A "parareal" time discretization for non-linear PDE's with application to the pricing of an American put

Guillaume Bal[1] and Yvon Maday[2]

[1] Department of Applied Physics & Applied Mathematics, Columbia University, New York, NY 10027, USA
[2] Laboratoire d'Analyse Numérique, Université Pierre et Marie Curie, Boîte courrier 187, 75252 Paris Cedex 05, France

Abstract. In this paper we introduce a new implementation of the "parareal" time discretization aimed at solving unsteady nonlinear problems more efficiently, in particular those involving non-differentiable partial differential equations. As in the former implementation [3], the main goal of this scheme is to parallelize the time discretization to obtain an important speed up. As an application in financial mathematics we consider the Black-Scholes equations for an American put. Numerical evidence of the important savings in computational time is also presented.

1 Introduction

In a recent note [3], the concept of "parareal" time discretization has been introduced to speed up the simulation of time-dependent partial differential equations through the use of parallel architectures. This discretization is based on

- the decomposition of the time interval $(0, T)$ into a sequence of sub-intervals (T^n, T^{n+1}) of size ΔT ($\Delta T = T/N$ for some integer N)
- the iterative use of two time discretization schemes, the first one with time step ΔT used over $(0, T)$, the second one with a smaller time step δt ($\delta t = \Delta T/M$ for some integer M) used over each sub-interval (T^n, T^{n+1}) independently, thus possibly in parallel.

As shown in [3], the global accuracy of the iterative process after a few iterations is comparable to that of the fine discretization of time step δt used over the whole interval $(0, T)$. Assuming that a large number of processors are at hand, this new discretization may allow us to get real time solutions when the fine discretization fails to do so. We refer to [1] and [2] for other approaches of parallelization in time that seem limited to differential systems.

In [3], the method has been proposed and implemented on linear and nonlinear parabolic problems. Here we introduce a slightly modified implementation, which turns out to be equivalent for linear problems, but gives better answers for nonlinear problems and more importantly allows us to tackle non-differentiable problems (whereas the former implementation was based on the linearization of the PDE). An example of such non-differentiable problems is provided by the Black-Scholes equation for an American put in financial mathematics to which we dedicate our last section.

2 Problem formulation

We consider the following first-order time-dependent partial differential equation

$$F(\partial_t u, u) = 0, \tag{1}$$

where F is a functional of $\partial_t u$, u, and possibly derivatives of u with respect to other variables than time. To solve (1) on $(0,T)$ with initial condition $u(t=0) = u_0$ approximately, assuming that a unique solution exists, we have at our disposal a one step "coarse" solver

$$E_{\Delta T}(T^n, U^n; U^{n+1}) = 0, \tag{2}$$

which to an approximation U^n of $u(T^n)$ yields U^{n+1}, an approximation of $u(T^{n+1})$. It is standard that using this iterative scheme allows us to define an approximation of the solution $u(t)$ for each $t = T^n$, $n = 1, .., N$. We assume in addition that we have a fine, one step solver

$$\mathcal{F}_{\delta t}(s, \sigma, \varphi; \psi) = 0,$$

which, to any $s \in (0,T), \sigma \in (0, T-s)$ and φ associates an approximation ψ of the solution $u(s+\sigma)$ to (1) with initial condition $u(t=s) = \varphi$.

Our objective is to devise a method that combines the use of the coarse solver on the whole interval $(0,T)$ and the fine solver in parallel on each small intervals (T^n, T^{n+1}) and allows us to provide an approximation of u over the whole interval $(0,T)$ with the same accuracy as a global use of $\mathcal{F}$ (with $s = 0$ and $\sigma = T$). To simplify notation, we assume that the fine solver is exact in the remainder of this section .

The iterative scheme is defined as follows. For $k = 1$, we compute a coarse approximation of each $u(T^n)$:

$$\begin{aligned} &E_{\Delta T}(T^n, U_1^n; U_1^{n+1}) = 0 \\ &U_1^0 = u_0. \end{aligned} \tag{3}$$

Then for $k \geq 1$, we define the solutions

$$\begin{aligned} &F(\partial_t y_k^n, y_k^n) = 0 \\ &y_k^n(n\Delta T) = U_k^n, \end{aligned} \tag{4}$$

for $0 \leq n \leq N-1$; next the jumps

$$S_k^n = y_k^{n-1}(n\Delta T) - U_k^n, \tag{5}$$

for $1 \leq n \leq N$; and finally the corrected coarse solutions

$$\begin{aligned} &E_{\Delta T}(T^n, U^n_{k+1}; U^{n+1}_{k+1} - \sum_{\ell=1}^{k} S^{n+1}_\ell) = 0, \quad 1 \leq n \leq N \\ &U^0_{k+1} = u_0. \end{aligned} \tag{6}$$

Another definition of the scheme can be done by introducing a predictor and a corrector. We can say that we first apply the coarse solver,

$$E_{\Delta T}(T^n, U^n_{k+1}; \tilde{U}^{n+1}_{k+1}) = 0$$

and then add the successive corrective jumps, $U^{n+1}_{k+1} = \tilde{U}^{n+1}_{k+1} + \sum_{\ell=1}^{k} S^{n+1}_\ell$.

Remark 1. The iterative scheme is optimal in the sense that if U^m_{k-1} is exact for $m \leq n$, then so is U^{n+1}_k.

Remark 2. It is also interesting to understand what has to be done if a good approximation U^n_0 of $u(T^n)$ is available for $n = 1, .., N$. Before defining the fine solutions (4), we have to initialize the corrector by introducing $S^n_0 = \tilde{U}^n_0 - U^n_0$ with $\tilde{U}^n_0$ defined by

$$E_{\Delta T}(T^{n-1}, U^{n-1}_0; \tilde{U}^n_0) = 0,$$

and then proceed as before by replacing the sum in (6) by $\sum_{\ell=0}^{k} S^{n+1}_\ell$.

3 Analysis of the algorithm for linear PDE's

Let us consider the following linear partial differential equation

$$\frac{\partial u}{\partial t} + A(u) = 0,$$

where A is a linear operator over a Hilbert space X and u is searched in $\mathcal{C}^1(0, T; X)$. This problem can be put in the framework of (1) by setting $F(a, b) = a + A(b)$.

Let us first mention that, for this simple situation, the current formulation is actually equivalent to that in [3]. We can indeed show that

$$U^{n+1}_{k+1} = U^{n+1}_k + S^{n+1}_k + \mathrm{prop}\{S^m_{k+1}\}^n_{m=1}$$

where $\mathrm{prop}\{S^m_{k+1}\}^n_{m=1}$ is the propagated jump at step $k+1$ and time n by the coarse solver (see [3] for the notations and definition of the propagated "jump"). To avoid confusion between the different formulations of the "parareal" scheme, we refer to [3] for additional details.

In the remainder of this section we prove that the accuracy of the current formulation of the "parareal" scheme increases with the number of iterations, first for a linear scalar differential equation, and then for more general linear parabolic equations.

3.1 Analysis of the algorithm on a differential equation

Let us first consider the iterative scheme for the simple linear equation $F(a,b) = a + \lambda b$, where $\lambda \geq 0$, or equivalently

$$\begin{aligned} \partial_t u + \lambda u &= 0 \\ u(t=0) &= u_0. \end{aligned} \tag{7}$$

Since the equation is linear, we assume that $u_0 = 1$ to simplify. First-order approximations of this equation are obtained for instance by

$$E_{\Delta T}(t, b; a) = \frac{a-b}{\Delta T} + \lambda \begin{cases} a & \text{implicit scheme} \\ b & \text{explicit scheme.} \end{cases} \tag{8}$$

For the implicit scheme, introducing $\delta = \lambda \Delta T$, we easily check that $U_0^n = (1+\delta)^{-n} = e^{-\delta n} + O(n\delta^2)$, where $e^{-\delta n} = u(T^n)$ is the exact solution. We then observe that

$$S_k^n = e^{-\delta} U_{k-1}^{n-1} - U_{k-1}^n,$$

and therefore

$$U_k^{n+1} = \frac{1}{1+\delta} U_k^n + \sum_{l=1}^{k} (e^{-\delta} U_{l-1}^n - U_{l-1}^{n+1}), \tag{9}$$

hence, subtracting the same relation with k replaced by $k-1$,

$$U_k^{n+1} = \frac{1}{1+\delta}(U_k^n - U_{k-1}^n) + e^{-\delta} U_{k-1}^n. \tag{10}$$

Let us define $V_k^n = e^{n\delta} U_k^n$, which is supposed to converge to 1 as $k \to \infty$. Upon multiplying the above expression by $e^{n\delta}$, we have

$$V_k^{n+1} = V_{k-1}^n + f(\delta)(V_k^n - V_{k-1}^n), \tag{11}$$

where

$$f(\delta) = \begin{cases} e^{\delta}(1+\delta)^{-1} & \text{implicit scheme} \\ e^{\delta}(1-\delta) & \text{explicit scheme.} \end{cases} \tag{12}$$

We say that the scheme is of *order* m if

$$|1 - f(\delta)| \leq C_f \delta^{m+1}. \tag{13}$$

Both the explicit and implicit schemes in (12) are of order 1. We can however construct higher order schemes, for instance by following the Richardson rule. For U^n given, we apply the implicit scheme of time step $\Delta T/2$ twice to obtain V and the implicit scheme of time step ΔT once to obtain W. We then define

$U^{n+1} = 2V - W$. We easily check that this scheme is of order 2 and that $U^N - u(T) = O((\varDelta T)^2)$. The corresponding function f is

$$f(\delta) = e^{\delta}\Big(\frac{2}{\Big(1+\dfrac{\delta}{2}\Big)^2} - \frac{1}{1+\delta}\Big) = 1 + O(\delta^3).$$

Let us now define the error terms ε_n^k via

$$V_k^n = e^{\delta n} U_k^n = 1 + \varepsilon_k^n. \tag{14}$$

We have the following

Proposition 1. *Let $u(t)$ be the solution of (7) and U_k^n the solution of the iterative scheme (3)—(6), where $E_{\varDelta T}$ is a coarse scheme of order $m \geq 1$ according to the definition (13).*

Then the error terms ε_k^n satisfy the following estimate

$$|\varepsilon_k^n| \leq C_k n^{k+1} \delta^{(m+1)(k+1)}. \tag{15}$$

In particular, we have $U_k^N = u(T) + O(\delta^{m(k+1)})$ and $y_k^n(t) = u(t) + O(n^k \delta^{(m+1)k})$.

Proof. Let us denote $f = f(\delta)$. From (11) and (14), we have

$$\varepsilon_k^{n+1} = f(\varepsilon_k^n - \varepsilon_{k-1}^n) + \varepsilon_{k-1}^n,$$

or equivalently

$$\varepsilon_k^{n+1} = f\varepsilon_k^n + (1-f)\varepsilon_{k-1}^n. \tag{16}$$

Upon solving the above equations successively, we verify that

$$\varepsilon_k^{n+1} = (1-f)\sum_{p=1}^{n} f^{n-p}\varepsilon_{k-1}^p \tag{17}$$

because $\varepsilon_{k-1}^0 = 0$ for all $k \geq 1$. Let us now prove the assertion by induction. We verify that

$$\varepsilon_0^n = 1 - f^n = (1-f)\sum_{p=0}^{n-1} f^p.$$

Notice that f^n is bounded independent of n so $|\varepsilon_0^n| \leq C_0 n \delta^{m+1}$. Let us now assume that $|\varepsilon_{k-1}^n| \leq C_{k-1} n^k \delta^{(m+1)k}$. We then deduce from (17) that

$$|\varepsilon_k^{n+1}| \leq C\delta^{m+1}\sum_{p=1}^{n} p^k \delta^{(m+1)k} \leq C_k (n+1)^{k+1}\delta^{(m+1)(k+1)},$$

since $\sum_{p=1}^n p^k \leq n^{k+1}$. This concludes the proof of the proposition.

3.2 Analysis of the algorithm on a parabolic equation

We now generalize the above analysis to the following type of parabolic problems

$$\begin{aligned} &\partial_t u + P(u) = 0 \\ &u(t=0,x) = u_0(x), \end{aligned} \tag{18}$$

where $P(u)$ is a pseudo-differential operator

$$P(u) = \widehat{P(\xi)\hat{u}(\xi)},$$

with symbol $P(\xi) \geq 0$. Common examples are the heat equation, with $P(u) = -\partial^2_{xx} u$ and $P(\xi) = \xi^2$, and its centered spatial discretization of mesh size h, with $P(u) = h^{-2}(2u(x) - u(x+h) - u(x-h))$ and $P(\xi) = 2h^{-2}(1 - \cos h\xi)$. In the Fourier domain, the problem reads

$$\begin{aligned} &\partial_t \hat{u} + P(\xi)\hat{u} = 0 \\ &\hat{u}(t=0,\xi) = \hat{u}_0(\xi), \end{aligned} \tag{19}$$

whose solution is given by

$$\hat{u}(t,\xi) = e^{-P(\xi)t}\hat{u}_0(\xi).$$

This section analyzes the convergence of the iterative scheme for the implicit Euler time discretization. In the Fourier domain, the implicit discretization is then given by

$$\hat{U}^{n+1}(\xi) = \frac{1}{1 + P(\xi)\Delta T}\hat{U}^n(\xi).$$

We set $\delta = \delta(\xi) = P(\xi)\Delta T$. Different frequencies satisfy here uncoupled equations. We introduce the function

$$f(\delta) = e^{\delta}\frac{1}{1+\delta}.$$

Let us define the sequence $\hat{U}^n_k(\xi)$ as the solution to the iterative scheme (3)—(6) with initial condition $\hat{u}_0(\xi) = 1$ and define $\eta^n_k = \eta^n_k(\xi)$ as

$$\eta^n_k = e^{-\delta n} - \hat{U}^n_k. \tag{20}$$

Following the proof of proposition (1), we verify that

$$\begin{aligned} &\eta^{n+1}_k = e^{-\delta} f(\delta)\eta^n_k + e^{-\delta}(1 - f(\delta))\eta^n_{k-1} \\ &\eta^n_0 = e^{-\delta n} - (e^{-\delta} f(\delta))^n, \end{aligned}$$

with $\eta^0_k = 0$ for $k \geq 0$, assuming that $\hat{u}_0(\xi) = 1$. We will derive an estimate for η^n_k as $n \to \infty$ for small values of k. For this purpose, we introduce

$$\theta^n_k = -\frac{e^{n\delta}}{f^{n-k}(\delta)(1 - f(\delta))^k}\eta^n_k. \tag{21}$$

An easy calculation shows that

$$\begin{aligned} &\theta_k^{n+1} = \theta_k^n + \theta_{k-1}^n \\ &\theta_0^n = 1 - f^{-n}(\delta). \end{aligned} \tag{22}$$

We easily verify that $f(\delta) \geq 1$. The terms θ_n^k are all positive. Moreover, we have $\theta_k^n \leq (f-1)f^{-1}\tilde{\theta}_k^n$, where

$$\begin{aligned} &\tilde{\theta}_k^{n+1} = \tilde{\theta}_k^n + \tilde{\theta}_{k-1}^n, \qquad k \geq 0, \\ &\tilde{\theta}_{-1}^n = 1, \quad n \geq 0, \\ &\tilde{\theta}_k^0 = 0, \quad k \geq -1. \end{aligned}$$

Indeed, we recognize that

$$\tilde{\theta}_k^n = \binom{n}{k+1} = \frac{n!}{(k+1)!(n-k-1)!}$$

hence $\tilde{\theta}_0^n = n$. From this, we deduce the quite accurate estimate

$$0 \leq (-1)^k \eta_k^n \leq \Big(e^{-\delta} f(\delta)\Big)^n \Big(\frac{f(\delta)-1}{f(\delta)}\Big)^{k+1} \binom{n}{k+1}. \tag{23}$$

It is easy to check that the above estimate implies that of proposition 1 when $\delta = P(\xi)\Delta T$ with λ of order 1 and $n \leq N$. Moreover, we directly verify that $\eta_k^n = 0$ for $k \geq n$ (which means that U_n^k is exact for $k \geq n$). For a fixed λ of order 1, δ converges to 0 with ΔT and we obtain the results of proposition 1. For values of ξ such that $P(\xi)\Delta T$ is of order 1, the theory needs to be improved. There is no hope that such high frequencies are correctly calculated by the iterative scheme. However, we want to make sure that they do not pollute the global solution too much. This is done by obtaining a bound on η_k^n independent of the frequency ξ, or equivalently independent of δ. Let us introduce

$$\alpha_k^n = \sup_\delta |\eta_k^n(\delta)|.$$

We now show that for $k << n$, we have

$$\alpha_k^n \leq \Big(\frac{C(k+1)}{n}\Big)^{k+1}, \tag{24}$$

where C is a constant independent of k and n. We deduce therefore that the iterative scheme converges as $(\Delta T)^{k+1}$ for small values of k *independently* of $\delta = P(\xi)\Delta T$.

In order to bound α_k^n we are led to analyse the function

$$h(\delta) = \frac{1}{1+\delta}\Big(1 - \frac{1+\delta}{e^\delta}\Big)^{\dfrac{k+1}{n}}.$$

We verify that $h(0) = 0$ and $h(+\infty) = 0$. Maxima of h are obtained at those points δ verifying

$$\frac{(k+1)\delta}{n} = \frac{e^\delta}{1+\delta} - 1 = g(\delta). \tag{25}$$

We verify that $g''(\delta) > 0$ for $\delta > 0$, from which we deduce the existence of a unique point $\delta_0 > 0$ where $h(\delta_0)$ reaches its maximum. From (25), we obtain that for $(k+1)/n << 1$,

$$\delta_0 = \frac{2(k+1)}{n} + o(\frac{k}{n}),$$

hence, recalling (23)

$$\alpha_k^n \leq \Big(\frac{1}{1+2(k+1)/n}\Big)^n \Big(\frac{1}{2}\Big(\frac{2(k+1)}{n}\Big)^2\Big)^{k+1} \begin{pmatrix} n \\ k+1 \end{pmatrix}$$

for $k/n << 1$. Using

$$\begin{pmatrix} n \\ k+1 \end{pmatrix} \sim \frac{n^{k+1}}{(k+1)!},$$

and the Stirling formula

$$k! = \Big(\frac{2\pi}{k+1}\Big)^{1/2} e^{-(k+1)} (k+1)^{k+1} (1 + O(k^{-1})),$$

we deduce that

$$\alpha_k^n \leq \Big(\frac{2(k+1)}{e}\frac{1}{n}\Big)^{k+1},$$

for $1 << k << n$. For small values of k, the constant $2/e$ may need to be replaced by another constant. This proves our claim (24).

Let us return to the equation (19). Defining $\eta_k^n(\xi)$ as the error for a frequency ξ after k iterations of the iterative scheme, we have that

$$|\eta_k^n(\xi)| \leq \Big(\frac{C(k+1)}{n}\Big)^{k+1} |\hat{u}_0(\xi)|.$$

Hence, using the Parseval inequality, we deduce the main result of this section:

Proposition 2. *Let $u(t,x)$ be the solution of (18) and $U_k^n(x)$ the solution of the "parareal" scheme (3)—(6) with implicit Euler coarse discretization. Then we have the following estimate*

$$\|u(T,x) - U_k^n(x)\|_{L^2(\mathbb{R})} \leq \Big(\frac{C(k+1)}{n}\Big)^{k+1} \|u_0\|_{L^2(\mathbb{R})}. \tag{26}$$

This proves that the iterative scheme transforms the implicit Euler time discretization of order 1 into a time discretization of order $k+1$ for any linear parabolic problem.

In the actual implementation of the method, a scheme with fine time step δt is used in parallel over each (T^n, T^{n+1}). The gain in computation time obtained by using the iterative scheme is estimated as follows. The cost of the direct scheme, based on the sole use of the fine solver, is proportional to $T/\delta t$. The cost of the iterative scheme is proportional (with the same constant of proportionality) to $(k+1)(T/\Delta T + \Delta T/\delta t)$, with $n = (\Delta T)^{-1}$. For $1+k$ fixed, the iterative scheme cost is optimized provided $T/\Delta T = \Delta T/\delta t$ since the product $T/\delta t$ is fixed. This implies $\Delta T = \sqrt{T\delta t}$. The accuracy of the iterative scheme will be therefore comparable to the direct simulation provided $k+1=2$ since $\delta t = T^{-1}(\Delta T)^2$.

The maximal gain in time is then found to be

$$G = \frac{1}{4}\Big(\frac{T}{\delta t}\Big)^{1/2},$$

provided we have at our disposal $(T/\delta t)^{1/2}$ processors. For instance, with $T = 1$ and $\delta t = 10^{-6}$, we ideally obtain a gain of $G = 250$ provided we have 1000 processors and use 2 iterations of the “parareal” scheme.

4 Application to a nonlinear parabolic problem

Our purpose is now to apply the procedure to a non linear partial differential equation. In [3], we have considered the following problem

$$\frac{\partial u}{\partial t} - \Delta u + 5u^3 = 5 \cdot \sin(2 \cdot t), \tag{27}$$

over the unit disc Ω, with initial condition $u(t=0) = \tan(x^2 + y^2 - 1)$, and homogeneous boundary conditions.

We have compared the current implementation with the one of [3] since they do not coincide for nonlinear problems. The generalization to nonlinear problems was done in [3] through a linearization of the coarse time solver. The current scheme is designed to apply to any type of nonlinear equation.

The time interval is $(0, 10)$ and $\Delta T = 0.1, \delta t = 0.0008$. The results presented in Tab. 1 prove, on this example, that the current scheme is more efficient than the first one. The error is measured in the $L^\infty(0,T;L^2(\Omega))$–norm with respect to the exact solution (the latter is computed with a time step much smaller than δt) provided that we have 100 processors at our disposal, the solution with accuracy 2.8 10^{-4} comes about 14 times faster than if we would use a standard sequential scheme!

The purpose of the last section is to illustrate that this approach can indeed provide much faster solutions for nonlinear, non-differentiable problems.

5 Pricing of an American put

We consider in this section the pricing of an American option. Up to some change of variables, the American put solves in a simplified setting the fol-

k	1	2	3	4	5	6
current method	$7.4\,10^{-2}$	$1.0\,10^{-2}$	$1.3\,10^{-3}$	$2.8\,10^{-4}$	$2.8\,10^{-4}$	$2.8\,10^{-4}$
method [3]	$1.2\,10^{-2}$	$5.6\,10^{-3}$	$2.7\,10^{-3}$	$1.3\,10^{-3}$	$5.6\,10^{-4}$	$2.8\,10^{-4}$

Table 1. $L^\infty(O,T;L^2(\Omega))$ norm of the error for different values of k.

lowing non-linear equation [5]

$$\min(\partial_t u - \partial^2_{xx} u, u - g(x)) = 0, \tag{28}$$

with $u(t=0) = g(x) = \max(e^x - 1, 0)$. Notice that this equation can be written in the framework of the introduction $F(\partial_t u, u, \partial^2_{xx} u) = 0$, where $F(a,b,c) = \min(a-c, b-g(x))$ is a continuous non-differentiable functional.

It is possible to show that the above equation (28) admits a unique weak (viscosity) solution such that u and $\partial_t u$ are continuous. Moreover, there exists a *free boundary*, the optimal exercise boundary $\xi(t)$ such that $u(t,x) = g(x)$ for $x \le \xi(t)$ and $\partial_t u(t,x) - \partial^2_{xx} u(t,x) = 0$ for $x > \xi(t)$. The derivative $\partial_x u(t,x)$ is moreover continuous everywhere except at $x = \xi(t)$. An approximation of $u(t,x)$ (see below in which sense it is an approximation) is given in Fig. 1.

The discretization chosen to solve (28) approximately is a splitting method, based on solving the parabolic equation on an interval $(t, t+\Delta T)$ and then projecting the solution on the set of functions satisfying $u \ge g$. The implicit scheme is given by

$$\begin{array}{ll} \text{Step 1:} & \dfrac{U^{n+1/2} - U^n}{\Delta T} - \partial^2_{xx} U^{n+1/2} = 0 \\ \text{Step 2:} & U^{n+1} = \max\Big(g(x), U^{n+1/2}(x)\Big). \end{array} \tag{29}$$

Here again, $u^n(x)$ is an approximation of $u(n\Delta T, x)$.

Step 1 in (29) needs to be further discretized. We replace the space of "positions" $x \in I\!R$ by the interval $(-3,5)$ and impose the boundary conditions $u(t,-3) = g(-3)$ and $u(t,5) = g(5) = 0$. The latter condition is valid as long as $u(t,x)$ is small for x in the vicinity of 5. We see in Figure 1 that such is the case of t up to 1.25 at least but that this assumption no longer holds for $T = 2.5$. Solving the equation for $T \ge 2.5$ requires to solve the parabolic equation on a larger domain. We keep our domain $x \in (-3,5)$ for all times to simplify. We approximate the second-order derivative $\partial^2_{xx} u$ as

$$\partial^2_{xx} u \approx \frac{u_{m+1} + u_{m-1} - 2u_m}{(\Delta x)^2}, \tag{30}$$

where $u_m \approx u(-3 + m\Delta x)$ and Δx is a constant spatial step.

We are now ready to apply the iterative algorithm (3)—(6) to solve (28). We want to solve (28) for $(t,x) \in (0, 2.5) \times (-3, 5)$ using a time step $\Delta T = 0.05$. The "exact part" (4) is solved using the same splitting algorithm (29)

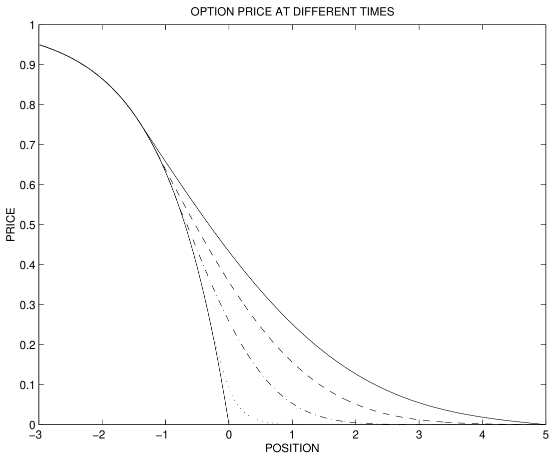

Fig. 1. Option price $u(t,x)$ for several values of time on the interval $(-3,5)$. Represented are the initial condition (solid line), the solution at $t=0.1$ (dotted line), at $t=0.5$ (dot-dashed line), at $t=1.25$ (dashed line) and $t=2.5$ (solid line).

with a time step $\delta t = 10^{-3}$. The number of points for the spatial discretization is 1500.

The errors on the interval (-3.5) for the four first iterations at final time $T=2.5$ are give in Fig. 2. A quantitative measure of the $L^\infty(-3,5)$ norm of

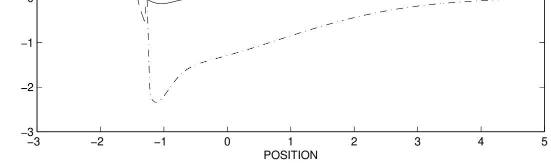

Fig. 2. Errors at time $T=2.5$ between the "exact solution" obtained by solving (28) with time step $\delta t = 10^{-3}$ and the iterative solutions $U_k^N(x)$ for $k=0$ (dotted line), $k=1$ (dot-dashed line), $k=2$ (dashed line), $k=3$ (solid line), and $k=4$ (gray dotted line).

the errors between the exact solution and $U_k^N(x)$, $\varepsilon_k(x) = u(T,x) - U_k^N(x)$ is presented in Tab. 2. These results confirm the exponential convergence of

k	0	1	2	3	4	5
$\|\varepsilon_k\|_{L^\infty}$	$5.5\,10^{-3}$	$2.3\,10^{-3}$	$5.7\,10^{-4}$	$1.3\,10^{-4}$	$2.9\,10^{-5}$	$6.5\,10^{-6}$
ρ_k	-	0.42	0.24	0.23	0.22	0.23

Table 2. L^∞ norm of the error $\varepsilon_k(x)$ for different values of k and radius of convergence of the method $\rho_k = \|\varepsilon_k\|_{L^\infty}/\|\varepsilon_{k-1}\|_{L^\infty}$.

$U_k^n(x)$ to the solution $u(t,x)$ as $k \to \infty$. However, unlike the convergence of order $O((\Delta T)^k)$ obtained for linear equations in section 3, the numerical results suggest a convergence of the form ρ^k, where ρ is independent of ΔT. There is no theoretical explanation to this fact yet.

The accuracy of solution calculated with the fine discretization $\delta t = 10^{-3}$ is $1.2\,10^{-4}$ in the L^∞ norm (it has been compared with a much finer discretization). We can therefore deduce that the solution of the iterative scheme is as accurate as the fine solution after 3 iterations of the "parareal" algorithm. The total gain in computational time is therefore $[T/\delta t]/[(k+1)(T/\Delta T + \Delta T/\delta t)] = 2500/400 = 6.25$ and requires $\Delta T/\delta t = 50$ processors.

An interesting information in the pricing of American options is the free boundary $\xi(t)$, separating the two regions where it is optimal to exercise the option $(u = g)$ and where it is not $(u > g)$. We also expect the free boundary $\xi_k(t)$ obtained at iteration k from $U_k^n(x)$ to converge to the exact free boundary $\xi(t)$ obtained from $u(t,x)$. This is confirmed in Fig. (3). Notice that $\xi_k(t)$ is exact at $T^k = k\Delta T$ since the solution $U_k^n(t)$ is exact for $n \le k$. This can be observed on the right-hand side of Fig. (3). The convergence of the free boundary is comparable to that of the solution (see Tab. 3). The radius of convergence of the method becomes less accurate for large values of k when errors of the order of the spatial discretization Δx are reached.

k	0	1	2	3	4	5
$\|\eta_k\|_{L^\infty}$	$1.7\,10^{-1}$	$4.8\,10^{-2}$	$1.8\,10^{-2}$	$4.0\,10^{-3}$	$1.3\,10^{-3}$	$8.6\,10^{-4}$
ρ_k	-	0.28	0.37	0.23	0.32	0.66

Table 3. L^∞ norm of the error $\eta_k(t) = \xi(t) - \xi_k(t)$ for different values of k and radius of convergence of the method $\rho_k = \|\eta_k\|_{L^\infty}/\|\eta_{k-1}\|_{L^\infty}$.

Acknowledgments. The numerical results in section 4 have been obtained thanks to the software FreeFem+ [4] and have benefitted from the help of Frédéric Hecht and Olivier Pironneau. The first author acknowledges partial support from NSF grant DMS-0072008.

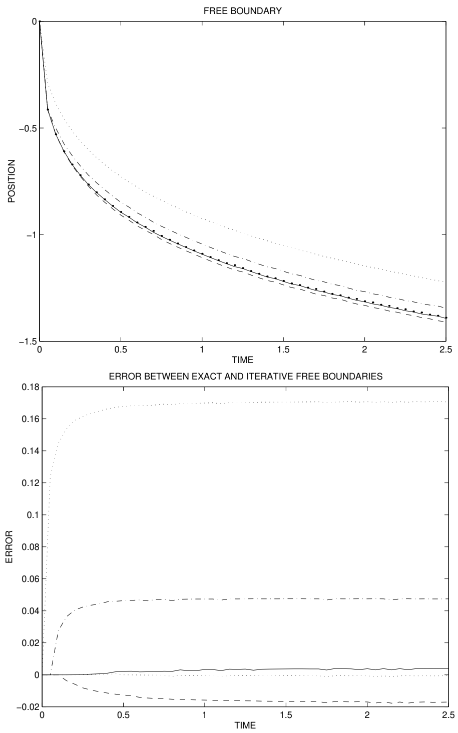

Fig. 3. Right figure: Free boundary $\xi(t)$ (solid line), $\xi_0(t)$ (dotted line), $\xi_1(t)$ (dot-dashed line), $\xi_2(t)$ (dashed line), $\xi_3(t)$ (points). Left figure: Error between the exact free boundary $\xi(t)$ and $\xi_k(t)$ for $k = 0$ (dotted line), $k = 1$ (dot-dashed line), $k = 2$ (dashed line), $k = 3$ (solid line), and $k = 4$ (gray dotted line).

References

1. P. Chartier and B. Philippe (1993) A parallel shooting technique for solving dissipative ODE's, Computing **51**, pp 209–236.
2. J. Erhel J. and S; Rault S. (2000) Algorithme parallèle pour le calcul d'orbites, Techniques et Sciences informatiques, **19**, pp 649–673.
3. J.-L. Lions, Y. Maday and G. Turinici (2001) Résolution d'EDP par un schéma en temps "pararéel", C. R. Acad. Sci. Paris Sér. I Math. **332** , no. 7, 661–668.
4. O. Pironneau, F. Hecht , D. Bernardi and K. Ohtsuka, FreeFem+, http://www-rocq.inria.fr/Frederic.Hecht/FreeFemPlus.htm.
5. P. Wilmott, H. Howison and J. Dewynne (1998) Option Pricing. Mathematical models and computation, Oxford financial press, Oxford.