

# A CFL-Like Constraint for the Fast Marching Method in Inhomogeneous Chemical Kinetics

RAMÓN ESCOBEDO

*Departamento de Matemática Aplicada y Ciencias de la Computación, E.T.S.I. de Industriales y de Telecomunicación, Universidad de Cantabria, Av. de Los Castros s/n, Santander 39005, Spain*

*Received 20 July 2007; accepted 20 September 2007*

*Published online 29 November 2007 in Wiley InterScience (www.interscience.wiley.com).*

*DOI 10.1002/qua.21554*

**ABSTRACT:** Level sets and fast marching methods are a widely used technique for problems with moving interfaces. Chemical kinetics has been recently added to this family, for the description of reaction paths and chemical waves in homogeneous media, in which the velocity of the interface is described by a given field. A more general framework must consider variable velocities due to inhomogeneities induced by chemical changes. In this case, a constraint must be satisfied for the correct use of fast marching method. We deduce an analytical expression of this constraint when the Godunov scheme is used to solve the Eikonal equation, and we present numerical simulations of a case which must be enforced to obey the constraint. © 2007 Wiley Periodicals, Inc. *Int J Quantum Chem* 108: 848–857, 2008

**Key words:** chemical kinetics; fast marching method; godunov scheme

## 1. Introduction

Recently, the fast marching level set method of Sethian [1,2] has been widely used in chemical kinetics to solve fundamental problems such as finding reaction paths [3,4], calculating reaction trajectories [5] and computing tunneling paths [6], and has also been taken into account to construct geometrical methods to simulate travelling waves in inhomogeneous chemical media [7]. All these problems are recognized as fundamental challenges in Chemistry.

*Correspondence to:* R. Escobedo; e-mail: escobedo@unican.es

Fast marching methods (FMM) are especially suitable methods for tracking interfaces whose velocity is defined by a function  $F(\mathbf{x}, t)$ ,  $(\mathbf{x}, t) \in \mathbb{R}^n \times [0, +\infty)$ , which does not change sign, that is,  $F(\mathbf{x}, t) > 0$  (or  $F(\mathbf{x}, t) < 0$ ) for all  $(\mathbf{x}, t)$ . Typical examples are the propagation of an action potential along the axon of a nerve, the propagation of a grass fire on a prairie, pulse propagation through cardiac cells, reaction-diffusion or ecological systems, and pulse recycling and motion in semiconductor devices. See [8] and references therein.

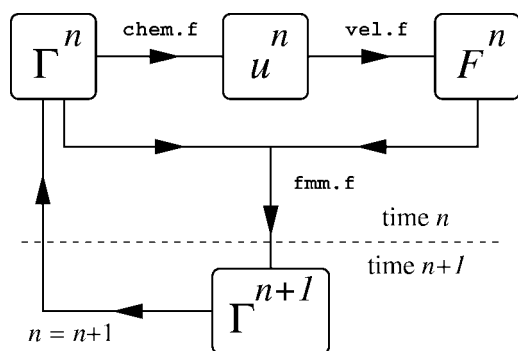
In the cited chemical works, the function  $F$  is a given field which remains constant along the problem, assuming an homogeneous or chemically isotropic medium. For instance,  $F$  is used as the

inverse of a cost function for travelling along a dynamical reaction or potential energy paths [3,4], or the inverse of  $\sqrt{2(E-V)}$ , where  $E$  and  $V$  are, respectively, the total and potential energy of the reactants [5,6].

Unfortunately, in the general case, the velocity field is not a given data of the problem and it often depends on the position of the reaction interface itself. This means that the values of  $F$  at a given time must be calculated by solving a (system of) partial differential equation(s) (PDEs) simultaneously with the interface evolution problem. In this more general framework, the FMM is coupled to a static chemical problem which gives the instantaneous velocity field, and a Courant-Friedrichs-Lewy (CFL)-like constraint for  $F$  must be taken into account.

In inhomogeneous or anisotropic chemical media, the mathematical model usually consists in describing the evolution of a chemical magnitude  $u(x, y, t)$  (e.g., a chemical substance, its concentration or its mass fraction) in a 2D or 3D domain  $\Omega$ . To do this, one must be able to calculate the magnitude  $u(x, y, t)$  for a given position of the interface  $\Gamma^n$  at a given time  $t^n$ , and the velocity  $F^n$  of (at least) all the points of  $\Gamma^n$  at time  $t^n$ . Then, the FMM can obtain the new position of the interface after a short interval of time  $\Delta t$ . It is then possible to start again for  $t^{n+1} = t^n + \Delta t$ . See this algorithm in Figure 1.

The algorithm consists of two parts. The first part is the *chemical* part, where the solution  $u^n$  and the velocity  $F^n$  are calculated in the domain  $\Omega^n$  characterized by  $\Gamma^n$ , which is evolving in time. This part is done by solving the PDEs system with a main routine called `chem.f` in Figure 1, and by calculating the velocity with a second routine `vel.f`.



**FIGURE 1.** General algorithm for a moving front problem, consisting of a static chemical part (`chem.f` + `vel.f`) and a time-evolving geometrical part (`fmm.f`).

The second part of the algorithm is the *geometric* part, which consists in actualizing the geometry of the problem to the next time step by moving the interface to its new position  $\Gamma^{n+1}$ . This part is carried out by the FMM routine, called `fmm.f` in Figure 1.

In inhomogeneous problems, the velocity  $F$  must be renewed after each interval of time  $\Delta t$  by solving the chemical problem assuming a fixed interface. The consistency of the general algorithm is based on the assumption that  $\Delta t$  is short enough to ensure that the velocity field remains almost constant during this interval of time.

This article presents the derivation of an explicit expression which acts as a CFL-like constraint for the renewal of  $F$  and imposes a restriction for the FMM, namely in the Godunov method, which is the one used in [3–6].

## 2. The Fast Marching Method

### 2.1. SETUP OF THE ALGORITHM

Let  $\{(x_i, y_j)\}_{i=1, j=1}^{N_x, N_y}$  be a rectangular space discretization of a two-dimensional domain  $\Omega$ . At time  $t^n$ , the interface  $\Gamma^n$  is given by a set of  $N_l^n$  nodes of  $\Omega$ :

$$\Gamma^n = \{(x_i^n, y_l^n)\}_{l=1}^{N_l^n}. \quad (1)$$

Assume that  $F^n \geq 0$  for all  $t \geq 0$  (recall that the velocity does not change sign). The idea of the FMM consists in describing the evolution of the interface (also called *front* or *free boundary*) by means of an *arrival time* function  $\phi(x, y)$  defined as the *time it takes to the interface to arrive to the point*  $(x, y)$ .

At time  $t^n$ , the interface is given by the points such that

$$\phi^n(x, y) = t^n. \quad (2)$$

Starting from these points, the FMM provides the arrival time  $\phi$  for  $t \geq t^n$ . At the end of the  $n$ -FMM step, the arrival time is  $t^{n+1}$  and the position of the interface is given by the points in  $\Omega$  such that

$$\phi^n(x, y) = t^{n+1}. \quad (3)$$

In principle,  $\phi^n(x, y)$  can be constructed beyond  $t^{n+1}$ , but the validity of this calculation is subject to physical or chemical restrictions imposed by  $F^n$ , especially in inhomogeneous media; in this case,  $\phi^n(x, y)$  must often be reconstructed after each  $\Delta t$ .

Using the classical notation introduced by Sethian, three disjoint sets can be defined to classify

## ESCOBEDO

each point of  $\Omega$  in terms of the history of the position of the front: (for clearness, the index  $n$  has been omitted)

$$A = \{(x, y) \in \Omega / \phi(x, y) \leq 0\};$$

the front is or has been here;

$$C = \{(x, y) \in \Omega / \phi(x, y) > 0$$

and **at least one** of its neighbors is in A};

$$F = \{(x, y) \in \Omega / \phi(x, y) > 0$$

and **none** of its neighbors is in A}.

The set  $A$  is called the set of "Accepted points," and  $C$  and  $F$  are, respectively, the sets of "Close" and "Far" points with respect to their distance to  $\Gamma$ .

Numerically, the interface  $\Gamma^{n+1}$  (at time  $t^n + \Delta t$ , using again the notation with index  $n$ ) is given by the set of points of  $\Omega$  such that

$$|\phi^n(x, y) - \Delta t| < \varepsilon, \quad (4)$$

where  $\varepsilon$  is a small tolerance of the width of the front that can be tuned to obtain a "one-point width" front. Then,  $\Gamma^n$  is approximated by the set  $C$ ; see Figure 2.

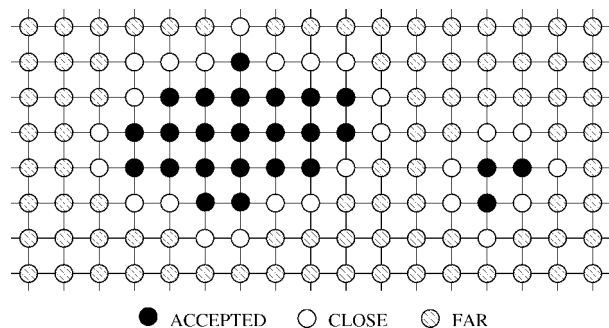
The level sets formulation consists in identifying the front  $\Gamma^n$ , which is a 2D collection of points of  $\Omega^n$ , with the zero level set of a 3D surface  $W$  in the 3D space  $\Omega^n \times [0, +\infty)$ , where time is used as a third dimension:

$$W(x, y, t) = 0. \quad (5)$$

The interface  $\Gamma^n$  at time  $t^n$  is then given by the set of points in  $\Omega$  such that

$$\Gamma^n = \{(x, y) \in \Omega^n / W(x, y, t^n) = 0\}. \quad (6)$$

The points  $(x, y)$  of the interface are depending on time, so let us consider  $x(t)$  and  $y(t)$  as variable



**FIGURE 2.** Accepted, close, and far points for a nonconnected interface.

in time. Then, the variation of  $W(x(t), y(t), t)$  with respect to time is given by the *total derivative* of Eq. (5) with respect to  $t$ :

$$\frac{\partial W}{\partial x} \frac{dx}{dt} + \frac{\partial W}{\partial y} \frac{dy}{dt} + \frac{\partial W}{\partial t} = 0, \quad (7)$$

$$\text{i.e. } \vec{\nabla} W \cdot (\dot{x}(t), \dot{y}(t)) + \frac{\partial W}{\partial t} = 0. \quad (8)$$

Here  $\partial/\partial t$  denotes partial derivative with respect to  $t$  (resp.  $x$  and  $y$ ),  $\vec{\nabla} W$  is used to denote the gradient of the scalar field  $W$ ,  $\vec{\nabla} W = (\partial W/\partial x, \partial W/\partial y)$ ,  $\dot{x}(t)$  denotes the variation in time of coordinate  $x$ , i.e., the velocity of change in coordinate  $x$ , and the central dot denotes the scalar product between vectors.

Let the vector  $\vec{N}$  be the unit outward directed normal to the zero level set:  $\vec{N} = \vec{\nabla} W / |\vec{\nabla} W|$ . The velocity  $F$  of a particle  $(x, y)$  in the direction  $\vec{N}$  is then

$$F(x, y, t) = (\dot{x}(t), \dot{y}(t)) \cdot \vec{N} = (\dot{x}(t), \dot{y}(t)) \cdot \frac{\vec{\nabla} W}{|\vec{\nabla} W|}. \quad (9)$$

Using this in Eq. (8) shows that the zero level set of  $W(x, y, t)$  evolves with a velocity  $F(x, y, t)$  according to the following Hamilton-Jacobi equation:

$$\frac{\partial W}{\partial t} + F|\vec{\nabla} W| = 0. \quad (10)$$

Let us now deduce an equation for  $\phi(x, y)$ . By definition, for all  $(x, y)$ ,  $\phi(x, y)$  denotes the time at which  $(x, y)$  is in the zero level set of  $W$ , so

$$\forall (x, y) \in \Omega, \quad W(x, y, \phi(x, y)) = 0. \quad (11)$$

Then, the total derivative of  $W(x, y, \phi(x, y))$  with respect to time is zero,

$$\left( \vec{\nabla} W + \frac{\partial W}{\partial t} \vec{\nabla} \phi \right) \cdot (\dot{x}, \dot{y}) = 0, \quad (12)$$

and this, for all  $(x, y)$ , which means that  $\vec{\nabla} W + \partial W/\partial t \vec{\nabla} \phi$  must be the null vector. Then, replacing  $\partial W/\partial t$  using the Hamilton-Jacobi Eq. (10) yields

$$\vec{\nabla} W = F|\vec{\nabla} W| \vec{\nabla} \phi, \quad (13)$$

which means that  $\vec{\nabla} W$  and  $\vec{\nabla} \phi$  are colinear vectors with modules related by

$$|\vec{\nabla} W| = F|\vec{\nabla} W||\vec{\nabla} \phi|. \quad (14)$$

Eliminating  $|\vec{\nabla}W|$  from each side gives finally the following Eikonal equation for the arrival time  $\phi$ , in terms of the outwards normal velocity  $F$ :

$$|\vec{\nabla}\phi| = \frac{1}{F}. \quad (15)$$

The Eq. (15) is the equation which is solved by the FMM. The effectiveness of the FMM lies in the fact that  $\phi$  is constructed in the upwind direction (i.e., from low to high values of  $\phi$ ), in order to guarantee the increasing evolution of  $\phi$ , according to the fact that the time it takes to the front to arrive to a point depends only on the history. Recall again that the velocity is always positive, so the interface can not move backwards.

The FMM step finishes as soon as a point  $(x, y)$  is found such that

$$\phi(x, y) > \Delta t + \varepsilon.$$

The new front is then given by the new set C, after what the algorithm returns to the chemical problem to obtain the solution and the new velocity at these points. For completeness, let us now describe the FMM step.

## 2.2. FMM STEP: FROM $t^n$ TO $t^{n+1}$ :

The FMM first initializes  $\phi(x, y)$  and then corrects its values by solving (15) with an iterative process. The initialization must be done at least for the first time step, and can be omitted in successive time steps

if no reinitialization of  $\phi$  is needed. The constraint we present here is in fact a test which can enforce this reinitialization.

### 2.2.1. FMM algorithm

- 1.1 At all points of A, set  $\phi = 0$ .
- 1.2 At all points of C (the interface), assign to  $\phi$  the value of the time obtained by dividing the distance  $d$  from the point to the set A by the velocity of the interface at this point:  $\phi = d/F$  (see later a better way to do this).
- 1.3 At all points of F, assign to  $\phi$  the value  $+\infty$ . These points are far and they don't have any influence in the correction of the points of C (*upwind*).

Once  $\phi$  is initialized, the *fast marching* starts. The interface evolves point by point by correcting the initial estimation by means of the following iterative process:

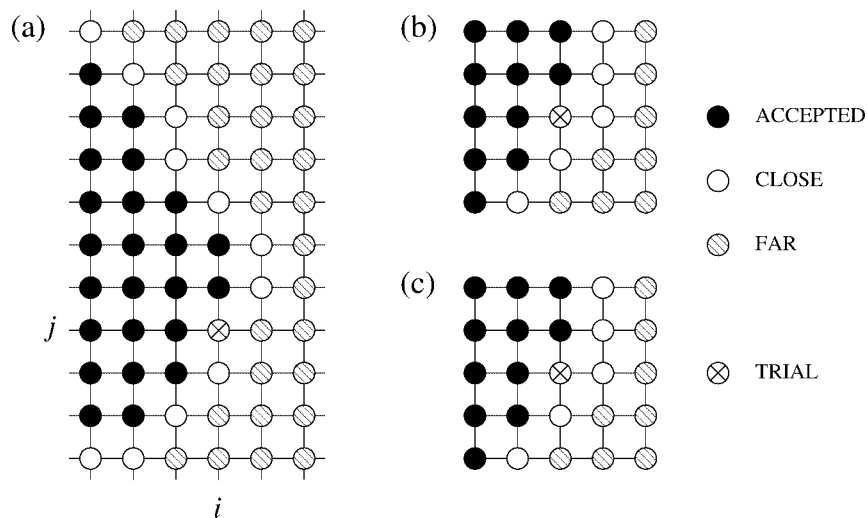
- 2.1 Obtain the point TRIAL  $(x_T, y_T)$  from set C which has the smallest value of  $\phi$ . That is, find  $(x_T, y_T) \in C$  such that

$$\forall(x, y) \in C, \quad \phi(x, y) \geq \phi(x_T, y_T).$$

See Figure 3(a).

- 2.2 Perform the following test of time-step termination:

- i. If  $\phi(x_T, y_T) > \Delta t + \varepsilon$ ,  $\phi$  has been constructed for all the points verifying (4).



**FIGURE 3.** The interface is moved one point: (a) Find the point  $(x_T, y_T)$ . (b) Move  $(x_T, y_T)$  to A. (c) Move the neighbors of  $(x_T, y_T)$  from F to C.

- Then the FMM step is finished and we return to the chemical problem.
- ii. If not, there still exists points at which the interface will arrive at a time lesser or equal to  $\Delta t + \varepsilon$  and the FMM have to continue.
- 2.3 Move the trial point  $(x_T, y_T)$  to  $A$  (and delete it from  $C$ ). The value of  $\phi$  at this point is definitive because it cannot be improved with this algorithm. See Figure 3(b).
  - 2.4 Move to  $C$  the neighbors of  $(x_T, y_T)$  that are in  $F$ , because these points have now a neighbor in  $A$ . See Figure 3(c).
  - 2.5 Actualize the value of  $\phi$  at all the neighbors of  $(x_T, y_T)$  that are in  $C$ , by solving Eq. (15). See Figure 4. This is the *gordian knot* of the FMM, and is detailed in the next section.
  - 2.6 The interface has been moved one point; see Figure 5. Goto 2.1.

### 2.3. GODUNOV'S METHOD

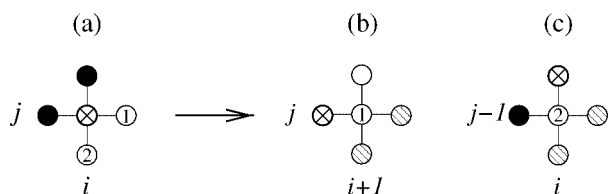
A suitable way to solve the eikonal equation in the context of the fast marching algorithm is the Godunov method [1], which is precisely the method used in Refs. [3–6]. The Godunov method makes use of the following approximation of the gradient, already used by Sethian:

$$\sqrt{\max(D_{ij}^{-x}\phi, -D_{ij}^{+x}\phi, 0)^2 + \max(D_{ij}^{-y}\phi, -D_{ij}^{+y}\phi, 0)^2} = \frac{1}{F_{ij}}, \quad (16)$$

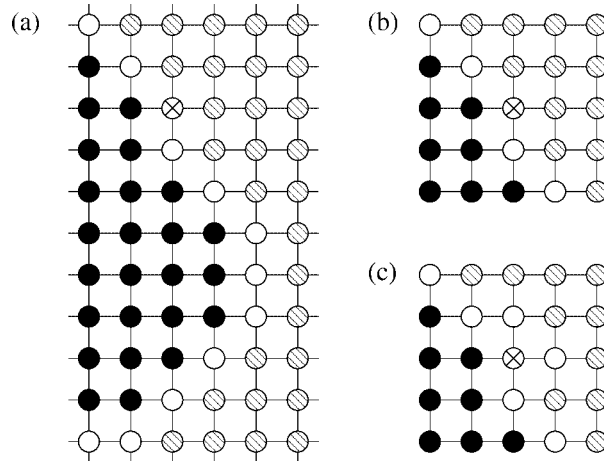
where

$$D_{ij}^{-x}\phi = \frac{\phi_{ij} - \phi_{i-1,j}}{\Delta x} \quad \text{and} \quad D_{ij}^{+x}\phi = \frac{(\phi_{i+1,j} - \phi_{ij})}{\Delta x}. \quad (17)$$

The numerical resolution of Eq. (16) is not trivial because unknown values must be compared a priori to calculated the maxima. Typically, expensive



**FIGURE 4.** Actualization of  $\phi$  with the Godunov scheme in the neighbors of  $(x_T, y_T)$  that are in  $C$ : (a) the two neighbors; (b) First neighbor of  $(x_T, y_T)$  and its respective neighbors: the information comes only from the left. (c) Second neighbor: the information comes in both directions, from the left and from above.



**FIGURE 5.** Next FMM step: (a) new position of the interface and localization of the new point  $(x_T, y_T)$ ; (b) and (c): steps 2.3–2.5 in this new situation. The actualization of  $\phi$  in the three neighbors of  $(x_T, y_T)$  is shown in Figure 6.

iterative methods are used (Rouy-Tourin, 1992). Fortunately, it is possible to take advance from the upwind character of the Godunov approximation to solve Eq. (16) in a very effective computational way.

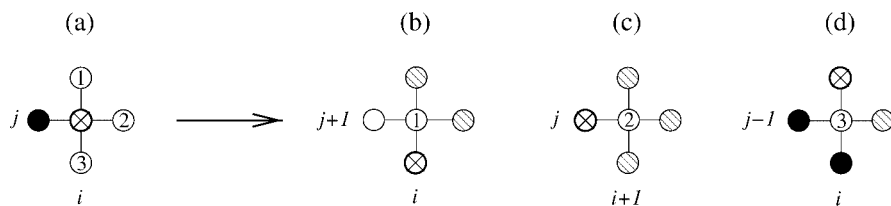
Before to explain the step 2.5 of the FMM algorithm, notice that the step 1.2 is equivalent to solve (16) assigning to  $\phi$  a zero value in the points of  $A$  and the value  $+\infty$  in the points of  $F$ .

**Step 2.5:** The actualization of  $\phi$  at the neighbors of  $(x_T, y_T)$  is done by solving Eq. (16) by assigning to  $\phi$  the value  $+\infty$  at the neighbors [of the neighbor of  $(x_T, y_T)$  which is being actualized] that are not in  $A$ . Once the calculation is done, the previous values of  $\phi$  at the neighbors of the neighbor of  $(x_T, y_T)$  must be restituted. The neighbors that are in  $A$  are used with their value and they remain unchanged.

How we do this? Assume a uniforme spatial discretization in both directions  $x$  and  $y$ :  $h = \Delta x = \Delta y$ . Then, Eq. (16) becomes

$$\max(\phi_{ij} - \phi_{i-1,j}, \phi_{ij} - \phi_{i+1,j}, 0)^2 + \max(\phi_{ij} - \phi_{i,j-1}, \phi_{ij} - \phi_{i,j+1}, 0)^2 = \frac{h^2}{F_{ij}^2}. \quad (18)$$

These maxima cannot be calculated without knowing  $\phi_{i,j}$ , which is precisely the unknown of the equation. Let us replace maxima by minima,



**FIGURE 6.** Actualization of the neighbors of the new point  $(x_T, y_T)$ . In this case there are three neighbors to be actualized, each one with a different kind of neighborhood.

$$\begin{aligned} \max(\phi_{i,j} - \phi_{i-1,j}, \phi_{i,j} - \phi_{i+1,j}, 0) \\ = \phi_{i,j} - \min(\phi_{i-1,j}, \phi_{i,j}, \phi_{i+1,j}), \end{aligned}$$

$$\begin{aligned} \max(\phi_{i,j} - \phi_{i,j-1}, \phi_{i,j} - \phi_{i,j+1}, 0) \\ = \phi_{i,j} - \min(\phi_{i,j-1}, \phi_{i,j}, \phi_{i,j+1}), \end{aligned}$$

and define the values

$$\alpha_1 = \min(\phi_{i-1,j}, \phi_{i+1,j}) \text{ and } \alpha_2 = \min(\phi_{i,j-1}, \phi_{i,j+1}). \quad (19)$$

The values  $\alpha_{1,2}$  are known because they correspond to nodes of A or they have been assigned to  $+\infty$  for this calculation. Then Eq. (16) is equivalent to

$$(\phi_{i,j} - \min(\phi_{i,j}, \alpha_1))^2 + (\phi_{i,j} - \min(\phi_{i,j}, \alpha_2))^2 = \frac{h^2}{F_{i,j}^2}. \quad (20)$$

As we are actualizing  $\phi_{i,j}$ , at least one of the neighbors of  $(x_i, y_j)$  must be in A, that is, the information comes from at least one direction  $x$  or  $y$ , or from both directions:

- If the information comes only from direction  $x$  (respectively  $y$ ), then we can calculate  $\min(\phi_{i,j}, \alpha_2) = \phi_{i,j}$ , because  $(x_i, y_{j-1})$  and  $(x_i, y_{j+1})$  are not in A, so  $\phi$  must be at  $+\infty$  in these points. The equation to solve is simply

$$\phi_{i,j} = \alpha_1 + \frac{h}{F_{i,j}} \quad (21)$$

$$\left( \text{Respectively, } \phi_{i,j} = \alpha_2 + \frac{h}{F_{i,j}} \right).$$

- If the information comes from both directions, then there exists at least one neighbor in each direction where the value of  $\phi$  is lower than  $\phi_{i,j}$ :  $\alpha_{1,2} < +\infty$ . This way,  $\phi_{i,j}$  can be dropped from minima calculations in (20) and minima calculations are redundant. The Godunov equation

becomes  $(\phi_{i,j} - \alpha_1)^2 + (\phi_{i,j} - \alpha_2)^2 = h^2/F_{i,j}^2$ , which can be rewritten as

$$\phi_{i,j}^2 - (\alpha_1 + \alpha_2)\phi_{i,j} + \frac{1}{2} \left( \alpha_1^2 + \alpha_2^2 - \frac{h^2}{F_{i,j}^2} \right) = 0. \quad (22)$$

The discriminant of this second degree equation is

$$\Delta = -(\alpha_1 - \alpha_2)^2 + \frac{2h^2}{F_{i,j}^2},$$

and it can not be negative. This imposes a first condition on the velocity, which consequently must be such that

$$F_{i,j} \leq \frac{\sqrt{2}h}{|\alpha_1 - \alpha_2|}. \quad (23)$$

This is often called the condition for “classically allowed points.” When  $\Delta > 0$ , the higher root must be used to preserve the upwind.

For the consistency of the FMM algorithm, a second upwind condition must be verified by the solution of Eq. (22), which is that  $\phi_{i,j}$  must be greater than  $\alpha_1$  and  $\alpha_2$ :  $\phi_{i,j} > \max(\alpha_1, \alpha_2)$ . This means that

$$\phi_{i,j} = \frac{\alpha_1 + \alpha_2}{2} + \frac{1}{2}\sqrt{\Delta} > \alpha_{1,2},$$

$$\text{i.e. } \Delta > 4 \left[ \alpha_{1,2} - \frac{\alpha_1 + \alpha_2}{2} \right]^2 = (\alpha_1 - \alpha_2)^2,$$

and then

$$F_{i,j} < \frac{h}{|\alpha_1 - \alpha_2|}. \quad (24)$$

This condition is more restrictive than the one deduced for “classically allowed points,” and is consequently the one which must be used as a constraint.

#### 2.4. THE CONDITION (24) IS A CONSTRAINT

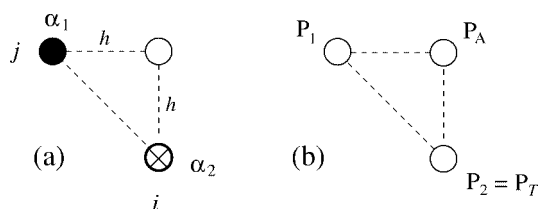
By construction, the FMM is such that expression (24) is always verified during the geometric part of the problem. This can be found in the seminal papers of Sethian. Let us give here a qualitative description of this fact, which will help us to illustrate the meaning of condition (24).

Figure 7 shows the numerical cell of a point  $P_T = (x_T, y_T)$  whose upper neighbor  $P_A = (x_i, y_j)$  is being updated. Assume that information comes in both directions  $x$  and  $y$ : from the left from  $P_1 = (x_{i-1}, y_j)$ , and from below from  $P_2 = (x_i, y_{j-1})$ . According to the algorithm presented here, Eq. (22) must be used and the value of  $\phi$  at  $P_A$  will depend on  $\phi(P_1)$  and  $\phi(P_2)$ . Then, the trial point  $P_T$  must be  $P_1$  or  $P_2$ ; assume that  $P_T = P_2$  (which means that  $P_1$  is an already accepted point, i.e., the front  $\Gamma$  has already passed through  $P_1$ ). An equivalent argument can be written for the case in which  $P_T = P_1$ .

It is now possible to calculate the minima in (19) and found that  $\alpha_1 = \phi_{i-1,j}$  and  $\alpha_2 = \phi_{i,j-1}$ . The value  $\alpha_2 - \alpha_1$  (which is positive, because the front arrives first to  $P_1$  than to  $P_2$ ) is the time elapsed between the arrivals of the front to  $P_1$  and to  $P_2$ . In other words,  $\alpha_2 - \alpha_1$  is the time spent by the front in going from  $P_1$  to  $P_2$ . Note that  $\Gamma$  can be described by a not connected set of points, in such a way that  $\Gamma$  can arrive to  $P_2$  from far from  $P_1$ . For simplicity, assume that  $\Gamma$  is a connected set.

In terms of velocities, the value  $h/(\alpha_2 - \alpha_1)$  is the velocity at which the interface covers the distance  $h$  separating  $P_1$  and  $P_2$  from  $P_A$  in a time  $\alpha_2 - \alpha_1$ . The restriction (24) means that  $F_{i,j}$  must be lower than this value, something that obviously makes sense, because  $F_{i,j}$  is the value of the velocity of the front at  $P_A = (x_i, y_j)$ .

Numerically,  $F_{i,j}$  is the value used to actualize  $\phi_{i,j}$ , that is, the value of  $F$  in the numerical cell of  $P_A$ . It is then necessary to  $F_{i,j}$  to be lower than the velocity at which the interface goes from  $P_1$  to  $P_2$ ; if not, once



**FIGURE 7.** Numerical cell in the actualization of  $P_A = (x_i, y_j)$ .

in  $P_1$ , the interface will arrive first to  $P_A$ , and then to  $P_2$ , something that is not possible.

In terms of times, the time it takes to the interface to arrive to  $P_A = (x_i, y_j)$  coming from its neighbors (located at a distance  $h$ ) at a velocity  $F_{i,j}$  must be greater or equal than the time spent in going from one neighbor to the other, a time given by  $|\alpha_1 - \alpha_2|$ . If this is not so, the interface will arrive first to the point that is being actualized than to a neighbor which is being used for this actualization, something that is an absurdity.

As we have shown, the FMM preserves the upwind during the geometric part of the numerical resolution. However, this may not be the case during the chemical part of the problem, where the velocity is renewed after each time step and the expression (24) can be no longer satisfied. Let us show how this may happen.

The function  $F^n$  is defined as the velocity at the points of  $\Gamma^n$  at time  $t^n$ . Notice that nothing is said about the value of  $F^n$  outside the interface. In fact, it may happen (especially in inhomogeneous problems) that a *velocity* does not make sense outside the interface. However, these values are needed in the FMM. This problem is solved by extending  $F^n$  to a narrow band around  $\Gamma^n$  (several methods can be used [9], e.g. by using the velocity at the nearest point of  $\Gamma^n$ ). Once a FMM step has been done, and before the renewal of  $F^n$ , the interface at the following time step  $t^{n+1}$  is given by a new set of points:

$$\Gamma^{n+1} = \{(x_l^{n+1}, y_l^{n+1})\}_{l=1}^{N_l^{n+1}}.$$

The velocity at these points at time  $t^n$  is called, actually,  $F_{\text{ext}}^n$ , which is an extension of  $F^n$ . As shown before, the condition (24) is still preserved for  $F_{\text{ext}}^n$ , but this can be false for the renewed velocity  $F^{n+1}(\Gamma^{n+1})$ . In fact, the velocity at  $\Gamma^{n+1}$  can be very different from the extension of the velocity at time  $t^n$ , depending on the smoothness of the simulated chemical process (e.g., in shock waves arising in explosions, detonations, etc.). The velocity is not supposed to be constant between time-steps of size  $\Delta t$ , and important chemical changes may happen from one interval of time to the other, which can affect considerably the velocity:

$$F^{n+1}(\Gamma^{n+1}) \neq F_{\text{ext}}^n(\Gamma^{n+1}).$$

From the chemical or physical point of view, the velocity of the interface at a given point is unique (recall that  $F$  does not change sign), but the numerical resolution makes use of two values of the velocity

at the points at which the arrival of the interface coincides with the change of time-step: the extended value  $F_{\text{ext}}^n$  and the renewed value  $F^{n+1}$ . At these points, the condition (24) can be not satisfied by  $F^{n+1}$ ; there is no reason why (neither chemical nor mathematical, nor numerical)  $F^{n+1}$  could be enforced to verify (24).

A special care must be taken with what is often called "nonclassical points". When some points are considered as such for some physical or chemical reason (e.g., they are unattainable points because an always positive value such as kinetic energy becomes negative), the algorithm must be adapted without losing the fidelity of the simulation to the real problem. Both the existence of real solutions of Eq. (16), that is,  $\Delta \geq 0$ , and the upwind condition (24) must be ensured by the algorithm along the numerical resolution of the problem.

In conclusion, we have described the way in which the condition (24) acts as a constraint for  $\phi$ . In fact, in the algorithm we have proposed here, the expression (24) must be introduced at some step as a numerical test for the validity of the arrival time function  $\phi$  for the next FMM step. When the condition is not satisfied, a reinitialization of  $\phi$  is needed, or, in a more complicated way, an adaptive time-step algorithm must be implemented. The refinement of the spatial mesh makes things smoother, but can also contribute to render more demanding the condition (24) in points located near abrupt changes of the velocity field.

### 3. A numerical example

We have considered a general situation in which a chemical reaction takes place in a domain characterized by the geometry described in Figure 8.

Assume that the chemical reactant interface arrives to a region in which the substrate is chemically inhomogeneous. Inhomogeneities can be due to different properties of the substrate, but also to dynamical effects induced by the reaction (variation in concentration, electromagnetic effects, temperature...). The reactant can then evolve with a different speed in different parts of the substrate.

Assume that the channel (with two entrances, A and B) described by two nonreactive and noncatalytic obstacles, is a fast reactant region ( $F_{\text{ch}} = 1$ ), compared to the rest of the substrate ( $F_s = 0.1$ ). This figure shows the (frequent) case in which the interface arrives first to one of the entrances (e.g., A).

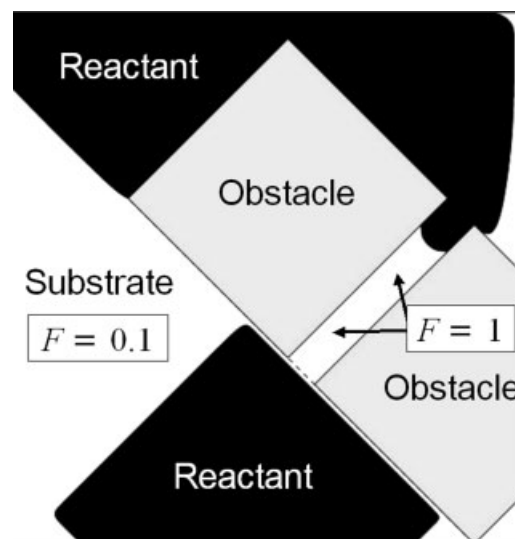


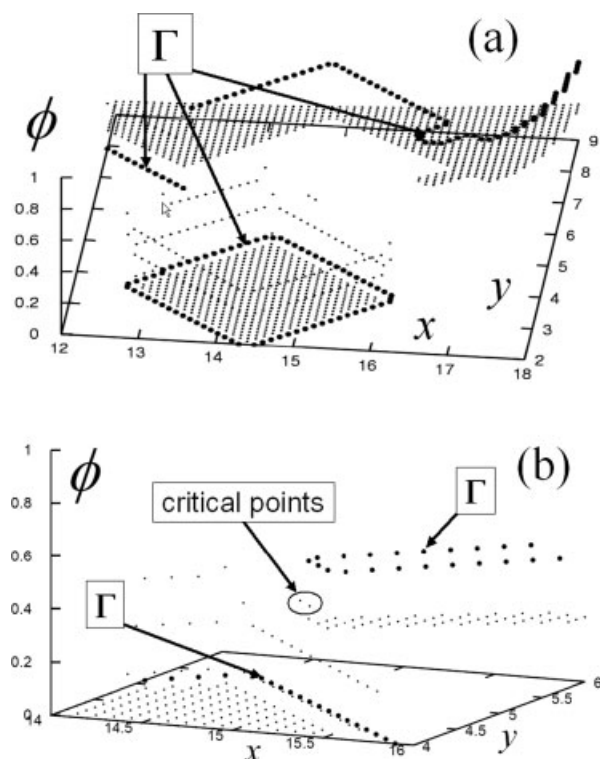
FIGURE 8. A chemically inhomogeneous domain.

Inside the channel, the advance of the interface is faster than outside. Figure 9(a) shows that the arrival time at the most advanced point of the interface inside the channel is lower than its value at the entrance B; see the two isolated points located approximately at (15, 4.5, 0.4).

During the next FMM step, the interface will arrive approximately at the same time to the entrance B from inside and outside of the channel.

Level sets and fast marching methods are especially suitable for this kind of situation in which topological changes may appear during the evolution of the interface. This is a typical situation which can be produced, e.g. by four-well potentials as in [3] (see Fig. 1 therein).

Figure 9(b) shows the precise instant in which both fronts of the interface encounter each other. The interface  $\Gamma$  is given by two nonconnected sets of points (at least in the region we are focusing on). The actualization of  $\phi$  (step 2.5 of the FMM) may require the use of close points located at different regions of the substrate, i.e. in regions of different velocities: outside the channel, the interface is just arriving (high arrival time), whereas inside the channel the interface has a low arrival time. Then  $|\alpha_1 - \alpha_2|$  becomes large with respect to its typical value when points are all in the same region. When the velocity is renewed, chemical processes can be such that  $F$  can adopt a large value; the combination of these two effects can lead to a violation of the constraint.



**FIGURE 9.** (a) The not-connected interface  $\Gamma$  when just entering into the channel ( $F_{\text{ch}} = 1$ ), and at the other end, outside the channel ( $F_s = 0.1$ ). Depicted is arrival time  $\phi(x, y)$ . Narrow points are accepted points, boldface points are close points (which define the interface, and in which the value of  $\phi$  is provisional at this stage of the algorithm). (b) Same as in (a), after a FMM step. The (definitive) arrival time at critical points near (15, 4.5) is larger than inside the channel, producing a large value of  $|\alpha_1 - \alpha_2|$ .

It is important to note that even if the constraint (24) can be seen as a CFL-like condition, where the value of a parameter ( $F$ ) is restricted by a *space over time* fraction, it is not a CFL condition. Even if the space value is  $h$  (the space step), the time value  $|\alpha_1 - \alpha_2|$  is absolutely not related to the time step of the algorithm.

## Conclusion

In this article, we have derived a constraint for the Godunov scheme when this scheme is used to solve the Eikonal equation during the fast marching method, when this later is coupled to a front propagation problem in time-dependent inhomogeneous media producing non-constant velocity fields.

Such systems have been found to be ubiquitous in chemical engineering in flow reactors, in nature in wind- and water-driven pollutants, and in explosions and detonations problems.

From the computational point of view, we have proposed to solve these problems by dividing the algorithm in two parts: a chemical part, which is the static part of the problem, which accounts for the calculations of the main physical or chemical magnitudes (such as the amount of some reactant) and the velocity of the moving front at a fixed instant of time, and a geometrical part, which accounts for the evolution in time of the interface and, consequently, the time-evolution of the chemical domain.

When the Godunov method is used in the FMM to evolve the interface along one interval of time, a constraint exists and must be fulfilled. For time-independent velocity fields, the constraint is, by construction of the FMM, automatically verified. When the velocity is time-dependent, which means that the velocity field is not known from initial data, the constraint must be tested at each time step. We have obtained an explicit expression of this constraint, which arises from the reformulation of Godunov's scheme by replacing maxima with minima, together with a short analysis of the neighborhood of each point of the interface in each iteration of the FMM.

Finally, we have presented a generic chemical example in order to illustrate the facility with which the restriction can be violated. When the velocity field exhibits abrupt changes in short distances, the discretized points of the chemical domain can be distributed in such a way that, at a given point and instant of time, the upwind evolution of the front is not conserved, something that can not be possible.

The constraint can be used as a direct test of the validity of the numerical velocity field used at the end of a given time-step. When the constraint is not fulfilled, the general algorithm must be stopped until this step is solved, by reinitializing the arrival time function, by changing the time or space discretizations or both, or by introducing an adaptive time-step routine.

## ACKNOWLEDGMENTS

We thank F. Bernal for fruitful discussions. This work has been carried out under a Ramón y Cajal contract at the University of Cantabria.

---

**References**

1. (a) Sethian, J. A. Proc Natl Sci USA 1996, 93, 1591; (b) Sethian, J. A. SIAM Rev 1999, 41, 199.
2. Chopp, D. L. SIAM J Sci Comput 2001, 23, 230.
3. Dey, B. K.; Janicki, M. R.; Ayers, P. W. J Chem Phys 2004, 121, 6667.
4. Dey, B. K.; Ayers, P. W. Mol Phys 2006, 104, 541.
5. Dey, B. K.; Bothwell, S.; Ayers, P. W. J Math Chem 2007, 41, 1.
6. Dey, B. K.; Ayers, P. W. Mol Phys 2007, 105, 71.
7. Kály-Kullai, K. J Math Chem 2003, 34, 163.
8. Bonilla, L. L.; Escobedo, R.; Higuera, F. J. Phys Rev E 2003, 67, 036202.
9. Adalsteinsson, D.; Sethian, J. A. J Comput Phys 1998, 148, 2.