

# Optimal Control of Level Sets Generated by the Normal Flow Equation

Angelo Alessandri, Patrizia Bagnerini, Roberto Cianci, Mauro Gaggero

**Abstract** The goal of this paper is the optimal control of level sets generated by the normal flow equation. The problem consists in finding the normal velocity that minimizes a given performance index expressed by means of a cost functional. In this perspective, a sufficient condition of optimality requiring the solution of a system of partial differential equations is derived. As in general it is difficult to solve such a system, an approximation scheme based on the extended Ritz method is proposed to find suboptimal solutions. The control law is forced to take on a fixed structure depending nonlinearly on a finite number of parameters to be suitably chosen. The selection of the parameters is accomplished by using a gradient-based technique. To this end, the adjoint equation is derived to compute the gradient of the cost functional with respect to the parameters of the control law. The effectiveness of the proposed approach is shown by means of simulations.

**Key words:** level set methods, normal flow, optimal control, approximation, extended Ritz method, adjoint equation

## 1 Introduction

Many engineering and scientific problems are characterized by a number of different regions interacting and depending on various factors, such as physical laws and

---

A. Alessandri, P. Bagnerini, R. Cianci  
Department of Mechanical Engineering, University of Genoa, P.le Kennedy Pad. D, I-16129  
Genoa, Italy, e-mail: alessandri@dime.unige.it, bagnerini@dime.unige.it, cianci@dime.unige.it

M. Gaggero  
Institute of Intelligent Systems for Automation, National Research Council of Italy, Via De Marini  
6, I-16149 Genoa, Italy, e-mail: mauro.gaggero@cnr.it

This work has been partially supported by the AFOSR with grant no. FA9550-15-1-0530, by  
INDAM-GNCS and by INDAM-GNFM

geometry. A lot of examples exist in cross-disciplinary contexts, such as fluid dynamics, material science, computational biology, biomedicine, land protection, and energy engineering. The literature presents a variety of methods to simulate front propagation. Among the various alternatives, level set methods are widely used to propagate fronts in two or three dimensional domains in many fields, such as image processing, computational fluid dynamics, detonation and deflagration waves, seismic analysis, optimal path planning, and shape design [4, 5, 7, 13]. In level set methods, the front is implicitly represented at each time by a certain level set (typically, the zero level) of a given function [18]. They have a number of advantages over Lagrangian approaches. In fact, they rely on typical geometric quantities that can be easily computed, such as the curvature or the normal to the front. Furthermore, changes of topology are handled (the surfaces can merge, divide, etc.), and the extension to three or more dimensions is straightforward.

The different level set methods depend on the choice of the velocity field in the dynamic equation. Two examples used in many applications are the normal flow, where the velocity field is along the normal direction to the front, and the mean curvature flow, where the velocity is proportional to the curvature at each point of the front. Based on the preliminary results of [1, 2], the aim of this work is to optimally drive a moving front using the normal flow (NF) equation, in order to overcome the computational difficulties that have prevented to attack this problem up to now.

Though there exists a vast literature concerning the control of systems described by partial differential equations (PDEs), very little has been done on the control of level sets. The lack of contribution may be due to the difficulty in treating the problem, both theoretically and numerically, and to the poor awareness of the potential application of the resulting control paradigm, which only in very recent times has emerged to some extent. Among the few available results, in [10] a prey-predator model inspired by biology is presented. The control of level sets resulting from the two-phase Stefan problem is the topic addressed in [8] and [11], where the solution is searched for by using gradient-based methods.

In this context, the present work focuses on the optimal control of the normal flow equation by regarding the speed as a control action. Unfortunately, finding a solution to such a problem is almost impossible since it would require to solve analytically a system of PDEs. Thus, we rely on suitable finite-dimensional approximations based on the extended Ritz method (ERIM) [15, 16, 21]. The ERIM can be applied to any functional optimization problem, even in high-dimensional settings [22]. It is based on the idea of constraining the control law to take on a fixed structure, where a finite number of free parameters can be suitably chosen. Then, the original problem is turned into a mathematical programming one that consists in optimizing the parameters. We exploit the ERIM to trade between complexity of the approximating structure, performance, and computational effort required to solve optimal control problems of fronts generated by level sets methods. The optimal parameters of the control policies are chosen by means of gradient-descent methods that rely on either a numerically-approximate gradient based on finite differences or the exact gradient obtained by solving the related adjoint equations [1, 2]. Simulations show that the

latter approach ensures great savings in the computational time as compared to the former one with the same approximation accuracy. A similar use of adjoint methods to compute the gradient of cost functionals for the purpose of identification is described in [19, 20].

This paper is organized as follows. In Section 2, the problem of the optimal level set control is formulated. The approximate solution method based on the ERIM is described in Section 3. The adjoint equation for the computation of the gradient of the cost functional is presented in Section 4. Simulation results are shown in Section 5. Conclusions are drawn in Section 6.

## 2 Optimal control of level set dynamics

Level set methods are based on a very intuitive idea. Let  $\Omega \subset \mathbb{R}^q$  and  $t \geq 0$  be a space domain and the time, respectively. Consider a front or boundary in two or three dimensions separating two regions (two phases). Suppose to move this front with a know speed. Level set methods consist in considering the front represented at each time  $t$  as the zero level set of a function  $\phi : \Omega \times [0, T] \rightarrow \mathbb{R}$ , where  $T > 0$  is a given time horizon. The front  $x(t, s)$  is then given at time  $t$  by the set of points such that  $\phi(x(t, s), t) = 0$ , where  $s$  is the arc-length parameter of the initial curve  $x(0, s)$  (see Figure 1). Differentiating with respect to  $t$ , we obtain the Hamilton-Jacobi (HJ) equation

$$\phi_t(x, t) + v(x, t) \cdot \nabla \phi(x, t) = 0 \quad \text{in } \Omega \times [0, T], \quad (1)$$

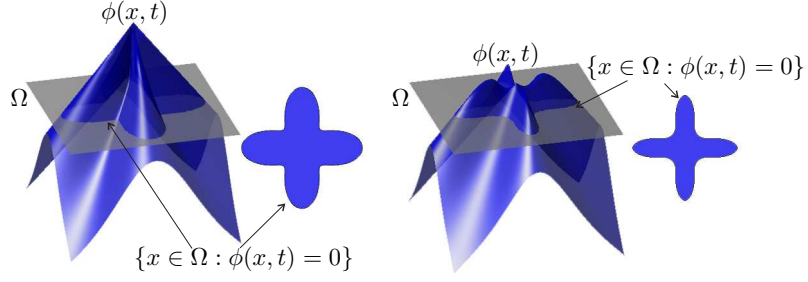
where  $v(x, t) := \frac{d}{dt}x(t, s)$ , i.e., the Lagrangian material particle velocity, gives the direction of propagation of the front at the point  $x(t, s)$  and  $\nabla$  denotes the spatial gradient. Specifically, we focus on the choice of a velocity  $v(x, t)$  that is proportional to the normal to the front, i.e.,  $v(x, t) = u \nabla \phi(x, t) / |\nabla \phi(x, t)|$ , where  $u$  represents the propagation speed. By replacing the expression of  $v$  in (1), we obtain the NF equation:

$$\phi_t(x, t) + u |\nabla \phi(x, t)| = 0 \quad \text{in } \Omega \times [0, T]. \quad (2)$$

Of course, we need to fix the initial conditions  $\phi_0 : \Omega \rightarrow \mathbb{R}$ , i.e.,  $\phi(x, 0) = \phi_0(x)$ , for all  $x \in \Omega$ . Usually,  $\phi_0$  is the signed distance to the initial front.

Based on the aforesaid, one can control the time evolution of (2) via a suitable choice of  $u$ , which plays the role of a control action. It can be chosen either as a function of time or both of time and space. In the former case we have  $u : [0, T] \rightarrow \mathbb{R}$ , whereas in the latter we deal with  $u : \Omega \times [0, T] \rightarrow \mathbb{R}$ . The level set control of (2) consists in choosing either  $u(t)$  (space-independent policy) or  $u(x, t)$  (space-dependent policy) in such a way to move the front associated with some level set of  $\phi(x, t)$  as desired.

From now on, let us denote the set-valued mapping associated with the level set  $l \in \mathbb{R}$  of the function  $\phi$  by  $\Gamma_l : [0, T] \rightrightarrows \mathcal{C}$ , where  $\Gamma_l(t) := \{x \in \Omega : \phi(x, t) = l\}$ . Moreover, we denote by  $\mathcal{U}$  the set of admissible control functions like, for instance, the set of measurable bounded functions. This set is made of the smooth functions



**Fig. 1** Examples of fronts described by zero level sets.

$t \mapsto u(t)$  or  $(x, t) \mapsto u(x, t)$  such that there exist solutions to the Cauchy problem for (2) in a space denoted by  $\mathcal{F}$ . The evolution of the level set over time may be associated with a performance index depending either on the boundary or on the interior of the shape. In particular, let us focus on the following cost:

$$J(t_0, \phi, u) := \int_{t_0}^T L(\phi, u)(t) dt + K(\phi),$$

with  $t_0 \in [0, T]$ , where  $L$  and  $K$  are functionals defined on  $\mathcal{F} \times \mathcal{U}$  and  $\mathcal{F}$ , respectively. For example, consider

$$\begin{aligned} L(\phi, u)(t) &= \int_{\Omega} g(\phi(x, t), u(x, t), t) H(\phi(x, t)) dx \\ K(\phi) &= \int_{\Omega} \bar{g}(\phi(x, T)) H(\phi(x, T)) dx, \end{aligned}$$

where  $\bar{g} : \mathbb{R} \rightarrow \mathbb{R}$  is a final penalty term. Since we search for the control action  $u \in \mathcal{U}$  that minimizes  $J(t_0, \phi, u)$ , we define

$$V(t_0, \phi) := \inf_{u \in \mathcal{U}} J(t_0, \phi, u). \quad (3)$$

Following [9][Theorem 1, p. 6], the solution of (3) can be obtained by finding a smooth-enough  $V(t, \phi)$  that solves the system of PDEs

$$\begin{cases} V_t + V_{\phi} \phi_t + L(\phi, u^{\circ}) + u^{\circ} V_{\phi} |\nabla \phi| = 0 \\ \phi_t + u^{\circ} |\nabla \phi| = 0, \end{cases} \quad (4)$$

where

$$u^{\circ} \in \arg \min_{u \in \mathcal{U}} (L(\phi, u) + u V_{\phi} |\nabla \phi|).$$

Unfortunately, it is not easy to find the analytic solution of (4). This motivates the use of numerical methods to search for an approximate solution to the level set optimal control problem (3). From now on, with a little abuse of notation, we will simply write  $J(\phi, u)$  instead of  $J(0, \phi, u)$ .

### 3 Approximate solutions with the Extended Ritz Method

We propose an approach based on the ERIM to find an approximate solution to problem (3). The basic idea consists in constraining the optimal control law  $u^\circ(x, t)$  to take on a fixed structure, denoted by  $\gamma$ , where a finite number of parameters to be suitably tuned are inserted. This approach was originally proposed for the optimal control of lumped parameter systems [15, 16]. Recently, it has been used also for the optimal control of systems described by PDEs [3]. In this context, it may be regarded as a compromise between the classical paradigms “discretize-then-optimize” and “optimize-then-discretize” [17]. Thus, it appears well-suited also to treating the optimal control of level sets. The considered fixed-structure functions  $\gamma$  are the following:

$$\gamma(\cdot, w) = \sum_{i=1}^n c_i \psi(\cdot, \kappa_i) + b, \quad c_i \in \mathbb{R}, \quad b \in \mathbb{R}, \quad \kappa_i \in \mathbb{R}^l, \quad (5)$$

where  $\psi(\cdot, \kappa_i)$  is a parametrized basis function,  $n$  is the overall number of basis functions, and all the parameters are collected in the vector  $w := (c, b, \kappa)$ , where  $\kappa := (\kappa_1, \kappa_2, \dots, \kappa_l)$ , with the dimension of  $w$  equal to  $O(n) = n(1 + l) + 1$ . Most of commonly-used approximating functions belongs to the class (5), such as feed-forward neural networks, free-node splines, radial-basis-function networks with adjustable centers and widths, and trigonometric polynomials with free frequencies and phases. Approximating structures like (5) are known to be endowed with better approximation capabilities than classical linear combinations of fixed basis functions, corresponding to the Ritz method of the calculus of variations (see, e.g., [6, 12]).

Here the goal is to approximate the mapping  $(x, t) \mapsto u^\circ(x, t)$  that solves problem (3) by imposing the control law

$$u(x, t) = \gamma(x, t, w) \quad (6)$$

in the NF equation (2), where  $w \in \mathbb{R}^{O(n)}$ . Both  $\phi$  and  $u$  are functions of  $w$ , hence also the cost functional depends on  $w$ . From now on, according to the context, we will highlight the dependence of  $J$  either on  $\phi$  and  $u$  or simply on  $w$ . Thus, problem (3) reduces to find the optimal weights  $w^\circ$  that minimize the cost functional  $J$ , i.e.,

$$w^\circ \in \arg \min_{w \in \mathbb{R}^{O(n)}} J(w). \quad (7)$$

In the following, we will address the problem of finding a solution of (7). As proposed in [1, 2], we will use gradient-based methods to solve (7) subject to (2).

#### 4 Adjoint equation and optimization procedure

As discussed in the previous section, we need to compute an analytic form of the gradient of the cost functional  $J$  with respect to the parameters  $w$  to solve (7) and therefore find the optimal weights of the approximating function  $\gamma$  in (5). However, due to the nonlinear dependence of  $J$  on the parameters, this is usually a difficult task. For the sake of generality, let us consider the generic cost functional

$$J(\phi, u) = \int_0^T \int_{\Omega} h(\phi(x, t), u(x, t), t) dx dt + \int_{\Omega} \bar{h}(\phi(x, T)) dx, \quad (8)$$

where  $h : \mathbb{R} \times \mathbb{R} \times [0, T] \rightarrow \mathbb{R}$  and  $\bar{h} : \mathbb{R} \rightarrow \mathbb{R}$ , which we want to minimize subject to the NF equation (2). Clearly, the use of (6) in (2) and (8) makes the cost functional  $J$  depend on  $w$  both directly and indirectly since also the solution of (2) is affected by  $w$ . To reduce the notational burden, from now on we will drop the dependence on  $x$  and  $t$  and write explicitly the dependence on  $w$ . Moreover, let  $\tilde{\phi}(w, \tilde{w}) := \phi(w + \tilde{w}) - \phi(w) \in \mathcal{F}$ , where  $\tilde{w} \in \mathbb{R}^{O(n)}$ . Of course, if  $\tilde{w} \rightarrow 0$ , also  $\tilde{\phi}$  tends to zero.

Since the problem is of functional type, we need to deal with the Fréchet derivative of the cost functional along the direction  $(\tilde{\phi}, \tilde{w})$ . To this end, from (2) we have

$$\phi_t(w + \tilde{w}) + \gamma(w + \tilde{w}) |\nabla \phi(w + \tilde{w})| = 0 \quad (9a)$$

$$\phi_t(w) + \gamma(w) |\nabla \phi(w)| = 0. \quad (9b)$$

After replacing  $\gamma(w + \tilde{w})$  with a Taylor expansion of the first order centered in  $w$  and using the same approximation for the norm of the gradient of  $\phi$ , i.e.,

$$|\nabla \phi(w + \tilde{w})| = |\nabla \phi(w)| + \frac{\nabla \phi(w)}{|\nabla \phi(w)|} (\tilde{\phi}_x, \tilde{\phi}_y) + r_0,$$

where  $\tilde{w} \mapsto r_0(w, \tilde{w})$  is a remainder of order higher than one, it follows from (9) that

$$\tilde{\phi}_t + |\nabla \phi| \nabla_w \gamma \tilde{w} + \gamma F \cdot \tilde{\Phi} + r_1 = 0, \quad (10)$$

where, adopting the same notation proposed by [19], we let  $F := (F_1, F_2)$  with  $F_1 := \phi_x / |\nabla \phi|$ ,  $F_2 := \phi_y / |\nabla \phi|$ ,  $\tilde{\Phi} := (\tilde{\phi}_x, \tilde{\phi}_y)$ , and  $\tilde{w} \mapsto r_1(w, \tilde{w})$  accounts for all the remainders of order higher than one.

To compute the derivative of the cost functional in  $(\phi, w)$  along the direction  $(\tilde{\phi}, \tilde{w})$ , consider

$$\begin{aligned} J(\phi + \tilde{\phi}, w + \tilde{w}) - J(\phi, w) &= \int_0^T \int_{\Omega} h_{\phi}(\phi, \gamma) \tilde{\phi} dx dt + \int_0^T \int_{\Omega} h_u(\phi, \gamma) \nabla_w \gamma \tilde{w} dx dt \\ &\quad + \int_{\Omega} \bar{h}_{\phi}(\phi) \tilde{\phi} dx + r_2, \end{aligned} \quad (11)$$

**Algorithm 1** Pseudo-code of the algorithm for the selection of the optimal weights

---

```

1: procedure OPTIMIZATION
2: Inputs:
3:   equations (2), (12), (13), and  $N$ 
4: Outputs:
5:   vector of optimal parameters  $w^\circ$ 
6: Main loop:
7:    $bestCost \leftarrow \infty$ 
8:   for  $k$  from 1 to  $N$  do
9:     generate a random initial choice of  $w_k$ 
10:    while (stopping criterion is not satisfied) do
11:      solve the NF equation (2)
12:      solve the adjoint equation (12)
13:      compute the gradient (13)
14:      perform a step of the descent algorithm by using the gradient (13) to update  $w_k$ 
15:    end while
16:    compute the cost  $J(w_k)$  corresponding to  $w_k$ 
17:    if ( $J(w_k) < bestCost$ ) then
18:       $bestCost \leftarrow J(w_k)$ 
19:       $w^\circ \leftarrow w_k$ 
20:    end if
21:  end for
22: end procedure

```

---

where we have highlighted the dependence of  $u$  on  $w$ , and  $\tilde{w} \mapsto r_2(w, \tilde{w})$  is a remainder of order higher than one. The goal is to find the first-order necessary conditions of optimality by using the first variation with (10) as a constraint. Introducing the Lagrange multiplier  $(x, t) \mapsto \mu(x, t)$  and adding the product between  $\mu(x, t)$  and (10) to the right hand side of (11), after easy steps we obtain the adjoint equation

$$\begin{cases} -\mu_t = (\mu\gamma F_1)_x + (\mu\gamma F_2)_y - h_\phi(\phi, \gamma) & \text{in } \Omega \times [0, T) \\ \mu(x, T) = -\bar{h}_\phi(\phi(x, T)) & \text{in } \Omega. \end{cases} \quad (12)$$

Thus, the gradient of the cost functional turns out to be

$$\nabla_w J(\phi, u) = \int_0^T \int_\Omega \left( h_u(\phi, \gamma) + \mu |\nabla \phi| \right) \nabla_w \gamma dx dt. \quad (13)$$

When performing the search for the optimal weights, the initial guess of  $w$  may give rise to different final results due to local minima that may affect (7). In order to mitigate this risk, one may consider multistart techniques, by randomly-chosing  $N$  initial weights, as described in Algorithm 1, where we take as stopping criterion the standard conditions for descent methods.

The approximation of the solution of the optimal control problem requires the use of efficient numerical techniques for the solution of both the forward HJ equation and the backward adjoint equation. Clearly, the schemes for the forward and backward equations cannot be the same, as the structure of the adjoint equation is often very different from its counterpart in the forward system. It is known that the

use of adjoint equations in the control of PDEs drastically decreases the computational time, but often at the cost of a considerable increase in the effort to stabilize the adjoint equation. In our case, both (2) and (12) are of hyperbolic type. Therefore, we have used high-order finite-difference schemes for hyperbolic PDEs. Moreover, imposing the correct boundary conditions is not a trivial task for the adjoint equation (12), since the source term  $h_\phi(\phi, \gamma(w))$  may have a large variability for values of  $w$  far from the global optimum, thus creating spurious and non-physical reflections at the boundary.

## 5 Simulation results

In this section, we present the numerical results in 2D and 3D obtained in two different simulation examples. In the first one, the goal is inducing level sets to vanish as fast as possible, whereas in the second one we want to track a given reference curve over time. For the sake of brevity, in the following we will refer to the two examples with the terms “vanishing” and “tracking,” respectively.

All the simulations were performed on a personal computer with a 2.6 GHz Intel Xeon CPU with 64 GB of RAM. We used the routine *fmincon* of the Matlab Optimization Toolbox to compute the optimal values of the parameters. The NF equation (2) and the adjoint equation (12) were solved numerically by using the Matlab toolbox for HJ equations developed by Mitchell [14]. In more details, we used an upwind second-order essentially non-oscillatory (ENO) scheme [13, chap. 3] in space and a second-order total variation diminishing Runge-Kutta scheme in time. It is worth noting that the convective flux terms in the adjoint equation (12) depend explicitly on  $x$ . Since the Mitchell’s toolbox does not take into account this dependency, we modified the numerical scheme by deriving the flux and adding a source term.

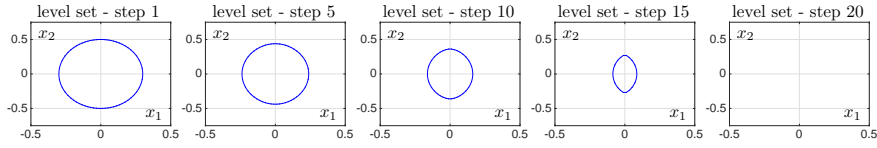
In all the examples, we focused on the nonlinear approximating structure (5) for the control law, using one-hidden-layer feedforward neural networks with sigmoidal activation functions. As it will be detailed later on, we considered different values for the number  $n$  of basis functions, i.e., we chose  $n = 5, 10, 15,$  and  $20$ .

As said, the first example deals with making level sets to vanish as fast as possible. In this case, we adopted the following cost functional to minimize:

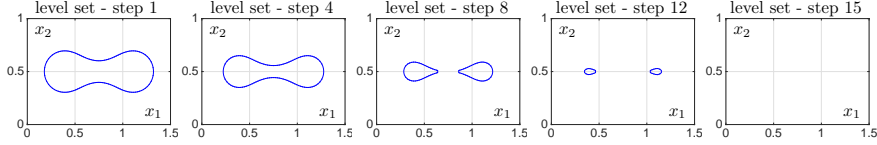
$$J = \int_0^T \int_{\Omega} \hat{H}(\phi(x,t))^2 dx dt, \quad \hat{H}(z) = \frac{1}{2} + \frac{1}{2} \tanh\left(\frac{z}{\tau}\right)$$

where  $\hat{H}$  is a smooth approximation of the Heaviside step function  $H(\cdot)$  and  $\tau$  is a positive constant [19]. In all the simulations,  $\tau$  was fixed equal to  $10^{-2}$ . We focused on two different shapes for the level sets, denoted as “shape 1” and “shape 2,” respectively. In both cases, the final time instant  $T$  was fixed to 1.5, the regular grid was composed of  $50 \times 75$  nodes, and the interval  $[0, T]$  was sampled with 50 subintervals of length  $\Delta t = 0.03$ . We chose suitable lower and upper bounds for the





**Fig. 2** Simulation results for the 2D “shape 1” vanishing example with  $n = 5$  basis functions.



**Fig. 3** Simulation results for the 2D “shape 2” vanishing example with  $n = 5$  basis functions.

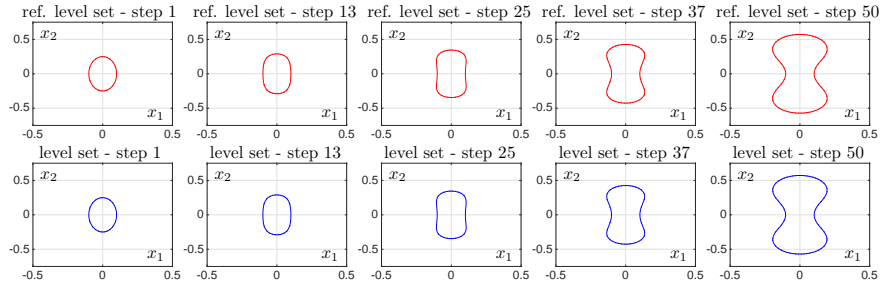
control input  $u(x, t)$ , fixed to  $-0.5$  and  $+0.5$ , respectively. Figures 2 and 3 show the results at selected time steps. In the first case, the initial ellipse vanishes at the time step 20, whereas in the second one there is a change of topology of the front, with the level set vanishing at the time step 15.

The second example concerns the tracking of a given reference curve over time. In other words, the goal is to find a control policy  $u(x, t)$  such that the zero level set of the function  $\phi(x, t)$  tracks a reference front  $\phi_{\text{ref}}(x, t)$ . Likewise in [1, 2], we considered the following cost function measuring the difference between the reference and actual level sets:

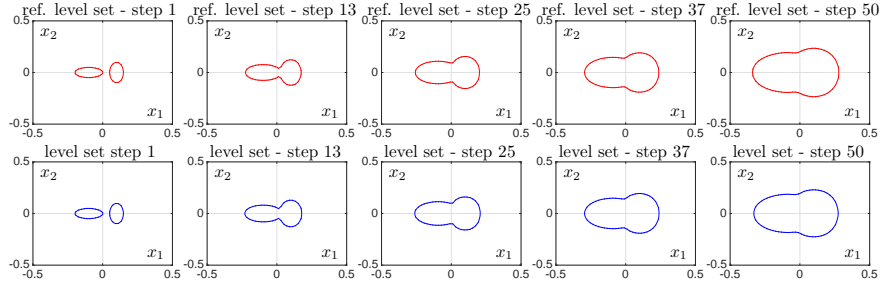
$$J = \int_0^T \int_{\Omega} (\hat{H}(\phi(x, t)) - \hat{H}(\phi_{\text{ref}}(x, t)))^2 dx dt, \quad (14)$$

where  $\hat{H}(\cdot)$  is the previously-introduced approximation of the Heaviside step function. As in the vanishing example, in the 2D case we focused on two different shapes, denoted again with a little abuse of notation as “shape 1” and “shape 2,” respectively. In both cases, the final time instant  $T$  was fixed equal to 1.5, and a sampling time  $\Delta t = 0.03$  was adopted. The mesh used for “shape 1” was a regular grid made up by  $75 \times 50$  nodes, while for the “shape 2” we used  $50 \times 50$  nodes. We constrained  $u(x, t)$  to lie in the ranges  $[0, 0.9]$  and  $[0, 0.3]$  for the “shape 1” and “shape 2,” respectively. Concerning the 3D case, we chose a final time  $T$  equal to 1.5, a regular grid of  $90 \times 60 \times 60$ , and a sampling time  $\Delta t = 0.05$ . The lower and upper bounds for the control inputs were chosen equal to 0 and 0.2, respectively. Figures 4, 5, and 6 show the results. Notice that a change of topology occurs in the 2D “shape 2” example, as two ellipses join into a unique curve.

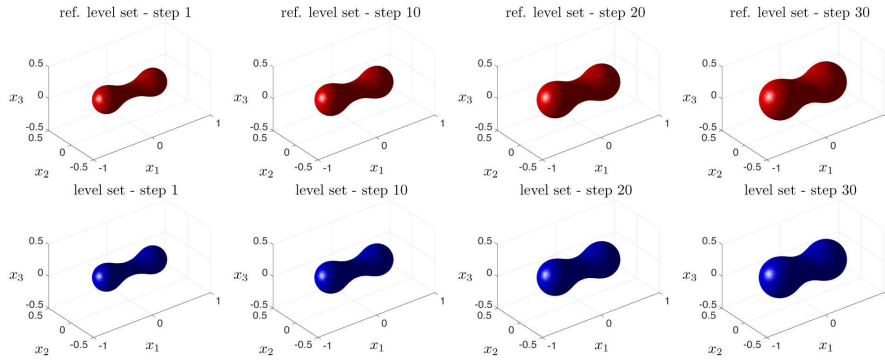
In both the vanishing and tracking examples, we applied the ERIM with different numbers of basis functions, i.e.,  $n = 5, 10, 15,$  and  $20$ , to find approximate solutions by using both the exact adjoint-based gradient of the cost functionals and the gradient obtained by means of finite-difference approximations. The accuracy of the approximation was almost the same for all the considered cases, thus showing that  $n = 5$  or  $n = 10$  are convenient choices since they ensure a sufficient accuracy with



**Fig. 4** Simulation results for the 2D “shape 1” tracking example with  $n = 5$  basis functions.



**Fig. 5** Simulation results for the 2D “shape 2” tracking example with  $n = 5$  basis functions.



**Fig. 6** Simulation results for the 3D tracking example with  $n = 5$  basis functions.

quite simple approximating structures. Notice that the accuracy of the solutions using either the adjoint-based gradient and the numerical approximate one is quite similar. However, the use of the finite-difference gradient requires much more computational time. To this end, the simulation times in all the considered examples are reported in Table 1. The time required to perform the simulations using the adjoint-based gradient is about ten times lower than the corresponding time needed by the finite-difference gradient.

**Table 1** Summary of the mean simulation times (in seconds).

	$n$	Adjoint-based gradient	Finite-difference gradient
vanishing 2D shape 1	5	$1.48 \cdot 10^2$	$1.46 \cdot 10^3$
	10	$1.98 \cdot 10^2$	$3.54 \cdot 10^3$
	15	$2.55 \cdot 10^2$	$6.29 \cdot 10^3$
	20	$3.27 \cdot 10^2$	$7.94 \cdot 10^3$
vanishing 2D shape 2	5	$1.28 \cdot 10^2$	$1.08 \cdot 10^3$
	10	$1.51 \cdot 10^2$	$2.68 \cdot 10^3$
	15	$1.89 \cdot 10^2$	$5.01 \cdot 10^3$
	20	$2.90 \cdot 10^2$	$7.29 \cdot 10^3$
tracking 2D shape 1	5	$5.65 \cdot 10^1$	$3.80 \cdot 10^3$
	10	$1.48 \cdot 10^2$	$7.38 \cdot 10^3$
	15	$1.78 \cdot 10^2$	$1.66 \cdot 10^4$
	20	$2.24 \cdot 10^2$	$2.09 \cdot 10^4$
tracking 2D shape 2	5	$7.44 \cdot 10^2$	$7.67 \cdot 10^3$
	10	$4.38 \cdot 10^2$	$5.44 \cdot 10^4$
	15	$1.26 \cdot 10^3$	$1.32 \cdot 10^4$
	20	$1.04 \cdot 10^3$	$8.73 \cdot 10^4$
tracking 3D	5	$2.39 \cdot 10^3$	$6.35 \cdot 10^4$
	10	$2.60 \cdot 10^3$	$6.37 \cdot 10^4$
	15	$2.35 \cdot 10^3$	$7.47 \cdot 10^4$
	20	$2.05 \cdot 10^3$	$7.95 \cdot 10^4$

## 6 Conclusions

Since the optimal control of fronts generated by NF level set methods cannot be solved analytically, an approach based on the ERIM has been developed to find computationally-tractable approximate solutions. The optimization of the cost functional has been performed subject to the NF dynamics as a constraint. The parameters of the approximate controller have been selected by descent techniques requiring the computation of the gradient of the cost functional with respect to the parameters of the control law, obtained by solving the related adjoint equations backwards in time. The proposed approach enables to solve problems not yet attacked because of both theoretical and practical difficulties. As a future work, we plan to improve the approach by reducing the optimization deficiency due to local minima trapping. Another direction of research will be the investigation of a cascade of dynamics governed by PDEs connected to front propagation, i.e., one or more equations describing a physical phenomenon in addition to the normal flow one.

## References

1. A. Alessandri, P. Bagnerini, and M. Gaggero. Optimal control of level sets dynamics. In *Proc. American Control Conf.*, pages 2208–2213, 2014.
2. A. Alessandri, P. Bagnerini, M. Gaggero, and A. Traverso. Further results on the optimal control of fronts generated by level set methods. In *Proc. American Control Conf.*, pages 5225–5230, 2016.
3. A. Alessandri, M. Gaggero, and R. Zoppoli. Feedback optimal control of distributed parameter systems by using finite-dimensional approximation schemes. *IEEE Trans. Neural Networks and Learning Systems*, 23(6):984–996, 2012.
4. L. Almeida, P. Bagnerini, and A. Habbal. Modeling actin cable contraction. *Comput. Math. Appl.*, 64(3):310–321, 2012.
5. L. Almeida, P. Bagnerini, A. Habbal, S. Noselli, and F. Serman. A mathematical model for dorsal closure. *J. Theor. Biol.*, 268(1):105–119, 2011.
6. A.R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Inf. Theory*, 39(3):930–945, 1993.
7. J.M. Berg and K. Holmström. On parameter estimation using level sets. *SIAM J. Control Optim.*, 37(5):1372–1393, 1999.
8. M.K. Bernauer and R. Herzog. Optimal control of the classical two-phase Stefan problem in level set formulation. *SIAM J. Sci. Comput.*, 33(1):342–363, 2011.
9. I. Ekeland and T. Turnbull. *Infinite-Dimensional Optimization and Convexity*. Chicago Lectures in Mathematics. The University of Chicago Press, 1983.
10. M. Haque, A. Rahmani, M. Egerstedt, and A. Yezzi. Efficient foraging strategies in multi-agent systems through curve evolutions. *IEEE Trans. Autom. Control*, 59(4):1036–1041, 2014.
11. M. Hinze and S. Ziegenbalg. Optimal control of the free boundary in a two-phase Stefan problem. *J. Comput. Phys.*, 223(2):657–684, 2007.
12. V. Kůrková and M. Sanguineti. Comparison of worst-case errors in linear and neural-network approximation. *IEEE Trans. Inf. Theory*, 48(1):264–275, 2002.
13. R. Kimmel. *Numerical Geometry of Images*. Springer-Verlag, New York, 2004.
14. I.M. Mitchell. The flexible, extensible and efficient toolbox of level set methods. *J. Scientific Comput.*, 35(2-3):300–329, 2008.
15. T. Parisini and R. Zoppoli. Neural networks for feedback feedforward nonlinear control systems. *IEEE Trans. Neural Netw.*, 5(3):436–449, 1994.
16. T. Parisini and R. Zoppoli. Neural approximations for multistage optimal control of nonlinear stochastic systems. *IEEE Trans. Autom. Control*, 41(6):889–895, 1996.
17. A. Quarteroni. *Numerical Models for Differential Problems*. Springer-Verlag, Milan, Italy, 2009.
18. J.A. Sethian. *Level set methods and fast marching methods*, volume 3 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, second edition, 1999.
19. I. Yang and C.J. Tomlin. Identification of surface tension in mean curvature flow. In *Proc. American Control Conf.*, pages 3290–3295, 2013.
20. I. Yang and C.J. Tomlin. Regularization-based identification for level set equations. In *Proc. IEEE Conf. on Decision and Control*, pages 1058–1064, 2013.
21. R. Zoppoli and T. Parisini. Learning techniques and neural networks for the solution of N-stage nonlinear nonquadratic optimal control problems. In A. Isidori and T. J. Tarn, editors, *Systems, Models and Feedback: Theory and Applications*, pages 193–210. Birkhäuser, Boston, 1992.
22. R. Zoppoli, M. Sanguineti, and T. Parisini. Approximating networks and extended Ritz method for the solution of functional optimization problems. *J. Optim. Theory Appl.*, 112(2):403–440, 2002.