algorithms is presented and in [20] a parallel forward-backward splitting is used.

The aim of this work is to show that sparse source identification can be also achieved by means of the optimal control and adjoint methods. We shall discuss both the strengths of the approach and also its limitations, arising mainly when the diffusivity is large or in long time horizons.

Optimal control problems with sparsity properties have been widely discussed earlier. In [6] for elliptic problems and in [7] for parabolic problems, controls with strong sparsity properties are obtained by considering optimal control problems in spaces of measures. Moreover, in [9] and in the latter version using finite elements [8], the adjoint methodology for inverse source identification of problems governed by parabolic equations was introduced. As a novelty, he optimal control viewpoint was taken and the initial condition considered to play the role of a control term. In this manner it was proved that the initial condition can be reached by minimizing a suitable functional that contains the difference between the solution of the forward problem and the given target. This methodology was later developed numerically in [24]. Later on, the efficiency of the numerical method was tested for the inverse design of the Burgers equation in [1]. Furthermore, the method was applied to the sonic-boom minimization problem [2] and to Doswell frontogenesis governed by linear hyperbolic transport equations in heterogeneous media [22].

The algorithm to be presented in this work for the sparse source identification of the linear diffusion-advection equation based on the adjoint methodology consists of two steps. Firstly, we use the adjoint methodology to identify the locations of the sources. Secondly, a least squares fitting is applied to find the corresponding intensities of the sources. In the numerical results section, we show several test cases, both in 1D and 2D, where the algorithm allows identifying the sparse initial sources very successfully, even for some heterogeneous materials or coupled models.

An outline of the paper now follows. In section 2, we describe the model problem and its discretization. The adjoint methodology for identification of the source locations is presented in section 3 together with its difficulties to find the corresponding source intensities. In section 4, the least squares fitting to get the source weights is explained and the complete algorithm is illustrated. Numerical results both in 1D and 2D are included in section 5 and conclusions can be found in the last section.

2 Model problem

We consider the numerical approximation of the inverse problem for the linear advection-diffusion equation,

$$\begin{cases} \partial_t u - d\Delta u + v \cdot \nabla u = 0, \quad \mathbf{x} \in \Omega, t \in [0,T], \\ u(\mathbf{x},t) = 0, \quad \mathbf{x} \in \partial\Omega, t \in [0,T], \\ u(\mathbf{x},0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \end{cases} \tag{1}$$

$d > 0$ being the diffusivity of the material and v the direction of the advection. We consider d and v to be constant for simplicity in the coming numerical simulations. But similar techniques can be applied for variable diffusivity and velocity fields.

Given a final time $T > 0$ and a target function u^* the aim is to identify the initial condition u_0 such that the solution, at time $t = T$, reaches the target u^*. Of course, due to the smoothing effect of parabolic equations, this cannot be done for irregular targets u^*. But whenever u^* is reachable, by backward uniqueness we know that the corresponding initial source u^* is unique (see [11], [12]).

In order to approximate a reachable target it is sufficient to consider initial c u_0 characterized as a combined set of sparse sources. This means that u_0 is a linear combination of unitary deltas with certain and possibly different weights, i.e:

$$u_0 = \sum_{i=1}^{l} \alpha_i \delta(x_i). \tag{2}$$

We formulate the inverse problem using optimal control techniques. In particular, we consider the minimization of the following functional:

$$\begin{aligned} J(u_0) := J(u(\cdot,0)) &= \frac{1}{2}\int_\Omega (u(\cdot,T) - u^*)^2 d\Omega + \tau \int_\Omega |u(\cdot,0)| d\Omega \\ &= \frac{1}{2}\int_\Omega (u(\cdot,T) - u^*)^2 d\Omega + \tau \sum_{i=1}^{l} |\alpha_i|. \end{aligned} \tag{3}$$

Note that the first term of the functional $J(u_0)$ in (3) is seeking for a initial condition u_0 such that its corresponding $u(T)$, according to the dynamics described by (1), is as close as possible to the target u^*. Additionally, the second term of the functional to be minimized (3) ensures that the initial condition u_0 that one is looking for, is a L^1 function, of minimal norm. More specifically, this second term of the functional filters out possible regular solutions that can be given by the first term of the function. It allows solutions including corner non smooth points.

2.1 Space and time discretization

We now describe a rather general space discretization of the model problem (1). Letting $\mathbf{u} : [0, T] \to \mathbb{R}^s$ where s is the number of grid points on Ω, we can write a general discretization of the diffusion–advection equation in (1) in a compact form as:

$$\mathbf{M}\dot{\mathbf{u}}(t) + d\mathbf{A}\mathbf{u}(t) + v\mathbf{V}\mathbf{u}(t) = 0. \tag{4}$$

In order to get a time discretized version of (4), we apply the implicit Euler method with stepsize $\Delta t := T/N$ where N is the total number of time steps. The numerical approximations to the solution are given by the vectors $\mathbf{u}^n \approx \mathbf{u}(t_n) \in \mathbb{R}^s$ with respect to the index $n = i\Delta t$ for $i = 1, 2, .., N$. Therefore, the fully discrete version of (1) is as follows,

$$(\mathbf{M} + d\Delta t\mathbf{A} + v\Delta t\mathbf{V})\mathbf{u}^{n+1} = \mathbf{M}\mathbf{u}^n. \tag{5}$$

Note that the general compact from in (5) will hold for any choice of the spatial discretization method (finite differences - FD, finite elements - FE or finite volumes - FD). The discretization scheme chosen will only change the specific entries of the matrices $\mathbf{M}$, $\mathbf{A}$ and $\mathbf{V}$. Algorithm 1 summarizes the numerical solution of the forward diffusion-advection equation.

Algorithm 1 Solution of the forward diffusion-advection equation.

```
procedure ForwardSolution($\mathbf{u}$, $N$)
    for $n = 1, 2, .., N$ do
        $\mathbf{u} \leftarrow (\mathbf{M} + d\Delta t\mathbf{A} + v\Delta t\mathbf{V})\backslash\mathbf{M}\mathbf{u}$
    return $u$
```

3 Adjoint methodology

In this section, the classical adjoint methodology that minimizes (3) subject to the diffusion–advection equation (1) using the gradient descent method is going to be explained.

First of all, we reformulate the constrained optimization problem (3) subject to (1) into an unconstrained optimization problem using the Lagrangian formulation:

$$\begin{aligned} L(u,\psi) = \frac{1}{2}\int_\Omega (u(\cdot,T)-u^*)^2 d\Omega + \tau \int_\Omega |u(\cdot,0)| d\Omega \\ + \int_0^T \int_\Omega \psi(-\partial_t u + d\Delta u - v\nabla \cdot u) d\Omega dt, \end{aligned} \tag{6}$$

with

$$\begin{cases} u(\mathbf{x},t) = 0 & \text{on} \quad \partial\Omega \times [0,T], \\ u(\mathbf{x},0) = u_0(\mathbf{x}) & \text{on} \quad \Omega. \end{cases} \tag{7}$$

In order to obtain the iterative scheme of the gradient descent method, we need to calculate the directional derivative $\delta L(u,\psi)$:

$$\begin{aligned} \delta L(u,\psi) = \int_\Omega (u(\cdot,T)-u^*)\delta u(\cdot,T) d\Omega + \tau \int_\Omega \operatorname{sign}(u(\cdot,0)) u(\cdot,0) \delta u(\cdot,0) d\Omega \\ + \int_0^T \int_\Omega \psi(-\partial_t \delta u + d\Delta \delta u - v\nabla \cdot \delta u) d\Omega dt, \end{aligned} \tag{8}$$

with

$$\begin{cases} \delta u = 0 & \text{on} \quad \partial\Omega \times [0,T] \\ u(\cdot,0) = \delta u_0 & \text{on} \quad \Omega \end{cases} \tag{9}$$

We now integrate by parts the last term in (8), but first of all, let's split it:

$$\begin{aligned} &\int_0^T \int_\Omega \psi(-\partial_t \delta u + d\Delta \delta u - v\nabla \cdot \delta u) d\Omega dt \\ &= -\int_0^T \int_\Omega \psi \partial_t \delta u d\Omega dt + d\int_0^T \int_\Omega \psi \Delta \delta u d\Omega dt - v\int_0^T \int_\Omega \psi \nabla \cdot \delta u d\Omega dt \end{aligned} \tag{10}$$

In the first place, we now integrate by parts in time the first term in (10):

$$\begin{aligned} -\int_0^T \int_\Omega \psi \partial_t \delta u d\Omega dt = -\int_\Omega \psi(\cdot,T)\delta u(\cdot,T) d\Omega + \int_\Omega \psi(\cdot,0)\delta u(\cdot,0) d\Omega \\ + \int_0^T \int_\Omega \partial_t \psi \delta u d\Omega dt. \end{aligned} \tag{11}$$

In the second place, we now integrate by parts in space twice the second term in (10):

$$\begin{aligned} &\int_0^T \int_\Omega \psi \Delta \delta u d\Omega dt = \int_0^T \int_\Omega \psi(\nabla \cdot \nabla \delta u) d\Omega dt \\ &= \int_0^T \int_{\partial\Omega} \psi \frac{\partial \delta u}{\partial n} d\Omega dt - \int_0^T \int_\Omega \nabla \delta u \cdot \nabla \psi d\Omega dt \\ &= \int_0^T \int_{\partial\Omega} \psi \frac{\partial \delta u}{\partial n} d\Omega dt - \int_0^T \int_{\partial\Omega} \delta u \frac{\partial \psi}{\partial n} d\Omega dt + \int_0^T \int_\Omega \delta u \Delta \psi d\Omega dt \\ &= \int_0^T \int_\Omega \delta u \Delta \psi d\Omega dt. \end{aligned} \tag{12}$$

where the last equality holds because $\delta u = 0$ as stated in (9).

Finally, we now integrate by parts in space once the third term in (10):

$$-v\int_0^T\int_\Omega \psi\nabla\cdot\delta u d\Omega dt = -v\int_0^T\int_\Omega \psi\delta u d\Omega dt + v\int_0^T\int_\Omega \delta u\nabla\psi d\Omega dt = \int_\Omega \delta u\nabla\psi d\Omega dt, \tag{13}$$

where the last equality holds because $\delta u = 0$ as stated in (9).

Hence, inserting (11), (12) and (13) into (8) we get

$$\begin{aligned}\delta L(u,\psi) &= \int_\Omega (\psi(\cdot,0) + \tau\text{sign}(u(\cdot,0))u(\cdot,0))\delta u(\cdot,0)d\Omega \\ &+ \int_0^T\int_\Omega (\partial_t\psi + d\Delta\psi + v\nabla\psi)\delta u d\Omega dt + \int_\Omega (u(\cdot,T) - u^* - \psi(\cdot,T))\delta u(\cdot,T)d\Omega.\end{aligned} \tag{14}$$

Finally, (14) can be written in the constrained optimization form with respect to the adjoint variable ψ:

$$\delta L(u,\psi) = \int_\Omega (\psi(\cdot,0) + \tau\text{sign}(u(\cdot,0)))\delta u(\cdot,0)d\Omega, \tag{15}$$

subject to

$$\begin{cases} -\partial_t\psi - d\Delta\psi + v\nabla\cdot\psi = 0 & \text{on} \quad \Omega\times[0,T] \\ \psi = 0 & \text{on} \quad \partial\Omega\times[0,T] \\ \psi(\cdot,T) = u(\cdot,T) - u^* & \text{on} \quad \Omega \end{cases} \tag{16}$$

Thus, the adjoint diffusion-advection equation (16) can be discretized using implicit Euler in the same way as the forward diffusion-advection was discretized in section 2.1. as summarized in algorithm 2.

Algorithm 2 Solution of the adjoint diffusion-advection equation.

```
procedure AdjointSolution($\mathbf{u}$, $N$)
    for $n = 1, 2, .., N$ do
        $\mathbf{u} \leftarrow (\mathbf{M} + d\Delta t\mathbf{A} - v\Delta t\mathbf{V})\backslash\mathbf{Mu}$
    return $u$
```

Now, if we take $\delta u_0 = -(\psi_0 + \tau\text{sign}(u_0)u_0)$ we get the following iterative scheme:

$$u_0^{k+1} = u_0^k - \epsilon(\psi_0^k + \tau\text{sign}(u_0^k)u_0^k) \tag{17}$$

where ϵ is the step size of the gradient descent method. The complete gradient descent procedure using adjoint methodology is presented in algorithm 3.

Figure 1 shows the numerical output of algorithm 3 for a two-dimensional example. Classical linear finite elements on an structured triangular mesh have

Algorithm 3 Gradient descent by adjoint methodology

```
procedure GradientDescent($u_0$, $\psi$, $u^*$, $TOL$, $N$)
    while $\|\text{ForwardSolution}(u_0, N) - u^*\|_2 > TOL$ do
        $u \leftarrow \text{ForwardSolution}(u_0, N)$
        $\psi \leftarrow \text{AdjointSolution}(u - u^*, N)$
        $u_0 \leftarrow u_0 - \epsilon(\psi + \tau\text{sign}(u_0))$
    return $u_0$
```

been used to produce these results. The left plots in Figure 1 correspond to the reference results (given target) and the right plots to the output of algorithm 3. One can rapidly observe that the recovered initial state is far from the given target. First of all, the gradient descent using the adjoint methodology does not recover the sparse character of the initial solution. Furthermore, the recovered maxima are far below from the actual maxima. It is not surprising that algorithm 3 does not find an sparse initial condition because it comes from solving the adjoint equation which is basically a diffusive process that smoothes out its state. Therefore, the adjoint methodology summarized in algorithm 3 finds an initial condition u_0 that is outside the sparse ansatz (2). Consequently, a second procedure is needed to project the obtained non sparse initial condition into the set of admissible sparse solutions.

In conclusion, algorithm 3 is not a suitable option when searching for linear combinations of locations and intensities. Nevertheless, one can observe that the local maxima of the resulting initial condition using algorithm 3 (see Figures 1b and 1d) fall into the exact locations where the actual initial sources are placed (see Figures 1a and 1c). Therefore, we can keep those locations and post-process them to find the corresponding intensities of the initial sources.

4 Source intensities identification

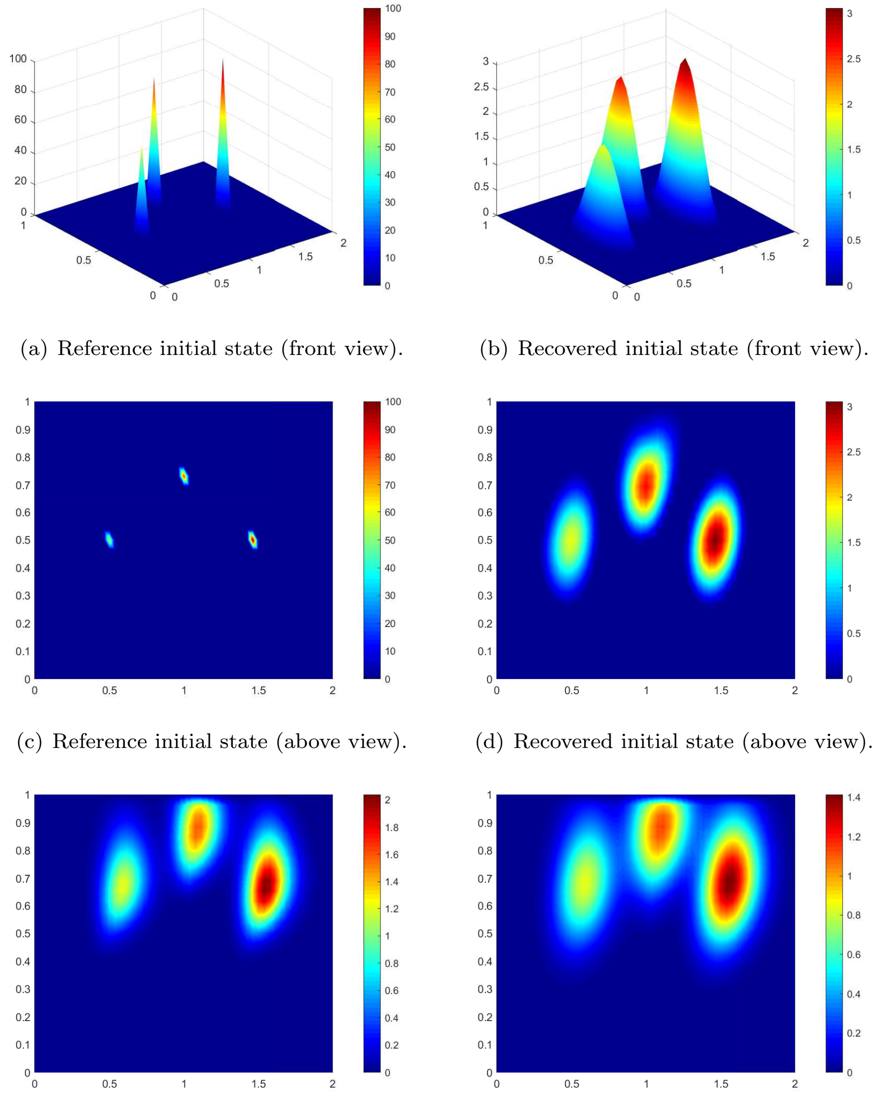
In this section we explain how to find the intensities of the initial sources once we have identified their locations using the descent gradient by the adjoint methodology procedure presented in the previous section. Once we have fixed the locations using algorithm 3, we can solve a least squares problem to get the corresponding intensities.

In particular, we have to assemble a matrix $\mathbf{L} \in \mathbb{R}^{s \times l}$ where at each column we have the forward solution obtained using algorithm 1 for a single unitary delta placed at each of the locations identified by the adjoint algorithm 3. We then solve the following linear system of equations for the vector of unknowns $\alpha = (\alpha_1, \alpha_2, ..., \alpha_l)^T$:

$$\mathbf{L}\alpha = u^*. \tag{18}$$

As the matrix $\mathbf{L}$ is not square, we solve the corresponding normal equations:

$$\mathbf{L}^T\mathbf{L}\alpha = \mathbf{L}^T u^*, \tag{19}$$



(a) Reference initial state (front view).

(b) Recovered initial state (front view).

(c) Reference initial state (above view).

(d) Recovered initial state (above view).

(e) Given target u^*.

(f) Recovered final state.

Figure 1: Sources identification using gradient descent by adjoint methodology in algorithm 3. Left plots correspond to the reference solution (Figures 1a and 1c) and the given target (Figure 1e) and right plots to the output of algorithm 3. Observe that the obtained initial data does not recover the sparse nature of the reference solution. A second procedure is needed for that. Parameters: $d = 0.05$, $v = (1, 2)$, $T = 0.1$, $N = 10$, $\Delta x = 0.0385$, $\tau = \Delta x^4$ and $\epsilon = 0.1$.

to find the vector of intensities α.

The complete numerical method for sparse source identification of the linear diffusion-advection equation under consideration, including the least squares fitting to find the intensities is summarized in algorithm 4.

Algorithm 4 Adjoint algorithm for sparse source identification

```
procedure SparseIdentification($u^*$, $TOL$, $N$)
    $u_0 = 0$, $\psi = 0$
    while $\|Su_0 - u^*\|_2 > TOL$ do
        $u_0 \leftarrow$ GradientDescent($u_0, \psi, u^*, TOL, N$)
        $x \leftarrow$ LocalMaxima($u_0$)
        for $i = 1, 2, .., l$ do
            $\mathbf{L}(:, i) \leftarrow$ ForwardSolution($\delta(x_i), N$)
        $\alpha = (\mathbf{L}^T\mathbf{L})\backslash\mathbf{L}^T u^*$
        $u_0 \leftarrow \sum_{i=1}^{l} \alpha_i \delta(x_i)$
    return $u_0$
```

5 Numerical results

In this section we present several numerical examples of the sparse source identification algorithm 4 for the linear diffusion-advection equation introduced in the previous section. All the numerical results in this work have been produced by implementing the aforementioned method in MATLAB R2018b on equidistant structured meshes and using finite elements (FE) and the implicit Euler method for the space and time discretizations respectively. For the FE discretization, we use triangular elements distributed as sketched in Figure 2 and the following pyramidal test functions (for details regarding the FE discretization matrices check [21])

$$\phi_k(x,y) = \begin{cases} \frac{x+y}{\Delta x} - 1, & \text{if} \quad \mathbf{x} = (x,y) \in \text{Region 1}, \\ \frac{y}{\Delta x}, & \text{if} \quad \mathbf{x} \in \text{Region 2}, \\ \frac{\Delta x - x}{\Delta x}, & \text{if} \quad \mathbf{x} \in \text{Region 3}, \\ 1 - \frac{x+y}{\Delta x}, & \text{if} \quad \mathbf{x} \in \text{Region 4}, \\ \frac{\Delta x - y}{\Delta x}, & \text{if} \quad \mathbf{x} \in \text{Region 5}, \\ \frac{x}{\Delta x}, & \text{if} \quad \mathbf{x} \in \text{Region 6}, \\ 0, & \quad \text{otherwise}. \end{cases} \tag{20}$$

Figure 3 shows a one-dimensional example with three heat sources to be identified. Left plot shows a comparison between the reference and the recovered initial condition using algorithm 4. The right plot shows the heat distribution at the final time T both using the reference and the recovered initial heat sources shown in the left plot. One can observe that both the locations and intensities of the heat sources are computed very successfully.

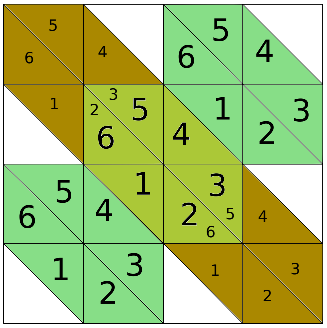
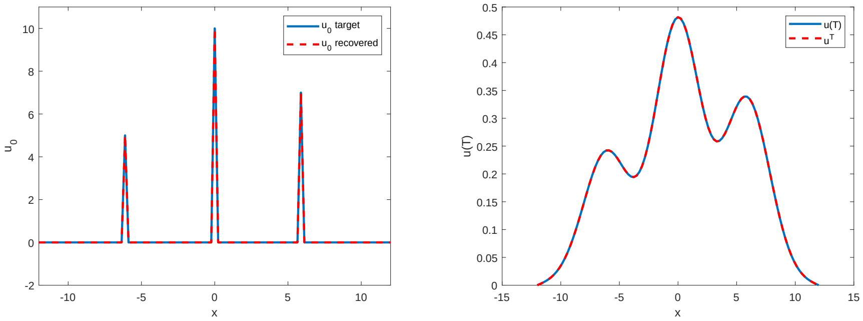


Figure 2: Sketch of the regions for the pyramidal test functions defined in (20).

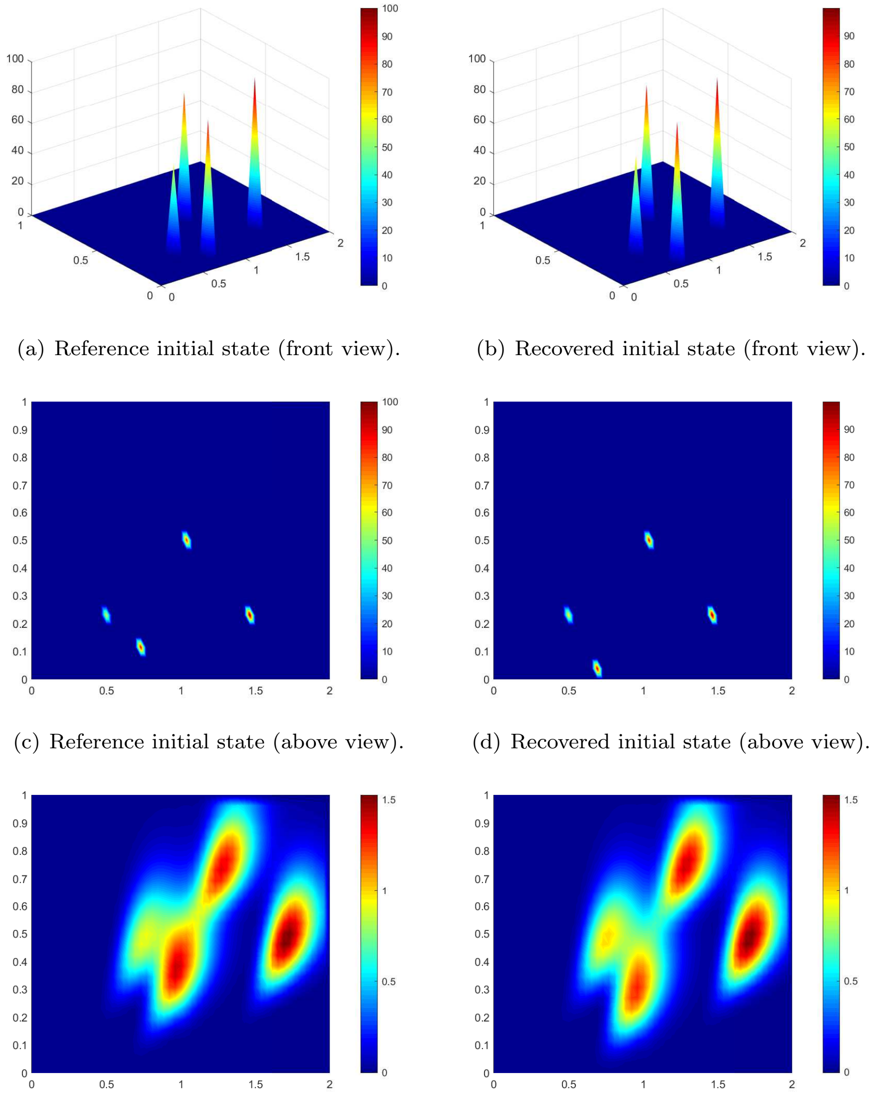


(a) Initial heat sources identification.

(b) Heat distribution at the final time T.

Figure 3: Sources identification using algorithm 4. Left plot shows a comparison between the reference and the recovered initial conditions and then right plot a comparison between the target function and the distribution generated by the recovered initial heat sources. Parameters: $d = 1$, $v = 0$, $T = 2$, $N = 30$, $\Delta x = 12/51$, $\tau = \Delta x^4$ and $\epsilon = 0.1$.

Figure 4 shows a two-dimensional example with several sources to be identified. Plots on the left side show the reference initial solution (Figures 4a and 4c) and the given target (Figure 4e). Similarly, plots on the right side show the recovered initial condition (Figures 4b and 4d) and the distribution at the final time T produced by the recovered initial sources (Figure 4f). Wind, produced by the advection term, is moving the initial sources from the bottom left corner to the upper right corner. By comparing the plots on the right side to the plots on the left side one can observe that both the locations and the intensities of the sparse sources are recovered very accurately. However, as usual in inverse problems, when either the diffusivity d is high or the time window is large the recovery is way harder and the algorithm fails. Therefore, algorithm 4 applies for reasonable small diffusivities and time intervals. This is definitely an open



(a) Reference initial state (front view).

(b) Recovered initial state (front view).

(c) Reference initial state (above view).

(d) Recovered initial state (above view).

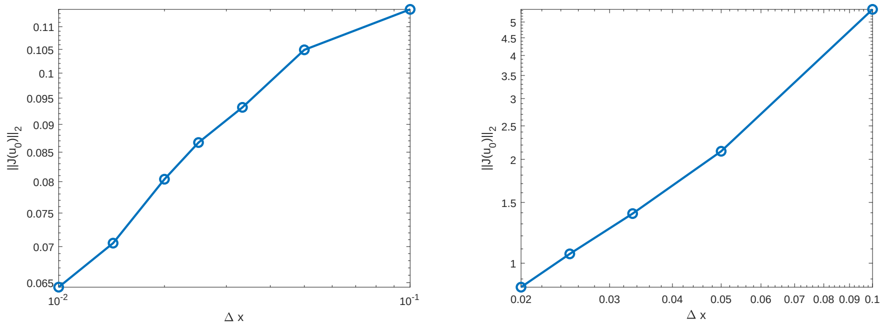
(e) Given target u^*.

(f) Recovered final state.

Figure 4: Sources identification using gradient descent by adjoint methodology in algorithm 4. Left plots correspond to the reference solution (Figures 4a and 4c) and the given target (Figure 4e) and right plots to the output of algorithm 4. One can observe how the plots on the right side behave as the reference plots on the left side. Parameters: $d = 0.05$, $v = (3, 3)$, $T = 0.1$, $N = 10$, $\Delta x = 0.0385$, $\tau = \Delta x^4$ and $\epsilon = 0.1$.

question for future research. The problem is that when the time interval is large or the diffusivity is high, the solution decays rapidly and the adjoint is not able to recover that behavior. One option to solve this problem could be to split the large time interval into smaller subintervals (possibly time adaptive) and apply the algorithm to each of them. This strategy has two main advantages: the successful application of the adjoint algorithm in each subinterval and a huge memory saving as both the solution of the forward and the adjoint problem will be saved only during the simulation of each subinterval.

Figure 5 shows the evolution of the norm of the functional $\|J(u_0)\|_2$ in (3) with respect to Δx for both the 1D and 2D cases shown above in Figures 3 and 4 respectively. One can observe how the iterative algorithm 4 gets a more accurate numerical approximation when the mesh resolution Δx decreases. However, the error evolution is quite slow which tells us that acceleration techniques like relaxation or preconditioning could be introduced in future research to achieve a faster iterative behavior. Another option would be to find an alternative to the gradient descent method like for instance faster gradient descent methods. In [17], the faster gradient descent method shown advantages with respect to the classical gradient descent method in the context of inverse problems for image deblurring and denoising tasks.

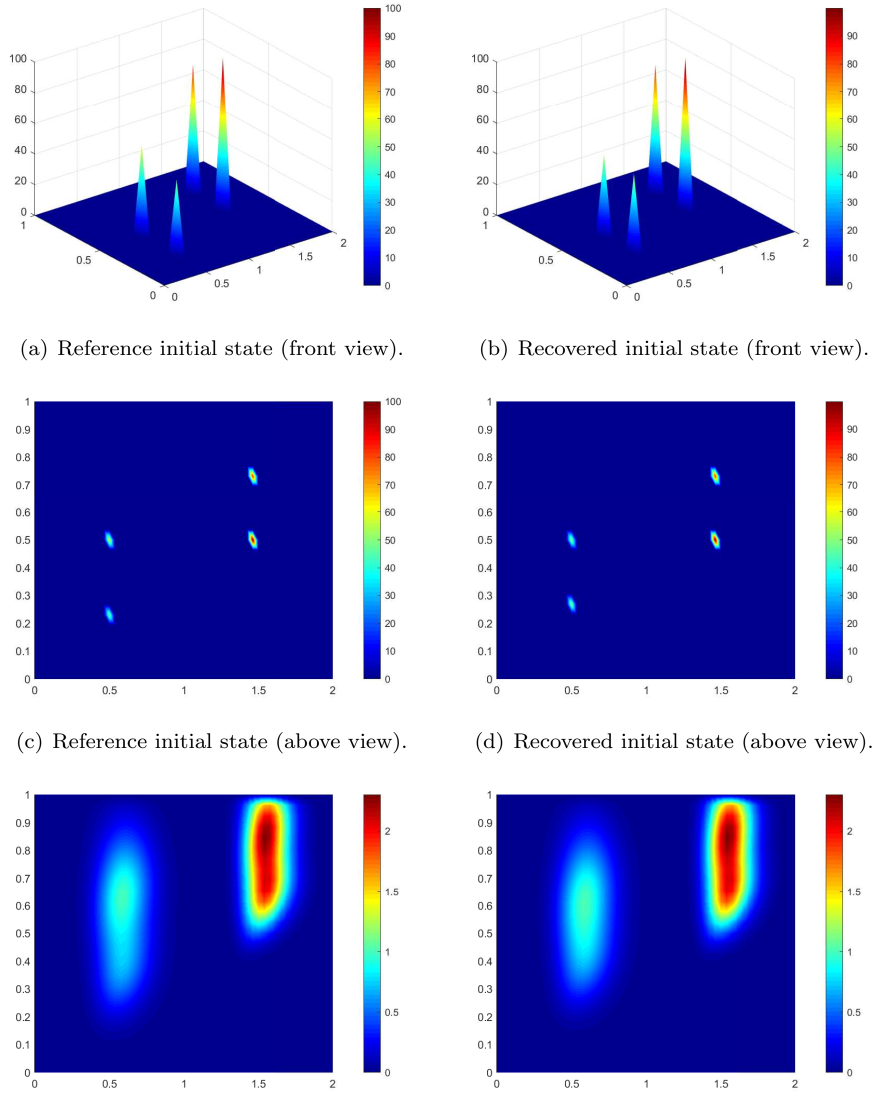


(a) 1D case.

(b) 2D case.

Figure 5: Norm of the functional $\|J(u_0)\|_2$ in (3) as a function of the mesh resolution Δx. Left plot corresponds to the one-dimensional case shown in figure 3. There $\Delta x = 1/100, 1/70, 1/50, 1/40, 1/30, 1/20, 1/10$, and the total number of iterations is 5000. Right plot corresponds to the two-dimensional case shown in figure 4. There $\Delta x = 1/50, 1/40, 1/30, 1/20, 1/10$, and the total number of iterations is 1000.

We now show in Figure 6 a two-dimensional example with several sources to be identified in an heterogeneous media. The left half ($\Omega_1 = [0,1] \times [0,1]$) and the right half ($\Omega_2 = [1,2] \times [0,1]$) subdomains are constituted by materials with different diffusivity constants. Consequently, the dynamics of the problem behaves differently in each of them. By comparing the plots on the right side to the plots on the left side one can observe that both the locations and the



(a) Reference initial state (front view).

(b) Recovered initial state (front view).

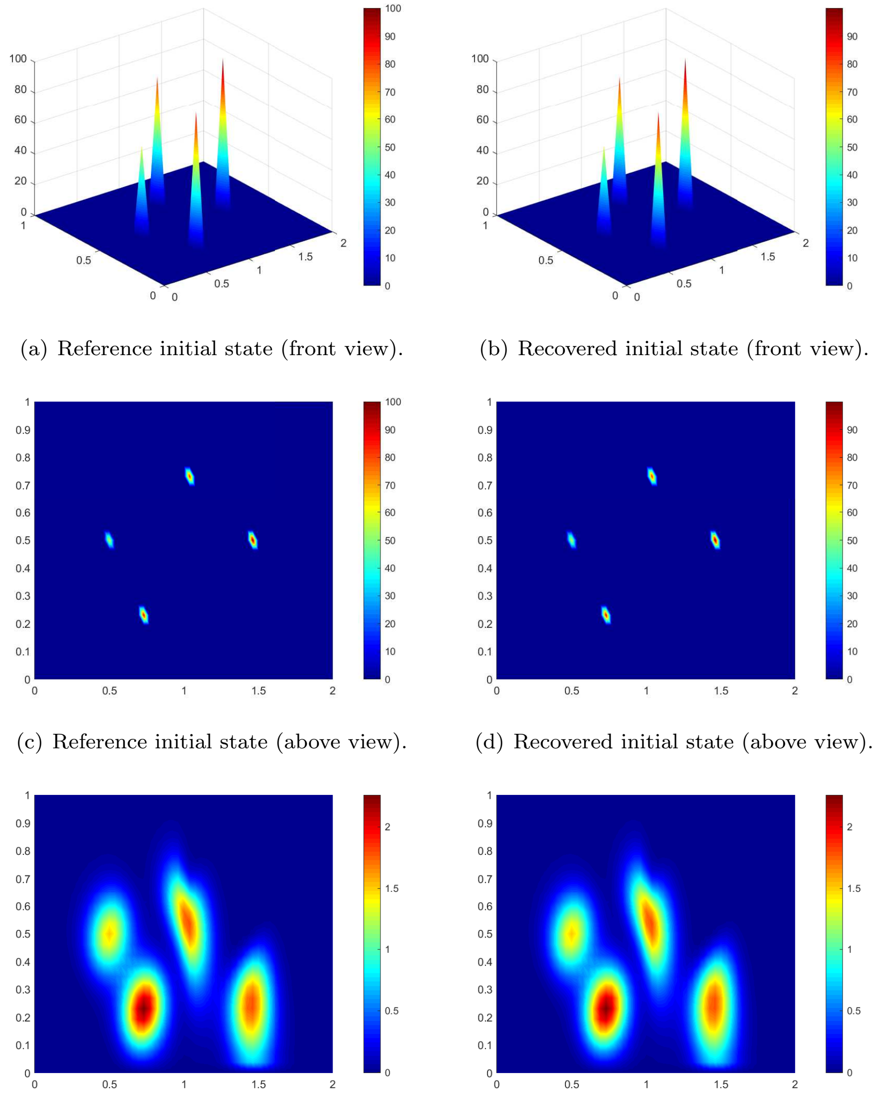
(c) Reference initial state (above view).

(d) Recovered initial state (above view).

(e) Given target u^*.

(f) Recovered final state.

Figure 6: Sources identification using gradient descent by adjoint methodology in algorithm 4 where left half and right half have different material coefficients. Left plots correspond to the reference solution (Figures 6a and 6c) and the given target (Figure 6e) and right plots to the output of algorithm 4. Parameters: $d = 0.08$ on $\Omega_1 = [0, 1] \times [0, 1]$ and $d = 0.05$ on $\Omega_2 = [1, 2] \times [0, 1]$, $v = (1, 2)$, $T = 0.1$, $N = 10$, $\Delta x = 0.0385$, $\tau = \Delta x^4$ and $\epsilon = 1$.



(a) Reference initial state (front view).

(b) Recovered initial state (front view).

(c) Reference initial state (above view).

(d) Recovered initial state (above view).

(e) Given target u^*.

(f) Recovered final state.

Figure 7: Sources identification by algorithm 4 where left half is governed by the heat equation and right half is modelled by the diffusion–advection equation. Left plots correspond to the reference solution (Figures 7a and 7c) and the given target (Figure 7e) and right plots to the output of algorithm 4. Parameters: $v = (0, 0)$ on $[0, 1] \times [0, 1]$ and $v = (0, -3)$ on $[1, 2] \times [0, 1]$, $d = 0.05$, $T = 0.1$, $N = 10$, $\Delta x = 0.0385$, $\tau = \Delta x^4$ and $\epsilon = 0.1$.

intensities of the sparse sources are recovered very accurately. However in this example the difference between the two diffusivities is not very strong. In case of strong jumps between the subdomains the algorithm (4) will have to be modified adding a splitting or a domain decomposition method to handle the interaction between the heterogeneous areas.

Finally, Figure 7 shows a two-dimensional example with several sources to be identified in a multi-model environment. This means that the left half ($\Omega_1 = [0,1] \times [0,1]$) and the right half ($\Omega_2 = [1,2] \times [0,1]$) of the domain are modelled with different equations. In particular, the heat equation is used on Ω_1 and the diffusion–advection equation is used on Ω_2. One can observe this difference between the two model equations in Figures 7e and 7f where the initial sources on Ω_2 move downwards at the same time as they dissipate while the initial sources on Ω_1 only dissipate without displacement. Even in this multi-model environment, the algorithm 4 is able to recover very accurately the locations and intensities of the initial sources.

6 Summary and conclusions

We have introduced an algorithm to recover the initial data of the linear diffusion-advection equation given a certain final target distribution. Our main interest is to identify moving pollution sources traveling in either a compressible or incompressible fluid and, as a consequence, we assumed that the initial condition is a linear combination of unitary deltas indicating the location of the sources, with their weights representing the intensity of the sources. In that context, we have presented an algorithm in two steps. Firstly, we use the adjoint methodology to identify the locations of the sources. Secondly, a least squares fitting is applied to find the corresponding intensities of the sources.

We have shown several test cases both in 1D and 2D where the algorithm identifies the initial sources very successfully even in heterogeneous media. However, a wide range of future directions are to be investigated. For instance, more research needs to be done in the case of strong jumps in the material coefficients. When the diffusivities of the coupled materials are very different from each other, the method presented fails. A possibility would be to combine it with splitting methods in order to parallelize the computations at each of the subdomains. Another future direction is would be to find out the maximum final time at which the recovery is still feasible. This is highly related to the diffusivity parameter as high diffusivities allow for a shorter final time that low diffusivities. Finally, it would be of interest to use the algorithm for more complicated geometries or nonlinear models.

Acknowledgements

This work has been partially funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 694126-DyCon), grant MTM2017-92996 of MINECO (Spain), the Marie Curie Training Network "Conflex", the ELKARTEK project KK-2018/00083 ROAD2DC of the Basque Government, ICON of the French ANR and "Nonlocal PDEs: Analysis, Control and Beyond", AFOSR Grant FA9550-18-1-0242.

References

[1] N. Allahverdi, A. Pozo, and E. Zuazua, *Numerical aspects of large-time optimal control of Burgers equation*, ESAIM: Mathematical Modelling and Numerical Analysis, 50 (2016), pp. 1371–1401.

[2] ——, *Numerical aspects of sonic-boom minimization*, Contemporary Mathematics, 658 (2016), pp. 267–279.

[3] H. Attouch, A. Cabot, P. Frankel, and J. Peypouquet, *Alternating proximal algorithms for linearly constrained variational inequalities: Application to domain decomposition for PDEs*, Nonlinear Analysis: Theory, Methods and Applications, 74(18) (2011), pp. 7455–7473.

[4] M. Bertero and M. Piana, *Inverse problems in biomedical imaging: modeling methods of solution*, Complex Systems in Biomedicine, pp. 1–33.

[5] K. Bredies and H. Pikkarainen, *Inverse problems in spaces of measures*, ESAIM: Control, Optimisation and Calculus of Variations, 19(1) (2013), pp. 190–218.

[6] E. Casas, C. Clason, and K. Kunisch, *Approximation of Elliptic Control Problems in Measure Spaces with Sparse Solutions*, SIAM J. Control Optim., 50(4) (2012), pp. 1735–1752.

[7] ——, *Parabolic Control Problems in Measure Spaces with Sparse Solutions*, SIAM J. Control Optim., 51(1) (2013), pp. 28–63.

[8] E. Casas, B. Vexler, and E. Zuazua, *Sparse initial data identification for parabolic PDE and its finite element approximations*, AIMS, 5(3) (2015), pp. 377–399.

[9] E. Casas and E. Zuazua, *Spike controls for elliptic and parabolic PDEs*, Systems and Control Letters, 62(4) (2013), pp. 311–318.

[10] H. Engl, A. Louis, and W. E. Rundell, *Inverse Problems in Medical Imaging and Nondestructive Testing*, Proceedings of the Conference in Oberwolfach, Germany, February 4-10,1996, Springer, Berlin, 1997.

[11] C. Fabre, J.-P. Puel, and E. Zuazua, *On the density of the range of the semigroup for semilinear heat equations*, in Control and optimal design of distributed parameter systems (Minneapolis, MN, 1992), vol. 70 of IMA Vol. Math. Appl., Springer, New York, 1995, pp. 73–91.

[12] E. Fernández-Cara and E. Zuazua, *The cost of approximate controllability for heat equations: the linear case*, Adv. Differential Equations, 5 (2000), pp. 465–514.

[13] R. Gilbert, L. Zhongyan, and J. Buchanan, *Direct and inverse problems in ocean acoustics*, Nonlinear Analysis: Theory, Methods, Applications, 30(3) (1997), pp. 1535–1546.

[14] R. Glowinski, S. Oshwer, and W. E. Yinl, *Splitting Methods in Communication, Imaging, Science and Engineering*, Springer, 2016.

[15] R. Glowinski, Y. Song, and X. Yuang, *An ADMM Numerical Approach to Linear Parabolic State Constrained Optimal Control Problems*, submitted, (2019).

[16] G. Gurarslan and H. Karahan, *Solving inverse problems of groundwater-pollution-source identification using a differential evolution algorithm*, Hydrogeology Journal, 23(6) (2015), pp. 1109–1119.

[17] H. Huang and U. Ascher, *Faster Gradient Descent and the Efficient Recovery of Images*, Vietnam Journal of Mathematics, 42(1) (2014), pp. 115–131.

[18] Y. Li, S. Osher, and R. Tsai, *Heat source identification based on ℓ_1 constrained minimization*, AIMS, 8(1) (2014), pp. 199–221.

[19] A. Mamonov and Y.-H. Tsai, *Point source identification in nonlinear advection-diffusion-reaction systems*, Inverse Problems, 29 (2013).

[20] C. Molinari and J. Peypouquet, *Lagrangian Penalization Scheme with Parallel Forward-Backward Splitting*, Journal of Optimization Theory and Applications, 117(2) (2018), pp. 413–447.

[21] A. Monge, *Partitioned methods for time–dependent thermal fluid–structure interaction*, PhD Thesis, Lund University, 2018.

[22] M. Morales and E. Zuazua, *Adjoint computational methods for 2D inverse design of linear transport equations on unstructured grids*, Computational and Applied Mathematics, (2019, to appear).

[23] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, *An Iterative Regularization Method for Total Variation-Based Image Restoration*, Multiscale Model. Simul., 4(2) (2006), pp. 460–489.

[24] A. Pozo, *Mathematical and numerical aspects of some scalar conservation laws: application to control*, PhD Thesis, University of the Basque Country, 2014.

[25] M. Prato and L. Zanni, *Inverse problems in machine learning: An application to brain activity interpretation*, J. Phys.: Conf. Ser., 135 (2008).

[26] R. Snieder and J. Trampert, *Inverse Problems in Geophysics*, Wavefield Inversion, pp. 119–190.

[27] M. Taroudakis and G. Makrakis, *Inverse Problems in Underwater Acoustics*, Springer, Berlin, 2001.

[28] W. Thacker, *Oceanographic inverse problems*, Physica D: Nonlinear Phenomena, 60(1-4) (1993), pp. 16–37.