Fast Marching Methods for Front Propagation
Lecture 1/3

M. Falcone

Dipartimento di Matematica
SAPIENZA, Università di Roma

Autumn School
"Introduction to Numerical Methods for Moving Boundaries"
ENSTA, Paris, November, 12-14, 2007

Outline

1. Introduction
2. Front Propagation and Minimum Time Problem
3. Global Numerical Approximations
4. Fast Marching Methods
   - A CFL-like condition for FM method

Outline

1. Introduction
2. Front Propagation and Minimum Time Problem
3. Global Numerical Approximations
4. Fast Marching Methods
   - A CFL-like condition for FM method

Level Set Method

The Level Set method has had a great success for the analysis of front propagation problems for its capability to handle many different physical phenomena within the same theoretical framework.
One can use it for isotropic and anisotropic front propagation, for merging different fronts, for Mean Curvature Motion (MCM) and other situations when the velocity depends on some geometrical properties of the front.
It allows to develop the analysis also after the on set of singularities.

Level Set Method

The big disadvantage with respect to "shock fitting" numerical methods is that it adds one dimension to the problem.
Infact our unknown is the "representation" function $u : \mathbb{R}^n \times [0, T] \rightarrow \mathbb{R}$ and looking at the 0-level set of u we can back to the front, i.e.

$$\Gamma_t \equiv \{x : u(x, t) = 0\}$$

The model equation corresponding to the LS method is

$$\begin{cases} u_t + c(x)|\nabla u(x)| = 0 & x \in \mathbb{R}^n \times [0, T] \\ u(x) = u_0(x) & x \in \mathbb{R}^n \end{cases} \tag{1}$$

where u_0 must be a representation function for the front (i.e.

$$\begin{cases} u_0 > 0, & x \in \mathbb{R}^n \setminus \Omega_0 \\ u(x) = 0 & x \in \mathbb{R}^n \\ u(x) < 0 & x \in \Omega_0 \end{cases} \tag{2}$$

More General Models

In the standard model the normal velocity $c : \mathbb{R}^n \to \mathbb{R}$ is given, but the same approach applies to other scalar velocities

$$\begin{cases} c(x,t) & \text{isotropic growth with time varying velocity} \\ c(x,\eta) & \text{anisotropic growth, cristal growth} \\ c(x,k(x)) & \text{Mean Curvature Motion} \end{cases} \tag{3}$$

Fast Marching Method

In this process we do not want loose accuracy and we would like to keep the derivative of u big enough (ideally $|Du(x)| = 1$) to get an accurate representation of the front (re-initialization).

The Fast Marching method has been conceived to speed up the computations and save CPU time.

The crucial point is to concentrate the computational effort around the front at every iteration avoiding useless computations.

Questions

In order to set up and analyze the FMM we have to answer several questions:

- what is the initial configuration of the front in our numerical scheme?
- what is the position of the front at every iteration?
- which nodes will be necessary to compute the front configuration at the following time step?
- does the procedure converge to the correct viscosity solution?
- how many operations will be needed to get the right solution?

Outline

1. Introduction
2. Front Propagation and Minimum Time Problem
3. Global Numerical Approximations
4. Fast Marching Methods
   - A CFL-like condition for FM method

Monotone evolution of fronts

Assume that the sign of the normal velocity is fixed, f.e. take $c(x) > 0$.

The evolution of the front is monotone increasing in the sense that the measure of the domain entoured by the front Γ_t is increasing. Defining

$$\Omega_t \equiv \{x : u(x,t) < 0\}$$

this means that $\Omega_t \subset \Omega_{t+s}$ for every $s > 0$.

If the evolution of the front is monotone we have an interesting link with the minimum time problem.

The minimum time problem

Let us consider the nonlinear system

$$\begin{cases} \dot{y}(t) = f(y(t), a(t)), \quad t > 0, \\ y(0) = x \end{cases} \tag{D}$$

where
$y(t) \in \mathbb{R}^N$ is the state
$a(\cdot) \in \mathcal{A}$ is the control

$$\begin{aligned} \mathcal{A} &= \text{admissible control functions} \\ &= \{\, a : [0, +\infty[\,\to A, \ \text{measurable} \,\} \end{aligned}$$

(e.g. $\mathcal{A}$ = piecewise constant functions with values in A),

The model problem

$A \subset \mathbb{R}^M$ is a given compact set.
Assume f is continuous and

$$|f(x,a) - f(y,a)| \leq L\,|x-y| \quad \forall x,y \in \mathbb{R}^N,\ a \in A.$$

Then, for each $a(\cdot) \in \mathcal{A}$, there is a unique trajectory of (D), $y_x(t;a,b)$ (Caratheodory Theorem).

Payoff

The *payoff* of the Minimum Time Problem is

$$t_x(a(\cdot)) = \min\{\, t : y_x(t;a) \in \mathcal{T}\,\} \leq +\infty,$$

where $\mathcal{T} \subseteq \mathbb{R}^N$ is a given closed target.

Goal

We want to minimize the payoff, i.e. the time to transfer the system from its initial position x to the target $\mathcal{T}$. The value function is

$$T(x) \equiv \inf_{a(\cdot)\in\mathcal{A}} t_x(a)\ .$$

(T is the minimum time function).

Example 1: the distance function

This is a very simple example, take

$$\begin{cases} \dot{y} = a, \\ y(0) = x. \end{cases} \qquad A = \{\, a \in \mathbb{R}^N : |a| = 1 \,\},$$

Then, $t_x(a^*)$ is equal to the length of the optimal trajectory joining x and the point $y_x(t_x(a^*))$, thus

$$t_x(a^*) = \min_{a \in \mathcal{A}} t_x(a) = \mathrm{dist}(x, \mathcal{T})$$

and any optimal trajectory is a straight line!

Reachable set

Definition

$\mathcal{R} \equiv \{ x \in \mathbb{R}^N : T(x) < +\infty \}$, *i.e. the set of starting points from which it is possible to reach the target (a priori this can be empty).*

WARNING

The reachable set $\mathcal{R}$ depends on the target and on the dynamics in a rather complicated way. It is *NOT* a datum in our problem.

Dynamic Programming

Lemma (Dynamic Programming Principle)

For all $x \in \mathcal{R}$, $0 \le t < T(x)$ *(so that* $x \notin \mathcal{T}$*),*

$$T(x) = \inf_{a(\cdot)\in\mathcal{A}} \{\, t + T(y_x(t;a))\,\} . \qquad \text{(DPP)}$$

"Proof"

The inequality "$\le$" follows from the intuitive fact that $\forall a(\cdot)$

$$T(x) \le t + T(y_x(t;a)).$$

DP Principle

The proof of the opposite inequality "$\geq$" is based on the fact that the equality holds if $a(\cdot)$ is optimal for x.
For any $\varepsilon > 0$ we can find a minimizing control a_ε such that

$$T(x) + \varepsilon \geq t + T(y_x(t; a_\varepsilon))$$

split the trajectory and pass to the limit for $\varepsilon \to 0$.

Sketch of the proof

To prove rigorously the above inequalities the following two properties of $\mathcal{A}$ are crucial:

1. $a(\cdot) \in \mathcal{A} \Rightarrow \forall s \in \mathbb{R}$ the function $t \mapsto a(t+s)$ is in $\mathcal{A}$;
2. $a_1, a_2 \in \mathcal{A} \Rightarrow a(\cdot) \in \mathcal{A}\ \forall s > 0$, where

$$a(t) \equiv \begin{cases} a_1(t) & t \leq s, \\ a_2(t) & t > s. \end{cases}$$

Note that the DPP works for

$$\mathcal{A} = \{\, \text{piecewise constants functions into } A \,\}$$

but not for

$$\mathcal{A} = \{\, \text{continuous functions into } A \,\}.$$

because joining together two continuous controls we are not guaranteed that the resulting control is continuous.

Bellman equation

Let us derive the Hamilton-Jacobi-Bellman equation from the DPP. Rewrite (DPP) as

$$T(x) - \inf_{a(\cdot)} T(y_x(t;a)) = t$$

and divide by $t > 0$,

$$\sup_{a(\cdot)} \left\{ \frac{T(x) - T(y_x(t;a))}{t} \right\} = 1 \qquad \forall t < T(x) \; .$$

We want to pass to the limit as $t \to 0^+$.

Bellman equation

Assume T is differentiable at x and $\lim_{t\to 0^+}$ commute with $\sup_{a(\cdot)}$. Then, if $\dot{y}_x(0;a)$ exists,

$$\sup_{a(\cdot)\in\mathcal{A}} \{ -\nabla T(x) \cdot \dot{y}_x(0,a) \} = 1,$$

and then, if $\lim\limits_{t\to 0^+} a(t) = a_0$, we get

$$\sup_{a_0 \in A} \{ -\nabla T(x) \cdot f(x, a_0) \} = 1 \,. \qquad \text{(HJB)}$$

This is the Hamilton-Jacobi-Bellman partial differential equation (first order, fully nonlinear PDE).

Bellman equation

Let us define the Hamiltonian,

$$H_1(x,p) := \max_{a \in A}\{ -p \cdot f(x,a) \} - 1,$$

we can rewrite (HJB) in short as

$$H(x, \nabla T(x)) = 0 \text{ in } \mathcal{R} \setminus \mathcal{T}.$$

A natural *boundary condition* on $\partial\mathcal{T}$ is

$$T(x) = 0 \qquad \text{for } x \in \partial\mathcal{T}$$

T verifies the HJB equation

Proposition

If $T(\cdot)$ is C^1 in a neighborhood of $x \in \mathcal{R} \setminus \mathcal{T}$, then $T(\cdot)$ satisfies (HJB) at x.

Proof

We first prove the inequality "$\leq$".

Fix $\bar{a}(t) \equiv a_0 \ \forall t$, and set $y_x(t) = y_x(t; \bar{a})$. (DPP) gives

$$T(x) - T(y_x(t)) \leq t \qquad \forall \ 0 \leq t < T(x).$$

We divide by $t > 0$ and let $t \to 0^+$ to get

$$-\nabla T(x) \cdot \dot{y}_x(0) \leq 1,$$

where $\dot{y}_x(0) = f(x, a_0)$ (since $\bar{a}(t) \equiv a_0$).

T verifies the HJB equation

Then,

$$-\nabla T(x) \cdot f(x, a_0) \leq 1 \qquad \forall a_0 \in A$$

and we get

$$\max_{a \in A} \{ -\nabla T(x) \cdot f(x, a) \} \leq 1 \, .$$

Next we prove the inequality "$\geq$".

T verifies the HJB equation

Fix $\varepsilon > 0$. For all $t \in]0, T(x)[$, by DPP there exists $\alpha \in \mathcal{A}$ such that

$$T(x) \geq t + T(y_x(t;\alpha)) - \varepsilon t \ .$$

Then

$$\begin{aligned}
1 - \varepsilon \leq & \frac{T(x) - T(y_x(t;\alpha))}{t} \\
& = -\frac{1}{t}\int_0^t \frac{\partial}{\partial s} T(y_x(s;\alpha))\, ds = -\frac{1}{t}\int_0^t \nabla T(y_x(s)) \cdot \dot{y}_x(s)\, ds \\
& = -\frac{1}{t}\int_0^t \nabla T(x) \cdot f(x, \alpha(s))\, ds + o(1) \quad \text{as } s \to 0^+ \\
& \leq \sup_{a \in A}\{ -\nabla T(x) \cdot f(x,a) \} + o(1) \ .
\end{aligned}$$

T verifies the HJB equation

Letting $s \to 0^+$, $\varepsilon \to 0^+$ we get

$$\sup_{a \in A}\{ -\nabla T(x) \cdot f(x,a) \} \geq 1 \, .$$

$\square$

We have proved that if T is regular then it satisfies (pointwise) the Bellman equation in the reachable set $\mathcal{R}$

Is T regular?

The answer is NO even for simple cases.

Let us go back to Example 1 where $T(x) = \text{dist}(x, \mathcal{T})$. Note that T is not differentiable at x if there exist two distinct points of minimal distance.
Let us take $N = 1$, $f(x, a) = a$, $A = B(0, 1)$ and choose

$$\mathcal{T} =]-\infty, -1] \cup [1, +\infty[\ .$$

Then,

$$T(x) = 1 - |x|$$

which is *not* differentiable at $x = 0$.
The correct framework to find a unique Lipschitz continuous solution is to look for a viscosity solution (see Barles book).

Evolutive vs Stationary Models

We are now able to state the link between the solution of the evolutive HJ equation connected to the Level Set Method and the minimum time

Theorem

Let T be the minimum time problem where

$$\mathcal{T} = \Omega_0, \quad f(x,a) \equiv -c(x)a, \quad a \in A \equiv B(0,1)$$

then the viscosity solution of the evolutive problem (1) is $u(x,t) = T(x) - t$.

This allows to replace the evolutive problem by a stationary problem and opens the way to the FMM.

Outline

1. Introduction
2. Front Propagation and Minimum Time Problem
3. Global Numerical Approximations
4. Fast Marching Methods
   - A CFL-like condition for FM method

A DP Scheme for the Minimum Time Problem

Let us examine a discretization procedure which is based on Dynamic Programming.

FD approximation

It is interesting to note that the up-wind corrected FD approximation of the corresponding equation is a special case of the DP discretization corresponding to a structured uniform grid and to the choice of only 4 control directions (N, S, E, W).

Time Discretization

By applying the change of variable (Kružkov)

$$v(x) = 1 - e^{-T(x)}$$

and rewrite the equation in the new variable

$$v(x) + \sup_{a \in A} [-f(x,a) \cdot \nabla v] = 1 \qquad \text{(HJ)}$$

$$v(x) = 0 \text{ on } \mathcal{T}$$
$$v(x) = 1 \text{ on } \partial\mathcal{R}$$

As seen, we can drop the second boundary condition.

Time discretization

Time step $h > 0$
Discrete times $t_j = jh$, $j \in \mathbb{N}$
Discrete dynamical system

$$\begin{cases} x_{j+1} = x_j + hf(x_j, a_j) \\ x_0 = x \end{cases}$$

We define

$$\mathcal{R}_h \equiv \{x \in \mathbb{R}^N : \exists \{a_j\} \text{ and } j \in \mathbb{N} \text{ such that } x_j \in \mathcal{T}\}$$

Discrete Minimum Time Function

Let us define

$$n_h(\{a_j\}, x) = \begin{cases} +\infty & x \notin \mathcal{R}_h \\ \min\{\in \mathbb{N} : x_j \in \mathcal{T}\} & \forall\, x \in \mathcal{R}_h \end{cases}$$

and

$$N_h(x) = \min_{\{a_j\}} n_h(\{a_j\}, x)$$

The discrete analogue of the minimum time function is $N_h(x)h$.

The discrete Bellman equation

We change the variable

$$v_h(x) = 1 - e^{-h\,N_h(x)}$$

Note that $0 \le v_h \le 1$.
By the Discrete Dynamic Programming Principle we get

$$v_h(x) = S(v_h)(x) \qquad \text{on } \mathcal{R}_h \setminus \mathcal{T}. \qquad (HJ_h)$$

$$S(v_h)(x) \equiv \min_{a \in A} \left[e^{-h} v_h(x + hf(x,a)) \right] + 1 - e^{-h}$$

$$v_h(x) = 0 \qquad \text{on } \mathcal{T} \qquad \text{(BC)}$$

Characterization of v_h

Since $x \in \mathbb{R}^N \setminus \mathcal{R}_h \Rightarrow x + hf(x,a) \in \mathbb{R}^N \setminus \mathcal{R}_h$ we can extend v_h to $\mathbb{R}^N$ setting

$$v_h(x) = 1 \qquad \text{on } \mathbb{R}^N \setminus \mathcal{R}_h \; .$$

Theorem

v_h is the unique bounded solution of $(HJ_h) - (BC)$.

Local Controllability

Assumptions on $\mathcal{T}$:

- (i) $\mathcal{T} \equiv \{x : g_i(x) \leq 0 \quad \forall\, i = 1, \ldots, M\}$ where $g_i \in C^2(\mathbb{R}^N)$ and $|\nabla g_i(x)| > 0$ for any x such that $g_i(x) = 0$.
- (ii) $\forall\, x \in \mathcal{T}\ \exists\, a \in A$ such that $g_i(x) = 0$ implies

$$f(x, a) \cdot \nabla g_i(x) < 0.$$

Bounds

$$\text{Let } \mathcal{T}_\delta \equiv \partial\mathcal{T} + \delta B \ , \qquad d(x) \equiv \text{ dist } (x, \partial\mathcal{T})$$

Lemma

Under our assumptions on f and local controllability, there exist some positive constants $\overline{h}$, δ, δ' such that

$$h\, N_h(x) \leq C\, d(x) + h \ , \qquad \forall\, h < \overline{h} \ , \quad x \in \mathcal{T}_\delta$$

and

$$T(x) \leq c\, d(x) \ , \qquad \forall\, x \in \mathcal{T}_{\delta'} \ .$$

Convergence

Theorem

Let the assumptions of the Lemma be satisfied and let $\mathcal{T}$ be compact with nonempty interiour.
Then, for $h \to 0^+$

$$v_h \to v \text{ locally uniformly in } \mathbb{R}^N$$

$$h\, N_h \to T \text{ locally uniformly in } \mathcal{R}$$

.

Error estimate

Let us assume Q is a compact subset of $\mathcal{R}$ where the following condition holds:

$$\exists\ C_0 > 0:\ \forall x \in Q \text{ there is a time optimal control with total variation less than } C_0 \text{ bringing the system to } \mathcal{T}. \qquad \text{(BV)}$$

Theorem

Let the assumptions of Theorem 7 hold true and let Q be a compact subset of $\mathcal{R}$ where (BV) holds. Then $\exists\, \overline{h}, C > 0$:

$$|T(x) - h\, N_h(x)| \leq Ch\,, \qquad \forall x \in Q, \quad \forall h \leq \overline{h}$$

Sketch of the Proof

1. Our assumptions imply that
   T is continuous on $\partial\mathcal{T}_n$ and V is continuous in $\mathbb{R}^N$
2. $h\,N_h(x) \leq d(x) + h\,, \quad \forall\, x \in \mathcal{T}_\delta$
3. $T(x) \leq C\,d(x)$.

That implies $T(x) - h\,N_h(x) \leq Ch$
Finally, (BV) implies $h\,N_h(x) - T(x) \leq Ch$. □

First order scheme

Corollary

Under the same hypotheses there exists two positive constants $\overline{h}$ and C:

$$|v(x) - v_h(x)| \leq Ch \qquad \forall\, x \in Q, h \leq \overline{h} \qquad \text{(E)}$$

The rate of convergence of the approximation scheme is 1.

Space Discretization

We build a triangulation of a rectangle Q in $\mathbb{R}^2$, $Q \supset \mathcal{T}$.

$$
\begin{aligned}
x_i =& \quad \text{nodes of the grid} \\
L =& \quad \# \text{ of nodes} \\
I_{\mathcal{T}} \equiv& \{i \in \mathbb{N} : x_i \in \mathcal{T}\} \\
I_{out} \equiv& \{i \in \mathbb{N} : x_i + hf(x_i, a) \notin Q, \forall\, a\} \\
I_{in} \equiv& \{i \in \mathbb{N} : x_i + hf(x_i, a) \in Q\} \\
k \equiv& \text{ max diameter of the cells (or triangles)}
\end{aligned}
$$

Fully discrete scheme

We want to solve

$$\begin{aligned} v(x_i) &= \min_{a\in A}[\beta v(x_i + hf(x_i,a))] + 1 - \beta, \quad \forall x_i \in I_{in} \\ v(x_i) &= 0 \qquad \forall x_i \in I_{\mathcal{T}} \\ v(x_i) &= 1 \qquad \forall x_i \in I_{out} \end{aligned}$$

in the space

$$W^k \equiv \{w : Q \to [0,1] : w \text{ is continuous in } Q \\ \nabla w = const, \text{ in } S_j\}$$

Fixed Point Problem

For any $i \in I_{\text{in}}$, $x_i + hf(x_i, a) \in Q$ there exists a vector of coefficients,$\lambda_{ij}(a)$:

$$0 \leq \lambda_{ij}(a) \leq 1$$

$$x_i + hf(x_i, a) = \sum_{j=1}^{L} \lambda_{ij}(a) x_j$$

$$\sum_{j=1}^{L} \lambda_{ij}(a) = 1$$

Fixed Point Problem

We define the operator $S : \mathbb{R}^L \to \mathbb{R}^L$

$$[S(U)]_i \equiv \begin{cases} \min_{a \in A}[\beta \Lambda_i(a) U] + 1 - \beta \ , & \forall\, i \in I_{in} \\ 0 & \forall\, i \in I_{\mathcal{T}} \\ 1 & \forall\, i \in I_{\text{out}} \end{cases}$$

$$S : [0,1]^L \to [0,1]^L$$

has a unique fixed point.

Properties of the scheme S

Theorem

$S : [0,1]^L \to [0,1]^L$ and

$$\|S(U) - S(V)\|_\infty \leq \beta \|U - V\|_\infty$$

Sketch of the proof. S is monotone, i.e.

$$U \leq V \Rightarrow S(U) \leq S(V)$$

Then, for any $U \in [0,1]^L$

$$1 - \beta = S_i(0) \leq S_i(U) \leq S_i(\mathbf{1}) = 1, \qquad \forall\, i \in I_{in}$$

where $\mathbf{1} \equiv (1, 1, \dots, 1)$. This implies, $S : [0,1]^L \to [0,1]^L$

S is a contraction

For any $i \in I_{in}$

$$S_i(U) - S_i(V) \leq \beta \Lambda_i(\hat{a})(U - V)$$

and $\|\Lambda_i(a)\| \leq 1$, $\forall\, a \in A$ which implies

$$\|S_i(U) - S(V)\|_\infty \leq \beta \|U - V\|_\infty.$$

□

Monotone convergence

We choose $U^0 \in [0,1]^L$

$$U_i^0 = \begin{cases} 0 & \forall\, i \in I_{\mathcal{T}} \\ 1 & elsewhere \end{cases}$$

$$U^0 \in U^+ \equiv \{U \in [0,1]^L : U \geq S(U)\}$$

Monotone convergence

By the monotonicity of S the sequence

$$\begin{aligned} &U^0 \\ &U^{n+1} \equiv S(U^n) \end{aligned}$$

is monotone decreasing, at least when a sufficiently coarse grid is chosen, and

$$U^n \searrow U^*$$

by the fixed point argument.

How the information flows

The information flows from the target to the other nodes of the grid.
In fact, on the nodes in $Q \setminus \mathcal{T}$, $U_i^0 = 1$. But starting from the first step of the algorithm their value immediately decreases since, by the local controllability assumption, the Euler scheme drives them to the target.
This is the most important fact will allows for a local implementation of the scheme and to its FM version

Outline

Front Propagation Problem

The main idea of Fast Marching method is based on the front propagation point of view. Let $\partial\Omega$ be a closed curve (the front) in $\mathbb{R}^2$ and suppose that each of its point moves in the normal direction with speed $c(x)$.

Front Propagation Problem

Then, the evolution of the front at every time is given by the level sets of the function $T(x)$ solution of the

Eikonal equation

$$\begin{cases} c(x)|\nabla T(x)| = 1 & x \in \mathbb{R}^n \backslash \Omega \\ T(x) = 0 & x \in \partial\Omega \end{cases} \qquad (4)$$

$T(x)$ is the arrival time of the front at x.

The FD discretization

Let us write equation (4) as

$$T_x^2 + T_y^2 = \frac{1}{c^2(x,y)}.$$

The standard up-wind first order FD approximation is

$$\left(\max\left\{ \max\left\{ \frac{T_{i,j} - T_{i-1,j}}{\Delta x}, 0 \right\}, -\min\left\{ \frac{T_{i+1,j} - T_{i,j}}{\Delta x}, 0 \right\} \right\} \right)^2 +$$

$$+ \left(\max\left\{ \max\left\{ \frac{T_{i,j} - T_{i,j-1}}{\Delta y}, 0 \right\}, -\min\left\{ \frac{T_{i,j+1} - T_{i,j}}{\Delta y}, 0 \right\} \right\} \right)^2 = \frac{1}{c_{i,j}^2}$$

where, as usual, $T_{i,j} = T(x_i, y_j)$.

The Fast Marching method

The FM method was introduced by J. A. Sethian in 1996 [S96]. It is an acceleration method for the classical iterative FD scheme for the eikonal equation.

Main Idea (Tsitsiklis (1995), Sethian (1996))

Processing the nodes in a special ordering one can compute the solution in just 1 iteration.
This special ordering corresponds to the increasing values of T.

The FMM is able to find the ordering corresponding to the increasing values of (the unknown) T, while computing. This is done introducing a *NARROW BAND* which locates the front.

The Fast Marching method

The FM method was introduced by J. A. Sethian in 1996 [S96]. It is an acceleration method for the classical iterative FD scheme for the eikonal equation.

Main Idea (Tsitsiklis (1995), Sethian (1996))

Processing the nodes in a special ordering one can compute the solution in just 1 iteration.
This special ordering corresponds to the increasing values of T.

The FMM is able to find the ordering corresponding to the increasing values of (the unknown) T,while computing. This is done introducing a *NARROW BAND* which locates the front.

The Fast Marching method

The FM method was introduced by J. A. Sethian in 1996 [S96]. It is an acceleration method for the classical iterative FD scheme for the eikonal equation.

Main Idea (Tsitsiklis (1995), Sethian (1996))

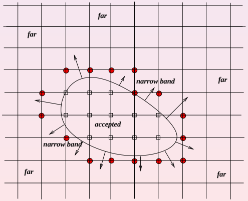
Processing the nodes in a special ordering one can compute the solution in just 1 iteration.
This special ordering corresponds to the increasing values of T.

The FMM is able to find the ordering corresponding to the increasing values of (the unknown) T,while computing. This is done introducing a *NARROW BAND* which locates the front.

The Fast Marching method

Just the nodes in the NB are computed at each step.
When the NB has passed through the whole grid, the algorithm ends.

The computation of T in every node is performed using the FD-discretization mentioned above.



Fast Marching method... marching

In the movie you can see the FM method at work.
Once a node is computed, a red spot turns on.

Front Representation on the Grid

Let us consider the simple case of a structured uniform grid in $\mathbb{R}^2$,

$$Z \equiv \{x_{ij} : x_{ij} = (x_i, y_j), x_i = i\Delta x, y_j = j\Delta y\}$$

we compute on $Q \cap Z$ where $\Omega_0 \subset Q$.
We set

- $T_{ij} = 0$ for every $x_{ij} \in \Omega_0$
- $T_{ij} = +\infty$ elsewhere

The Basic Local Rule

Let us consider the simple case of a structured uniform grid in $\mathbb{R}^2$,

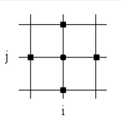
$$\left(\max\left\{\max\left\{\frac{T_{i,j}-T_{i-1,j}}{\Delta x},0\right\},-\min\left\{\frac{T_{i+1,j}-T_{i,j}}{\Delta x},0\right\}\right\}\right)^2+$$

$$+\left(\max\left\{\max\left\{\frac{T_{i,j}-T_{i,j-1}}{\Delta y},0\right\},-\min\left\{\frac{T_{i,j+1}-T_{i,j}}{\Delta y},0\right\}\right\}\right)^2=\frac{1}{c_{i,j}^2}$$

The Basic Local Rule

The points involved in this formula are the stencil of the scheme and they are the "first neighbors" of the node where we are computing, i.e.

$$N_{FD}(x_{ij}) = \{x_{i+1,j}, x_{i,j-1}, x_{i-1,j}, x_{i,j+1}\}$$



Partitioning the nodes

At every iteration (but the last one) we will have three sets of nodes:

- the Accepted nodes, where the values has been already computed and fixed
- the Narrow Band nodes, where the algorithm is computing
- the Far nodes, where the algorithm will compute in the next iterations

We will denote by $A(k)$, $NB(k)$ and $F(k)$ these subsets at the k-th iteration.

Definition

At the iteration k the $NB(k)$ is the set of nodes which are first neighborsöf the nodes in the Accepted region of the previous iteration, i.e. $A(k-1)$.

Initialization

Let us consider the simple case of a structured uniform grid in $\mathbb{R}^2$,

$$Z \equiv \{x_{ij} : x_{ij} = (x_i, y_j), x_i = i\Delta x, y_j = j\Delta y\}$$

we compute on $Q \cap Z$ where $\Omega_0 \subset Q$.
We set

- $T_{ij} = 0$ for every $x_{ij} \in \Omega_0$
- $T_{ij} = +\infty$ elsewhere

FMM Algorithm

The algorithm step-by-step, initialization

1. The nodes belonging to the initial front Γ_0 are located and labeled as *Accepted*. They form the set $\widetilde{\Gamma}_0$. The value of T of these nodes is set to 0.
2. $NB(1)$ is defined as the set of the nodes belonging to $N_{FD}(\widetilde{\Gamma}_0)$, external to Γ_0.
3. Set $T_{i,j} = +\infty$ for any $(i,j) \in NB(1)$.
4. The remaining nodes are labeled as *Far*, their value is set to $T = +\infty$.

FMM Algorithm

The algorithm step-by-step, main cycle

Repeat

1. Compute $T_{i,j}$ by the FD scheme on $NB(k)$
2. Find the minimum value of $T_{i,j}$ in $NB(k)$.
3. Label (i,j) as Accepted, i.e. $A(k+1) = A(k) \cup \{x_{ij}\}$
4. Remove (i,j) from $NB(k)$.
5. Up-date the $NB(k)$ to $NB(k+1)$adding the first neighbors of the NEW accepted node.
6. Set k=k+1

Until ALL the nodes have been accepted.

FMM Algorithm

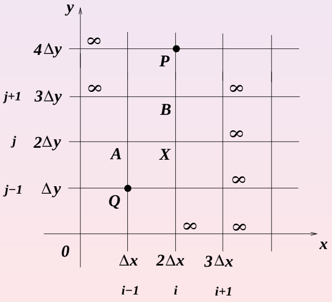
On a finite grid this give the solution after a finite number of operations, the solution coincides with the numerical solution of the global scheme.

How much it costs ?

We have to compute the solution at every point at most 4 times and we have to search for the minimum in NB at every iteration. Using a heap-sort method this search costs $O(\ln(N_{nb}))$. The global cost is dominated by $O(N\ln(N))$ (N represents the total number of nodes in the grid).

Problems with FM technique

However, there are examples which shows that the FM technique is not always suited to approximate the solution of (4). In fact it can produce complex solutions if no compatibility condition is introduced between the velocity and Δx !



Convergence of FM method

Theorem

Let the following assumptions hold true:

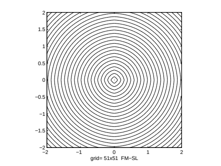
- $c \in Lip(\mathbb{R}^n) \cap L^\infty(\mathbb{R}^n)$
- $c(x) > 0$
- $\Delta x \leq (\sqrt{2} - 1)\frac{c_{min}}{L_c}$ where $c_{min} = \min_{\mathbb{R}^n \setminus \Omega} c(x)$ and L_c is the Lipschitz constant of c.

Then, FD-FM method computes a *real* approximate solution of (4). Moreover, this solution is exactly the same solution of the FD classical iterative scheme.

TEST 1. Numerical comparison

Distance from the origin. $\Omega = \{(0,0)\}\,, \quad c(x,y) \equiv 1$

Exact solution: $T(x,y) = \sqrt{(x^2+y^2)}$



TEST 1. Numerical comparison

Distance from the origin. $\Omega = \{(0,0)\}\,, \quad c(x,y) \equiv 1$

Exact solution: $T(x,y) = \sqrt{(x^2+y^2)}$

method	Δx	L^∞ error	L^1 error	CPU time (sec)
SL (46 its.)	0.08	0.0329	0.3757	8.4
FM-FD	0.08	0.0875	0.7807	0.5
FM-SL	0.08	0.0329	0.3757	0.7

TEST 1. Numerical comparison

Distance from the origin. $\Omega = \{(0,0)\}$, $\quad c(x,y) \equiv 1$

Exact solution: $T(x,y) = \sqrt{(x^2 + y^2)}$

method	Δx	L^∞ error	L^1 error	CPU time (sec)
SL (46 its.)	0.08	**0.0329**	**0.3757**	**8.4**
FM-FD	0.08	0.0875	0.7807	0.5
FM-SL	0.08	**0.0329**	**0.3757**	**0.7**

TEST 1. Numerical comparison

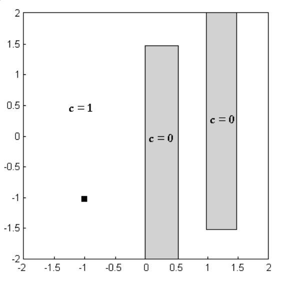
Distance from the origin. $\Omega = \{(0,0)\}\,, \quad c(x,y) \equiv 1$

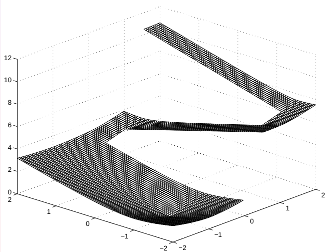
Exact solution: $T(x,y) = \sqrt{(x^2 + y^2)}$

method	Δx	L^∞ error	L^1 error	CPU time (sec)
SL (46 its.)	0.08	0.0329	0.3757	8.4
FM-FD	0.08	**0.0875**	**0.7807**	**0.5**
FM-SL	0.08	**0.0329**	**0.3757**	**0.7**

TEST 2. Minimum time problem

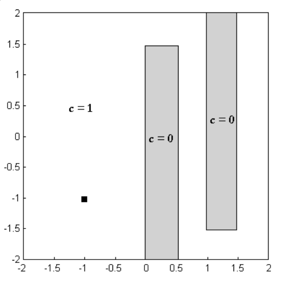
Presence of obstacles

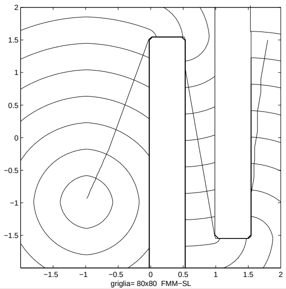




TEST 2. Minimum time problem

Presence of obstacles





Some references

[1] E. Cristiani, M. Falcone, *Fast semi-Lagrangian schemes for the Eikonal equation and applications*, SIAM J. Numer. Anal. (2007), .

[2] J. A. Sethian, *A fast marching level set method for monotonically advancing fronts*, Proc. Natl. Acad. Sci. USA, **93** (1996), 1591-1595.

[3] J.A. Sethian, *Fast Marching Methods*, SIAM Review, No.2 **41**, 199–235 (1999)

[4] J. A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*, Cambridge University Press, 1999.

[5] J. A. Sethian, A. Vladimirsky, *Ordered upwind methods for static Hamilton-Jacobi equations: theory and algorithms*, SIAM J. Numer. Anal., 41 (2003), pp. 325–363.

[6] J. N. Tsitsiklis, *Efficient algorithms for globally optimal trajectories*, IEEE Tran. Automatic. Control, **40** (1995), 1528-1538.

[7] A. Vladimirsky, *Static PDEs for time-dependent control problems*, Interfaces and Free Boundaries **8/3** (2006), 281-300.

Coming soon

- Sweeping
- Group Marching
- Extensions to more general problems and equations
- The case when $c(x, t)$ can change sign (Lecture 3)