Control Theory and Hyperbolic Partial Differential Equations

Adriano Festa
DISIM, L'Aquila, IT

Doctorate school Mathematics and Models

2019

Outline of the themes of the lectures

1. Bellman's approach to **Optimal Control**
2. Hamilton Jacobi Equations: generalities
3. Weak solutions
4. How to build a simple FD scheme
5. the '**Course of dimensionality**'
6. Applications

One starting motivation:
Using the Dynamical Programming Principle we show that the value function of a OC problem

$$a(\cdot) \in L^\infty([0,+\infty[,\mathcal{A}); \qquad \begin{cases} \dot{y}(t) = f(y_x(t), a(t)), \\ y_x(0) = x. \end{cases}$$

$$v(x) = \inf_a \int_0^{\tau_x(a)} l(y_x(s), a(s)) e^{-\lambda s}\, ds.$$

where $\tau_x(a)$ is the time of first exit from the set Ω, is the **viscosity solution** of the HJ equation

$$\begin{cases} \lambda v(x) + H(x, Dv(x)) = 0 & x \in \Omega \\ v(x) = 0 & x \in \partial\Omega \end{cases}$$

Bellman's Approach to OC

Where the **Hamiltonian** is defined in the usual way

$$H(x,p) := \max_a \{-f(x,a)\cdot p - l(x,a)\},$$

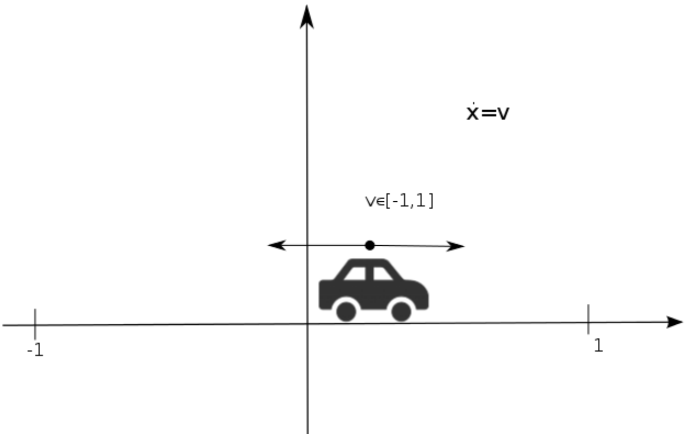
and $\tau_x(a)$ is the first time of exit from the domain.

Afterwards, with the Pontryagin's condition, it is possible the **synthesis** of a feedback control

$$a(t) = S(y_{x_0}(t)); \quad S(x) \in \arg\min_{a\in\mathcal{A}}\{f(x,a)\cdot Dv(x) + l(x,a)\}$$

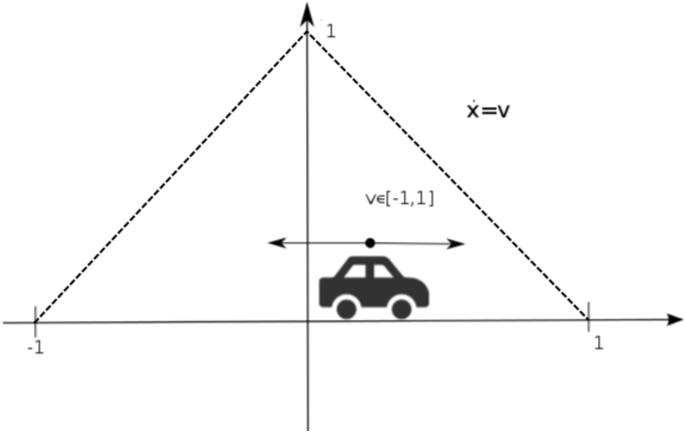
Optimal escape with a bounded speed car

Let us suppose to want to escape from the set $\Omega = [-1, 1]$, with a car of speed $v \in [-1, 1]$.



Optimal escape with a bounded speed car

Let us suppose to want to escape from the set $\Omega = [-1, 1]$, with a car of speed $v \in [-1, 1]$.



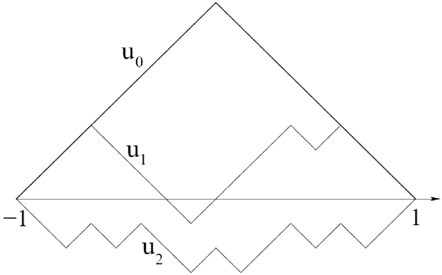
Typical issues

Eikonal Equation - minimum time problem

$$\max_{a\in B(0,1)} (a\cdot\nabla u(x)) = |\nabla u(x)| = 1, \quad u(-1) = u(1) = 0.$$

$$\Updownarrow$$

$$\begin{cases} \text{Minimize } T(x) := \int_0^\tau 1dt \\ \dot{y}(t) = a(t) \text{ a.e. } t\in[0,\tau] \\ a(t)\in[-1,1], \text{ a.e. } t\in[0,\tau], \quad y(0)=y_0 \text{ and } y(\tau)\in\{-1,1\}, \end{cases}$$



A natural question

Is it possible to select, among the a.e. solutions
the **‘interesting’** one?

A natural question

Is it possible to select, among the a.e. solutions
the **‘interesting’** one?

Answer: Viscosity solutions theory [Crandall, Lions 1983].

HJ equations

Let Ω be a subset of R^N. In general we consider the following **Hamilton-Jacobi equation**

$$\begin{cases} H(x, u(x), Du(x)) = 0, & x \in \Omega \\ u(x) = \psi(x), & x \in \partial\Omega \end{cases} \tag{1}$$

Hamiltonians with a special interest are:

- $H(x,p) := |p| - l(x)$ (Eikonal equation (distance))
- $H(x,r,p) := \lambda r + \sup_{a \in A} \{-f(x,a) \cdot p - l(x,a)\}$ (**OC problem**)
- $H(x,r,p) := \lambda r + \sup_{a \in A} \inf_{b \in B} \{-f(x,a,b) \cdot p - l(x,a,b)\}$ (Differential Game)

Viscosity solutions

Definition

A continuous function $v : \Omega \to \mathbb{R}$ is a viscosity solution of the equation (1) if the following conditions are satisfied:

- for any test function $\phi \in C^1(\Omega)$, if $x_0 \in \Omega$ is a local maximum point for $v - \phi$, then

$$H(x_0, v(x_0), \nabla\phi(x_0)) \leq 0 \quad \textbf{(viscosity subsolution)}$$

- for any test function $\phi \in C^1(\Omega)$, if $x_0 \in \Omega$ is a local minimum point for $v - \phi$, then

$$H(x_0, v(x_0), \nabla\phi(x_0)) \geq 0 \quad \textbf{(viscosity supersolution)}$$

Computing viscosity solutions

Viscosity solutions are typically **uniformly continuous and bounded**.
This means that the numerical methods should be able to reconstruct kinks in the solution (jumps in the derivative).

Computing viscosity solutions

Viscosity solutions are typically **uniformly continuous and bounded**.
This means that the numerical methods should be able to reconstruct kinks in the solution (jumps in the derivative).

The goal is to have:

- stability
- convergence
- very small 'numerical viscosity', to avoid the smearing of kinks
- the schemes should also be able to compute the solution after the onset of singularities

Model problem: 1D eikonal equation

Let us consider the following problem in $[-1, 1]$

$$\begin{cases} |u_x| = 1 & x \in (-1, 1) \\ u(-1) = u(1) \end{cases}$$

The problem can be reformulate as

$$\begin{cases} \max\limits_{a \in \{-1,1\}} \{a\, u_x\} = 1 & x \in (-1, 1) \\ u(-1) = u(1) \end{cases}$$

Model problem: 1D eikonal equation

Which is the advantage? To explicit the direction of the characteristic of the problem.

Indeed when $a > 0$, $u_x \approx D^- U_i = \frac{U_{i-1} - U_i}{\Delta x}$ will have some chances to be **stable**.

Instead when $a < 0$, $u_x \approx D^+ U_i = \frac{U_{i+1} - U_i}{\Delta x}$ is our candidate

Model problem: 1D eikonal equation

Which is the advantage? To explicit the direction of the characteristic of the problem.

Indeed when $a > 0$, $u_x \approx D^- U_i = \frac{U_{i-1} - U_i}{\Delta x}$ will have some chances to be **stable**.

Instead when $a < 0$, $u_x \approx D^+ U_i = \frac{U_{i+1} - U_i}{\Delta x}$ is our candidate

A good idea?

What about $u_x \approx D^+ U_i - D^- U_i = \frac{U_{i+1} - U_{i-1}}{\Delta x}$?

Model problem: 1D eikonal equation

Which is the advantage? To explicit the direction of the characteristic of the problem.

Indeed when $a > 0$, $u_x \approx D^- U_i = \frac{U_{i-1} - U_i}{\Delta x}$ will have some chances to be **stable**.

Instead when $a < 0$, $u_x \approx D^+ U_i = \frac{U_{i+1} - U_i}{\Delta x}$ is our candidate

A good idea?

What about $u_x \approx D^+ U_i - D^- U_i = \frac{U_{i+1} - U_{i-1}}{\Delta x}$?
Centered differences can be proven to be always unstable.

Model problem: 1D eikonal equation

An easy upwind scheme

Called

$$f^-(U_i) = \min\{-D^- U_i, 0\}, \quad f^+(U_i) = \max\{D^+ U_i, 0\},$$

we have that the scheme

$$\left[f^+ - f^-\right](U_i) = 1$$

is **stable** and consistent with (1).

Convergence estimate: 1D eikonal equation

A classic convergence result for monotone schemes (like upwind) is due to Crandall and Lions (1984)

convergence bounds

They proved for the upwind scheme above an **a-priori error estimate** in $L\infty$

$$\sup_i |u(x_i) - U_i| \leq C(\Delta x^{1/2}).$$

Computational issues

R. Bellman, *Dynamical Programming*, 1957.

DYNAMIC PROGRAMMING

BY

RICHARD BELLMAN

PRINCETON UNIVERSITY PRESS
PRINCETON, NEW JERSEY

mizing point.

There are, however, some details to consider. In the first place, the effective analytic solution of a large number of even simple equations as, for example, linear equations, is a difficult affair. Lowering our sights, even a computational solution usually has a number of difficulties of both gross and subtle nature. Consequently, the determination of this maximum is quite definitely not routine when the number of variables is large.

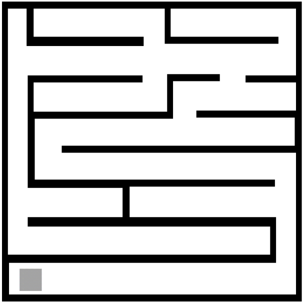
All this may be subsumed under the heading "the curse of dimensionality." Since this is a curse which has hung over the head of the physicist and astronomer for many a year, there is no need to feel discouraged about the possibility of obtaining significant results despite it.

However, this is not the sole difficulty. A further characteristic of these problems, as we shall see in the ensuing pages, is that calculus is not always sufficient for our purposes, as a consequence of the perverse fact that quite frequently the solution is a boundary point of the region

Some (partial answers) to the computational issues

- Fast marching methods (it plays with the dependency of the nodes of the grid)
- **Fast sweeping methods** (Gauss Seidel iteration on a squared grid)
- Parallel computing
- **Domain Decomposition methods** (with dependent or independent domains)

Applications I - Labyrinths



We consider the labyrinth $I(x)$ as a digital image with $I(x) = 0$ if x is on a wall, $I(x) = 0.5$ if x is on the target, $I(x) = 1$ otherwise.
We solve the eikonal equation

$$|Du(x)| = f(x) \quad x \in \Omega$$

with the discontinuous running cost

$$f(x) = \begin{cases} \frac{1}{4} & \text{if } I(x) = 1 \\ M & \text{if } I(x) = 0. \end{cases}$$

Applications I - Labyrinths

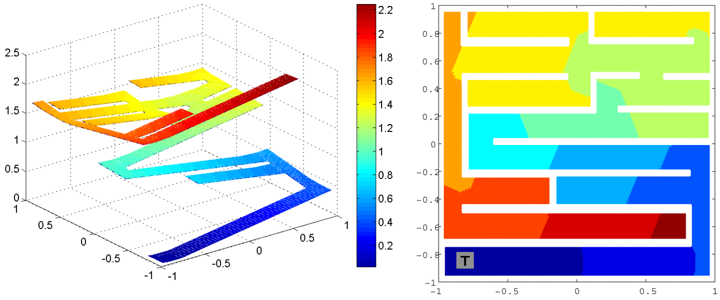


Figure: Mesh and level sets of the value function for the labyrinth problem ($dx = dt = 0.0078$, $M = 10^{10}$).

Applications II - Shape-From-Shading

The partial differential equation related to the Shape-from-Shading model is

$$R(n(x,y)) = I(x,y)$$

where I is the brightness function measured at all points (x,y) in the image, R is the reflectance function. If the surface is smooth we have

$$n(x;y) = \frac{(-v_x(x,y), -v_y(x,y), 1)}{\sqrt{1+|\nabla v(x,y)|^2}}.$$

If the light source is vertical, i.e. $\omega = (0,0,1)$, then equation simplifies to the eikonal equation

$$|\nabla u(x,y)| = \left(\sqrt{\frac{1}{I(x,y)^2} - 1}\right), \quad (x,y) \in \Omega.$$

Applications II - Shape-From-Shading

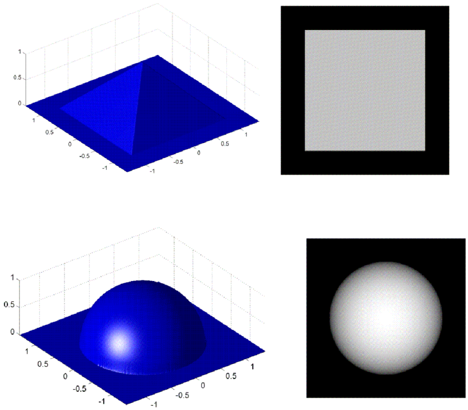


Figure: Reconstructed surface from its shading data.

Applications II - Shape-From-Shading

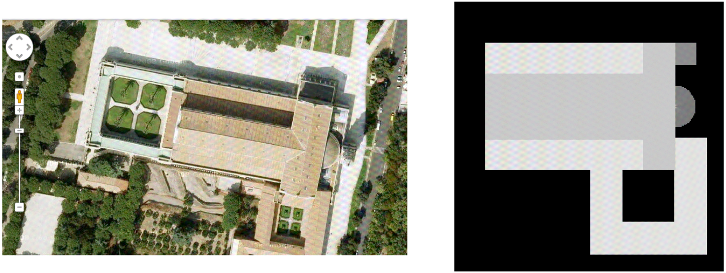


Figure: Basilica of Saint Paul Outside the Walls: satellite image and simplified sfs-datum.

Applications II - Shape-From-Shading

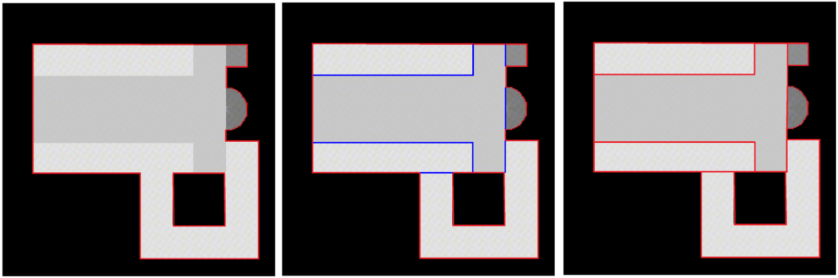


Figure: Choosing boundary conditions.

Applications II - Shape-From-Shading

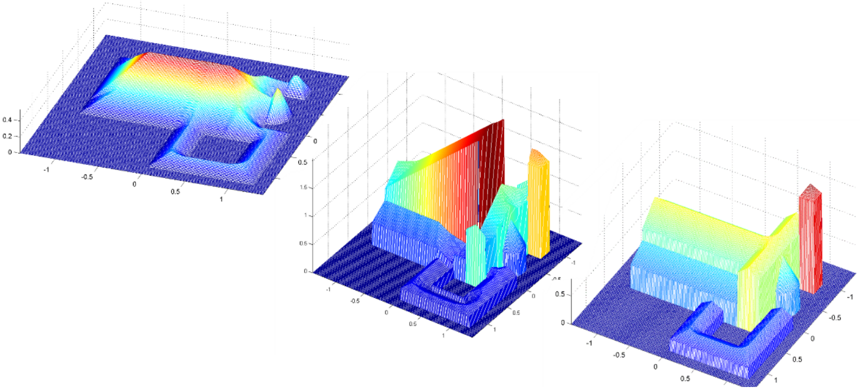
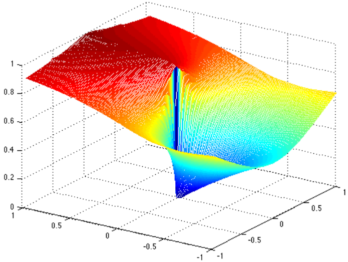
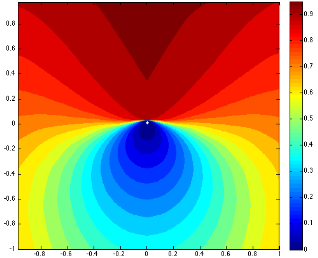


Figure: Solutions with various BC.

Target problems

Setting: a trajectory is driven to arrive in a *Target set* $\mathcal{T} \subset \Omega$.
Example: Zermelo Navigation Problem

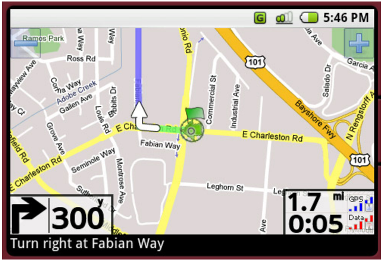




The target is a ball of radius equal to 0.005 centred in the origin, the control is in $A = B(0,1)$.

$$f(x,a) = a + \begin{pmatrix} 1 - x_2^2 \\ 0 \end{pmatrix}, \quad \Omega = [-1,1]^2, \quad \lambda = 1, \quad l(x,y,a) = 1.$$

Application III: Walking on a graph



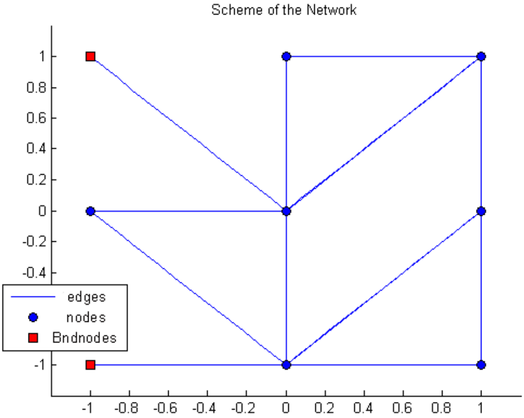
Problem

Optimization problems on a graph (shortest path, cheaper path etc. etc.)

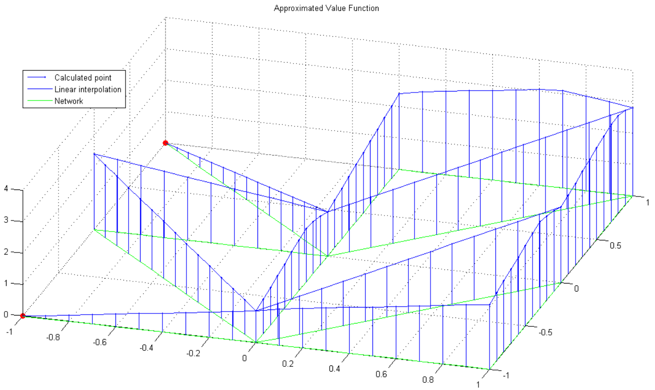
Some classical approches

- Combinatory (Dijkstra algorithm)
- Differential (Eikonal equation with constraints)

Test 1: structure of the graph



Test 1: solution and experimental convergence



$\Delta x = h$	$\|\| \cdot \|\|_\infty$	$Ord(L_\infty)$	$\|\| \cdot \|\|_2$	$Ord(L_2)$
0.2	0.1716		0.0820	
0.1	0.0716	1.2610	0.0297	1.4652
0.05	0.0284	1.3341	0.0127	1.2256
0.025	0.0126	1.1611	0.0072	0.8188

Application IV: Differential Game

Let $\Omega \subset \mathbb{R}^n$, we are considering the system

$$\begin{cases} y'(t) = f(y(t), a(t), b(t)), & t > 0, \\ y(0) = x, \end{cases}$$

- $f : \Omega \times A \times B \to \mathbb{R}^n$ is enough regular
- A, B are compact metric spaces,
- $b \in \mathcal{B} := \{$ measurable functions $[0, +\infty[\to B\}$
- $a \in \mathcal{A} = \{$ measurable functions $[0, +\infty[\to A\}$

$$J(x, a, b) = \int_0^{\tau_x} dt, \quad \tau_x = \text{time of first arrival in } \mathcal{T}$$

Lower value function

In the following we will refer to the problem
Find

$$u(x) := \inf_{\phi \in \Phi} \sup_{a \in \mathcal{A}} J(x, a, b)$$

where $\Phi : \mathcal{A} \to \mathcal{B}$ is a non anticipating strategy for the second player, this is the **lower value** of the game

Theorem (Evans Souganidis 84)

If $\mathcal{R} \setminus \mathcal{T}$ is open and $u(.)$ is continuous, then $u(.)$ is a viscosity solution of

$$\begin{cases} H(x, Du) = \min_{a \in A} \max_{b \in B} \{-f(x, a, b) \cdot Du\} - 1 = 0 & \text{in } \mathcal{R} \setminus \mathcal{T} \\ u(x) = 0 & \text{in } \mathcal{T} \end{cases}$$

Pursuit-Evasion game w. mult. purs.:

The dynamic is the following

$$\begin{cases} \dot{y}_1 = a_1 - b \\ \dot{y}_2 = a_2 - b \\ \vdots \\ \dot{y}_n = a_n - b \\ y(0) = y_0 \end{cases}$$

where a_i is the velocity of the $i - pursuer$ and b is the velocity of the evader.
We introduce also the following spaces of controls.

$$a \in A := B(0,1)^n$$
$$b \in B := B(0,1)$$

Tag-Chase: example 1

Examples: test with 5 pursuers 1 evader
case1,
case2,
case3.

Plan of the lecture:

1. Today: A general introduction
2. **13.02: Dynamical programming principle and Hamilton Jacobi equations**
3. **14.02: Viscosity solutions and well-position**
4. 19.02: Numerical Methods
5. 20.02: Lab Programming a 1D code in Matlab

Infos

My contact

Adriano FESTA
adriano.festa@univaq.it

Lecture Notes

Available on

http://adrianofesta.altervista.org/insegnamentoeng.html

Possible applications to handle with these tools

1. Optimal control: Robotic navigation, labyrinths
2. Optimal control: Optimal path on a Network, Traffic
3. Imaging 1: Segmentation
4. Imaging 2: Shape-from-shading
5. Geo/Bio applications: Sand piles geometry
6. Geo/Bio applications: Optimal path through an inhomogeneous media
7. Finance: Optimal pricing, mean Field Games
8. Programming: Parallel computing using MPI

... and more.

Questions?