Local minimization algorithms for dynamic programming equations

Dante Kalise, Axel Kröner, Karl Kunisch

LOCAL MINIMIZATION ALGORITHMS FOR DYNAMIC PROGRAMMING EQUATIONS

DANTE KALISE*, AXEL KRÖNER†, AND KARL KUNISCH‡

Abstract. The numerical realization of the dynamic programming principle for continuous-time optimal control leads to nonlinear Hamilton-Jacobi-Bellman equations which require the minimization of a nonlinear mapping over the set of admissible controls. This minimization is often performed by comparison over a finite number of elements of the control set. In this paper we demonstrate the importance of an accurate realization of these minimization problems and propose algorithms by which this can be achieved effectively. The considered class of equations includes nonsmooth control problems with ℓ_1-penalization which lead to sparse controls.

Key words. dynamic programming, Hamilton-Jacobi-Bellman equations, semi-Lagrangian schemes, first order primal-dual methods, semismooth Newton methods

AMS subject classifications.

1. Introduction. Since its introduction by Bellman in the 50's, dynamic programming has become a fundamental tool in the design of optimal control strategies for dynamical systems. It characterizes the value function of the corresponding optimal control problem in terms of functional relations, the so-called Bellman and Hamilton-Jacobi-Bellman (henceforth HJB) equations. We begin by briefly recalling this setting in the context of infinite horizon optimal control.
We make the following assumptions. We equip $\mathbb{R}^n$ for $n \in \mathbb{N}$ with the Euclidean norm $\|\cdot\|_2$. Furthermore, let

$$\begin{aligned} &f\colon \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}, \quad |f(x,u) - f(y,u)| \leq \omega_R \|x-y\|_2, \\ &l\colon \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}, \qquad |l(x,u) - l(y,u)| \leq \omega_R \|x-y\|_2, \end{aligned} \tag{1.1}$$

for $x, y \in \mathbb{R}^d$ with $\|x-y\|_2 \leq R$ and modulus $\omega_R\colon [0,\infty) \to [0,\infty)$ of polynomial growth satisfying $\lim_{r\to 0^+} \omega_R(r) = 0$ (cf. Ishii [14]), $d, m \in \mathbb{N}$. Let the dynamics be given by

$$\begin{cases} \dot{y}(t) = f(y(t), u(t)), \\ y(0) = x \end{cases} \tag{1.2}$$

for $t > 0$, where $x \in \mathbb{R}^d$ and $u \in \mathcal{U} \equiv \{u\colon \mathbb{R}_+ \to U \,\text{measurable}\}$, $U \subset \mathbb{R}^m$ compact. We introduce the following cost functional $J : \mathcal{U} \to \mathbb{R}$

$$J(u) = \int_0^\infty l(y(s), u(s)) e^{-\lambda s} ds\,, \quad \lambda > 0\,,$$

where y is the solution of (1.2) depending on x and u. By the application of the dynamic programming principle, the value function

$$v(x) \equiv \inf_{u\in\mathcal{U}} J(u)$$

*Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Altenberger Straße 69, 4040 Linz, Austria (dante.kalise@oeaw.ac.at).

†INRIA Saclay and CMAP, École Polytechnique, Route de Saclay, 91128 Palaiseau Cedex, France, (axel.kroener@inria.fr).

‡University of Graz, Institute of Mathematics and Scientific Computing, Heinrichstr. 36, A-8010 Graz, Austria and Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Altenberger Straße 69, 4040 Linz, Austria (karl.kunisch@uni-graz.at).

is characterized as the viscosity solution [4, Chapter 3] of the HJB equation

$$\lambda v(x) + \sup_{u \in U}\{-f(x,u)\cdot\nabla v(x) - l(x,u)\} = 0, \quad x \in \mathbb{R}^d.$$

There exists an extensive literature concerning the construction of numerical schemes for static HJB equations. The spectrum of numerical techniques includes ordered upwind methods [20, 3], high-order schemes [24], domain decomposition techniques [7] and geometric approaches [6], among many others (we refer to [13, Chapter 5, p.145] for a review of classical approximation methods). In this paper we follow a semi-Lagrangian approach [11], which is broadly used to approximate HJB equations arising in optimal control problems, see,e.g., [13]. We illustrate the basic steps in the formulation of a semi-Lagrangian scheme for our model problem.

The construction of a first-order semi-Lagrangian scheme begins by considering an Euler discretization of the system dynamics with time step $h > 0$

$$\begin{cases} y^{n+1} = y^n + hf(y^n, u^n), \\ \quad y^0 = x, \end{cases}$$

for $n \in \mathbb{N}^0$, $x \in \mathbb{R}^d$, and controls $u^n \in U$. Then, the application of the dynamic programming principle on the discrete-time dynamics leads to the Bellman equation

$$v(x) = \min_{u\in U}\{(1-\lambda h)v(x + hf(x,u)) + hl(x,u)\}, \quad x \in \mathbb{R}^d.$$

To discretize this equation in space we introduce a bounded domain $\Omega = [a,b]^d \subset \mathbb{R}^d$, $a, b \in \mathbb{R}$, where we define a regular quadrangular mesh with N nodes and mesh parameter k. We denote the set of nodes by $\Omega_k \subset \Omega$. Let the discrete value function be defined in all grid points, $V := \{v(x)\}_{x\in\Omega_k}$. However, note that $x + f(x,u)$ for $x \in \Omega_k$ is not necessarily a grid point, and therefore the value function has to be evaluated by interpolation which is chosen as a linear one here. The interpolant $I[\cdot](x)$ is defined on the basis of the dataset V. The resulting fully discrete scheme then reads

$$V(x) = \min_{u\in U}\{(1-\lambda h)I[V](x + hf(x,u)) + hl(x,u)\} =: G(V), \quad x \in \Omega_k, \tag{1.3}$$

which can be solved by a fixed point iteration starting from an initial guess V^0 by

$$V^{i+1} = G(V^i)\,.$$

An alternative to the fixed point approach is the use of Howard's algorithm or iteration in the policy (control) space [5, 1], which is faster but also utilizes the parametric minimization of the Hamiltonian.
Finally, once the value function is computed this allows to derive a feedback control for a given state $x \in \mathbb{R}^d$ by

$$u(x) \in \operatorname*{argmin}_{u\in U}\{(1-\lambda h)I[V](x + hf(x,u)) + hl(x,u)\}.$$

A characteristic feature of the class of HJB equations arising in optimal control problem is its nonlinear Hamiltonian,

$$\sup_{u\in U}\{-f(x,u)\cdot\nabla v(x) - l(x,u)\}\,,$$

which requires a parametric maximization (minimization) over the control set U. For its discrete analogue $G(V)$, a common practice in the literature is to compute the minimization by comparison, i.e., by evaluating the expression in a finite set of elements of U (see for instance [1, 12, 17] and references therein). In contrast to the comparison approach, the contribution of this paper is to demonstrate that an accurate realization of the min-operation on the right hand side of (1.3) can have an important impact on the optimal controls that are determined on the basis of the dynamic programming principle. In this respect, the reader can take a preview to Figure 4.2, where differences between optimal control fields obtained with different minimization routines can be appreciated. Previous works concerning the construction of minimization routines for this problem date back to [8], where Brent's algorithm is proposed to solve high dimensional Hamilton-Jacobi -Bellman equations and to [10], where the authors consider a fast semi-Lagrangian algorithm for front propagation problems. In this latter reference, the authors determine the minimizer of a specific Hamiltonian by means of an explicit formula. Moreover, for local optimization strategies in dynamic programming we refer to [16] for Brent's algorithm and to [22] for a Bundle Newton method.

In this article, a first-order primal-dual method (also known as Chambolle-Pock algorithm [9]) and a semismooth Newton method [15, 21] are proposed within the semi-Lagrangian scheme for the evaluation of the right hand side in (1.3). In contrast to the minimization by the comparison approach, the proposed algorithms leads to more accurate solutions for the same CPU time. Since we preserve the continuous nature of the control set, in some specific settings it is also possible to derive convergence results for our minimization strategies. Furthermore, it provides a solid framework to address challenging issues, such as nonsmooth optimal control problems with ℓ_1 control penalizations in the cost functional.

The paper is organized as follows. In Section 2 we begin by recasting the discretized Hamiltonian as a minimization problem explicitly depending on the control u. In Section 3 we introduce and adapt the Chambolle-Pock and semismooth Newton methods for the problem under consideration, and in Section 4 we present numerical examples assessing the accuracy and performance of the proposed schemes.

2. Explicitly control-dependent Hamiltonians. In this section we study the numerical evaluation of the discretized nonlinear Hamiltonian

$$\min_{u\in U}\{(1-\lambda h)I[V](x+hf(x,u))+hl(x,u)\}\,. \tag{2.1}$$

This is a non – standard minimization problem requiring the evaluation of the nonlinear mapping $u \to I[V](x+hf(x,u))$ which depends on the discrete dataset V and the system dynamics $f(x,u)$. Therefore, a first step towards the construction of local minimization strategies is to recast (2.1) by assuming specific structures for $I[V], f(x,u)$, and $l(x,u)$, leading to explicit piecewise linear or quadratic optimization problems on u. This section is split between the treatment of the interpolation $I[V](x+hf(x,u))$, and the evaluation of $l(x,u)$. For the sake of simplicity we restrict the presentation to the two dimensional case $d=2$, although the presented framework can be generalized to higher dimensions.

2.1. The interpolation operator and local subdivisions of the control space. To analyze the interpolation operator we consider for every node $x=(x_1,x_2)\in \Omega_k$ the patch of four triangles defined by the neighboring nodes, cf. Fig. 2.1. Assume that the arrival point $x+hf(x,u)$ for $x\in\Omega_k$, $u\in U$, is located in a triangle defined by

the points $x = x^1$, x^2, and x^3 in $\mathbb{R}^2$ with associated values V_1, V_2, and V_3 as indicated in Fig. 2.1. The linear interpolation formula then reads for a mesh point $x \in \Omega_k$

$$I[V](x) = cx_1 + dx_2 + e \tag{2.2}$$

with

$$\begin{pmatrix} x_1^1 & x_2^1 & 1 \\ x_1^2 & x_2^2 & 1 \\ x_1^3 & x_2^3 & 1 \end{pmatrix} \begin{pmatrix} c \\ d \\ e \end{pmatrix} = \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix}. \tag{2.3}$$

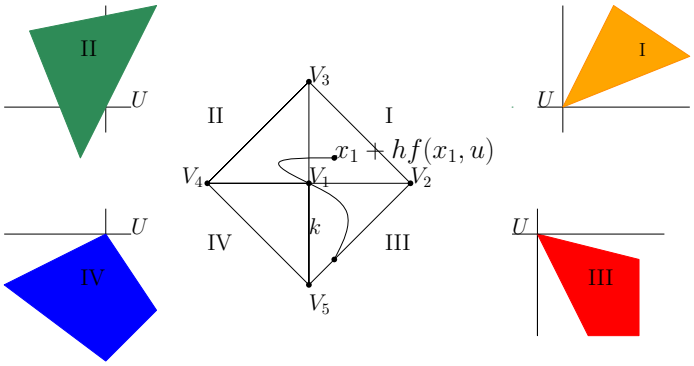

Fig. 2.1: Arrival points and related control sets.

Remark 2.1. Note that in the considered case of first order approximations the passage from the interpolation on one triangular sector to another one is continuous. For the first order interpolation schemes in a point $x \in \Omega_k$, one might also think of considering one interpolant for a macro-cell defined by a larger set of neighboring nodes. This could be computed by using 2^d suitable points. However, in practice this is similar to consider a grid of size $2h$, which leads to less accuracy, which is, in particular, in higher dimensions a crucial issue. Alternatively one could consider one interpolant for the macro-cell which is defined piecewise over all the triangulars. However, this piecewise smooth interpolation is not convenient for numerical optimization. As a further alternative one can resort to higher order interpolation schemes. □

According to (2.1), the interpolation operator $I[V](x)$ is evaluated at the arrival point $x + hf(x, u)$. If this point is sufficiently close to $x \in \Omega_k$, it is in general possible to know a priori in which of the four triangular sectors it will be located. Our goal is to establish a consistent procedure to locally divide the control space in sets that generate arrival point related to a single triangular sector. For instance, in the case of the eikonal equation there is a simple correspondence between the computed triangulars in control and state spaces, see Section 2.3. In the following we restrict ourselves to dynamics of the form

$$f(x, u) = g(x) + Bu, \tag{2.4}$$

for $B \in \mathbb{R}^{d\times m}$, i.e. nonlinear in the state, and linear in the control. We assume that $\bar{h} > 0$ is chosen such that

$$\bar{h} \sup_{x\in\Omega, u\in U} \|g(x) + Bu\|_2 \leq \frac{\sqrt{2}}{2} k. \tag{2.5}$$

Let the rows of B be denoted by $\{b_i\}_{i=1}^d$. Further, let x_D denote a grid point in Ω_k, let $\mathcal{I} \subset \{1,\dots,d\}$ be an index set with complement $\mathcal{I}^\complement$, and define a triangle $Q_\mathcal{I} \subset \mathbb{R}^d$ associated to x_D for the interpolation and $\mathcal{I}$ by

$$Q_\mathcal{I} = \{\tilde{x} = x_D + x \,|\, \|x\|_1 \le k,\ x_i \ge 0 \ \text{ for } \ i \in \mathcal{I},\ x_i \le 0 \ \text{ for } \ i \in \mathcal{I}^\complement\}.$$

Then

$$\mathcal{U}_\mathcal{I} = \{u \in U \,|\, g(x_D)_i + b_i u \ge 0 \ \text{ for } \ i \in \mathcal{I},\ g(x_D)_i + b_i u \le 0 \ \text{ for } \ i \in \mathcal{I}^\complement\}$$

has the property that $x_A = x_D + h(g(x_D) + Bu) \in Q_\mathcal{I}$ for $u \in \mathcal{U}_\mathcal{I}$, and $0 \le h \le \overline{h}$. In fact, $x = x_A - x_D = h(g(x_D) + Bu)$, and we obtain with (2.5) that $\|x\|_1 \le k$, $x_i \ge 0$ for $i \in \mathcal{I}$ and $x_i \le 0$ for $i \in \mathcal{I}^\complement$. Note also that

$$U = \bigcup_{\mathcal{I}\in\mathcal{W}} \mathcal{U}_\mathcal{I},$$

where $\mathcal{W}$ is the set of all subsets of index sets in $\{1,\dots,d\}$. The associated interpolation operator on $Q_\mathcal{I}$ is denoted by $I_\mathcal{I}$. Note that evaluation of $I_\mathcal{I}[V](x + f(x,u))$ for $x \in Q_\mathcal{I}$ and $u \in \mathcal{U}_\mathcal{I}$ leads to a linear dependence on u of the form

$$c_\mathcal{I} u_1 + d_\mathcal{I} u_2 + e_\mathcal{I}\,, \tag{2.6}$$

where the coefficients $c_\mathcal{I}$, $d_\mathcal{I}$ and $\epsilon_\mathcal{I}$ are uniquely determined by the vertices of $Q_\mathcal{I}$.

2.2. Evaluation of the cost term. Having approximated the interpolation term of the Hamiltonian, what is left is to provide an approximation of the running cost $l(x,u)$. If this term is defined in a pointwise manner, this imposes no additional difficulty. For instance, in some of the examples we shall utilize

$$l(x,u) = \|x\|_2^2 + \frac{\gamma}{2}\|u\|_2^2 = x_1^2 + x_2^2 + \frac{\gamma}{2}(u_1^2 + u_2^2) \quad x \in \Omega_k, \quad u \in U. \tag{2.7}$$

This introduces a constant and a quadratic term in u to be added to the above presented expressions for the interpolant (2.6). For l given as in (2.7) we will consider in the scheme (1.3) the explicit dependence on u. For a given node $x \in \Omega_k$ we have

$$[V]_x = \min_{\mathcal{I}\in\mathcal{W}} \{H_\mathcal{I}(x,V,u,\mathcal{I})\}\,, \tag{2.8}$$

where $[V]_x$ denotes the value of the discrete value function at node x. Further, for each set $\mathcal{I}$ we have

$$\begin{aligned} H_\mathcal{I}(x,V,u,\mathcal{I}) &= \min_{u\in U_\mathcal{I}} \{\beta I_\mathcal{I}[V](x + hf(x,u)) + hl(x,u)\}\,, \qquad (2.9)\\ &= \min_{u\in U_\mathcal{I}} \left\{\beta(c_\mathcal{I} u_1 + d_\mathcal{I} u_2 + e_\mathcal{I}) + h\left(x^2 + y^2 + \frac{\gamma}{2}(u_1^2 + u_2^2)\right)\right\}\,,\\ &= \min_{u\in U_\mathcal{I}} \{\frac{\tilde{a}_\mathcal{I}}{2} u_1^2 + \frac{\tilde{b}_\mathcal{I}}{2} u_2^2 + \tilde{c}_\mathcal{I} u_1 + \tilde{d}_\mathcal{I} u_2 + \tilde{e}_\mathcal{I}\}\,, \qquad (2.10) \end{aligned}$$

with

$$\begin{aligned} &\beta = 1 - \lambda h\,, \tilde{a}_\mathcal{I} = h\gamma\,, \tilde{b}_\mathcal{I} = h\gamma\,,\ \tilde{c}_\mathcal{I} = \beta c_\mathcal{I}\,,\\ &\tilde{d}_\mathcal{I} = \beta d_i\,, \tilde{e}_\mathcal{I} = \beta e_\mathcal{I} + h(x^2 + y^2)\,. \end{aligned}$$

REMARK 2.2. For minimum time optimal control problems, the resulting semi-Lagrangian scheme for points $x \in \Omega_k$ reads

$$[V]_x = \min_{u\in U}\{\beta I[V](x + hf(x + u)) + 1 - \beta\}\,, \text{ where } \beta = e^{-h}\,, \quad x \in \Omega_k,$$

see [13], and the Hamiltonian takes the simplified form

$$H_{\mathcal{I}}(x, V, u, \mathcal{I}) = \min_{u\in U_{\mathcal{I}}}\{\tilde{c}_{\mathcal{I}}u_1 + \tilde{d}_{\mathcal{I}}u_2 + \tilde{e}_{\mathcal{I}}\}.$$

□

To evaluate the right hand side in (2.8) numerically, we compute (2.9) for $\mathcal{I} \in \mathcal{W}$ by applying numerical optimization methods and then determine the minimizer on the macro-cell by comparison over the sets $U_{\mathcal{I}}$.

To determine the minimizer of the right hand side in (2.10) we define

$$F(u) = \frac{\tilde{a}_{\mathcal{I}}}{2}u_1^2 + \frac{\tilde{b}_{\mathcal{I}}}{2}u_2^2 + \tilde{c}_{\mathcal{I}}u_1 + \tilde{d}_{\mathcal{I}}u_2 + \tilde{e}_{\mathcal{I}}.$$

Then the minimization problem is given by

$$\min_{u\in U_{\mathcal{I}}} F(u), \quad \mathcal{I} \in W. \tag{2.11}$$

In particular, the following types of cost functionals are of interest for scalars $a, b, c, d, e, r, l, s \in \mathbb{R}$, where $a, c \geq 0$. Thereby we include also a bilinear term bu_1u_2 to allow also bilinear interpolants over quadrangular cells. For the triangular interpolation used throughout this paper, we take $b = 0$.

1. Infinite horizon problem with quadratic control cost:

$$F_2(u_1, u_2) = \frac{a}{2}u_1^2 + bu_1u_2 + \frac{c}{2}u_2^2 + du_1 + eu_2 + r. \tag{2.12}$$

   The functional is smooth and convex if $ac - b^2 > 0$.
2. Minimum time problem:

$$F_{\mathrm{MT}}(u_1, u_2) = bu_1u_2 + du_1 + eu_2 + r. \tag{2.13}$$

   The functional is non-convex if $b \neq 0$ but smooth.
3. Minimum time/Infinite horizon problem with ℓ_1-control cost:

$$F_{\mathrm{MT/IH}}(u_1, u_2) = a|u_1| + bu_1u_2 + c|u_2| + du_1 + eu_2 + r. \tag{2.14}$$

   The functional is non-convex if $b \neq 0$ and non-smooth.
4. Infinite horizon problem with ℓ_2- and ℓ_1-control cost:

$$F_{\mathrm{IH}}(u_1, u_2) = \frac{a}{2}u_1^2 + l|u_1| + bu_1u_2 + \frac{c}{2}u_2^2 + s|u_2| + du_1 + eu_2 + r. \tag{2.15}$$

   The functional is convex if $ac - b^2 > 0$ and non-smooth.

REMARK 2.3. Since U is compact all four functionals allow a minimum. If $ac - b^2 > 0$ it is unique for (2.12) and (2.13). For a discussion of optimal control problems of dynamical systems for functionals (2.14) and (2.15) we refer to [2, 18]. □

2.3. Special case: eikonal dynamics. We consider the relation between the subdivision of the state and control space in the context of minimum time problems with eikonal dynamics. To set up the control problem we introduce a closed target $\mathcal{T} \subset \mathbb{R}^n$, with $\mathrm{int}(\mathcal{T}) \neq \emptyset$, and smooth boundary. We consider the minimum time function

$$t(x,\tilde{u}) = \begin{cases} \inf\{t \in \mathbb{R}_+ : y(t,\tilde{u}) \in \mathcal{T}\} & \text{if } y(t,u(t)) \in \mathcal{T} \text{ for some } t, \\ +\infty & \text{otherwise,} \end{cases} \tag{2.16}$$

where $\tilde{u} \in \mathcal{U}$ and $y(\cdot, u(\cdot))$ denotes the solution of (1.2) depending on the control. Furthermore we assume a small-time local controllability assumption as formulated in [13, p. 216]. The minimum time problem is defined as

$$T(x) = \inf_{\tilde{u}\in\mathcal{U}} t(x,\tilde{u})$$

and the minimum time function can be characterized as the viscosity solution of

$$\begin{aligned} \sup_{u\in U}\left\{-f(x,u)^T\nabla v(x)\right\} &= 1 \quad \text{in } \mathcal{R}\setminus\mathcal{T}, \\ v(x) &= 0 \quad \text{on } \partial\mathcal{T}, \end{aligned} \tag{2.17}$$

where $\mathcal{R}$ are all points in the state space for which the time of arrival is finite.

We consider the dynamics

$$f(x,u) = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}, \quad U = \{u \in \mathbb{R}^2 : \|u\|_2 \leq 1\}, \quad x \in \mathbb{R}^2, \tag{2.18}$$

which leads to an equation of eikonal type. For this problem we have a direct relation between the direction of the control u and the identification of the arrival point in the state space; for instance, all the arrival points in the first quadrant will correspond to controls belonging to the set $U_{\{1,2\}} = \{u \in U \mid 0 \leq u_1\,, 0 \leq u_2\,\}$. It is possible to express the interpolation operator (1.3) for given node x piecewise as

$$I[V](x + hf(x,u)) = I_{\mathcal{I}}[V](x + hf(x,u)) \tag{2.19}$$

with $x + hf(x,u)$ in quadrant $U_{\mathcal{I}}$. For every quadrant we obtain control-dependent formulas of the form

$$I_{\mathcal{I}}[V](x + hf(x,u)) = c_{\mathcal{I}}u_1 + d_{\mathcal{I}}u_2 + e_{\mathcal{I}} \tag{2.20}$$

with coefficients $c_{\mathcal{I}}, d_{\mathcal{I}}, e_{\mathcal{I}} \in \mathbb{R}$.

3. Algorithms for solving minimization problems of the form (2.11)**.** We consider the following approaches:

a) Minimization by comparison over a finite subset $U_{\text{finite}} \subset U_{\mathcal{I}}$.
b) First order primal-dual method: Chambolle-Pock algorithm, see [9].
c) Second-order method: Two different types of semismooth Newton methods depending on the smoothness of the cost functional.
d) If the functional is of type (2.13), the controls are located on the sphere or in the origin and can be found by a classical Newton method with a suitable chosen initialization over the parameterized sphere.

The minimization by comparison approach a) is broadly used in the literature. It consists in choosing a finite subset $U_{\text{finite}} \subset U$ where the cost function is evaluated and the minimum is selected among the corresponding values. Such a procedure induces a different optimization paradigm, in the sense that the continuous nature of the control space is replaced by a discrete approximation. If a parameter-dependent discretization of the control set is considered, where errors with respect to the continuous set can be estimated, the discretization has to be fine enough such that the error is negligible compared to the errors introduced in the different discretization steps of the overall scheme. Furthermore, accurate discretization of the control space are far from trivial even in simple cases, as in the three-dimensional eikonal dynamics with $U = \{u \in \mathbb{R}^3 : \|u\|_2 \leq 1\}$, where a spherical coordinate-based discretization of the control introduces a concentration of points around the poles. In contrast to this approach, we propose several numerical algorithms for the treatment of problems of the form (2.11), which preserve the continuous nature of the optimization problem at a similar computational cost.

3.1. The smooth case: Cost functionals of type (2.12) **and** (2.13)**.** In this section we consider control constraints of the type $U = \{u \in \mathbb{R}^m : \|u\|_2 \leq 1\}$. To solve the optimization problem for smooth cost functionals of type (2.12) and (2.13) we present a first-order primal-dual algorithm and a semismooth Newton method.

3.1.1. Primal-dual algorithm. Here we assume that problem (2.11) can be reformulated as

$$\min_{u \in \mathbb{R}^m} F(u) + I_K(u), \tag{3.1}$$

where F is smooth, and convex and $I_K(u)$ is the indicator function of $K = U_{\mathcal{I}}$. This fits in the setting presented in [9], where a primal-dual algorithm (also known as Chambolle-Pock algorithm) is formulated, when using the same F, and by choosing the mappings denoted by K and G in the reference by id and I_K. We recall the algorithm in the following.

Algorithm 1: Chambolle-Pock algorithm

Data: Choose $n = 0$, $\tau > 0$, $\sigma > 0$, $\eta > 0$, $\vartheta \in [0, 1]$, $(u_0, y_0) \in \mathbb{R}^m \times \mathbb{R}^d$, $\bar{u}_0 = u_0$.

repeat

Compute $x_{n+1} = (y_{n+1}, u_{n+1}, \bar{u}_{n+1})$ *by*

$$\begin{cases} y_{n+1} = (I + \sigma \partial F^*)^{-1}(y_n + \sigma \bar{u}_n), \\ u_{n+1} = (I + \tau \partial G)^{-1}(u_n - \tau y_{n+1}), \\ \bar{u}_{n+1} = u_{n+1} + \vartheta(u_{n+1} - u_n). \end{cases}$$

Set $n = n + 1$.

until $\|x_n - x_{n-1}\|_{\mathbb{R}^d \times \mathbb{R}^m \times \mathbb{R}^m} < \eta$;

From identities from convex analysis, see [19], we have

$$(I + \tau \partial G)^{-1}(y) = \underset{u \in \mathbb{R}^m}{\operatorname{argmin}} \left\{ \frac{\|u - y\|_2^2}{\tau} + I_K(u) \right\} = P_K(y), \tag{3.2}$$

$$u - (I + \tau \partial F)^{-1}(u) = \tau \left(I + \frac{1}{\tau} \partial F^* \right)^{-1} \left(\frac{u}{\tau} \right), \tag{3.3}$$

where P_K denotes the projection on K. From (3.3) we have with $\sigma = \frac{1}{\tau}$

$$(I + \sigma \partial F^*)^{-1}(\sigma u) = \sigma u - \sigma \left(I + \frac{1}{\sigma} \partial F \right)^{-1} (u),$$

and hence

$$(I + \sigma \partial F^*)^{-1}(u) = u - \sigma \left(I + \frac{1}{\sigma} \partial F \right)^{-1} \left(\frac{1}{\sigma} u \right).$$

In the case $\mathcal{I} = \{1, 2\}$ the projection P_K is given by

$$P_K \colon \mathbb{R}^m \to \mathbb{R}^m, \quad P_K(p) = \frac{\max(0, p)}{\max(1, \| \max(0, p) \|_2)}. \tag{3.4}$$

This leads to Algorithm 2.

Algorithm 2: Chambolle-Pock algorithm (variant)

Data: Choose $n = 0$, $\tau > 0$, $\sigma > 0$, $\eta > 0$, $\vartheta \in [0, 1]$, $(u_0, y_0) \in \mathbb{R}^m \times \mathbb{R}^d$, $\bar{u}_0 = u_0$.

repeat

Compute $x_{n+1} = (y_{n+1}, u_{n+1}, \bar{u}_{n+1})$ *by*

$$\begin{cases} y_{n+1} = y_n + \sigma \bar{u}_n - \sigma \left(I + \frac{1}{\sigma} \partial F \right)^{-1} \left(\frac{y_n}{\sigma} + \bar{u}_n \right), \\ u_{n+1} = P_K(u_n - \tau y_{n+1}), \\ \bar{u}_{n+1} = u_{n+1} + \vartheta(u_{n+1} - u_n). \end{cases}$$

Set $n = n + 1$.

until $\|x_n - x_{n-1}\|_{\mathbb{R}^d \times \mathbb{R}^m \times \mathbb{R}^m} < \eta$;

If $(I + \frac{1}{\sigma} \partial F)^{-1}$ is linear, the first step can be reformulated as

$$y_{n+1} = \left(I + \frac{1}{\sigma} \partial F \right)^{-1} \partial F \left(\frac{y_n}{\sigma} + \bar{u}_n \right).$$

REMARK 3.1. Since the cost functional has only quadratic and linear terms it is more convenient to use a linear interpolant rather than a bilinear one in the semi-Lagrangian scheme.This does not produce any bilinear terms. If F has a bilinear term it is very challenging to compute $(I + \frac{1}{\sigma} \partial F)^{-1}$ in higher dimensions, since then ∂F depends nonlinearly on u. □

3.2. Semismooth Newton method. The Chambolle-Pock algorithm is a first order algorithm which requires convexity of the functional. We refer to [23] for an extension to a class of non-convex problems. As a second algorithm that we propose we turn to the semismooth Newton method which does not rely on global convexity. We recall some main aspects of semismooth functions, cf. [21].

DEFINITION 3.2 (Semismoothness). *Let $V \subset \mathbb{R}^\mu$ be nonempty and open, $\mu \in \mathbb{N}$. Then function $f \colon V \to \mathbb{R}^\mu$ is semismooth at $x \in V$ if it is Lipschitz continuous near x and if the following limit exists for all $s \in \mathbb{R}^\nu$, $\nu \in \mathbb{N}$:*

$$\lim_{M \in \partial f(x + \tau d)_{d \to s, \tau \to 0^+}} Md.$$

Furthermore, there holds the following chain rule.

LEMMA 3.3 (Chain rule). *Let $V \subset \mathbb{R}^\mu$ and $W \subset \mathbb{R}^\nu$ be nonempty open sets, $g\colon V \to W$ be semismooth at $x \in V$, and $h\colon W \to \mathbb{R}^\eta$, $\eta \in \mathbb{N}$, be semismooth at $g(x)$ with $g(V) \subset W$. Then the composite map $f := h \circ g\colon V \to \mathbb{R}^\eta$ is semismooth at x. Moreover,*

$$f'(x,\cdot) = h'(g(x), g'(x,\cdot)).$$

To set up a semismooth Newton algorithm for (2.11) we proceed as follows. For simplicity, we focus on the case $\mathcal{I} = \{1,2\}$. The first-order optimality condition for cost functionals of type (2.12) and (2.13) can be formulated as

$$u = P_K(u - \vartheta \nabla F(u)) \quad \forall \vartheta > 0 \tag{3.5}$$

for $u \in \mathbb{R}^m$, $F = F_2$ or $F = F_{\mathrm{MT}}$ with projection P_K as in (3.4). We introduce $p = y - \vartheta \nabla F(y)$ and choose ϑ such that $\vartheta \nabla F(u)$ is of the same scale as y and rewrite (3.5) as

$$\begin{aligned} u - P_K(p) &= 0, \\ u - \vartheta \nabla F(u) - p &= 0. \end{aligned}$$

Setting $\beta = \max(1, \|\max(0,p)\|_2)$ we define

$$E(u,p,\beta) = \begin{pmatrix} \beta u - \max(0,p) \\ u - \vartheta \nabla F(u) - p \\ \beta - \max(1, \|\max(0,p)\|_2) \end{pmatrix}.$$

Now the optimality condition can be formulated as

$$E(z) = 0$$

with $z = (u,p,\beta)$. Then we set up a semismooth Newton method as presented in Algorithm 3.

Algorithm 3: Semismooth Newton algorithm

Data: Choose $n = 0$, initialization $z_0 \in U_{\mathcal{I}} \times \mathbb{R}^m \times \mathbb{R}$, $\eta > 0$.

repeat

Solve the Newton equation for $\delta z \in U_{\mathcal{I}} \times \mathbb{R}^m \times \mathbb{R}$ given by

$$\begin{pmatrix} \beta_n I & D_{\chi_{p_n \geq 0}} & y_n \\ I - \vartheta \nabla^2 F(u_n) & -I & 0 \\ 0 & \chi_{m_n \geq 1} \frac{-p_n^T D_{\chi_{p_n \geq 0}}}{m_n} & 1 \end{pmatrix} \delta z = -E(z_n) \tag{3.6}$$

(with matrix $D_{\chi_{p_n \geq 0}} = \mathrm{diag}(\chi_{p_n \geq 0})$ and $m_n = \|\max(0,p_n)\|_2$).

Update

$$z_{n+1} = z_n + \delta z. \tag{3.7}$$

Set $n = n + 1$.

until $\|z_n - z_{n-1}\|_{\mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}} < \eta$;

We address solvability of (3.6).

LEMMA 3.4. *Set $M = I - \vartheta\nabla^2 F(u_n)$ and $q^+ = \max(0, q)$ for $q \in \mathbb{R}^m$. The Newton matrix is regular if one of the following two conditions is satisfied:*

1. *$m_n < 1$: $\beta_n \neq 0$, $\left(\frac{1}{\beta_n} M D_{\chi_{p_n \geq 0}} + I\right)$ regular.*
2. *$m_n \geq 1$: $\beta_n \neq 0$,*

$$I - \frac{1}{\beta_n} M \quad \text{regular} \tag{3.8}$$

and $\left\{ -\frac{1}{m_n}(p^+)^T (I - \frac{1}{\beta_n} M^+)^{-1} \left(-M^+ \frac{u_n^+}{\beta_n} + z_4^+ \right) \right)^+ + 1 \right\} \neq 0$, *where the components of $p = (p^+, p^-)$ are ordered with respect to active and inactive sets and M^+ denotes the submatrix corresponding to p^+.*

REMARK 3.5. For $\beta_n \approx 1$ condition (3.8) is closely related to the regularity of the Hessian of F. Moreover, if we set $K = \sup_{u \in U_{\{1,2\}}} \|\nabla^2 F(u)\|$ and assume that β_n are bounded away from 1 (i.e. $\beta_n \geq \kappa > 1$) then the choice $\vartheta < \frac{\kappa - 1}{K}$ implies (3.8). □

Proof of Lemma 3.4. The two cases are considered separately. For $m_n < 1$, the Newton-Matrix is given by

$$DE(z_n) = \begin{pmatrix} \beta_n I & D_{\chi_{p_n \geq 0}} & u_n \\ M & -I & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The regularity of the matrix follows from the regularity of $\begin{pmatrix} \beta_n I & D_{\chi_{p_n \geq 0}} \\ M & -I \end{pmatrix}$ and hence of $\left(\frac{1}{\beta_n} M D_{\chi_{p_n \geq 0}} + I\right)$.

For $m_n \geq 1$ the Newton-matrix is given by

$$DE(z_n) = \begin{pmatrix} \beta_n I & D_{\chi_{p_n \geq 0}} & u_n \\ M & -I & 0 \\ 0 & \frac{-(p^+)^T}{m_n} & 1 \end{pmatrix}$$

and leads to the Newton equations

$$\beta_n \delta u - \delta p^+ + \delta\beta u_n = z_1, \tag{3.9}$$

$$M\delta u - \delta p = z_2, \tag{3.10}$$

$$-\frac{1}{m_n}(p^+)^T \delta p^+ + \delta\beta = z_3 \tag{3.11}$$

for suitable $z_1, z_2 \in \mathbb{R}^m$, $z_3 \in \mathbb{R}$ independent of $(\delta u, \delta p, \delta\beta)$. From the first and second equations we obtain

$$-\frac{1}{\beta_n} M(\delta p^+ - \delta\beta u_n) + \delta p = z_4$$

for $z_4 \in \mathbb{R}^m$. Without loss of generality we assume that the components of $\delta p = (\delta p^+, \delta p^-)$ are ordered with respect to active and inactive sets. Then we have

$$(I - \frac{1}{\beta_n} M^+)\delta p^+ + \delta\beta M^+ \frac{u_n^+}{\beta_n} = z_4 \tag{3.12}$$

and therefore

$$\delta p^+ = \delta\beta(I - \frac{1}{\beta_n}M^+)^{-1}\left(-M^+\frac{u_n^+}{\beta_n} + z_4^+\right). \tag{3.13}$$

With (3.11) we obtain

$$\left\{-\frac{1}{m_n}(p^+)^T(I - \frac{1}{\beta_n}M^+)^{-1}\left(-M^+\frac{u_n^+}{\beta_n} + z_4^+)\right)^+ + 1\right\}\delta\beta = \bar{\beta}. \tag{3.14}$$

□

THEOREM 3.6. *Let $\bar{u}$ be a strict local minimizer of* (2.11). *Under the assumptions of Lemma 3.4 the semismooth Newton method converges locally superlinearly or terminates after a finite number of steps at $\bar{u}$.*

Proof. The operator E is semismooth and is as a composition of Lipschitz continuous functions again Lipschitz continuous. Furthermore from Lemma 3.4 we obtain the boundedness of the inverse derivative in a neighbourhood of $\bar{u}$.

Consequently, the assertion follows from [21, p. 29] and [15, p. 220, Thm. 8.3]. □

3.3. Approach for cost functionals of type (2.13)**.** To treat the problem (2.11) for cost functionals of type (2.13) we present an alternative approach. We show that all possible minimizers are located on the sphere or in the origin; cf. also the discussion in [10] where a linear functional is considered. To minimize over the sphere we parameterize the sphere by polar coordinates and consider the restriction of the functional on the sphere. There the minimizer can be found by applying a classical Newton method.

LEMMA 3.7. *For control problems* (2.11) *with cost functional* (2.13) *for $r = 0$ all minimizers are located either on the sphere or in the origin.*

Proof. For simplicity we consider the case (2.11) for $d = 2$ and $\mathcal{I} = \{1, 2\}$. Let $F(u) = du_1 + bu_1u_2 + eu_2$, $u \in \mathbb{R}^2$, $d, b, e \in \mathbb{R}$. Then we have

$$\nabla F(u) = \begin{pmatrix} d + bu_2 \\ e + bu_1 \end{pmatrix}, \quad \nabla^2 F(u) = \begin{pmatrix} 0 & b \\ b & 0 \end{pmatrix}.$$

This implies, we have for $b \neq 0$ a saddle point with eigenvalues ± 1 in

$$u_1 = -\frac{e}{b}, \quad u_2 = -\frac{d}{b}.$$

For $b = 0$ the minimum is reached in $u = (0, 0)^T$. Consequently, all minimizers are on the boundary of $\partial U_{\{1,2\}}$. However, since the restriction of the bilinear functional to the axes is linear the assertion follows immediately. □

In two dimensions the functional F restricted to the sphere is given by

$$\tilde{F}(\varphi) = a\cos(\varphi) + \frac{b}{2}\sin(2\varphi) + c\sin(\varphi), \quad 0 \leq \varphi \leq \frac{\pi}{2}.$$

3.4. The non-smooth case: Cost functionals of type (2.14) **and** (2.15)**.** In this section we consider two types of control constraints, box constraints as well as Euclidean constraints. Finally, we also present a splitting approach.

3.4.1. Semismooth Newton in case of Euclidean norm constraints. In this section we restrict the consideration to equations of eikonal type and present the approach for a two dimensional problem. Therefore the control set is given by $U = \{u \in \mathbb{R}^2 : \|u\|_2 \leq 1\}$. Let us consider the equivalent problem formulation

$$\min_{u\in\mathbb{R}^2} E(u) + h(u), \tag{3.15}$$

where $E\colon \mathbb{R}^2 \to \mathbb{R}$ is smooth and

$$h(u) = \begin{cases} \alpha(|u_1| + |u_2|), & \text{if } u \in U_{\mathcal{I}}, \\ \infty, & \text{if } u \notin U_{\mathcal{I}}. \end{cases} \tag{3.16}$$

The optimality condition for the nonsmooth problem (3.15) is given by

$$0 \in \nabla E(u) + \partial h(u)\,,$$

where ∂h denotes the subdifferential of the convex function h. Setting $q = -\nabla E(u)$ the optimality condition can be equivalently written as

$$\begin{cases} q = -\nabla E(u), \\ q \in \partial h(u). \end{cases} \tag{3.17}$$

With the convex conjugate we obtain equivalently

$$\begin{cases} q = -\nabla E(u), \\ u \in \partial h^*(q)\,, \end{cases} \tag{3.18}$$

with

$$h^*(q) = \sup_{v\in U_{\mathcal{I}}} (v \cdot q - h(v)).$$

Note that on $U_{\mathcal{I}}$

$$v \cdot q - h(v) = (q_1 - \alpha_1)v_1 + (q_2 - \alpha_2)v_2$$

with

$$\alpha_1 = \operatorname{sgn}(L_1)\alpha, \quad \alpha_2 = \operatorname{sgn}(L_2)\alpha \tag{3.19}$$

and

(3.20)

$$L_1 = \begin{cases} 1, & \text{if } \mathcal{I} = \{1,2\} \text{ or } \mathcal{I} = \{1\}, \\ -1, & \text{if } \mathcal{I} = \{2\} \text{ or } \mathcal{I} = \emptyset, \end{cases}, \quad L_2 = \begin{cases} 1, & \text{if } \mathcal{I} = \{1,2\} \text{ or } I = \{2\}, \\ -1, & \text{if } I = \{1\} \text{ or } \mathcal{I} = \emptyset. \end{cases}$$

For fixed q and $v \in U_{\mathcal{I}}$ we define

$$l(v) = (q_1 - \alpha_1)v_1 + (q_2 - \alpha_2)v_2.$$

The sup of this linear function is attained on $\partial U_{\mathcal{I}}$. We distinguish four cases

$$\begin{aligned}
&\operatorname{sgn}(q_1 - \alpha_1) \neq \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) \neq \operatorname{sgn}(L_2): \quad h^*_{L_1,L_2}(q) = 0, \\
&\operatorname{sgn}(q_1 - \alpha_1) = \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) \neq \operatorname{sgn}(L_2): \quad h^*_{L_1,L_2}(q) = q_1 - \alpha, \\
&\operatorname{sgn}(q_1 - \alpha_1) \neq \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) = \operatorname{sgn}(L_2): \quad h^*_{L_1,L_2}(q) = q_2 - \alpha, \\
&\operatorname{sgn}(q_1 - \alpha_1) = \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) = \operatorname{sgn}(L_2): \\
&\qquad\qquad h^*_{L_1,L_2}(q) = \sup_{\varphi\in\Lambda} (q_1 - \alpha_1)\cos\varphi + (q_2 - \alpha_2)\sin\varphi\,,
\end{aligned}$$

with

$$\Lambda = \begin{cases} 0 \leq \varphi \leq \frac{\pi}{2} & \text{if } \mathcal{I} = \{1,2\}, \\ \frac{\pi}{2} < \varphi \leq \pi & \text{if } \mathcal{I} = \{2\}, \\ \pi < \varphi \leq \frac{3\pi}{2} & \text{if } \mathcal{I} = \emptyset, \\ \frac{3\pi}{2} < \varphi \leq 2\pi & \text{if } \mathcal{I} = \{1\}. \end{cases} \tag{3.21}$$

In the following for simplicity we only consider the problem for $\mathcal{I} = \{1,2\}$. From the first order condition we obtain the following relation between q and φ

$$\tan\varphi = \frac{q_2 - \alpha}{q_1 - \alpha} > 0. \tag{3.22}$$

Summarizing we have

$$h^*(q) = \begin{cases} (q_1 - \alpha)\cos\varphi + (q_2 - \alpha)\sin\varphi, & \text{if } q_1, q_2 \geq \alpha, \\ q_2 - \alpha, & \text{if } q_1 < \alpha, \quad q_2 > \alpha, \\ 0, & \text{if } q_1, q_2 < \alpha, \\ q_1 - \alpha, & \text{if } q_1 > \alpha, q_2 < \alpha\,, \end{cases} \tag{3.23}$$

with φ given by (3.22) and for the generalized derivative

$$\partial h^*(q) = \begin{cases} (w_1(q), w_2(q))^T, & \text{if } q_1,\ q_2 > \alpha, \\ (0,1)^T, & \text{if } q_1 < \alpha,\ q_2 > \alpha, \\ (0,0)^T, & \text{if } q_1 < \alpha,\ q_2 < \alpha, \\ (1,0)^T, & \text{if } q_1 > \alpha,\ q_2 < \alpha \end{cases} \tag{3.24}$$

where

$$(w_1(q), w_2(q)) = \nabla((q_1 - \alpha)\cos\varphi + (q_2 - \alpha)\sin\varphi)\,.$$

To set up a semismooth Newton method we introduce a piecewise affine, continuous regularization of the gradient in a ε-tube around the discontinuity and define

$$(\partial h^*)_\varepsilon(q) = \begin{cases} (w_1(q), w_2(q))^T, & \text{if } q_1,\ q_2 > \alpha + \varepsilon, \\ (0,1)^T, & \text{if } q_1 < \alpha,\ q_2 > \alpha + \varepsilon, \\ \left(0, \dfrac{q_2 - \alpha}{\varepsilon}\right)^T, & \text{if } q_1 < \alpha,\ \alpha < q_2 < \alpha + \varepsilon, \\ (0,0)^T, & \text{if } q_1 < \alpha,\ q_2 < \alpha, \\ \left(\dfrac{q_1 - \alpha}{\varepsilon}, 0\right)^T, & \text{if } \alpha < q_1 < \alpha + \varepsilon,\ q_2 < \alpha, \\ (1,0)^T, & \text{if } q_1 > \alpha + \varepsilon,\ q_2 < \alpha, \\ \dfrac{r}{\varepsilon}(w_1(q_p), w_2(q_p))^T, & \text{if } \alpha < q_1,\ \alpha < q_2, \text{ and } \|q - (\alpha,\alpha)^T\|_2 \leq \varepsilon, \end{cases} \tag{3.25}$$

where

$$r = \|q - (\alpha,\alpha)\|_2\,, \quad \theta = \tan^{-1}\left(\frac{q_2 - \alpha}{q_1 - \alpha}\right)\,, \text{ and } \ q_p = \varepsilon(\cos(\theta), \sin(\theta)) + (\alpha,\alpha).$$

The semismooth Newton method is presented in Algorithm 4.

Algorithm 4: Semismooth Newton algorithm

Data: Choose $n = 0$, initialization $z_0 = (q_0, u_0) \in \mathbb{R}^2 \times U_{\mathcal{I}}$, $\eta > 0$.
repeat

Solve the Newton equation for $\delta z \in \mathbb{R}^2 \times U_{\mathcal{I}}$ given by

$$\begin{pmatrix} I & \nabla^2 E(u_n) \\ -D(\partial h^*)_\varepsilon(q_n) & I \end{pmatrix} \delta z = \begin{pmatrix} q_n + \nabla E(u_n) \\ -(\partial h^*)_\varepsilon(q_n) + u_n \end{pmatrix}, \tag{3.26}$$

where $D(\cdot)$ denotes the Newton derivative.
Update

$$z_{n+1} = z_n + \delta z. \tag{3.27}$$

Set $n = n + 1$.
until $\|z_n - z_{n-1}\|_{\mathbb{R}^2 \times \mathbb{R}^2} < \eta$;

We can conclude the following theorem.

THEOREM 3.8. *Let $u \in \mathbb{R}^2$ be a strict local minimum of* (3.15). *If*

$$I - \nabla^2 E(u_n) D(\partial h^*)_\varepsilon(q_n) \tag{3.28}$$

is regular for all k, the semismooth Newton converges locally superlinearly.

Proof. We introduce the piecewise affine function

$$G\colon \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^2 \times \mathbb{R}^2, \quad G(q, u) = (q_n + \nabla E(u), -(\partial h^*)_\varepsilon(q) + u_n).$$

G is semismooth and together with condition (3.28) we get directly the regularity of the Hessian in the Newton equation given in (3.26). From both conditions we derive locally superlinear convergence as in Theorem 3.6. ∎

3.4.2. Semismooth Newton in case of box constraints. Now, let the control set be given by $U = \{u \in \mathbb{R}^m : u_a \le u \le u_b\}$ with $u_a \le 0 \le u_b$, $u_a, u_b \in \mathbb{R}^2$. As in the previous section, we consider a cost functional of the type

$$\min_{u \in \mathbb{R}^2} E(u) + h(u), \tag{3.29}$$

where E is smooth and

$$h(u) = \begin{cases} \alpha(|u_1| + |u_2|), & \text{if } u \in U_{\mathcal{I}}, \\ \infty, & \text{if } u \notin U_{\mathcal{I}}. \end{cases} \tag{3.30}$$

Proceeding as in the previous section we distinguish four cases

$$\begin{aligned}
&\operatorname{sgn}(q_1 - \alpha_1) \neq \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) \neq \operatorname{sgn}(L_2): && h^*_{L_1,L_2}(q) = 0, \\
&\operatorname{sgn}(q_1 - \alpha_1) = \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) \neq \operatorname{sgn}(L_2): && h^*_{L_1,L_2}(q) = |q_1 - \alpha_1| u_a, \\
&\operatorname{sgn}(q_1 - \alpha_1) \neq \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) = \operatorname{sgn}(L_2): && h^*_{L_1,L_2}(q) = |q_2 - \alpha_2| u_b, \\
&\operatorname{sgn}(q_1 - \alpha_1) = \operatorname{sgn}(L_1) \ \wedge \ \operatorname{sgn}(q_2 - \alpha_2) = \operatorname{sgn}(L_2): && h^*_{L_1,L_2}(q) = |q_1 - \alpha_1| u_a \\
& && \qquad + |q_2 - \alpha_2| u_b \,,
\end{aligned}$$

with α_1, α_2 as in (3.19) and L_1, L_2 as in (3.20). Further, there holds

$$\nabla h^*(q) = \begin{cases} (\operatorname{sgn}(q_1)\, u_a, \operatorname{sgn}(q_2)\, u_b)^T, & \text{if } q_1 > \alpha,\ q_2 > \alpha, \\ (0, \operatorname{sgn}(q_2)\, u_b)^T, & \text{if } q_1 < \alpha,\ q_2 > \alpha, \\ (0,0)^T, & \text{if } q_1 < \alpha,\ q_2 < \alpha, \\ (\operatorname{sgn}(q_1)\, u_a, 0)^T, & \text{if } q_1 > \alpha,\ q_2 < \alpha\,, \end{cases} \tag{3.31}$$

for $\mathcal{I} = \{1, 2\}$ and accordingly for the other cases. To set up a semismooth Newton method we regularize the first component of the gradient with ramps of width ε at $|q_1| = \alpha$ and the second component with ramps at $|q_2| = \alpha$. The overall algorithm has the same form as the semismooth Newton method in (3.26).

By a similar consideration as in the previous section Theorem 3.8 holds also for the semismooth Newton method described above.

3.4.3. A splitting approach. To treat the cost functionals (2.14) and (2.15) with ℓ_1-terms we can reformulate the problem without non-differentiable terms by doubling the number of variables. This is illustrated for the two dimensional case with two dimensional control $(u, v) \in U_{\mathcal{I}}$. For a scalar $z \in \mathbb{R}$ we define $z_+ = \max(0, z)$, $z_- = \min(0, z)$ (in particular $z_+ z_- = 0$ and $|z| = z_+ - z_-$). Then problem (2.11) with cost functional (2.15) is given by

$$\begin{cases} \min\limits_{u_+, u_-, v_+, v_-} a u_+^2 + a u_-^2 + c v_+^2 + c v_-^2 + b u_+ v_+ - b u_- v_+ - b u_+ v_- + b u_- v_- \\ \quad + l(u_+ - u_-) + s(v_+ - v_-) + d(u_+ + u_-) + e(v_+ + v_-) + r, \quad \text{s.t.} \\ g_i + b_i \begin{pmatrix} u_+ - u_- \\ v_+ - v_- \end{pmatrix} \geq 0, \\ u_+ u_- = 0, \quad v_+ v_- = 0, \\ u_+ \geq 0, \quad v_+ \geq 0, \quad u_- \geq 0, \quad v_- \geq 0, \\ \|u\|_2^2 + \|v\|_2^2 \leq 1, \quad u, v \in U_{\mathcal{I}} \end{cases}$$

with g chosen as in (2.4) and b_i as the columns of B. This problem formulation can be simplified, by distinguishing four cases depending on $\mathcal{I}$; for example, for $U_{\{1,2\}}$ we obtain the equivalent problem

$$\begin{cases} \min\limits_{u_+, v_+} a u_+^2 + c v_+^2 + b u_+ v_+ + (l + d) u_+ + (s + e) v_+ + r, \quad \text{s.t.} \\ g_i + b_i \begin{pmatrix} u_+ \\ v_+ \end{pmatrix} \geq 0, \quad i = 1, 2, \\ u_+ \geq 0, \quad v_+ \geq 0, \\ u_+^2 + v_+^2 \leq 1. \end{cases} \tag{3.32}$$

Problem (3.32) can be solved by using algorithms of Section 3.1.

4. Numerical examples. In this section we present a set of numerical tests aiming at studying the performance and accuracy of the algorithms presented in the previous sections. We begin by assessing the performance of the proposed numerical optimization routines as a separate building block.

4.1. Preliminary tests. We consider a generic two-dimensional minimization problem of the form

$$\min_{u\in U} \frac{1}{2}\|u\|_2^2 + L\cdot u + \gamma\|u\|_1 \tag{4.1}$$

subject to Euclidean norm or box constraints, i.e. $U = \{u \in \mathbb{R}^2 \mid \|u\|_2 \leq 1\}$ or $U = \{u \in \mathbb{R}^2 \mid 0 \leq u_1 \leq 1,\, 0 \leq u_2 \leq 1\}$, respectively. Results presented in Tables 4.1, 4.2 and 4.3 show the different performance scenarios found under different costs and constraints. For every setting we can observe that the minimization by the comparison algorithm (see Section 3) requires very fine discretizations of the control variable (and thus, higher CPU time) to achieve similar error levels as the optimization-based counterpart.

Algorithm	Tolerance	Iterations	CPU time	$\|\cdot\|_2$ error
Chambolle-Pock	1E-4	9	7.3E-5 [s]	4.31E-5
Semismooth Newton	1E-4	5	1.3E-2 [s]	7.74E-9
Comparison ($1E4$ evaluations)	–	–	1.1E-3 [s]	1.6E-2
Comparison ($2E3$ evaluations)	–	–	6.6E-4 [s]	4.1E-2

Table 4.1: Performance tests for an ℓ_2-cost (i.e. $\gamma = 0$) with Euclidean norm constraint.

Algorithm	Tolerance	Iterations	CPU time	$\|\cdot\|_2$ error
Semismooth Newton	1E-4	101	4.53E-3 [s]	1.51E-3
Comparison ($4E4$ evaluations)	–	–	5.18E-3 [s]	5.77E-3
Comparison ($2E5$ evaluations)	–	–	2.08E-2 [s]	2.80E-3

Table 4.2: Combined ℓ_2- and ℓ_1-cost ($\gamma = 0.1$) with box constraint.

Algorithm	Tolerance	Iterations	CPU time	$\|\cdot\|_2$ error
Semismooth Newton	1E-4	58	1.47E-2 [s]	1.23E-3
Comparison ($2E3$ evaluations)	–	–	8.56E-4 [s]	4.18E-2
Comparison ($1E5$ evaluations)	–	–	1.20E-2 [s]	1.47E-2

Table 4.3: Combined ℓ_2- and ℓ_1-cost ($\gamma = 0.1$) with Euclidean norm constraint.

4.2. Interplay with the semi-Lagrangian scheme. Having embedded the minimization routines within a semi-Lagrangian scheme, we show in Fig. 4.1 the evolution of the average iteration count per gridpoint (at every fixed point iteration

of the semi-Lagrangrian scheme), for both a minimum time and an infinite horizon optimal control problem subject to eikonal dynamics. Note the difference in the evolution of the subiteration count depending on the considered control problem. In the minimum time problem nodes that have not received information propagating from the optimality front are still minimized with irrelevant information until they are reached by the optimality front, whereas in the infinite horizon case "correct" information is available for every node from the first iteration due to the presence of a running cost. In this latter case, the impact of the available information from the previous semi-Lagrangian iteration is similar to a warm start of the optimization routine.

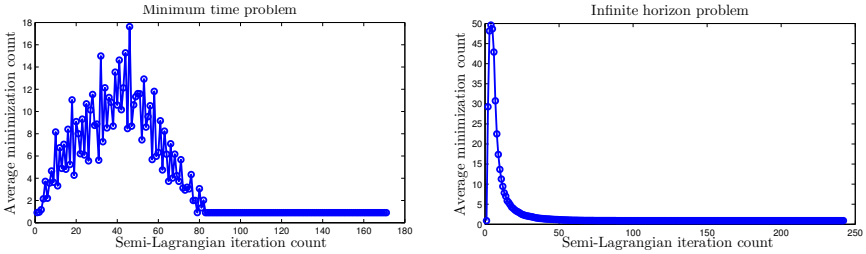

Fig. 4.1: Subiteration count for 2D control problems with eikonal dynamics

4.3. Infinite horizon problems with ℓ_2 running cost. We present three different numerical tests for infinite horizon optimal control problem with cost functional and running cost given by

$$J(u,x) = \int_0^\infty l(x(s),u(s))e^{-\lambda s}ds\,, \quad \text{and} \quad l(u,x) = \frac{1}{2}\|x\|_2^2 + \frac{\gamma_2}{2}\|u\|_2^2\,, \quad \gamma_2 > 0, \lambda > 0\,.$$

As common setting, the fixed point iteration is solved until

$$\|V^{n+1} - V^n\| \leq \frac{1}{5}k^2\,, \quad n = 1,2,3,\dots\,,$$

where k stands for the space discretization parameter and the stopping tolerance for the inner optimization routine is set to 10^{-4}. Further common parameters are $\lambda = 0.1$ and $\gamma_2 = 2$, and specific settings for every problem can be found in Table 4.4.

Test	Ω	U	h
Test 1	$[-1,1]^2$	$\|u\|_2 \leq 1$	$\frac{\sqrt{2}}{4}k$
Test 2	$[-1,1]^3$	$\|u\|_2 \leq 1$	$\frac{1}{2}k$
Test 3	$[-1,1]^3$	$\|u\|_\infty \leq 0.3$	$\frac{1}{5}k$

Table 4.4: Parameters for Tests 1-3.

Test 1: 2D eikonal dynamics. In this test we consider eikonal dynamics of the form

$$f(x,u) = (u_1, u_2)^T\,, \quad \|(u_1,u_2)\|_2 \leq 1\,. \tag{4.2}$$

We study the accuracy and performance of the semi-Lagrangian scheme with different minimization routines: discretization of the control set and minimization by comparison, a semismooth Newton method given in Algorithm 3, and the approach by Chambolle and Pock given in Algorithm 2. In order to make a fair study of the different routines, we choose the discrete set of controls for the comparison algorithm such that the CPU time for the semismooth Newton method and the Chambolle-Pock algorithm is almost the same. Table 4.5 shows errors in both the value function and in the optimal control between the exact solutions v and u and their numerical approximations V_h and U_h for the different schemes. Errors are computed with respect to the exact solution of the Hamilton-Jacobi-Bellman equation, which is not readily available in the literature. Since it is useful for numerical investigations, it is provided in Appendix B. The results show that we have similar CPU time of the approaches and independently of the meshsize, the optimization-based schemes yield more accurate approximations of the value function and the associated optimal control than the approach based on comparison. Further results are shown in Fig. 4.2, where we also consider nonhomogeneous eikonal dynamics

$$f(x,u) = (1+\chi_{x_2>0.5}(x))(u_1,u_2)^T\,, \quad \|(u_1,u_2)\|_2 \leq 1 \tag{4.3}$$

with $\chi_{x_2>0.5}(x)$ corresponds to the indicator function of the set $\{x=(x_1,x_2)\,|x_2 > 0.5\}$. The figure shows that both approaches, the SL-scheme with a semismooth inner optimization block and the one with a comparison-based routine, lead to very similar value functions. This is clearly not the case for the optimal control fields depicted in rows 2 and 3 of Fig. 4.2. Even by a post-processing step it would be difficult to obtain the results in the third row from those in the second row.

	$k=0.05$, $(40^2$ DoF)			$k=0.025$, $(80^2$ DoF)		
Algorithm	CPU time	$\|v-V_h\|_1$	$\|u-U_h\|_1$	CPU time	$\|v-V_h\|_1$	$\|u-U_h\|_1$
Comparison	63.52 [s]	3.12E-2	3.84E-2	5.76E2 [s]	1.96E-2	1.74E-2
Semismooth Newton	77.25 [s]	2.62E-2	1.61E-2	7.27E2 [s]	1.36E-2	7.21E-3
Chambolle-Pock	63.05 [s]	2.60E-2	1.42E-2	5.77E2 [s]	1.36E-2	6.83E-3

Table 4.5: Infinite horizon control of 2D eikonal dynamics. CPU time and errors for a semi-Lagrangian scheme with different minimization routines. The comparison algorithm was run with a discrete set of 1280 control points in every node.

Test 2: 3D eikonal dynamics. In order to illustrate that our approach can be extended to higher dimensions, we consider the infinite horizon optimal control of three dimensional eikonal dynamics

$$f(x,u) = (u_1,u_2,u_3)^T\,, \quad \|(u_1,u_2,u_3)\|_2 \leq 1\,.$$

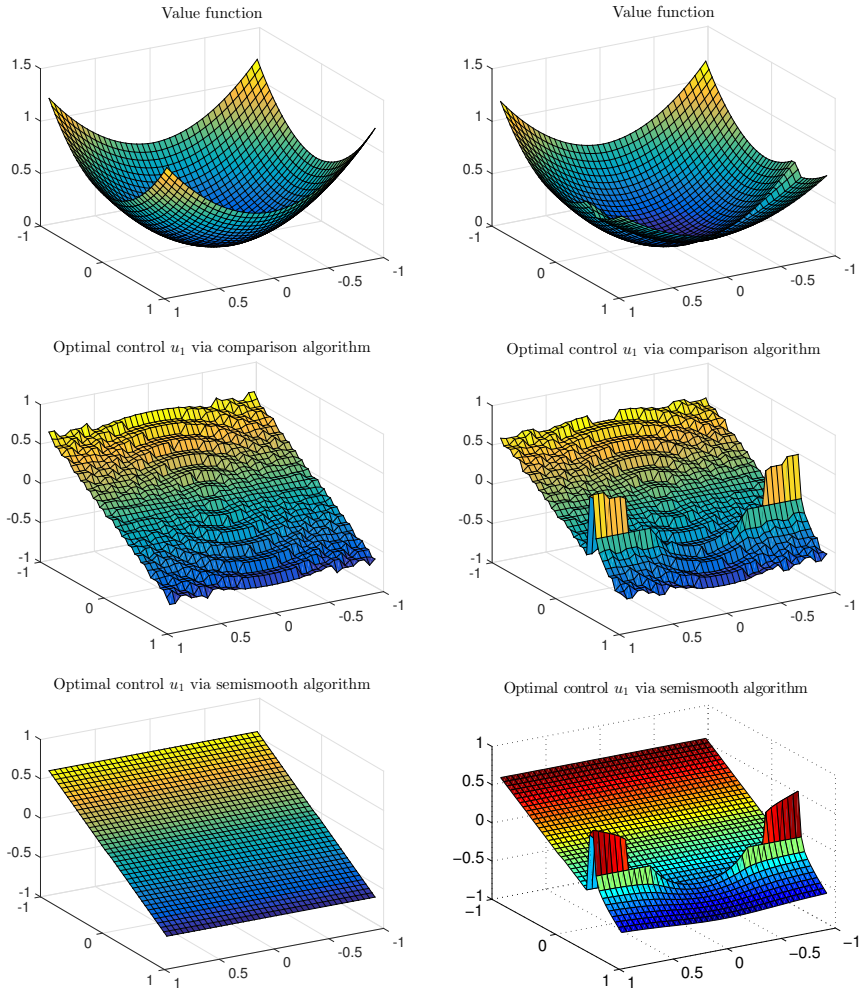

Fig. 4.2: Infinite horizon control with 2D eikonal dynamics. Left: continuous dynamics. Right: discontinuous dynamics.

Errors and CPU time are shown in Table 4.6. Since the values for the semismooth Newton and the Chambolle-Pock algorithms are similar, we only include the data for this latter one. With the Chambolle-Pock algorithm we obtain more accurate solutions of the control field for a similar amount of CPU time as the comparison approach.

Test 3: Triple integrator with two control variables. In this test the dynamics are given by a triple integrator with two control variables subject to box constraints

$$f(x,u) = (x_2, x_3 + u_1, u_2)^T, \quad |u_1| \leq a, \ |u_2| \leq b, \quad a, b \in \mathbb{R}.$$

	$k = 0.1, \quad (20^3$ DoF)			$k = 0.05, \quad (40^3$ DoF)		
Algorithm	CPU time	$\|v - V_h\|_1$	$\|u - U_h\|_1$	CPU time	$\|v - V_h\|_1$	$\|u - U_h\|_1$
Comparison	93.76 [s]	2.49E-2	2.46E-2	2.89E2 [s]	1.02E-2	1.89E-2
Chambolle-Pock	97.25 [s]	9.92E-3	2.07E-2	2.01E2 [s]	5.66E-3	1.22E-2

Table 4.6: Infinite horizon control of 3D eikonal dynamics. CPU time and errors for a semi-Lagrangian scheme with different minimization routines. The comparison algorithm was run with a discrete set of 5120 control points.

The purpose of this example is to stress that the minimization strategy that we have introduced can be also applied to non-eikonal dynamics, where the correspondence between octants in the state space and the control field is not trivial. For the sake of completeness the control space decomposition procedure for this example can be found in Appendix A. In this particular case, every octant will have associated a different rectangular sector in the control space for its arrival points. Results for the value function, optimal controls and trajectories are shown in Fig. 4.3. In the second row of this figure, we observe distinct differences between the optimal controls obtained from the Chambolle-Pock and comparison-based algorithms. These differences in the control lead to different approximated steady states as it can be seen in the left of the second row of Fig. 4.3. To highlight also the effect of closed-loop control we carried out an experiment with additive structural and output noise. In the third row of Fig. 4.3, the resulting states from open and closed-loop control can be compared.

4.4. Infinite horizon problems with combined ℓ_2 and ℓ_1-costs. We present two numerical tests for infinite horizon optimal control problems with cost function and running cost given by

$$J(u,x) = \int_0^\infty l(x(s),u(s))e^{-\lambda s}ds\,,$$
$$l(u,x) = \frac{1}{2}\|x\|_2^2 + \frac{\gamma_2}{2}\|u\|_2^2 + \gamma_1\|u\|_1\,, \quad \gamma_2 > 0,\ \gamma_1 > 0,\ \lambda > 0\,.$$

The stopping rule for the fixed point iteration is defined in the same way as in the previous set of examples. All the optimization-based routines have been solved with a semismooth Newton method as the inner block, with a regularization parameter $\varepsilon = 1e-3$. Further common parameters are $\lambda = 0.1$ and $\gamma_2 = 2$, and specific settings for every problem can be found in Table 4.7.

Test	Ω	U	h	k
Test 4	$[-1,1]^2$	$\|u\|_2 \leq 1$	$\frac{\sqrt{2}}{4}k$	0.025
Test 5	$[0,2\pi]^3$	$\|u\|_\infty \leq 0.3$	$0.2k$	0.2

Table 4.7: Parameters for Tests 4-5.

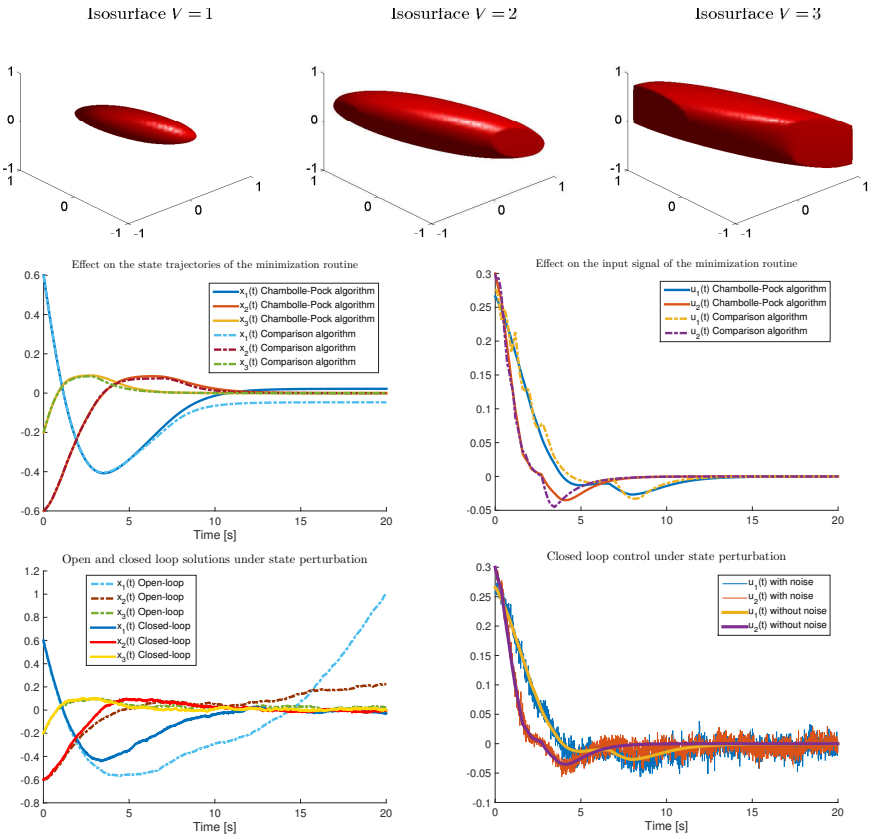

Fig. 4.3: Triple integrator with two controls, numerical results with $k = 0.05$. Top: different isosurfaces. Middle: differences in the optimal control lead to different trajectories. Bottom: the feedback approach leads to robustness with respect to noise in the dynamics.

Test 4: 2D eikonal dynamics. This test considers the same two dimensional dynamics as in Test 1, the only difference being the inclusion of an ℓ_1-term in the cost functional. For the inner optimization block we apply the semismooth Newton method presented in Algorithm 4. Results are shown in Fig. 4.4, where differences in the shape of the value function can be observed as the γ_1 weight increases. In the second row of Fig. 4.4, the effect of sparsity on the first control component can be seen from the fact that it is identically zero in a band around the origin. Moreover, the width of the band increases with γ_1. As for the value function, introducing the ℓ_1-term breaks its radially symmetric structure, as it is shown in the first row of Fig. 4.4.

Test 5: 3D car model. In our last test, the dynamics are given by a nonlinear 3D car model with two control variables:

$$f(x, u) = (u_1 \cos(x_3), u_1 \sin(x_3), u_2)$$

with $u_1 \in [-\omega_1, \omega_1]$, and $u_2 \in [-\omega_2, \omega_2]$, $\omega_1 > 0$, $\omega_2 > 0$. For this problem we implement the semismooth Newton algorithm with box constraints introduced in Section 3.4.2. Results are shown in Fig. 4.5. The first row shows same isovalues for different

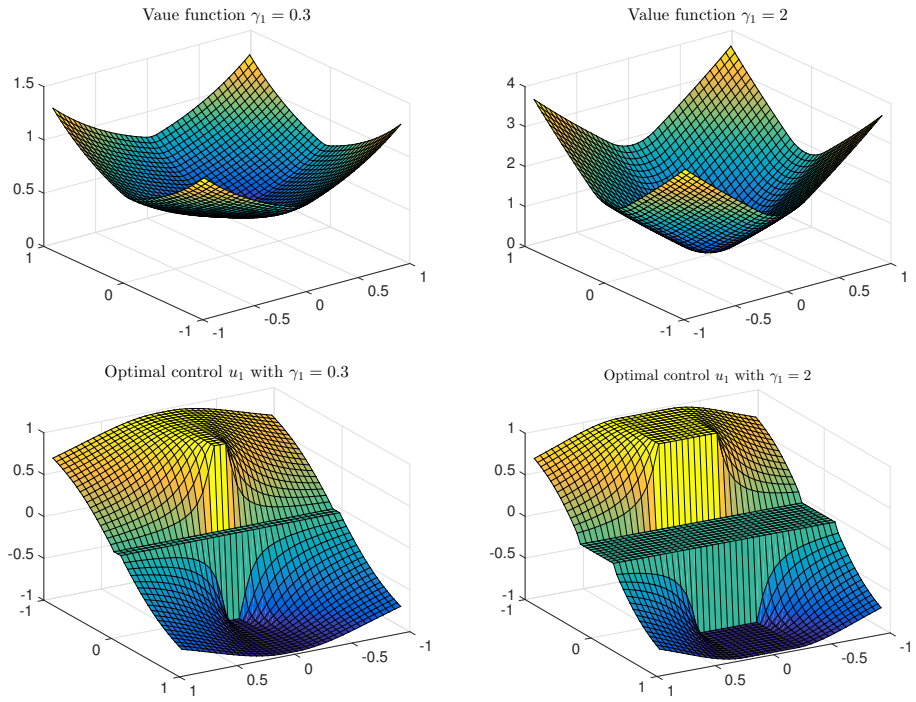

Fig. 4.4: Sparse control of eikonal dynamics. Top: inclusion of an ℓ_1-cost breaks the radially symmetric structure of the solution. Bottom: the sparsity of the optimal control translates into a zero band around of the origin, depending on the weight γ_1.

costs. The addition of an ℓ_1-term shrinks the region of a given isovalue. The second and third rows depict the effect of sparsity in both control variables, creating regions of zero action. A direct consequence of this can be seen at the bottom of Fig. 4.5, where the inclusion of the additional ℓ_1-cost generates optimal trajectories which do not reach the origin due to the interplay between the control cost and the discount factor of the optimal control problem.

Concluding remarks. We have presented a numerical approach for the solution of HJB equations based on a semi-Lagrangian discretization and the use of different local minimization strategies for the approximation of the corresponding numerical Hamiltonian. The numerical results show a more accurate resolution of the optimal control field at a similar computational cost as the currently used schemes. Furthermore, the proposed approach can be also adapted to treat non-differential costs such as ℓ_1-penalizations on the control. Since the numerical approximation of the Hamiltonian constitutes one building block within the construction of approximation schemes for HJB and related equations, the idea of using local minimization techniques can be conveniently recast in similar problems, such as front propagation problems and differential games, and in related approximation techniques, like fast marching schemes, policy iteration methods and high-order approximations.

Appendix A: Decomposition of the control space for Test 3. We make an explicit presentation of the decomposition of the control space of Test 3. The

dynamics is given by

$$f(x,u) = (x_2, x_3 + u_1, u_2)\,, \quad |u_1| \leq a,\ |u_2| \leq b, \quad a,b \in \mathbb{R}.$$

For a given departure point $x^d = (x_1^d, x_2^d, x_3^d) \in \mathbb{R}^3$ and a sufficiently small h, we want to identify a relation between subsets $U_{\mathcal{I}}$ of the control space $U = [-a,a] \times [-b,b]$ and the location of the corresponding arrival points

$$x^d + h(x_2^d, x_3^d + u_1, u_2)\,, \quad (u_1,u_2) \in U_{\mathcal{I}}.$$

In the three-dimensional case, the control set U can be decomposed into at most eight disjoint subsets, one per octant with

$$U = \bigcup_{\mathcal{I} \subset W} U_{\mathcal{I}}$$

for $W = \{1,2,3\}$. Since every octant defines a unique linear interpolant, we solve eight different minimization problems, and then compute the global nodal minimizer by comparison. For instance, let us consider the sector

$$Q_{\{1,2,3\}} = \{(x_1,x_2,x_3)\,|\,x_1^d \leq x_1,\ x_2^d \leq x_2,\ x_3^d \leq x_3,\ (x_1 - x_1^d) + (x_2 - x_2^d) + (x_3 - x_3^d) \leq k\}\,,$$

where the evaluation of the arrival point is defined by the linear interpolant $I_{\{1,2,3\}}$ depending on the points x^d, $x^d + (k,0,0)$, $x^d + (0,k,0)$, and $x^d + (0,0,k)$. The subset $U_{\{1,2,3\}}$ related to this interpolant reduces to

$$U_{\{1,2,3\}} = \{(u_1,u_2) \in U\,|\,x_3^d + u_1 \geq 0,\ u_2 \geq 0\}\,.$$

Note that for this definition to be meaningful, we need to assume that $x_2^d \geq 0$, otherwise $U_{\{1,2,3\}} = \emptyset$. Also note that the condition $(x_1 - x_1^d) + (x_2 - x_2^d) + (x_3 - x_3^d) \leq k$ is omitted by assuming a sufficiently small h ensuring it. The next subset $U_{\{2,3\}}$ relates to the sector

$$Q_{\{2,3\}} = \{(x_1,x_2,x_3)\,|\,x_1^d \geq x_1,\ x_2^d \leq x_2,\ x_3^d \leq x_3,\ -(x_1 - x_1^d) + (x_2 - x_2^d) + (x_3 - x_3^d) \leq k\}\,,$$

and is given by

$$U_{\{2,3\}} = \{(u_1,u_2) \in U\,|\,x_3^d + u_1 \geq 0,\ u_2 \geq 0\}\,.$$

Note that $U_{\{1,2,3\}} \equiv U_{\{2,3\}}$, however, $U_{\{2,3\}}$ is nonempty only when $x_2^d \leq 0$. Therefore, for a given departure point, depending on the coordinate x_2^d, only one of these two subsets will be active with arrival points in different sectors $Q_{\{1,2,3\}}$ or $Q_{\{2,3\}}$ (with different interpolation data). In a similar way, the remaining six subsets can be obtained. By assuming a linear control term, the identification of the control subsets is simple, as it will only require the resolution of linear inequalities where the departure point enters as a fixed data.

Appendix B: Exact value function for Test 1. The exact value function for the infinite horizon optimal control problem with eikonal dynamics in Test 1 is derived. The HJB-equation has the form

$$\lambda v + \max_{u \in U} \left\{ -u^T \nabla v - \left(\frac{\|x\|_2^2}{2} + \frac{\gamma}{2} \|u\|_2^2 \right) \right\} = 0, \quad x \in \mathbb{R}^n, \quad \gamma > 0, \tag{4.4}$$

where $U = \{u \in \mathbb{R}^2 \mid \|u\|_2 \leq 1\}$, for which we want to obtain an explicit solution. If $\frac{1}{\gamma}\|\nabla v\|_2 < 1$, then $u^* = -\frac{1}{\gamma}\nabla v$ provides a maximum for the expression in brackets, and the HJB-equation becomes

$$\lambda v + \frac{1}{2\gamma}\|\nabla v\|_2^2 - \frac{1}{2}\|x\|_2^2 = 0.$$

Switching to polar coordinates (r, φ) this equation can be expressed as

$$\lambda v(r,\varphi) + \frac{1}{2\gamma}(v_r(r,\varphi)^2 + \frac{1}{r^2}v_\varphi(r,\varphi)^2) - \frac{r^2}{2} = 0,$$

and assuming that the solution is radially symmetric

$$\lambda v(r) + \frac{1}{2\gamma}v_r(r)^2 - \frac{r^2}{2} = 0.$$

The ansatz $v_1 = Ar^2$, leads to $\lambda A r^2 + \frac{2}{\gamma}A^2r^2 - \frac{r^2}{2} = 0$ and hence $A = \frac{\gamma}{4}(\sqrt{\lambda^2 + \frac{4}{\gamma}} - \lambda)$. The resulting expression for v_1 is the solution of (4.4), if $(v_1)_r \leq \gamma$ which results in $r \leq 2((\lambda^2 + \frac{4}{\gamma})^{\frac{1}{2}} - \lambda)^{-1} =: \overline{r}$. We note that $v_1(\overline{r}) = \frac{\gamma}{2}\overline{r}$.

We turn to the case that $\frac{1}{\gamma}\|\nabla v\|_2 > 1$. In this case the maximum in (4.4) is achieved on the boundary of U at $u^* = -\frac{\nabla v}{\|\nabla v\|_2}$. Again we look for a radially symmetric solution in polar coordinates

$$\lambda v(r) + |v_r(r)| - \frac{r^2}{2} = \frac{\gamma}{2}. \tag{4.5}$$

Assuming that $v_r \geq 0$ we make the Ansatz $v_2(r) = ar^2 + br + c + de^{-\lambda r}$. Inserting into (4.5) and comparing coefficients we obtain

$$v_2(r) = \frac{1}{2\lambda}r^2 - \frac{1}{\lambda^2}r + \frac{\gamma}{2\lambda} + \frac{1}{\lambda^3} + de^{-\lambda r}.$$

Continuous concatenation with v_1 at $\overline{r}$ implies that

$$d = e^{\lambda\overline{r}}\left[\left(\frac{\gamma}{2} + \frac{1}{\lambda^2}\right)\overline{r} - \frac{1}{2\lambda}\overline{r}^2 - \frac{\gamma}{2\lambda} - \frac{1}{\lambda^3}\right]. \tag{4.6}$$

Note that this latter coupling yields that $v(r)$ is a $\mathcal{C}^1(\mathbb{R})$ function and therefore it is also a classical solution of eq. (4.4). Summarizing we have

$$v(r) = \begin{cases} \frac{\gamma}{4}(\sqrt{\lambda^2 + \frac{4}{\gamma}} - \lambda)r^2 & \text{for } r \leq \overline{r} \\ \frac{1}{2\lambda}r^2 - \frac{1}{\lambda^2}r + \frac{\gamma}{2\lambda} + \frac{1}{\lambda^3} + de^{-\lambda r} & \text{for } r > \overline{r}, \end{cases}$$

where d is given in (4.6).

REFERENCES

[1] A. Alla, M. Falcone, and D. Kalise, *An efficient policy iteration algorithm for dynamic programming equations*, SIAM J. Sci. Comput. **37** (2015), no. 1, 181–200.

[2] W. Alt and C. Schneider, *Linear-quadratic control problems with l^1-control cost*, Optim. Control Appl. Meth. (2014), published online, doi: 10.1002/oca.2126.
[3] K. Alton and I. Mitchell, *An ordered upwind method with precomputed stencil and monotone node acceptance for solving static convex Hamilton–Jacobi equations*, Journal of Scientific Computing **51** (2012), no. 2, 313–348.
[4] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Birkhäuser, Boston, 2008.
[5] O. Bokanowski, S. Maroso, and H. Zidani, *Some convergence results for Howard's algorithm*, SIAM J. Numer. Anal. **47** (2009), no. 4, 3001–3026.
[6] N. Botkin, K.H. Hoffmann, and V. Turova, *Stable numerical schemes for solving Hamilton–Jacobi–Bellman–Isaacs equations*, SIAM J. Sci. Comput. **33** (2011), no. 2, 992–1007.
[7] S. Cacace, E. Cristiani, M. Falcone, and A. Picarelli, *A patchy dynamic programming scheme for a class of Hamilton–Jacobi–Bellman equations*, SIAM J. Sci. Comput. **34** (2012), no. 5, A2625–A2649.
[8] E. Carlini, M. Falcone, and R. Ferretti, *An efficient algorithm for Hamilton-Jacobi equations in high dimension*, Comput. Visual. Sci. **7** (2004), no. 1, 15–29.
[9] A. Chambolle and T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vis. **40** (2011), no. 1, 120–145.
[10] E. Cristiani and M. Falcone, *Fast semi-Lagrangian schemes for the Eikonal equation and applications*, SIAM J. Numer. Anal. **45** (2007), no. 5, 1979–2011.
[11] M. Falcone, *A numerical approach to the infinite horizon problem of deterministic control theory*, Appl. Math. Opt. **15** (1987), no. 1, 1–13.
[12] ———, *Numerical methods for differential games based on partial differential equations*, Int. Game Theory Rev. **8** (2006), no. 2, 231–272.
[13] M. Falcone and R. Ferretti, *Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi Equations*, Society for Industrial and Applied Mathematics, 2014.
[14] H. Ishii, *Uniqueness of unbounded viscosity solution of Hamilton-Jacobi equations*, Indiana Univ. Math. J. **33** (1984), no. 5, 721–748.
[15] K. Ito and K. Kunisch, *Lagrange multiplier approach to variational problems and applications*, Society for Industrial and Applied Mathematics, 2008.
[16] M. Jarczyk, *Lokale Optimierungsstrategien in der dynamischen Programmierung*, Diploma thesis, Universität Bayreuth, 2005.
[17] A. Kröner, K. Kunisch, and H. Zidani, *Optimal feedback control of the undamped wave equation by solving a HJB equation*, ESAIM: Contr. Optim. Calc. Var. (2014), to appear. doi: 10.1051/cocv/2014033.
[18] H. Maurer and G. Vossen, *On L^1-minimization in optimal control and applications to robotics*, Optim. Control Appl. Meth. **27** (2006), 301–321.
[19] R.T. Rockafellar, *Convex analysis*, Convex Analysis, Princeton University Press, 1997.
[20] J. Sethian and A. Vladimirsky, *Ordered upwind methods for static Hamilton–Jacobi equations: Theory and algorithms*, SIAM J. Numer. Anal. **41** (2003), no. 1, 325–363.
[21] M. Ulbrich, *Semismooth Newton Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces*, Society for Industrial and Applied Mathematics, 2011.
[22] C. Unrath, *Nichtglatte Optimierungsstrategien in der dynamischen Programmierung*, Diploma thesis, Universität Bayreuth, 2006.
[23] T. Valkonen, *A primal-dual hybrid gradient method for non-linear operators with applications to MRI*, Inverse Probl. **30** (2014), no. 5.
[24] T. Xiong, M. Zhang, Y.-T. Zhang, and C.-W. Shu, *Fast sweeping fifth order WENO scheme for static Hamilton–Jacobi equations with accurate boundary treatment*, J. Sci. Comput. **45** (2010), no. 1-3, 514–536.

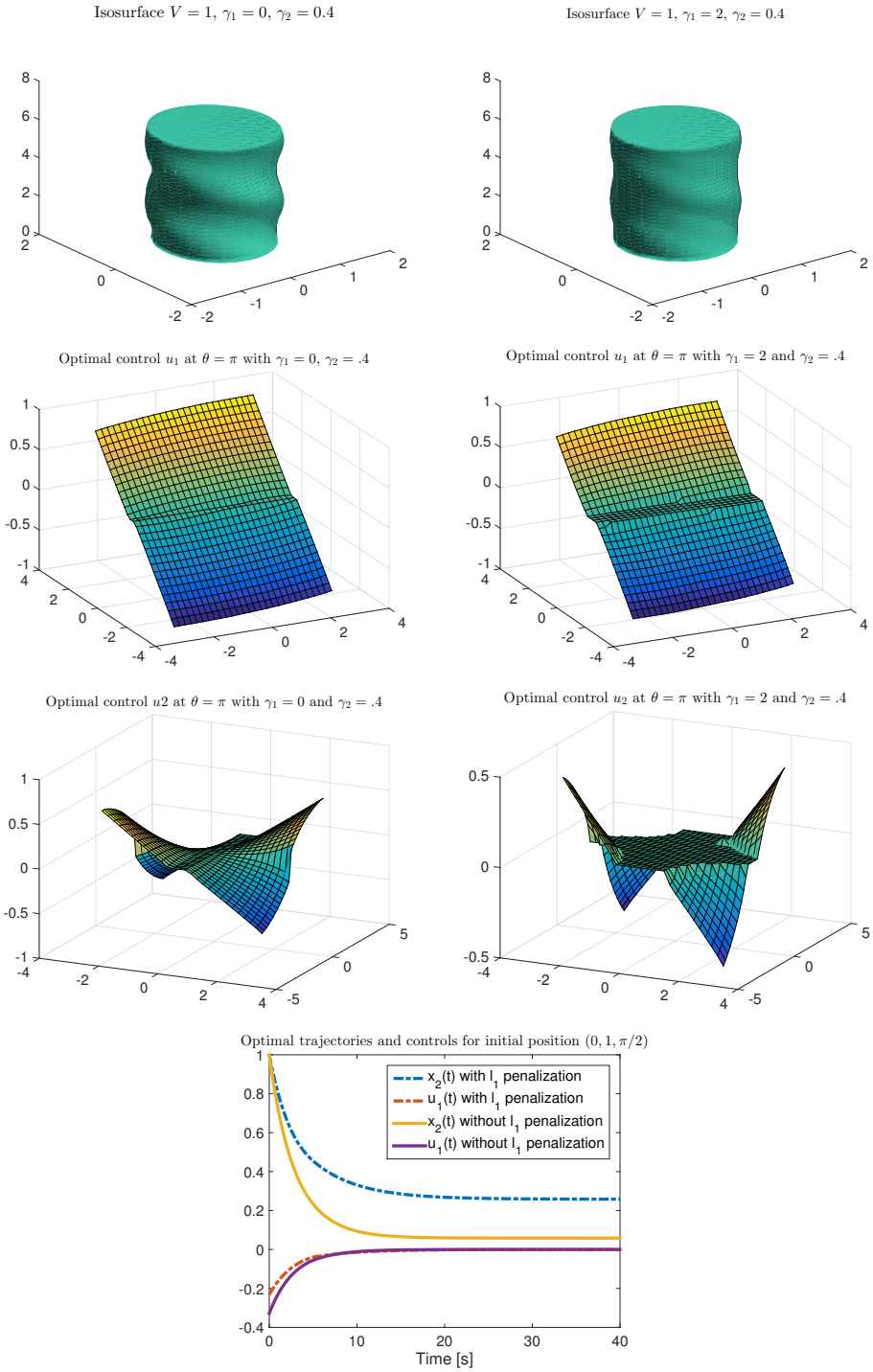

Fig. 4.5: Sparse control of a 3D car model. Top: different isosurfaces with different ℓ_1-cost. Rows 2 and 3: effect of sparsity on the optimal control field. Bottom: optimal trajectories with and without sparsity.