A TECHNIQUE FOR HIGH–ORDER TREATMENT OF DIFFUSION TERMS IN SEMI–LAGRANGIAN SCHEMES

Roberto FERRETTI(*)

Version: September 12, 2000

Abstract. We consider in this paper a semi–Lagrangian technique to treat advection–diffusion equations. The scheme is based on a stochastic representation formula for the solution which allows to avoid the splitting between advective and diffusive part of the evolution operator. This results in a somewhat reduced computational complexity, as well as the possibility to achieve higher consistency rates. A general theoretical analysis is carried out in the paper, and numerical tests are presented.

Key words. Advection–diffusion equations, stochastic representation, method of characteristics, semi–Lagrangian schemes, high–order schemes, convergence.

AMS(MOS) subject classification. 65M25, 65M12, 60H30.

(*) Dipartimento di Matematica, Università di Roma "Tor Vergata", v. della Ricerca Scientifica, 00133 Roma, Italy, e-mail: ferretti@mat.uniroma2.it

0. Introduction.

This paper is devoted to a semi–Lagrangian type treatment of second order terms in advection–dominated, possibly degenerate, parabolic equations. We will use the advection–diffusion equation,

$$\begin{cases} v_t(x,t) = \nu\Delta v(x,t) + f(x)\cdot\nabla v(x,t) + \lambda v(x,t) + g(x) \\ v(x,0) = v_0(x) \end{cases} \tag{0.1}$$

as a model problem to describe the tecnique and to carry out a general convergence analysis.

The semi–Lagrangian (SL) schemes are very popular in the Numerical Weather Prediction community and in this setting they have been introduced by Robert in the eighties (see [R]). In different frameworks, they are also used under names such as Modified Method of Characteristics, Eulerian–Lagrangian methods, Characteristic Galerkin methods and so on (see [DR], [HLB], [MPS], [P], [SW]). The general idea of semi–Lagrangian methods is to reconstruct the solution by integrating numerically the equation along the characteristics starting from any grid point, not on the whole time interval (as it would be the case in the particle method), but on a single time step. The solution is computed coupling a numerical method for ODEs (to compute the upwind points with respect to the grid nodes) with an interpolation formula (to recover the value of the solution in such points, which are not in general grid points themselves). An extensive introduction to this class of methods is given in [SC] along with a number of extensions. The comparison with more classical eulerian difference schemes shows that in general SL schemes have a higher computational cost per time step, but that they also allow larger time steps. It is generally agreed that semi–implicit imlementations of SL schemes perform remarkably well in the case of strongly advection–dominated problems.

If $\nu = 0$ (i.e. in the case of pure advection), the schemes rely on the representation formula

$$v(x,t) = \int_0^t e^{\lambda s} g(y(s))ds + e^{\lambda t} v_0(y(t)) \tag{0.2}$$

where $y(t)$ is the solution of

$$\begin{cases} \dot y(s) = f(y(s)) \\ y(0) = x. \end{cases} \tag{0.3}$$

Although using the representation formula (0.2) is very natural in the case of (linear) first order equations, the situation gets more complex when a diffusion term appears. In this situation, the usual response is to split the evolution operator and treat in semi–Lagrangian way only the first order part. This results either in very severe time–step bounds, or in the additional computational effort of solving an implicit scheme for the second order term. A further disadvantage is that the splitting itself introduces a limitation in the consistency rate of the scheme.

An alternative approach which is a natural extension of the technique used for pure advection equations has first been given by Camilli and Falcone in quite a different setting, namely stochastic dynamic programming (see [CF] for the original problem, [BF] for its adaptation to advecion–diffusion equations). This approach is based on the stochastic representation (Feynman–Kac) formula for the solution of (0.1):

$$v(x,t) = E\left\{\int_0^t e^{\lambda s} g(y(s))ds + e^{\lambda t} v_0(y(t))\right\} \tag{0.4}$$

where $y(t) = y(x,t)$ is now the solution of the stochastic differential equation (SDE):

$$\begin{cases} dy(s) = f(y(s))ds + \sqrt{2\nu}dW \\ y(0) = x \end{cases} \tag{0.5}$$

and $E\{\cdot\}$ is the expectation with respect to the Wiener measure (i.e. the probability measure of the brownian process W in (0.5)). It is easy to see that if $\nu = 0$, then (0.4) reduces to (0.2). It should also be noted that the solution of (0.5) must be understood in the Ito sense (although of course in the case of constant ν it coincides with the solution in the Stratonovich sense).

The purpose of this paper is precisely to describe and analyse a general form high order version of the first order algorithm in [BF]. To this end, we will make extensive use of the theory of approximation for the trajectories of stochastic differential equations, as well as the convergence results given in [FF] for the first order advection equation.

The outline of the paper is the following. Section 1 reviews some basic approximation results for Stochastic Differential Equations. Section 2 deals with time and space discretization, whereas in section 3 we prove convergence and a priori estimates for the fully discrete schemes and in section 4 we carry out the Fourier analysis of the scheme for the simplified one–dimensional, constant coefficient case. Section 5 discusses some extensions to more general problems, including nonconstant viscosity and boundary conditions. Finally, in Section 6 we present the results of some numerical tests in $I\!R^1$.

1. Approximation of Stochastic Differential Equations.

We will review in this section some useful points in the theory of approximation for SDEs. For an extensive treatment of the problem, the reader is referred to [KP].

To fix ideas, we assume that the functions $f : I\!R^N \to I\!R^N$ and $g : I\!R^N \to I\!R$ are bounded along with their derivatives, up to the order required by the desired consistency rate. Here and in the sequel, we will denote as usual by $\|\cdot\|_\alpha$ the norm in L^α ($L^\alpha(I\!R^N)$ or $L^\alpha(\Omega)$ depending on the context).

The approximation of the SDE (0.5) is a widely studied problem in probability theory. It turns out that (0.5) can be approximated at least in two different senses. The *strong* convergence (with order p) means that, using in the numerical scheme suitable increments of a particular trajectory of the brownian process, the numerical trajectory of (0.5) converges in probability to the exact trajectory (with order p), that is

$$E\{|y_k - y(k\Delta t)|\} \to 0 \quad (E\{|y_k - y(k\Delta t)|\} \le \Delta t^p). \tag{1.1}$$

The *weak* sense means that the same holds only for expected values, that is

$$|E\{h(y_k)\} - E\{h(y(k\Delta t))\}| \to 0 \qquad (|E\{h(y_k)\} - E\{h(y(k\Delta t))\}| \le \Delta t^p). \tag{1.2}$$

for any smooth function h. In general, it happens that the convergence rate of the approximation is higher for weak approximations. Moreover, under suitable assumptions, the averaging of the trajectories in (1.2) (which is in principle a Monte–Carlo type algorithm) needs only to be computed on a limited number of trajectories, this number depending on the order of the scheme.

Since our interest is in computing the expectation in (0.4), we will focus on weak approximations. The following examples (which are all from [KP]) show some stochastic version of well known one–step schemes. For notational simplicity, we will assume a scalar SDE ($N = 1$).

Example 1: *Stochastic Euler scheme.*

In the simplest case of the Euler approximation, we obtain the scheme

$$\begin{cases} y_{k+1} = y_k + \Delta t f(y_k) + \sqrt{2\nu}\Delta W_k \\ y_0 = x. \end{cases} \tag{1.3}$$

Although the increments ΔW_k should be understood as independent gaussian variables with zero mean and variance Δt, in practice (see [KP]) in order to achieve first order weak convergence it suffices to use two–point distributed variables with probability density

$$P(\Delta W_k = \pm\sqrt{\Delta t}) = \frac{1}{2}. \tag{1.4}$$

Example 2: *Stochastic Heun scheme.*

In the case of constant diffusion coefficient ν, the stochastic version of Heun's formula is given by:

$$\begin{cases} y_{k+1} = y_k + \dfrac{\Delta t}{2}(f(y_k) + f(y_k + \Delta t f(y_k) + \sqrt{2\nu}\Delta W_k)) + \sqrt{2\nu}\Delta W_k \\ y_0 = x. \end{cases} \tag{1.5}$$

In this case, second order weak convergence may be achieved provided the variables ΔW_k satisfy for example

$$P(\Delta W_k = \pm\sqrt{3\Delta t}) = \frac{1}{6}, \tag{1.6a}$$

$$P(\Delta W_k = 0) = \frac{2}{3}. \tag{1.6b}$$

Example 3: *Stochastic Crank–Nicolson scheme.*

Again in the case of constant diffusion coefficient ν, the stochastic Crank–Nicolson formula reads:

$$\begin{cases} y_{k+1} = y_k + \dfrac{\Delta t}{2}(f(y_k) + f(y_{k+1})) + \sqrt{2\nu}\Delta W_k \\ \quad y_0 = x. \end{cases} \tag{1.7}$$

This scheme is second order weakly convergent provided the variables ΔW_k satisfy (1.6).

In general, higher order schemes may be constructed either by some suitable generalization of deterministic schemes, or by extrapolation on second or third order schemes. In both cases, the increase in computational complexity is considerable. We give an example for extrapolation.

Example 4: *Extrapolation on a second–order scheme.*

Assume $y_k^{(\Delta t)}$ is the aproximate solution given at the k–th iteration of a second–order scheme with time step Δt; let moreover h be a smooth function. Then:

$$h(y(k\Delta t)) = \frac{1}{21}\left[32E(h(y_{4k}^{(\Delta t/4)})) - 12E(h(y_{2k}^{(\Delta t/2)})) + E(h(y_k^{(\Delta t)}))\right] + O(\Delta t^4)$$

where the expectation may be replaced by a finite sum with weights such as, for example, the ones given by (1.6).

2. Time and space discretization.

We will first obtain in this section a discrete time approximation of the representation formula (0.4), by a stochastic one–step time discretization of order $p \geq 1$, in the form:

$$\begin{cases} y_{k+1} = y_k + \Delta t \Phi(y_k, y_{k+1}, \Delta W_k) \\ \quad y_0 = x. \end{cases}$$

Following [FF], it is convenient to consider the dynamical system:

$$\begin{pmatrix} dy(s) \\ d\gamma(s) \end{pmatrix} = \begin{pmatrix} f(y(s))ds + \sqrt{2\nu}dW \\ e^{\lambda s}g(y(s))ds \end{pmatrix} \tag{2.1}$$

with the initial conditions $y(0) = x$, $\gamma(0) = 0$. Applying a stochastic one–step scheme to (2.1), and restricting to the first time step, we have:

$$\begin{pmatrix} \bar{y} \\ \bar{\gamma} \end{pmatrix} = \begin{pmatrix} x \\ 0 \end{pmatrix} + \Delta t \begin{pmatrix} \Phi_f(x, \bar{y}, \Delta W) \\ \Phi_g(x, \bar{y}, \bar{\gamma}, \Delta W) \end{pmatrix} \tag{2.2}$$

where we have splitted the function Φ according to (2.1). In (2.2), $\bar{y}$ is an approximation of $y(\Delta t)$, whereas $\bar{\gamma}$ approximates the integral between 0 and Δt in (0.4) (note that the diffusion appears only in the components of y). If an implicit scheme is used, $\bar{y}$ and $\bar{\gamma}$ are understood as the solutions of equation (2.2). In any case, we will possibly use in the sequel the notations $\bar{y}(x, \Delta W)$ or $\bar{\gamma}(x, \Delta W)$ to denote the left–hand side of (2.2). In order to make clear the previous construction, let us rewrite the schemes of section 1 in the form (2.2).

Example 1: *Euler scheme.*

In the case of the Euler scheme, we have:

$$\Phi_f(x, \bar{y}, \Delta W) = f(x) + \sqrt{2\nu}\Delta W$$

$$\Phi_g(x, \bar{y}, \Delta W) = g(x). \tag{2.3}$$

Example 2: *Heun scheme.*

In this case the scheme is given by:

$$\Phi_f(x, \bar{y}, \Delta W) = \frac{1}{2}(f(x) + f(x + \Delta t f(x) + \sqrt{2\nu}\Delta W)) + \sqrt{2\nu}\Delta W$$

$$\Phi_g(x, \bar{y}, \Delta W) = \frac{1}{2}(g(x) + g(x + \Delta t f(x) + \sqrt{2\nu}\Delta W)) \tag{2.4}$$

Example 3: *Crank–Nicolson scheme.*

Lastly, for the the Crank–Nicolson formula we have:

$$\Phi_f(x, \bar{y}, \Delta W) = \frac{1}{2}(f(x) + f(\bar{y})) + \sqrt{2\nu}\Delta W$$

$$\Phi_g(x, \bar{y}, \Delta W) = \frac{1}{2}(g(x) + g(\bar{y})). \tag{2.5}$$

If we denote by $\bar{v}^k$ a time–discrete approximate solution at time $t_k = k\Delta t$, obtained rewriting (0.4) on a single time step and introducing the previous one–step approximation of (0.5), we obtain:

$$\bar{v}^{k+1}(x) = E\left\{\bar{\gamma}(x, \Delta W) + e^{\lambda \Delta t}\bar{v}^k(\bar{y}(x, \Delta W))\right\} \tag{2.6}$$

Now, relying on the assumption of weak convergence, we further compute the expectation with respect to ΔW by working on a "small" number of determinations as shown in the examples of section 1. We associate a weight w_i to a determination Δ_i of the variable ΔW so as to have

$$P(\Delta W = \Delta_i) = w_i \quad (i = 1, \ldots, s) \tag{2.7}$$

with the obvious conditions

$$w_i \geq 0 \quad (i = 1, \ldots, s) \tag{2.8a}$$

$$\sum_{i=1}^{s} w_i = 1 \tag{2.8b}$$

and the further assumptions (see [KP]):

$$\sum_{i=1}^{s} w_i \Delta_i = 0 \tag{2.9a}$$

$$\sum_{i=1}^{s} w_i \Delta_i^2 = \Delta t \tag{2.9b}$$

Using the discrete probability density (2.7) in (2.6), we obtain a second time–discrete approximation in the form

$$v^{k+1}(x) = \sum_{i=1}^{s} w_i \left\{ \bar{\gamma}(x, \Delta_i) + e^{\lambda \Delta t} v^k(\bar{y}(x, \Delta_i)) \right\} \tag{2.10}$$

in which the initial condition $v^0(x)$ for $k = 0$ coincides with $v_0(x) = v(x, 0)$.

The following step to obtain a fully discrete scheme is to discretize (2.10) with respect to space variables. This is done as usual by setting up a space grid (with space step Δx and nodes x_j, $j = 1, \ldots, q$) in the computational domain and computing (2.10) at each node, replacing the upwind value $v^k(\bar{y}(x, \Delta_i))$ with a suitable polynomial reconstruction. We point out that in principle SL schemes do not require a structured grid, but this is by far the most common situation in practical implementations (an implementation on unstructured grid is given, for example, in [LLS]).

Let v_j^k be the desired approximation of $v(x_j, t_k)$, and define

$$V^k := \begin{pmatrix} v_1^k \\ \vdots \\ v_q^k \end{pmatrix}.$$

Let moreover $I[V](x)$ be a reconstruction (interpolation, extension) operator such that $I[V](x_j) = v_j$, and if $v(x)$ is a C^r function and

$$V := \begin{pmatrix} v_1 \\ \vdots \\ v_q \end{pmatrix} = \begin{pmatrix} v(x_1) \\ \vdots \\ v(x_q) \end{pmatrix},$$

then, for any x in the computational domain,

$$|I[V](x) - v(x)| \le C \Delta x^r. \tag{2.11}$$

As it has been done for time discretizations, we give some practical examples of reconstruction operators. Again, for the sake of simplicity, the examples will refer to the case of $I\!R^1$, with constant space step Δx.

Example 1: *Finite elements basis.*

The simplest choice of the interpolation operator is

$$I[V](x) = v(x_l)$$

with x_l defined as the grid node closest to x. This choice corresponds to P_0 (piecewise constant) finite elements and in this case $r = 1$. A more efficient choice, based on P_1 (piecewise linear) finite elements is defined by:

$$I[V](x) = \frac{x_{l+1} - x}{\Delta x} v_l + \frac{x - x_l}{\Delta x} v_{l+1}$$

with $[x_l, x_{l+1}]$ the grid interval containing x; in this case $r = 2$.

Example 2: *High–order polynomial interpolation.*

Higher order reconstructions may be carried out by Lagrange interpolation on a proper set of nodes around x. For example, a cubic reconstruction require four nodes (two on the right side of x and two on the left side). Defining l as the index of the node just on the left of x, it reads:

$$I[V](x) = \frac{(x - x_l)(x - x_{l+1})(x - x_{l+2})}{6\Delta x^3} v_{l-1} + \cdots + \frac{(x - x_{l-1})(x - x_l)(x - x_{l+1})}{6\Delta x^3} v_{l+2}$$

and the order of accuracy is $r = 4$.

It is also possible to use monotonic (see [RW]) or Essentially Non–Oscillatory interpolations. In this case the linearity of the mapping $I[\cdot]$ is lost, as the following example shows.

Example 3: *Essentially non–oscillatory (ENO) reconstruction.*

Let $V[x_{k_0}, \ldots, x_{k_m}]$ denote a divided difference of order m for the function v. Then a second order ENO reconstruction of v reads

$$I[V](x) = v_l + V[x_l, x_{l+1}](x - x_l) + \phi\big(V[x_{l-1}, x_l, x_{l+1}], V[x_l, x_{l+1}, x_{l+2}]\big)(x - x_l)(x - x_{l+1})$$

where

$$\phi(a, b) = \begin{cases} a & \text{if } |a| \le |b| \\ b & \text{if } |a| > |b| \end{cases}$$

In this case $r = 3$, and the interpolation has also a good behaviour with respect to the total variation. Higher order reconstructions are also possible.

The fully discrete scheme may then be written as

$$\begin{cases} v_j^{k+1} = \sum_{i=1}^{s} w_i \left\{ \bar{\gamma}(x_j, \Delta_i) + e^{\lambda \Delta t} I[V^k](\bar{y}(x_j, \Delta_i)) \right\} \\ v_j^0 = v_0(x_j). \end{cases} \tag{2.12}$$

3. Convergence and error estimates.

In examining the convergence of the fully discrete scheme (2.12), we should take into account that convergence theory for SL schemes applied to first–order equations is itself somewhat incomplete. More precisely (see [FF]), the unconditional stability of SL schemes, which is a widely observed fact in practice, has only been theoretically proved for some special case. We recall that setting $\nu = 0$ in (0.1), we obtain the first order advection equation

$$\begin{cases} v_t(x,t) = f(x)\cdot\nabla v(x,t) + \lambda v(x,t) + g(x) \\ v(x,0) = v_0(x). \end{cases} \tag{3.1}$$

Accordingly, the SL scheme for (3.1) is

$$\begin{cases} u_j^{k+1} = \bar{\gamma}(x_j,0) + e^{\lambda\Delta t} I[U^k](\bar{y}(x_j,0)) \\ u_j^0 = v_0(x_j). \end{cases} \tag{3.2}$$

While the study of a more general proof of stability is still in progress, what will be proved here is that the scheme (2.12) is *as stable as* its nonviscous counterpart (3.2) (in fact, (2.12) is slightly more stable).

Theorem 3.1. *Let f be a smooth vector field, g be a bounded function. Assume moreover that the scheme* (3.2) *is L^α–stable for any smooth f, that is*

$$\|I[U^{k+1}]\|_\alpha \le (1 + C\Delta t)\|I[U^k]\|_\alpha$$

for a positive constant C independent of Δx, Δt. Then,

$$\|I[V^{k+1}]\|_\alpha \le (1 + C\Delta t)\|I[V^k]\|_\alpha \tag{3.3}$$

with V^k solution of (2.12).

Proof. Define

$$u_{i,j}^{k+1} = \bar{\gamma}(x_j,\Delta_i) + e^{\lambda\Delta t} I[V^k](\bar{y}(x_j,\Delta_i)) \tag{3.4}$$

$$U_i^k := \begin{pmatrix} u_{i,1}^k \\ \vdots \\ u_{i,q}^k \end{pmatrix}.$$

Since the introduction of a constant Δ_i does not change the regularity of the vector field f, the scheme (3.4) is still L^α–stable, so that

$$\|I[U_i^{k+1}]\|_\alpha \le (1 + C_i\Delta t)\|I[V^k]\|_\alpha.$$

Using (3.4) and (2.8) in (2.12), we get

$$\|I[V^{k+1}]\|_\alpha \le \sum_{i=1}^s w_i \|I[U_i^{k+1}]\|_\alpha \le$$

$$\leq \sum_{i=1}^{s} w_i(1 + C_i\Delta t)\|I[V^k]\|_\alpha \leq$$

$$\leq (1 + (\sum_{i=1}^{s} w_i C_i)\Delta t)\|I[V^k]\|_\alpha$$

which proves (3.3).

■

Remark. The assumption on the smoothness of f seems to be necessary to avoid in (3.2) possible instabilities due to very disordered advecting vector fields.

We turn now to the determination of the consistency rate for the scheme (2.12). To this end, we define

$$V(t) := \begin{pmatrix} v_1(t) \\ \vdots \\ v_q(t) \end{pmatrix} = \begin{pmatrix} v(x_1, t) \\ \vdots \\ v(x_q, t) \end{pmatrix}.$$

Theorem 3.2. *Let f be a smooth vector field, g be a smooth bounded function, $v(\cdot,\cdot)$ be a smooth solution of* (0.1)*. Assume moreover that* (1.2) *and* (2.11) *hold for any smooth f and g. Then the local truncation error satisfies the bound*

$$\frac{1}{\Delta t}\left| v(x_j, t_{k+1}) - \sum_{i=1}^{s} w_i\{\bar\gamma(x_j, \Delta_i) + e^{\lambda\Delta t} I[V(t_k)](\bar y(x_j, \Delta_i))\}\right| \leq C\left(\Delta t^p + \frac{\Delta x^r}{\Delta t}\right). \tag{3.5}$$

Proof. We estimate the left–hand term of (3.5) as

$$\frac{1}{\Delta t}\left| v(x_j, t_{k+1}) - \sum_{i=1}^{s} w_i\{\bar\gamma(x_j, \Delta_i) + e^{\lambda\Delta t} I[V(t_k)](\bar y(x_j, \Delta_i))\}\right| \leq$$

$$\leq \frac{1}{\Delta t}\left| v(x_j, t_{k+1}) - \sum_{i=1}^{s} w_i\{\bar\gamma(x_j, \Delta_i) + e^{\lambda\Delta t} v(\bar y(x_j, \Delta_i), t_k)\}\right| +$$

$$+\frac{1}{\Delta t}\left|\sum_{i=1}^{s} w_i\{\bar\gamma(x_j, \Delta_i) + e^{\lambda\Delta t} v(\bar y(x_j, \Delta_i), t_k)\} - \right.$$

$$\left. - \sum_{i=1}^{s} w_i\{\bar\gamma(x_j, \Delta_i) + e^{\lambda\Delta t} I[V(t_k)](\bar y(x_j, \Delta_i))\}\right|. \tag{3.6}$$

We recall that the representation formula (0.4) gives for $v(x_j, t_{k+1})$:

$$v(x_j, t_{k+1}) = E\left\{\int_0^{\Delta t} e^{\lambda s} g(y(x_j, s))ds + e^{\lambda\Delta t} v(y(x_j, \Delta t), t_k)\right\} \tag{3.7}$$

The first term in the right–hand side of (3.6) can be estimated by (3.7) and the weak convergence assumption (1.2) as

$$\frac{1}{\Delta t}\left| v(x_j, t_{k+1}) - \sum_{i=1}^{s} w_i\{\bar\gamma(x_j,\Delta_i) + e^{\lambda\Delta t} v(\bar y(x_j,\Delta_i), t_k)\}\right| \le C\Delta t^p \tag{3.8}$$

(note that we are working on a single time step and that v is supposed to be smooth). Using (2.11), the second term may be estimated as

$$\frac{1}{\Delta t}\left|\sum_{i=1}^{s} w_i\{\bar\gamma(x_j,\Delta_i) + e^{\lambda\Delta t} v(\bar y(x_j,\Delta_i), t_k)\} - \right.$$

$$\left. - \sum_{i=1}^{s} w_i\{\bar\gamma(x_j,\Delta_i) + e^{\lambda\Delta t} I[V(t_k)](\bar y(x_j,\Delta_i))\}\right| \le$$

$$\le \frac{1}{\Delta t}\sum_{i=1}^{s} w_i e^{\lambda\Delta t} |v(\bar y(x_j,\Delta_i), t_k)\} - I[V(t_k)](\bar y(x_j,\Delta_i))| \le$$

$$\le \sum_{i=1}^{s} w_i e^{\lambda\Delta t} C \frac{\Delta x^r}{\Delta t} \tag{3.9}$$

which, together with (3.8), proves (3.5).

■

Remark. (3.5) is also an L^α estimate under natural assumptions (see [FF]).

Corollary 3.2. *Let f be a C^p vector field, g be a C^p bounded function, v be a C^r solution of (0.1) and assume that (1.2) and (2.11) hold for some $p, r \ge 1$. Assume moreover that the scheme (3.2) is L^α–stable. Then, as $\Delta t, \Delta x \to 0$,*

$$\|I[V^k] - v(k\Delta t)\|_\alpha \to 0 \tag{3.10}$$

for any k, and the following estimate holds:

$$\|I[V^k] - v(k\Delta t)\|_\alpha \le C\left(\Delta t^p + \frac{\Delta x^r}{\Delta t}\right). \tag{3.11}$$

■

Remark. The estimate (3.11) may converge to zero only provided the steps Δt, Δx satisfy the condition $\Delta x^r = o(\Delta t)$; nevertheless convergence of approximate solution is ensured by the Lax–Richtmeyer equivalence theorem (see [RM]) since the scheme is consistent and stable.

4. Fourier analysis for the fully discrete scheme.

In the previous section we have analysed the stability of the scheme into consideration starting from the assumption that it is stable for the pure advection equation. In this section we will carry out the Fourier analysis of the scheme, giving a direct proof of convergence (although in the oversimplified framework of Von Neumann analysis, i.e. in one space dimension, for the constant coefficient case, with an infinite space grid). As a byproduct of this analysis, we will also obtain the Feynman–Kac representation formula for this special case.

The problem considered in Von Neumann analysis is

$$\begin{cases} v_t = \nu v_{xx} + a v_x & x \in I\!R \\ v(x,0) = v_0(x) \end{cases} \tag{4.1}$$

whose exact solution is given by the convolution

$$v(x,t) = \frac{e^{-\frac{(x+at)^2}{4\nu t}}}{\sqrt{4\pi\nu t}} * v_0(x). \tag{4.2}$$

According to (2.12), the fully discrete approximation of (4.1) reads

$$V^{k+1} = \Psi V^k = \sum_{i=1}^{s} w_i \Psi^{(i)} V^k \tag{4.3}$$

where the banded infinite matrices $\Psi^{(i)}$ correspond to the matrices used in approximating the advection equations

$$\begin{cases} u_t = \left(a + \frac{\sqrt{2\nu}\Delta_i}{\Delta t}\right) u_x & x \in I\!R \\ u(x,0) = u_0(x). \end{cases} \tag{4.4}$$

In this setting, $\Psi^{(i)} = \left(\psi^{(i)}_{jl}\right)$, so that

$$I[V^k](x_j + a\Delta t + \sqrt{2\nu}\Delta_i)) = \sum_j \psi^{(i)}_{jl} v^k_l \tag{4.5}$$

with $x_j = j\Delta x$ and the condition $\psi^{(i)}_{jl} = \psi^{(i)}_{j+1,l+1}$ (this condition requires the linearity of the reconstruction operator $I[\cdot]$). Since the scheme has already been proved to be stable for P_1 interpolations (see [CF], [BF]), we will assume in addition that the reconstruction $I[\cdot]$ is exact on quadratic functions.

Carrying out the transformations in the Fourier domain, we obtain from (4.2):

$$\hat{v}(\omega,t) = \mathcal{F}[v](\omega,t) = e^{\boldsymbol{i}at\omega - \nu t\omega^2}\hat{v}_0(\omega). \tag{4.6}$$

where $\boldsymbol{i}$ denotes the imaginary unit. On the other hand, we get from (4.3)

$$\hat{V}^{k+1}(\omega) = \hat{\Psi}(\omega)\hat{V}^k(\omega) \tag{4.7}$$

and hence

$$\hat{V}^k(\omega) = \hat{\Psi}(\omega)^k \hat{V}^0(\omega) \tag{4.8}$$

where the Fourier transform of a sequence is understood as the transformation of its pulsed continuous equivalent, $\hat{\Psi}$ is the transform of the sequence ψ_{0l} and the matrix product in the right–hand side of (4.3) is treated as a convolution. Equation (4.8) gives the Fourier transform of the approximate solution at time $t = k\Delta t$, with $k \to \infty$ as $\Delta t \to 0$. Recovering the asymptotic behaviour of $(\hat{\Psi})^k$ is a well–known problem in probability theory, and the answer is given by the central limit theorem as far as the sequence ψ_{0l} is made of positive elements. This condition fails in our case, but we will follow similar steps.

Step 1. As a start, we show that $|\hat{\Psi}(\omega)| \leq 1$. Indeed, this is equivalent to the Von Neumann condition of L^2 stability. Recalling (4.3), we have

$$|\hat{\Psi}(\omega)| \leq \sum_{i=1}^{s} w_i |\hat{\Psi}^{(i)}(\omega)| \leq 1. \tag{4.9}$$

The last inequality in (4.9) follows from the L^2 stability of the schemes which approximate equation (4.4), this being proved in [FF]. We remark that (4.9) implies that $\hat{\Psi}(\omega)^k$ is finite for any k.

Step 2. Let us compute the Taylor series of $\hat{\Psi}(\omega)$ centered in $\omega = 0$. By known properties of the Fourier transform, the required derivatives can be computed starting from the moments of the sequence ψ_{0l}, so that

$$\hat{\Psi}(0) = \sum_l \psi_{0l} = \sum_{i=1}^{s} w_i \sum_l \psi_{0l}^{(i)} = 1, \tag{4.10}$$

which follows from the fact that the reconstruction is exact on constant functions. Again,

$$\hat{\Psi}'(0) = -\boldsymbol{i} \sum_l x_l \psi_{0l} = -\boldsymbol{i} \sum_{i=1}^{s} w_i \sum_l x_l \psi_{0l}^{(i)} =$$

$$= -\boldsymbol{i} \sum_{i=1}^{s} w_i (a\Delta t + \sqrt{2\nu}\Delta_i) = -\boldsymbol{i} a\Delta t - \boldsymbol{i}\sqrt{2\nu} \sum_{i=1}^{s} w_i \Delta_i = -\boldsymbol{i} a\Delta t. \tag{4.11}$$

In (4.11), we have used both the fact that $\sum_l x_l \psi_{0l}^{(i)}$ is nothing but the reconstruction (which has been supposed to be exact) of the linear function x in the point $x = a\Delta t + \sqrt{2\nu}\Delta_i$, and the condition (2.9$a$). From the second moment we have

$$\hat{\Psi}''(0) = -\sum_l x_l^2 \psi_{0l} = -\sum_{i=1}^{s} w_i \sum_l x_l^2 \psi_{0l}^{(i)} = -\sum_{i=1}^{s} w_i (a\Delta t + \sqrt{2\nu}\Delta_i)^2 =$$

$$= -a^2 \Delta t^2 - 2\nu \sum_{i=1}^{s} w_i \Delta_i^2 \tag{4.12}$$

in which we have used again the facts that $\sum_l x_l^2 \psi_{0l}$ is the reconstruction of the function x^2 at the point $x = a\Delta t + \sqrt{2\nu}\Delta_i$, that this reconstruction is supposed to be exact and that (2.9a) holds. Using (2.9b) we obtain at last

$$\hat{\Psi}''(0) = -2\nu\Delta t + o(\Delta t). \tag{4.13}$$

It is easy to show by means of the same arguments that, since $a\Delta t + \sqrt{2\nu}\Delta_i = O(\sqrt{\Delta t})$, successive derivatives satisfy $\hat{\Psi}^{(m)}(0) = o(\Delta t)$ as soon as $m > 2$. We obtain therefore:

$$\hat{\Psi}(\omega) = 1 + \Delta t(a\boldsymbol{i}\omega - \nu\omega^2) + o(\Delta t) \tag{4.14}$$

Step 3. Setting now $k = t/\Delta t$ and letting $\Delta t \to 0$:

$$\hat{\Psi}(\omega)^{\frac{t}{\Delta t}} \to e^{iat\omega - \nu t\omega^2} \tag{4.15}$$

which also ensures via (4.8) the convergence of the approximate solution to the exact one. We stress that the gaussian term in the Fourier transform (and hence, the possibility of treating the second–order derivative) comes from the presence of a nonzero centered second moment in the sequence ψ_{0l}.

5. Some remarks on boundary conditions and extensions.

We will sketch in this section some adaptations in order to apply the scheme to more realistic problems.

5.1 – More general forms of the advection–diffusion equation

We may consider nonconstant or anisotropic diffusion, nonconstant λ, time–dependent f and g. In such a general setting, the equation under consideration becomes

$$\begin{cases} v_t(x,t) = \displaystyle\sum_{i,j=1}^{N} a_{ij}(x,t)\frac{\partial^2}{\partial x_i \partial x_j}v(x,t) + f(x,t)\cdot\nabla v(x,t) + c(x,t)v(x,t) + g(x,t) \\ v(x,0) = v_0(x) \end{cases} \tag{5.1}$$

with $A(\cdot,\cdot) = (a_{ij}(\cdot,\cdot))$ a symmetric positive semidefinite matrix, thus including degenerate diffusions. The representation formula for the solution becomes now

$$v(x,t) = E\left\{\int_0^t e^{\int_0^s c(y(\sigma),\sigma)d\sigma} g(y(s),s)ds + e^{\int_0^t c(y(\sigma),\sigma)d\sigma} v_0(y(t),t)\right\}, \tag{5.2}$$

$y(t)$ being now the solution of the SDE:

$$\begin{cases} dy(\sigma) = f(y(\sigma), t-\sigma)ds + \sqrt{2}B(y(\sigma), t-\sigma)dW \\ y(0) = x. \end{cases} \tag{5.3}$$

where B is an $N \times N$ matrix such that $BB^t = A$.

The extension of the scheme (2.12) to this case requires to treat the nonautonomous SDE (5.3) instead of the autonomous SDE (0.5). This results in more complex schemes; however for the schemes presented in section 1 (including extrapolation) this is possible with a reasonable increase in computational complexity. Higher order schemes may lead to overly complex forms.

It is worth to point out that the case of a degenerate second order operator does not introduce special problems in the discretization of (5.2)–(5.3); simply, the diffusion added in (5.3) is itself degenerate. This is a remarkable advantage of this technique in treating strongly advection–dominated problems.

5.2 – Dirichlet boundary conditions

The theory exposed so far applies for equations posed in $\Omega = I\!R^N$. Assume now a Dirichlet boundary condition is imposed on the boundary of Ω, that is

$$v(x,t) = b(x) \qquad \text{for } x \in \partial\Omega,\ t \in [0,T]. \tag{5.4}$$

In terms of trajectories of the SDE (0.5) (or (5.3)), this means that defining the exit time of $y(x,t)$ from Ω as:

$$\theta(x) := \inf\{t \geq 0 : y(x,t) \in I\!R^N \setminus \Omega\}$$

the representation formula (5.2) must be changed into

$$v(x,t) =$$

$$= E\left\{\int_0^{t\wedge\theta(x)} e^{\int_0^s c(y(\sigma),\sigma)d\sigma} g(y(s),s)ds + e^{\int_0^{t\wedge\theta(x)} c(y(\sigma),\sigma)d\sigma} v_0(y(t\wedge\theta(x)), t\wedge\theta(x))\right\}, \tag{5.5}$$

where $a \wedge b = \min(a,b)$, finite or $+\infty$, and $v_0(y(\theta(x)),\theta(x)) = b(\theta(x))$.

In the discrete approximation, it is required to determine the point at which the discrete trajectory defined by (2.1) crosses the boundary of Ω. This can be done numerically by a variable time step (a detailed description of the variable time step technique is given in [FF]).

5.3 – Neumann boundary conditions

It is also possible to treat Neumann boundary conditions. In this case, the representation formula (5.2), (5.3) still holds, but the solution of (5.3) should now be *reflected* by the boundary instead of being *stopped* as in (5.5).

6. Numerical tests.

This section gives some simple one–dimensional tests intended to illustrate and check the previous convergence results. As discussed in [FFM], the error estimate (3.11) shows a nontrivial interplay between the two discretization steps; the efficiency of the scheme is maximized setting

$$\Delta t = \Delta x^{\frac{r}{p+1}} \tag{6.1}$$

and with this choice the convergence rate of the scheme is $\frac{rp}{p+1}$ (with respect to the only independent discretization step Δx).

Numerical solutions have been computed with respectively 65, 257 and 1025 nodes. In order to compute the convergence rate of the scheme, we use extrapolation on the errors ϵ between the steps Δx, $\Delta x/4$ and $\Delta x/16$, according to the relationship

$$\frac{\epsilon_{\Delta x} - \epsilon_{\Delta x/4}}{\epsilon_{\Delta x/4} - \epsilon_{\Delta x/16}} \approx 4^{\alpha}$$

that is,

$$\alpha = \frac{1}{\ln 4} \ln \frac{\epsilon_{\Delta x} - \epsilon_{\Delta x/4}}{\epsilon_{\Delta x/4} - \epsilon_{\Delta x/16}} \tag{6.2}$$

which holds if $\epsilon \approx C\Delta x^{\alpha}$, allowing to evaluate the convergence rate from the difference between numerical errors, that is, between numerical solutions (this does not require to know the exact solution). Results are summarized in tables 1–8. The values of the approximate solution are shown for different time steps and at different points in the computational grid. In the last row of the tables the approximate convergence rate is computed by means of (6.2) and compared with the theoretical one (in brackets). If (6.2) cannot be applied, the field is omitted. Typically, this happens if the numerical values do not converge monotonically, this possibly meaning that the scheme has not reached its theoretical asymptotic behaviour (but see also the discussion of example 3).

Example 1. We consider the equation

$$\begin{cases} v_t = \nu v_{xx} + x v_x + 1 \\ v_0(x) = x \end{cases}$$

posed in $\Omega = (-1, 1)$, with boundary conditions $v(-1, t) = -1$, $v(1, t) = 1$, and $\nu = 10^{-2}$, so that the equation is clearly advection–dominated. The approximations are computed at $t = 2$, with different coupling of a one–step scheme with a spatial reconstruction. Figure 1 shows the numerical solution obtained with Heun/P_2 coupling.

Table 1 refers, for comparison, to the first order scheme Euler/P_1. Tables 2–4 are obtained using a second order (Heun) scheme coupled with respectively P_1, P_2 and cubic reconstructions. The theoretical forecast is well respected, and the advantage in using higher order schemes is apparent from the comparison with the Euler/P_1 scheme.

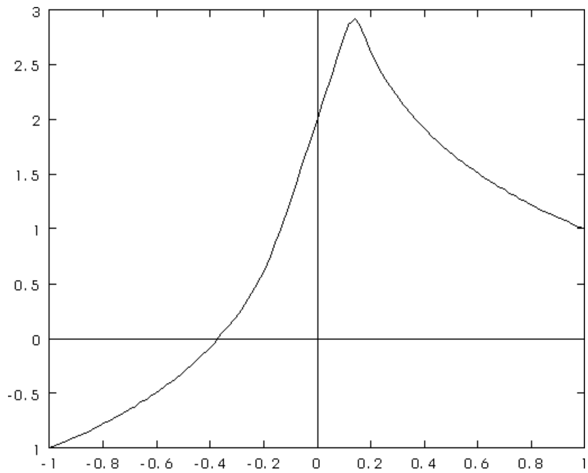

Figure 1. Approximate solution for Example 1.

Example 2. We consider again equation

$$\begin{cases} v_t = \nu v_{xx} + x v_x + 1 \\ v_0(x) = x \end{cases}$$

with $\Omega = (-1, 1)$, $v(-1, t) = -1$ and $v(1, t) = 1$, but now $\nu = 0.1$, so that the contributions of advective and diffusive parts are comparable. Figure 2 shows the numerical solution, again obtained with Heun/P_2 coupling.

In this case, experimental convergence rates do not match theoretical analysis. This suggests that the scheme has poorer performances in problems which are not advection–dominated.

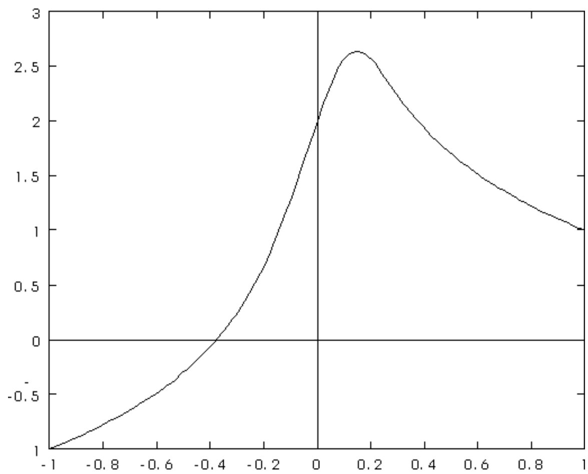

Figure 2. Approximate solution for Example 2.

Example 3. In order to analyse in greater detail the behaviour of the scheme for equations with a relevant diffusion, we consider now the heat equation

$$\begin{cases} v_t = v_{xx} \\ v_0(x) = \begin{cases} 1 & \text{if } -0.5 \le x \le 0.5 \\ 0 & \text{else} \end{cases} \end{cases}$$

again with $\Omega = (-1, 1)$, $v(-1, t) = v(1, t) = 0$. Figures 3–5 show the approximate solutions at $t = 1$ with a Heun/cubic discretization for different time steps, at fixed space step $\Delta x = 0.02$. The highly oscillating behaviour of the scheme for the larger time steps can be explained with an example. Assume the simplified situation of (4.1) with $\nu = 1/2$, $a = 0$, $\Delta t = \Delta x = 0.01$ and an Euler/P_1 approximation. Then, the two "upwind" points for the node x_j are $x_j \pm \sqrt{2\nu\Delta t} = x_{j\pm 10}$, which are the numerical domain of dependence at the previous time step. Going backwards one more time step, the numerical domain of dependence is made by the points x_j, x_{j-20} and x_{j+20}, and so forth. It is apparent that the domain of dependence develops "holes" which cause the oscillations seen in figures 3 and 4. However, the situation in which the upwind points precisely coincide with other grid nodes is unlikely. In a more realistic setting, the numerical domain of dependence would also include one node left and one node right of the point $x_j \pm \sqrt{2\nu\Delta t}$. Therefore, a "hole" of width $2\sqrt{2\nu\Delta t}$ could be filled up in a numer of steps

$$k = \frac{2\sqrt{2\nu\Delta t}}{2\Delta x}$$

so that keeping fixed the final time T, the condition $T \ge k\Delta t$ (which would be necessary to avoid such "holes" in the numerical domain of dependence) yelds

$$\Delta t \le \frac{T^{2/3}\Delta x^{2/3}}{(2\nu)^{1/3}}. \tag{6.3}$$

In practice, it would be reasonable to require the stronger condition

$$\Delta t << \frac{T^{2/3}\Delta x^{2/3}}{(2\nu)^{1/3}}. \tag{6.4}$$

in order to obtain a more regular dependence. Although this condition has been obtained for the Euler/P_1 coupling, a similar analysis in other cases would coincide up to a constant factor on the right–hand side. For example, the case of Heun/cubic scheme in example 3 would give

$$\Delta t << \frac{2T^{2/3}\Delta x^{2/3}}{(3\nu)^{1/3}} \approx 0.1 \tag{6.5}$$

which could well explain the situation of figures 3–5.

It is interesting to note that this condition is better satisfied with strongly advection–dominated problems (ν small) or in the large–time simulation (T large). In the limit, this condition is always satisfied when the stationary solution is computed ($T = \infty$), as in [CF] and [BF]. The relationship Δt vs. Δx seems not to be critical since the right–hand side of (6.4) is $O(\Delta x^{2/3})$; in particular, it is possible to work at constant or slightly increasing Courant numbers (the limiting case $\Delta t \sim \Delta x^{2/3}$ is precisely the situation of tables 2 and 6).

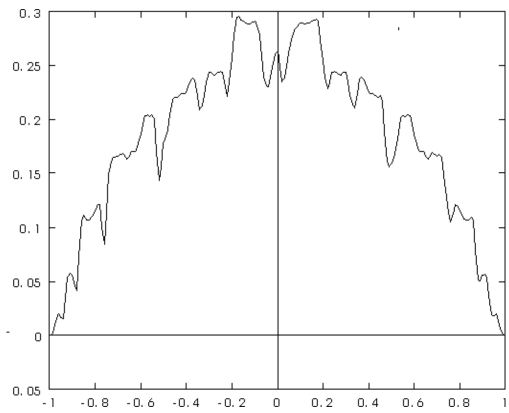

Figure 3. Approximate solution for Example 3, $\Delta t = 0.1$.

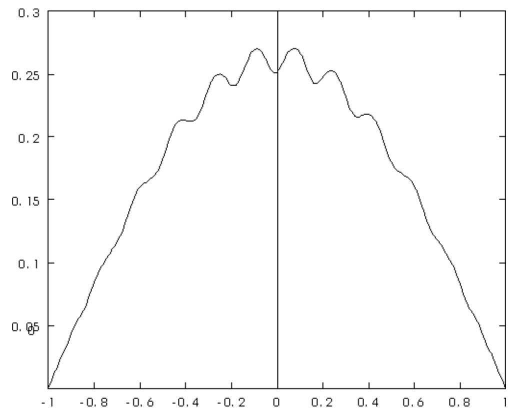

Figure 4. Approximate solution for Example 3, $\Delta t = 0.01$.

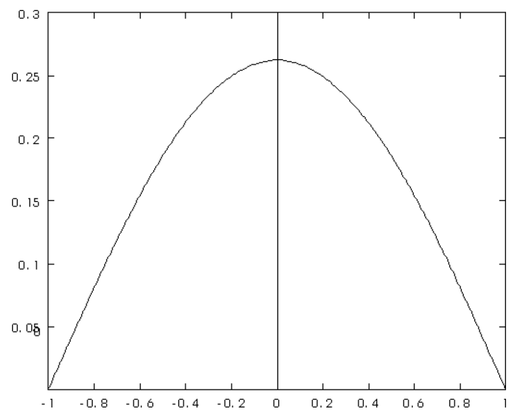

Figure 5. Approximate solution for Example 3, $\Delta t = 0.001$.

Euler / P_1 ($\Delta t = \Delta x$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.434038	1.999890	2.429296
$\Delta x = 0.0078125$	0.398112	1.999999	2.398112
$\Delta x = 0.001953125$	0.389260	1.999999	2.389260
rate	$\alpha = 1.01$ (1.0)		$\alpha = 0.91$ (1.0)

Heun / P_1 ($\Delta t = \Delta x^{2/3}$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.393379	1.999999	2.393375
$\Delta x = 0.0078125$	0.387978	2.000000	2.387978
$\Delta x = 0.001953125$	0.386861	1.999999	2.386861
rate	$\alpha = 1.14$ (1.33)		$\alpha = 1.14$ (1.33)

Heun / P_2 ($\Delta t = \Delta x$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.386787	2.000000	2.386791
$\Delta x = 0.0078125$	0.386674	1.999999	2.386674
$\Delta x = 0.001953125$	0.386669	1.999999	2.386669
rate	$\alpha = 2.25$ (2.0)		$\alpha = 2.27$ (2.0)

Heun / **cubic** ($\Delta t = \Delta x^{4/3}$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.382849	2.001707	2.404162
$\Delta x = 0.0078125$	0.386625	2.000004	2.386689
$\Delta x = 0.001953125$	0.386669	2.000003	2.386669
rate	$\alpha = 3.21$ (2.66)		$\alpha = 4.89$ (2.66)

Tables 1 − 4. Values of numerical solutions and approximate convergence rates for Example 1.

Euler / P_1 ($\Delta t = \Delta x$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.465006	1.972907	2.406712
$\Delta x = 0.0078125$	0.438381	1.988490	2.408092
$\Delta x = 0.001953125$	0.429544	1.990077	2.405930
rate	$\alpha = 0.79$ (1.0)	$\alpha = 1.65$ (1.0)	

Heun / P_1 ($\Delta t = \Delta x^{2/3}$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.427223	1.991334	2.401383
$\Delta x = 0.0078125$	0.425345	1.991883	2.404433
$\Delta x = 0.001953125$	0.425968	1.991479	2.405044
rate			$\alpha = 1.16$ (1.33)

Heun / P_2 ($\Delta t = \Delta x$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.424727	1.992111	2.405313
$\Delta x = 0.0078125$	0.426398	1.991238	2.405104
$\Delta x = 0.001953125$	0.426863	1.990982	2.405015
rate	$\alpha = 0.92$ (2.0)	$\alpha = 0.88$ (2.0)	$\alpha = 0.62$ (2.0)

Heun / **cubic** ($\Delta t = \Delta x^{4/3}$)			
step	$x = -0.25$	$x = 0.0$	$x = 0.25$
$\Delta x = 0.03125$	0.428523	1.997695	2.420991
$\Delta x = 0.0078125$	0.426955	1.991372	2.405737
$\Delta x = 0.001953125$	0.427003	1.990809	2.404994
rate		$\alpha = 1.74$ (2.66)	$\alpha = 2.18$ (2.66)

Tables 5 – 8. Values of numerical solutions and approximate convergence rates for Example 2.

REFERENCES

[BF] M. Briani and M. Falcone, *A semi–lagrangian scheme for advection–diffusion equations*, preprint

[CF] F. Camilli and M. Falcone, *An approximation scheme for the optimal control of diffusion processes*, RAIRO Model. Math. Anal. Numer. **29** (1995), 97–122

[DR] J. Douglas Jr. and T. F. Russell, *Numerical methods for convection–dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures*, SIAM J. Num. Anal. **19** (1982), 871-885

[FF] M. Falcone and R. Ferretti, *Convergence analysis for a class of semi–Lagrangian advection schemes*, SIAM J. Num. Anal. **35** (1998), 909–940

[FFM] M. Falcone, R. Ferretti and T. Manfroni, *Optimal discretization steps in semi–lagrangian approximation of first–order PDEs*, preprint

[KP] P.E. Kloeden and E. Platen, *Numerical solution of stochastic differential equations*, Springer–Verlag, Berlin, 1992

[HLB] Y. Hasbani, E. Livne and M. Bercovier, *Finite elements and characteristics applied to advection–diffusion equations*, Comput. & Fluids **11** (1983), 71–83

[LLS] D. Y. Le Roux, C. A. Lin and A. Staniforth, *An accurate interpolating scheme for semi–Lagrangian advection on an unstructured mesh for ocean modelling*, Tellus **49A** (1997), 119–138

[MPS] K. W. Morton, A. Priestley And E. Süli, *Stability of the Lagrange–Galerkin method with non–exact integration*, RAIRO Model. Math. Anal. Numer. **22** (1988), 625–653

[P] O. Pironneau, *On the transport–diffusion algorithm and its applications to the Navier–Stokes equations*, Num. Math. **38** (1982), 309–332

[RW] P.J. Rasch and D.L. Williamson, *On shape–preserving interpolation and semi–Lagrangian transport*, SIAM J. Sci. Stat. Comput. **11** (1990), 656–687

[RM] R. D. Richtmeyer and K. W. Morton, *Difference methods for initial value problems*, Interscience, New York, 1967

[R] A. Robert, *A stable numerical integration scheme for the primitive meteorological equations*, Atmos. Ocean. **19** (1981), 35–46

[SC] A. Staniforth and J. Coté, *Semi-Lagrangian integration schemes for atmospheric models - a review*, Monthly Weather Review **119** (1991), 2206–2223

[SW] E. Süli and A. Ware, *A spectral method of characteristics for hyperbolic problems*, SIAM J. Numer. Anal. **28** (1991), 423–445