Local optimization techniques in semi-Lagrangian schemes for Hamilton-Jacobi equations

Dante Kalise
Optimization and Optimal Control Group
Radon Institute for Computational and Applied Mathematics (Linz)

(joint work with A. Kröner and K. Kunisch)

Maurizio60 Conference, Sapienza University of Rome , 5.12.2014

Infinite horizon optimal control

Consider the minimization of

$$J_x(\alpha) = \int_0^\infty l(y_x(s), \alpha(s)) e^{-\lambda s}\, ds\,, \quad \lambda > 0\,,$$

subject to

$$\begin{aligned} &\dot{y}(t) = f(y, \alpha)\,, \quad y(\cdot) \in \mathbb{R}^n, \alpha(\cdot) \in A \subset \mathbb{R}^m, \\ &y(0) = x\,. \end{aligned}$$

For

$$v(x) = \inf_{\alpha \in \mathcal{U}} J_x(\alpha)\,,$$

HJB eq. for Infinite Horizon Control

$$\lambda v(x) + \sup_{a \in A}\{-f(x,a) \cdot Dv(x) - l(x,a)\} = 0\,.$$

The basic SL-FEM scheme

DPP with pseudotime parameter h, semi-discrete formulation

$$\begin{cases} v^h(x) = \min_{a \in A}\{(1-\lambda h)v^h(x+hf(x,a)) + hl(x,a)\} & \text{in } \Omega \subset \mathbb{R}^n, \\ v^h|_{\partial\Omega} = g(x). \end{cases}$$

We discretize Ω into a set of nodes x_i, $i = 1, \ldots, N$ and we consider the set of values $V := \{v^h(x_i)\}$. We define the first order interpolant

$$v^h(x) = I_1[V](x) + O(\Delta x^2), \qquad \Delta x = \max_i |x_i - x_{i-1}|.$$

Fully discrete SL-FEM

$$(V)_i = \min_{a \in A}\{(1-\lambda h)I_1[V](x_i + hf(x_i,a)) + hl(x_i,a)\}, \quad i = 1, \ldots, N.$$

The basic SL-FEM scheme

The standard way to solve this system is by a fixed point iteration of the value function.

Fully discrete SL-FEM/ Value Iteration (VI) scheme

$$\begin{aligned} V^{k+1} &= T(V^k), \quad \text{for } i = 1, \ldots, N \\ \left(T(V^k)\right)_i &\equiv \min_{a \in A} \left\{(1-\lambda h) I_1[V^k](x_i + hf(x_i, a)) + hl(x_i, a)\right\} . \end{aligned}$$

This scheme is fully-discrete except for the minimization routine.

- A standard practice in this context is to discretize the set A and compute the minimization by comparison. The computation of the optimal control

$$a_i^* = \underset{a \in A}{argmin} \left\{(1-\lambda h) I_1[V^k](x_i + hf(x_i, a)) + hl(x_i, a)\right\} ,$$

shall inherit this approach.

Local minimization in SL schemes

Previous attempts to embed other minimization techniques:

- Carlini, Cumpleañero and Ferretti (2004): Brendt algorithm.
- Grüne et al. (2005,06,09): Brendt , recursive search, Newton-Bundle.
- Bonnans et al. (2004): bilevel optimization problem HJB + parameters.

What we propose:

to replace the minimization-by-comparison approach with state of the art numerical optimization solvers, able to provide a more accurate representation of the control field at a similar computational cost.

The interpolation operator

For a triangulated 2D domain we consider local linear interpolants

$$I[V](x) = cx_1 + dx_2 + e\,,$$

with coefficients are given by

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

Under the assumptions

$$h \leq \|f(\cdot,\cdot)\|_\infty^{-1} \frac{\sqrt{2}}{2} k, \quad \dot{y}(t) = g(y) + Bu\,, \quad l(x,a) = w(x) + \frac{\gamma}{2}\|u\|_2^2\,,$$

Writing the local minimization problem

... and evaluating at the arrival point

$$x_a = x_d + hf(x_d, a)\,,$$

it is possible to rewrite

$$\min_{a \in A} \Big\{ (1 - \lambda h) I_1[V^k](x_i + hf(x_i, a)) + hl(x_i, a) \Big\}$$

as

$$\min_{a \in A_i \subset A} J(a)$$

with

$$J(a) := \frac{1}{2} a^t Q a + La + C\,.$$

Subdivision of the control set

Since different controls generate arrival points located at different triangles, it is necessary to establish a subdivision of the control set.

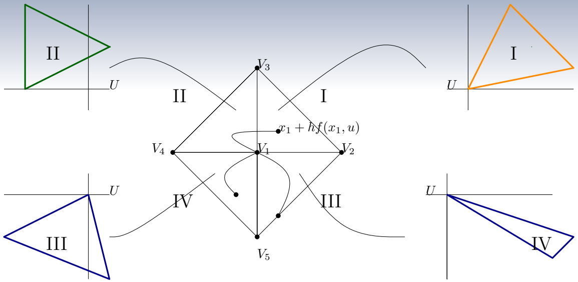


Figure: Arrival points and related control sets.

For eikonal dynamics, the relation is trivial, in other cases it requires some work.

A semismooth Newton method

We address the 2D eikonal case. We need to solve a problem of the form as

$$\min_{a \in A_1} \frac{1}{2} a^t Q a + L a + C$$

where

$$A_1 = \{(a_1, a_2) | a_1 \geq 0\,, a_2 \geq 0\,, \|a\|_2 \leq 1\}\,.$$

The first-order optimality condition can be formulated as

$$a = P(a - \vartheta \nabla J(a)) \quad \forall \vartheta > 0\,, a \in \mathbb{R}^2\,,$$

with projection

$$P : \mathbb{R}^m \to \mathbb{R}^m, \quad P(p) = \frac{\max(0,p)}{\max(1, \|max(0,p)\|_2)}\,.$$

A semismooth Newton method

We introduce $p = a - \vartheta \nabla J(a)$ and $\beta = \max(1, \| \max(0,p)\|)$. By defining the system

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

the optimality condition can be formulated as $E(z) = 0$ with $z = (a,p,\beta)$, which can be solved via a Newton iteration

$$\begin{aligned} DE(z_k)\delta z &= -E(z_k)\,, \\ z_{k+1} &= z_k + \delta z\,. \end{aligned}$$

Assessing the performance of the method

method	tolerance	iterations	cpu time	error
Chambolle-Pock	1E-4	9	7.3E-5 [s]	4.31E-5
Semi-smooth 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

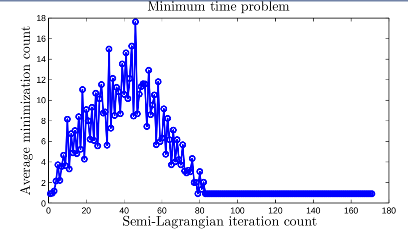
Assessing the performance of the method

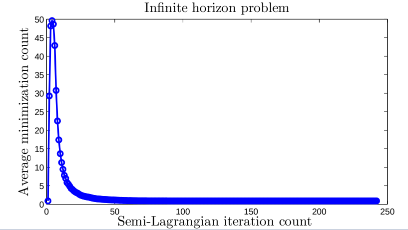
method	tolerance	iterations	cpu time	error
Semi-smooth Newton	1E-4	101	4.53E-3 [s]	1.51E-3
Comparison (4E4 evaluations)	–	–	5.18E-3 [s]	5.77E-3
Comparison (2E5 evaluations)	–	–	2.08E-2 [s]	2.80E-3

Assessing the performance of the method

method	tolerance	iterations	cpu time	error
Semi-smooth 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

Performance inside the SL scheme



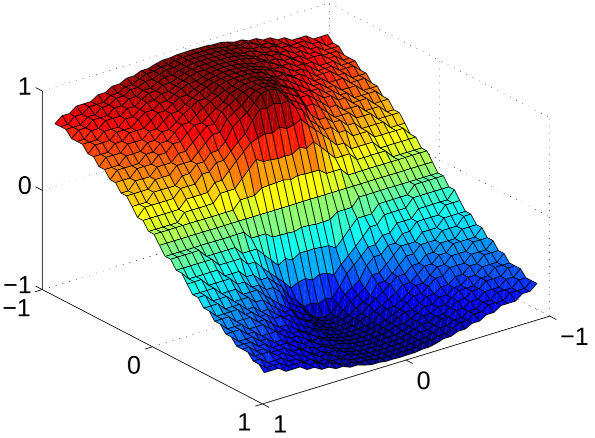


Performance inside the SL scheme

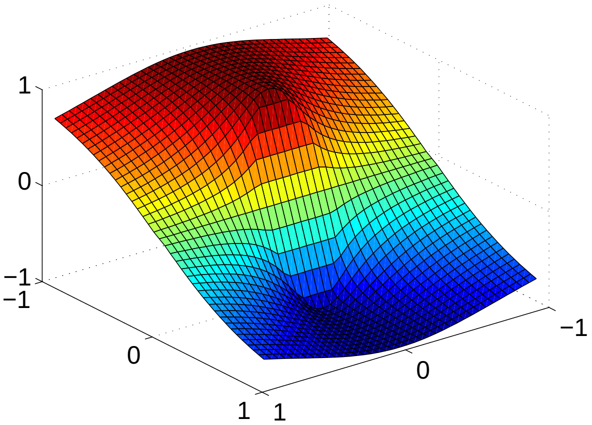
	$\Delta x = 0.05$ $(40^2$ DoF)			$\Delta x = 0.025$ $(80^2$ DoF)		
Minimization	CPU time	$\|v - V_h\|_1$	$\|u - U_h\|_1$	CPU time	$\|v - V_h\|_1$	$\|u - U_h\|_1$
Comparison	63.52 [s]	3.12E-2	3.84E-2	5.76E2 [s]	1.96E-2	1.74E-2
Semismooth	77.25 [s]	2.62E-2	1.61E-2	7.27E2 [s]	1.36E-2	7.21E-3
Chambolle-Pock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

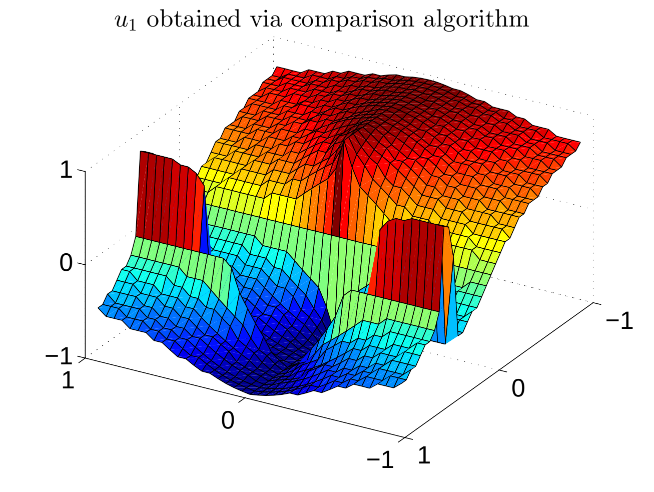
Table: Infinite horizon control of eikonal dynamics. CPU time and errors for a semi-Lagrangian scheme with different minization algorithms. The comparison algorithm was run with a discrete set of 1028 control points.

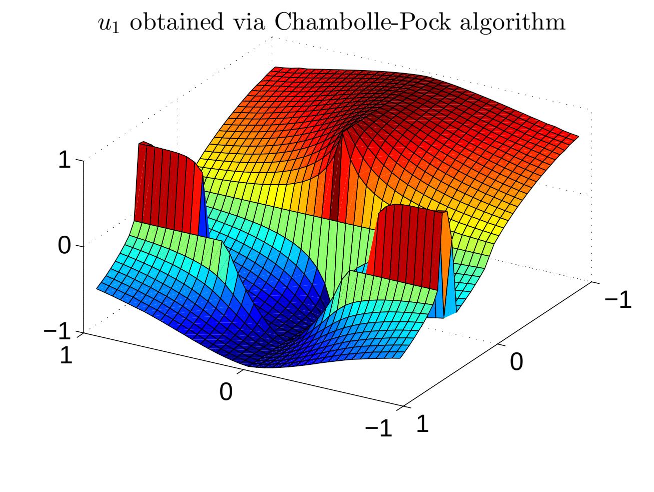
u_1 obtained via comparison algorithm



u_1 obtained via Chambolle-Pock algorithm




u_1 obtained via comparison algorithm
1
0
−1
1
0
−1
1
0
−1


u_1 obtained via Chambolle-Pock algorithm
1
0
−1
1
0
−1
1
0
−1

Concluding remarks

- We have implemented state of the art building blocks for numerical optimization in SL schemes.
- We obtain, a similar computational cost, a better resolution of the control field with different dynamics and costs.
- Similar ideas can be applied in other control problems and related numerical schemes.

Thanks for your attention.

Local optimization techniques in semi-Lagrangian schemes for Hamilton-Jacobi equations

Dante Kalise
Optimization and Optimal Control Group
Radon Institute for Computational and Applied Mathematics (Linz)

(joint work with A. Kröner and K. Kunisch)

Maurizio60 Conference, Sapienza University of Rome , 5.12.2014