

NON CONFORMING SPACE-TIME GRIDS FOR THE WAVE EQUATION : A NEW APPROACH

Laurence Halpern

Abstract. We present a new method to design non-conforming space-time algorithms for the wave equation. It relies on the use of the Schwarz Waveform Relaxation method, and allows strong velocity contrasts, together with the use of weakly dispersive meshes.

§1. Introduction

When solving evolution problems in heterogeneous media, it is often desirable to use non conforming grids in space and time, such a case may be when different time scales in different media are present. When dealing with wave propagation, this issue is most crucial, due to numerical dispersion. For example, let us consider the one dimensional wave equation

$$\frac{1}{c^2(x)} \frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0.$$

In order to minimize the dispersion, the mesh sizes need to be close to the CFL condition everywhere. When using the leap-frog scheme for instance, the CFL number $\gamma = c\Delta t/\Delta x$ must be smaller than 1. For $\gamma = 1$, the scheme is exact.

The question of designing a mesh refinement in time and space arises in two ways. The first application is when one wants to use a finer space discretisation in a small part Ω_1 , with a roughly constant speed. In order to keep γ close to 1, one has to discretize time in the same fashion. The second application is when the velocity is larger in a part Ω_1 , if a constant space mesh is required, a finer mesh in time is needed in Ω_1 . We summarize and generalize with I domains Ω_i such that

- (I) either c is constant in \mathbb{R} , the space meshes Δx_i are different, $c\Delta t_i/\Delta x_i$ is a constant ,
- (II) or the speed in each Ω_i is a constant c_i , the space mesh Δx is a constant, the time steps Δt_i are such that $c_i\Delta t_i/\Delta x$ is a constant.

In both cases γ has to be constant, as large as possible, depending on the scheme.

This problem has been widely studied in particular for the transport equation, in connection with problem (I). The first paper is [1], where the Lax-Wendroff scheme is proved to be stable with interpolatory transmission conditions. In [2, 3], the leapfrog scheme is studied. Stability problems for Dirichlet boundary data are reported. New transmission conditions using discrete energy estimates are proposed, and proved to be stable.

Our approach is totally different and addresses in particular problems (I) and (II). It relies on the use of Schwarz waveform relaxation algorithms (SWR). The wave equation in $\mathbf{R} \times (0, T)$ is first written as a collection of wave equations in $\Omega_i \times (0, T)$ with perfectly transmitting conditions on the boundaries between neighboring subdomains. The solution is calculated through a Schwarz algorithm. In the case of piecewise constant velocities, it converges in two iterations on appropriate time windows. We then discretize in time and space using finite volumes, which enables us to naturally take the transmission conditions into account. The space and time steps are chosen independently and optimally in each subdomain, and the solution is transmitted to the neighbour by a projection procedure. For the leapfrog scheme, the total procedure is stable and convergent in both cases. The tools are energy estimates in case (I) and Laplace transform in case (II). It is numerically shown to keep the second order accuracy.

§2. About the Schwarz Waveform relaxation method

We consider the second order , one dimensional wave equation with variable wave speed,

$$\mathcal{L}(u) = \frac{1}{c^2(x)} \frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0 \quad (1)$$

on the domain $\mathbb{R} \times (0, T)$ with initial conditions $u(\cdot, 0) = p$ and $\frac{\partial u}{\partial t}(\cdot, 0) = q$.

We first define the classical Schwarz waveform relaxation algorithm in the case of two subdomains. We introduce two overlapping subdomains $\Omega_1 = (-\infty, L)$ and $\Omega_2 = (0, +\infty)$. At step k , we solve two subproblems in $\Omega_i \times (0, T)$, with a Dirichlet data on the boundary given by the previous step in the other domain. The solution in $\Omega_i \times (0, T)$ at step k is called u_i^k . The classical Schwarz algorithm extended to space-time domains is then given by:

$$\begin{cases} (\frac{1}{c^2} \frac{\partial^2}{\partial t^2} - \frac{\partial^2}{\partial x^2}) u_i^{k+1} = 0 \text{ in } \Omega_i \times (0, T) \text{ for } i = 1, 2, \\ u_i^{k+1}(\cdot, 0) = p \text{ in } \Omega_i, \quad \frac{\partial u_i^{k+1}}{\partial t}(\cdot, 0) = q \text{ in } \Omega_i, \\ u_1^{k+1}(L, \cdot) = u_2^k(L, \cdot), \quad u_2^{k+1}(0, \cdot) = u_1^k(0, \cdot) \text{ in } (0, T). \end{cases} \quad (2)$$

Theorem 1. *For the Schwarz algorithm (2), convergence is achieved in a finite number of iterations, $k \geq \frac{T \sup c(x)}{L}$.*

Proof. The proof can be found in [5], and relies on the finite speed of propagation. We describe it here in the case of constant velocity. Consider the errors $U_i^k = u_i^k - u$. They satisfy system (2) with zero initial values. Using d'Alembert's formula, we have

$$\begin{aligned} \text{for } x - ct > 0, \quad U_2^{k+1}(x, t) &= 0, \\ \text{for } x - ct < 0, \quad U_2^{k+1}(x, t) &= U_1^k(0, t - \frac{x}{c}), \end{aligned} \quad (3)$$

and

$$\begin{aligned} \text{for } x + ct < L, \quad U_1^{k+1}(x, t) &= 0, \\ \text{for } x + ct > L, \quad U_1^{k+1}(x, t) &= U_2^k(L, t - \frac{L-x}{c}). \end{aligned} \quad (4)$$

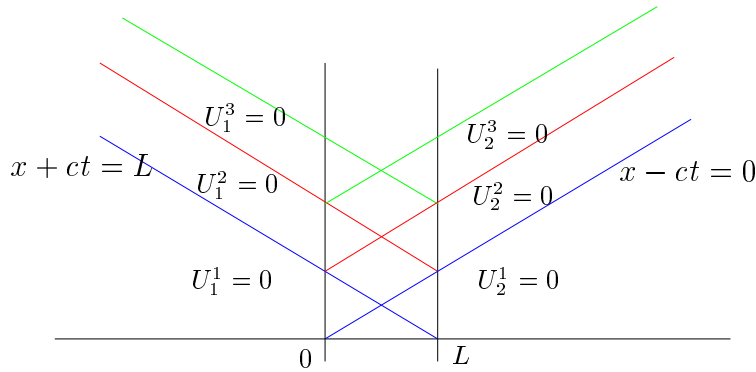


Figure 1: Evolution of the Schwarz algorithm with Dirichlet transmission conditions

We can now see the end of the proof on Figure 1 using (3) and (4) : at step k , U_1^k vanishes for $x + ct \leq kL$ and U_2^k vanishes for $x - ct \geq -(k-1)L$. Thus U_1^k and U_2^k vanish identically on the time interval $(0, T)$ if $T \leq kL/c$. \square

This shows first that the classical method, even with two subdomains, is extremely slow, and secondly that it needs a large overlap to increase the convergence speed. Note that for any transmission conditions,

the error U_1^k on the left is a function of $x - ct$ only, while on the right the error U_2^k is a function of $x + ct$ only. This in turn implies the following identities for $k \geq 1$ and any positive time

$$\begin{cases} \left(\frac{\partial}{\partial x} + \frac{1}{c} \frac{\partial}{\partial t} \right) U_2^k(L, t) = 0, \\ \left(\frac{\partial}{\partial x} - \frac{1}{c} \frac{\partial}{\partial t} \right) U_1^k(0, t) = 0. \end{cases}$$

This observation leads to the following simple but important

Theorem 2. *The transmission conditions defined by*

$$\begin{aligned} \left(\frac{\partial}{\partial x} + \frac{1}{c} \frac{\partial}{\partial t} \right) u_1^{k+1}(L, t) &= \left(\frac{\partial}{\partial x} + \frac{1}{c} \frac{\partial}{\partial t} \right) u_2^k(L, t), \\ \left(\frac{\partial}{\partial x} - \frac{1}{c} \frac{\partial}{\partial t} \right) u_2^{k+1}(0, t) &= \left(\frac{\partial}{\partial x} - \frac{1}{c} \frac{\partial}{\partial t} \right) u_1^k(0, t), \end{aligned} \quad (5)$$

lead to well-posed initial boundary value problems even without overlap and they are optimal: convergence in the Schwarz algorithm with these transmission conditions is achieved in two iterations, i.e. u_i^2 is identical to u in Ω_i .

Table 1 shows the convergence to the accuracy of the numerical scheme of the algorithm with transmission conditions (5), without overlap. The velocity is $c = 1$, the computation is done on $(0, T)$ with $T = 2$. The initial data are $u(x, 0) = e^{-50(0.5-x)^2}$, $\partial_t u(x, 0) = 0$. The domain $(0, 2)$ is divided into two subdomains $(0, 1)$ and $(1, 2)$. The initial guess $(u_i^0)_{i=1,2}$ is chosen to be 0. We use the scheme presented in section 3.

grid	error after 2 iterations	discretization error
50 x 50	2.6128e-04	2.1515e-02
100 x 100	2.7305e-05	4.9472e-03
200 x 200	3.2361e-06	1.2218e-03
400 x 400	3.9852e-07	3.0321e-04
800 x 800	4.9532e-08	7.5567e-05

Table 1: Convergence in two iterations to the accuracy of the numerical scheme

In view of Theorem 2, we introduce now a general method on non overlapping subdomains.

2.1. The general method

We decompose the domain \mathbb{R} into I non overlapping subdomains $\Omega_i = (a_i, a_{i+1})$, $a_j < a_i$ for $j < i$ and $a_1 = -\infty$, $a_{I+1} = \infty$ as given in Figure 2.

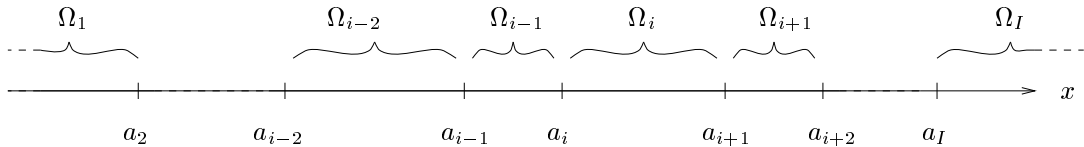


Figure 2: Domain decomposition into I non-overlapping subdomains.

We introduce a general non overlapping Schwarz waveform relaxation algorithm

$$\begin{aligned} \mathcal{L}(u_i^{k+1}) &= 0 && \text{in } \Omega_i \times (0, T) \\ \mathcal{B}_i^-(u_i^{k+1})(a_i, t) &= \mathcal{B}_i^-(u_{i-1}^k)(a_i, t) && t \in (0, T) \\ \mathcal{B}_i^+(u_i^{k+1})(a_{i+1}, t) &= \mathcal{B}_i^+(u_{i+1}^k)(a_{i+1}, t) && t \in (0, T) \\ u_i^{k+1}(x, 0) &= p(x) && x \in \Omega_i \\ \frac{\partial u_i^{k+1}}{\partial t}(x, 0) &= q(x) && x \in \Omega_i \end{aligned} \quad (6)$$

where \mathcal{B}_i^\pm are linear transmission operators which we will determine to get optimal performance of the algorithm. For ease of notation we defined here $u_0^k := 0$, $u_{I+1}^k := 0$, so that the index i in (6) ranges from $i = 1$ to I .

Definition 1. The Dirichlet to Neumann operators for the wave equation $\mathcal{S}^-(x_0)$ and $\mathcal{S}^+(x_0)$ are defined as follows. $\mathcal{S}^-(x_0)g(t) = \frac{\partial v}{\partial x}(x_0, t)$, where $v(x, t)$ is the solution of the exterior problem

$$\begin{aligned} \mathcal{L}(v) &= 0, & \text{in } (-\infty, x_0) \times (0, T) \\ \frac{\partial v}{\partial t}(x_0, t) &= g(t) & t \in (0, T) \end{aligned} \quad (7)$$

with zero initial data, and $\mathcal{S}^+(x_0)g(t) = \frac{\partial v}{\partial x}(x_0, t)$ where $v(x, t)$ is the solution of the exterior problem

$$\begin{aligned} \mathcal{L}(v) &= 0, & \text{in } (x_0, \infty) \times (0, T) \\ \frac{\partial v}{\partial t}(x_0, t) &= g(t) & t \in (0, T) \end{aligned} \quad (8)$$

with zero initial data.

The general result reads

Theorem 3. *Algorithm (6) converges in I iterations on $[0, T]$ to the solution u of the wave equation in \mathbb{R} such that $\partial_t u$ and $\partial_x u$ are continuous across the numerical interfaces, for the following choice of the operators \mathcal{B}_i^\pm :*

$$\mathcal{B}_i^- = (\partial_x - \mathcal{S}^-(a_i)\partial_t), \quad \mathcal{B}_i^+ = (\partial_x + \mathcal{S}^+(a_{i+1})\partial_t).$$

Proof. The proof is general and goes as follows. We consider vanishing initial data. At step 1, we have $\mathcal{L}(u_1^1) = 0$ in $\Omega_1 = (a_1, a_{i+1})$. Thus by definition we have on the left $\mathcal{B}_2^-(u_1^1)(a_2, \cdot) = 0$. At step 2, the problem in Ω_2 has as a left boundary condition $\mathcal{B}_2^-(u_2^2)(a_2, \cdot) = 0$. We define $\tilde{\Omega}_2 = (a_1, a_3)$ and \tilde{u}_2^2 in $\tilde{\Omega}_2$ as u_1^2 in Ω_1 and u_2^2 in Ω_2 . We define likewise $\tilde{\Omega}_{I-1}$ and \tilde{u}_{I-1}^2 . We now have an equivalent domain decomposition with $(\tilde{\Omega}_2, \Omega_3, \dots, \Omega_{I-2}, \tilde{\Omega}_{I-1})$, and the functions $(\tilde{u}_2^2, u_3^2, \dots, u_{I-2}^2, \tilde{u}_{I-1}^2)$. We proceed until the problem is reduced to two or three domains, depending on I . Suppose I is even. After $K = I/2$ steps we are led to two domains $\tilde{\Omega}_K = (a_1, a_{K+1})$ and $\tilde{\Omega}_{K+1} = (a_{K+1}, a_{I+1})$, and two functions \tilde{u}_K^K and \tilde{u}_{K+1}^K such that $\mathcal{L}(\tilde{u}_i^K) = 0$ in $\tilde{\Omega}_i$. Then due to the exact transmission conditions we have, at step $K+1$, vanishing boundary conditions for \tilde{u}_K^{K+1} and \tilde{u}_{K+1}^{K+1} , thus \tilde{u}_K^{K+1} and \tilde{u}_{K+1}^{K+1} vanish. We now have to go successively downwards with vanishing data to $u_{K-1}^{K+1} = 0, \dots, u_1^{2K} = 0$ and the same upwards. Thus at step I , the error vanishes in every subdomain. \square

2.2. The optimal method in a stratified medium and time windows

Suppose now the velocity to be constant in each subdomain : $c \equiv c_i$ in Ω_i . We then have

Theorem 4. *Suppose the subdomains coincide with the discontinuities. Then algorithm (6) converges in two iterations if*

$$\mathcal{B}_i^- = (\partial_x - \frac{1}{c_{i-1}}\partial_t), \quad \mathcal{B}_i^+ = (\partial_x + \frac{1}{c_{i+1}}\partial_t) \quad (9)$$

$$\text{and } T < T_1 = \min_{1 \leq i \leq I} \frac{|a_{i+1} - a_i|}{c_i}.$$

This analysis suggests to use time windows on intervals $[pT_1, (p+1)T_1]$, where T_1 is given in theorem 4 and p takes integer values. In each window the algorithm is exact in two iterations.

We now concentrate on the discretization of algorithm (6), with local transmission operators \mathcal{B}_i^\pm given by (9). In the case of discontinuous speed, we suppose that $T \leq T_1$, which implies by Theorem 3 that $u_i^2 \equiv u/\Omega_i$ for $1 \leq i \leq I$.

§3. The numerical algorithm

Each domain Ω_i is discretized with a mesh Δx_i , the points are numbered from 0 to $J_i + 1$. The time interval in Ω_i is discretized with a mesh Δt_i , and the time steps are numbered from 0 to $N_i + 1$. The discrete value in domain Ω_i , at point j and time n is written $U_i(j, n)$.

In order to handle more easily the boundary conditions, we choose a “vertex centered” finite volume scheme [4]. The displacement u is considered to be constant on the shell $]x - \Delta x/2, x + \Delta x/2[\times]t - \Delta t/2, t + \Delta t/2[$ and the derivatives are constant on the dual mesh.

3.1. Description of the scheme

We consider first the boundary value problem defined as the wave equation in each domain Ω_i , with boundary conditions $\mathcal{B}_i^- u_i(a_i) = g_i^-(a_i)$ and $\mathcal{B}_i^+ u_i(a_{i+1}) = g_i^+(a_{i+1})$. We write the interior scheme by integrating the equation on $(x_j - \Delta x_i/2, x_j + \Delta x_i/2) \times (t_n - \Delta t_i/2, t_n + \Delta t_i/2)$, and approximating the remaining derivatives by finite differences. We obtain the classical leapfrog scheme. We now integrate the equation on the half-shell $(a_i, a_i/2, x_j + \Delta x_i/2) \times (t_n - \Delta t_i/2, t_n + \Delta t_i/2)$. We get the same terms as before, but a boundary term which is the integral of $\partial u / \partial x$ on $(t_n - \Delta t_i/2, t_n + \Delta t_i/2)$. It is handled using the boundary condition, and we obtain for example on the left.

$$B_i^-(U_i)(0, n) := \left(\frac{\Delta x_i}{2C_i^2} D_t^+ D_t^- - D_x^+ + \frac{1}{C_{i-1}} D_t^0 \right) (U_i)(0, n) = G_i^-(n).$$

where $D_t^+ \phi(n) = \frac{1}{\Delta t_i} (\phi(n+1) - \phi(n))$, $D_t^- \phi(n) = \frac{1}{\Delta t_i} (\phi(n) - \phi(n-1))$, $D_t^0 \phi(n) = \frac{1}{2\Delta t_i} (\phi(n+1) - \phi(n-1))$ and the same for the x -derivatives. The right hand side is

$$G_i^-(n) := \frac{1}{\Delta t_i} \int_{t_n - \Delta t_i/2}^{t_n + \Delta t_i/2} g_i^-(\tau) d\tau,$$

Within the iterative algorithm, $G_i^-(n)$ is now a $G_i^{-,k}(n)$, whose value is extracted from Ω_{i-1} at step $k - 1$ using the same process. We define

$$\tilde{B}_i^-(U_{i-1})(J_{i-1} + 1, n) := \left(-\frac{\Delta x_{i-1}}{2C_{i-1}^2} D_t^+ D_t^- - D_x^- + \frac{1}{C_i} D_t^0 \right) (U_{i-1})(J_{i-1} + 1, n).$$

and $\tilde{G}_{i-1}^{-,k-1}(n) = \tilde{B}_i^-(U_{i-1}^{k-1})(J_{i-1} + 1, n)$. If the time steps are the same in Ω_i and Ω_{i-1} , the transmission condition leads to $G_i^{-,k} = \tilde{G}_{i-1}^{-,k-1}$. If we have different time grids Δt_i in Ω_i , $G_i^{-,k}$ is a vector in \mathbb{R}^{N_i+1} and $\tilde{G}_{i-1}^{-,k-1}$ is a vector in $\mathbb{R}^{N_{i-1}+1}$. We need a projection operator, defined as follows. Suppose we are given a vector $\mathbf{v} = (v_0, \dots, v_N) \in \mathbb{R}^{N+1}$ which represents the values of a step function on the corresponding intervals $I_n = (t_n, t_{n+1})$ where $t_0 = 0$, $t_{N+1} = T$ and $\cup_{n=0}^N \overline{I_n} = [0, T]$ and the intervals do not overlap. Then we define the scalar product on \mathbb{R}^{N+1} by

$$(\mathbf{v}, \mathbf{w})_{N+1} := \sum_{n=0}^N |I_n| v_n w_n$$

where $|I_n|$ denotes the length of the interval I_n . We thus obtain the induced norm on \mathbb{R}^{N+1}

$$\|\mathbf{v}\|_{N+1}^2 := (\mathbf{v}, \mathbf{v})_{N+1}.$$

We first define the operator $\mathbb{F} : \mathbb{R}^{N+1} \longrightarrow L^2(0, T)$ which constructs a piecewise constant function on the intervals I_n from the vector \mathbf{v} ,

$$\mathbb{F} : \mathbf{v} \longmapsto f(t) := v_n, \quad t \in I_n$$

Then we define the operator $\mathbb{E} : L^2(0, T) \longrightarrow \mathbb{R}^{N+1}$ which projects a given function $f(t)$ onto a vector $\mathbf{v} \in \mathbb{R}^{N+1}$ corresponding to a piecewise constant function in the intervals I_n

$$\mathbb{E} : f(t) \longmapsto v_n := \frac{1}{|I_n|} \int_{I_n} f(t) dt.$$

Denoting by \mathbb{F}_i and \mathbb{E}_i the corresponding operators using the grid of Ω_i , we define the operator $\mathbb{P}_{i,j} : \mathbb{R}^{N_i+1} \rightarrow \mathbb{R}^{N_j+1}$ by

$$\mathbb{P}_{i,j} := \mathbb{E}_j \circ \mathbb{F}_i. \quad (10)$$

The precise implementation of the projection procedure is described in [6]. With these notations we have on the left $G_i^{-,k} = \mathbb{P}_{i,i-1} \tilde{G}_{i-1}^{-,k-1}$, and the same on the right.

We obtain the discrete Schwarz waveform relaxation algorithm on subdomains Ω_i , $1 \leq i \leq I$ with non-matching grids:

Definition 2. The discrete algorithm corresponding to (6,9) is given by

$$\left(\frac{1}{C_i^2(j)} D_t^+ D_t^- - D_x^+ D_x^- \right) (U_i^{k+1})(j, n) = 0, \quad 1 \leq j \leq J_i, 1 \leq n \leq N_i, \quad (11)$$

$$B_i^-(U_i^{k+1})(0, \cdot) = \mathbb{P}_{i-1,i} \tilde{B}_i^-(U_{i-1}^k)(J_{i-1}+1, \cdot), \quad (12)$$

$$B_i^+(U_i^{k+1})(J_i+1, \cdot) = \mathbb{P}_{i+1,i} \tilde{B}_i^+(U_{i+1}^k)(0, \cdot), \quad (13)$$

with the discrete operators

$$\begin{aligned} B_i^-(U_i)(0, n) &= \left(\frac{\Delta x_i}{2C_i^2} D_t^+ D_t^- - D_x^+ + \frac{1}{C_{i-1}} D_t^0 \right) (U_i)(0, n) \\ \tilde{B}_i^-(U_{i-1})(J_{i-1}+1, n) &= \left(-\frac{\Delta x_{i-1}}{2C_{i-1}^2} D_t^+ D_t^- - D_x^- + \frac{1}{C_{i-1}} D_t^0 \right) (U_{i-1})(J_{i-1}+1, n) \end{aligned} \quad (14)$$

$$\begin{aligned} B_i^+(U_i)(J_i+1, n) &= \left(\frac{\Delta x_i}{2C_i^2} D_t^+ D_t^- + D_x^- + \frac{1}{C_{i+1}} D_t^0 \right) (U_i)(J_i+1, n), \\ \tilde{B}_i^+(U_{i+1})(0, n) &= \left(-\frac{\Delta x_{i+1}}{2C_{i+1}^2} D_t^+ D_t^- + D_x^+ + \frac{1}{C_{i+1}} D_t^0 \right) (U_{i+1})(0, n) \end{aligned} \quad (15)$$

provided with initial values derived in the same way.

We note on formulae (14,15) that the transport operators B_i^\pm are approximated by a Lax-Wendroff-type scheme, and that they are discretized differently depending on whether they apply to U_i or to $U_{i\pm 1}$.

3.2. The case of continuous wave speed across numerical interfaces

We denote by $V = \{V(j)\}_{0 \leq j \leq J+1}$ a sequence in \mathbb{R}^{J+2} , and we define for $V, W \in \mathbb{R}^{J+2}$ a bilinear form on \mathbb{R}^{J+2} by

$$a_h(V, W) = \frac{\Delta x}{2} \sum_{j=1}^{J+1} D_x^-(V)(j) \cdot D_x^-(W)(j). \quad (16)$$

Accordingly, for any positive n , $V(n)$ stands for the sequence $\{V(j, n)\}_{0 \leq j \leq J+1}$. The discrete energy E_n at time step n , global in space, is defined as the sum of a discrete kinetic energy $E_{K,n}$ and a discrete potential energy $E_{P,n}$ given by

$$\begin{aligned} E_{K,n} &= \frac{\Delta x}{2} \left[\frac{1}{2C^2(0)} (D_t^-(V)(0, n))^2 + \sum_{j=1}^J \frac{1}{C^2(j)} (D_t^-(V)(j, n))^2 + \frac{1}{2C^2(J+1)} (D_t^-(V)(J+1, n))^2 \right], \\ E_{P,n} &= a_h(V(n), V(n-1)), \\ E_n &= E_{K,n} + E_{P,n}. \end{aligned} \quad (17)$$

The quantity $E_{K,n}$ is clearly a discrete kinetic energy. E_n can be identified as an energy by the

Lemma 5. For any $n \geq 1$, we have

$$E_n \geq \left(1 - \left(C \frac{\Delta t}{\Delta x} \right)^2 \right) E_{K,n}, \quad (18)$$

where C is defined by $C = \sup_{1 \leq j \leq J+1} C(j)$. Hence, under the CFL condition

$$C \frac{\Delta t}{\Delta x} < 1, \quad (19)$$

E_n is bounded from below by an energy.

This provides in each subdomain an energy identity and we proved in [6] the following result.

Theorem 6. *Assume that the velocity is continuous on the interfaces a_i . If the CFL condition (19) is satisfied by the discretization in each subdomain, then the non-overlapping discrete Schwarz waveform relaxation algorithm (11,...,15) is well-posed and converges on any time interval $[0, T]$ to the solution of*

$$\begin{aligned} \left(\frac{1}{C_i^2(j)} D_t^+ D_t^- - D_x^+ D_x^- \right) (U_i)(j, n) &= 0, \quad 1 \leq j \leq J_i, 1 \leq n \leq N_i, \\ B_i^-(U_i)(0, \cdot) &= \mathbb{P}_{i-1,i} \tilde{B}_i^-(U_{i-1})(J_{i-1} + 1, \cdot), \\ B_i^+(U_i)(J_i + 1, \cdot) &= \mathbb{P}_{i+1,i} \tilde{B}_i^+(U_{i+1})(0, \cdot), \end{aligned} \quad (20)$$

in the energy norm, i.e.

$$\sum_{i=1}^I E_{N_i}(U_i^k - U_i) \rightarrow 0 \text{ as } k \rightarrow \infty.$$

The same calculations actually give a stability result for the limit

Theorem 7. *With the same assumptions as in Theorem 6, the limit of the iterates given by (20) satisfies the energy bounds, with a constant C depending only on the L^2 norms of the initial conditions:*

$$\sum_{i=1}^I E_{N_i}(U_i) \leq C.$$

3.3. The case of a stratified medium

We suppose again the velocity to be constant in each subdomain : $c \equiv c_i$ in Ω_i . We define a local CFL number to be $\gamma_i = c_i \Delta t_i / \Delta x_i$. The stability for the pure Cauchy problem implies $\gamma_i \leq 1$ (see for instance[7]).

3.3.1. The case $\gamma = 1$

We suppose here that $\gamma_i = 1$ in every Ω_i . The scheme resumes to

$$U_i(j, n+1) + U_i(j, n-1) - (U_i(j+1, n) + U_i(j-1, n)) = 0, \quad (21)$$

and the discrete operators are given for any solution of (21) by

$$\begin{aligned} \tilde{B}_2^-(U_1)(J_1+1, n) &= \frac{1}{\Delta x_1} (U_1(J_1, n) - U_1(J_1+1, n-1)), \\ \tilde{B}_1^+(U_2)(0, n) &= \frac{1}{\Delta x_2} (U_2(1, n) - U_2(0, n-1)). \end{aligned} \quad (22)$$

We consider first two media $\Omega_1 = (-\infty, 0)$ and $\Omega_2 = (0, +\infty)$.

Theorem 8. *In the case of two media Ω_1 and Ω_2 with the same CFL number $\gamma_1 = \gamma_2 = 1$, the discrete algorithm (11,...,15) converges in two iterations to the solution of (20).*

Proof. It is mimicked on the continuous case. By (21, 22), we have

$$\begin{aligned} \tilde{B}_2^-(U_1)(J_1+1, n) &= U_1(J_1 - n + 1, 1) - U_1(J_1 - n, 0), \\ \tilde{B}_1^+(U_2)(0, n) &= U_2(n, 1) - U_2(n-1, 0). \end{aligned}$$

Let now the initial values be null. It implies that we have at step 1 $\tilde{B}_2^-(U_1^1)(J_1+1, \cdot) \equiv 0$ and $\tilde{B}_1^+(U_2^1)(0, \cdot) \equiv 0$. By linearity we have $B_2^-(U_2^2)(0, \cdot) \equiv 0$ and $B_1^+(U_1^2)(J_1+1, \cdot) \equiv 0$. We conclude that $U_i^2 \equiv U_i$ for $i = 1, 2$. \square

When γ is equal to 1, the discrete velocity is equal to the continuous speed. Then Theorem 8 extends easily to

Theorem 9. *In the case of I media Ω_i with the same CFL number $\gamma_i = 1$, the discrete algorithm (11,...,15) converges in two iterations to the solution of (20) for $T < T_1$.*

3.3.2. The case $\gamma < 1$

We consider two media $\Omega_1 = (-\infty, 0)$ and $\Omega_2 = (0, +\infty)$, with velocities c_1 and $c_2 = qc_1$. The space mesh is constant, $\Delta x_1 = \Delta x_2 = \Delta x$, and $\Delta t_1 = q\Delta t_2$. Therefore $\gamma_1 = \gamma_2 = \gamma$.

We analyze here the case $q = 2$. The general case for any integer q will be done in a forthcoming paper. The discrete solution U_2 in Ω_2 is split into an even part $\{U_{2P}(\cdot, n)\}_{n \in \mathbb{N}} = \{U_2(\cdot, 2n)\}_{n \in \mathbb{N}}$ and an odd part $\{U_{2I}(\cdot, n)\} = \{U_2(\cdot, 2n+1)\}_{n \in \mathbb{N}}$. We define

$$\begin{aligned} V_1^k &= B_1^+(U_1^k)(J_1 + 1, \cdot) \quad ; \quad V_2^k = B_2^-(U_2^k)(0, \cdot), \\ \tilde{V}_1^k &= \tilde{B}_2^-(U_1^k)(J_1 + 1, \cdot) \quad ; \quad \tilde{V}_2^k = \tilde{B}_1^+(U_2^k)(0, \cdot). \end{aligned}$$

Hence the projection operators $\mathbb{P}_{i,j}$ have the very special form

$$(\mathbb{P}_{12}V_1)_{2P} = V_1; \quad (\mathbb{P}_{12}V_1)_{2I} = V_1; \quad \mathbb{P}_{21}V_2 = \frac{1}{2}(V_{2P} + V_{2I})$$

which gives the transmission conditions as

$$V_{2P}^{k+1} = \tilde{V}_1^k = V_{2I}^{k+1}; \quad V_1^{k+1} = \frac{1}{2}(\tilde{V}_{2P}^k + \tilde{V}_{2I}^k) \quad (23)$$

The discrete Laplace transform of a grid function $v = \{v_n\}_{n \geq 0}$ on a regular grid with time step δt is defined for $\eta > 0$ by (see for instance [7])

$$\mathcal{L}v(s) = \hat{v}(s) = \delta t \sum_{n \geq 0} e^{-sn\delta t} v_n, \quad s = \eta + i\tau, \quad |\tau| \leq \frac{\pi}{\delta t}, \quad (24)$$

Using the Laplace transform of U_1, U_{2P} and U_{2I} on the mesh $\delta t = 2\Delta t$, equation (11) becomes the difference equation

$$(e^{2s\Delta t} + e^{-2s\Delta t} - 2)\hat{U}_1 - \gamma_1^2 A \hat{U}_1 = 0, \quad (25)$$

with operator A acting on the j - variables given by

$$Ag(j) = g(j+1) - 2g(j) + g(j-1),$$

System in Ω_2 using the Laplace transform can be written as

$$\begin{aligned} (1 + e^{-2s\Delta t}) \hat{U}_{2I}(j, s) - 2\hat{U}_{2P}(j, s) - \gamma_2^2 A \hat{U}_{2P} &= 0, \\ (1 + e^{2s\Delta t}) \hat{U}_{2P}(j, s) - 2\hat{U}_{2I}(j, s) - \gamma_2^2 A \hat{U}_{2I} &= 0. \end{aligned} \quad (26)$$

According to the results in [6], we denote by $r_+(z, \gamma)$ (resp. r_-) the root whose modulus is > 1 (resp. < 1) for strictly positive η of the characteristic equation

$$\gamma^2(r - 2 + 1/r) = (z + 1/z - 2). \quad (27)$$

We then can solve (25) and (26) with $Z = e^{s\Delta t}$ as (see [1] and [3])

$$\begin{aligned} \hat{U}_1^k(j, s) &= a_1^k(s)(r_+(Z^2, \gamma_1))^j, \\ \begin{pmatrix} \hat{U}_{2I}(j, s) \\ \hat{U}_{2P}(j, s) \end{pmatrix} &= a_{2p}^k(s)(r_-(Z, \gamma_2))^j \begin{pmatrix} Z \\ 1 \end{pmatrix} + a_{2m}^k(s)(r_-(-Z, \gamma_2))^j \begin{pmatrix} -Z \\ 1 \end{pmatrix}. \end{aligned} \quad (28)$$

The boundary terms \hat{V}_j^k and $\tilde{\hat{V}}_j^k$ are given by

$$\begin{aligned} \Delta x_1 \hat{V}_1^k &= E(Z^2, \gamma, \frac{c_1}{c_2}) a_1^k; \\ \Delta x_1 \tilde{\hat{V}}_1^k &= \tilde{E}(Z^2, \gamma, 1) a_1^k; \end{aligned} \quad (29)$$

$$\begin{aligned}
\Delta x_2 \begin{pmatrix} \widehat{V}_{2I}^k(j, s) \\ \widehat{V}_{2P}^k(j, s) \end{pmatrix} &= a_{2p}^k E(Z, \gamma, \frac{c_2}{c_1}) \begin{pmatrix} Z \\ 1 \end{pmatrix} + a_{2m}^k E(-Z, \gamma, \frac{c_2}{c_1}) \begin{pmatrix} -Z \\ 1 \end{pmatrix} \\
\Delta x_2 \begin{pmatrix} \widehat{\widetilde{V}}_{2I}^k(j, s) \\ \widehat{\widetilde{V}}_{2P}^k(j, s) \end{pmatrix} &= a_{2p}^k \widetilde{E}(Z, \gamma, 1) \begin{pmatrix} Z \\ 1 \end{pmatrix} + a_{2m}^k \widetilde{E}(-Z, \gamma, 1) \begin{pmatrix} -Z \\ 1 \end{pmatrix}
\end{aligned} \tag{30}$$

with factors E and \widetilde{E} given by

$$\begin{aligned}
k(z) &= \frac{1}{2} \left(z - \frac{1}{z} \right), \quad \sigma(z, \gamma) = \frac{1}{2} (r_+(z) - r_-(z)), \\
E(z, \gamma, q) &= \sigma(z, \gamma) + \frac{q}{\gamma} k(z), \quad \widetilde{E}(z, \gamma, q) = -\sigma(z, \gamma) + \frac{q}{\gamma} k(z)
\end{aligned} \tag{31}$$

Transmission conditions (23) become

$$\widehat{V}_{2P}^{k+1} = \widehat{V}_{2I}^{k+1} = \widehat{V}_1^k, \quad \widehat{V}_1^{k+1} = \frac{1}{2} (\widehat{V}_{2P}^k + \widehat{V}_{2I}^k)$$

and thus we get the recursion relations

$$\begin{cases} a_{2m}^k = \frac{Z-1}{Z+1} \frac{E(-Z, \gamma, \frac{c_2}{c_1})}{E(Z, \gamma, \frac{c_2}{c_1})} a_{2p}^k, \\ a_{2p}^{k+1} = \frac{1}{4} \frac{Z+1}{Z} \frac{\widetilde{E}(Z^2, \gamma, 1)}{E(Z, \gamma, \frac{c_2}{c_1})} a_1^k, \\ a_1^{k+1} = \frac{1}{Z+1} \frac{E(Z, \gamma, \frac{c_2}{c_1})}{E(Z^2, \gamma, \frac{c_1}{c_2})} \left[(Z+1)^2 \frac{\widetilde{E}(Z, \gamma, 1)}{E(Z, \gamma, \frac{c_2}{c_1})} - (Z-1)^2 \frac{\widetilde{E}(-Z, \gamma, 1)}{E(-Z, \gamma, \frac{c_2}{c_1})} \right] a_{2p}^k. \end{cases} \tag{32}$$

There is a two-level recursion formula for each coefficient $a_j^{k+2} = R a_j^k$, with a convergence rate

$$R(Z, \gamma) = \frac{1}{4Z} \frac{\widetilde{E}(Z^2, \gamma, 1)}{E(Z^2, \gamma, \frac{c_1}{c_2})} \left[(Z+1)^2 \frac{\widetilde{E}(Z, \gamma, 1)}{E(Z, \gamma, \frac{c_2}{c_1})} - (Z-1)^2 \frac{\widetilde{E}(-Z, \gamma, 1)}{E(-Z, \gamma, \frac{c_2}{c_1})} \right]. \tag{33}$$

Defining

$$\rho(z, \gamma, q) = \frac{\widetilde{E}(z, \gamma, 1)}{E(z, \gamma, q)}; \quad \tilde{\rho}(Z, \gamma) = \frac{(Z+1)^2}{4Z} \rho(Z, \gamma, \frac{c_1}{c_2}) \rho(Z^2, \gamma, \frac{c_2}{c_1}), \tag{34}$$

we obtain for the convergence rate of the discrete Schwarz waveform relaxation algorithm

$$R(Z, \gamma) = \tilde{\rho}(Z, \gamma) - \tilde{\rho}(-Z, \gamma). \tag{35}$$

The properties of the solutions $r^\pm(z)$ and of the functions E and \widetilde{E} are given in great details in [6]. In particular it is proved that for any $z \neq 1$, and $\gamma < 1$, $E(z, \gamma, q) \neq 0$. Furthermore if $z \rightarrow 1$, we have

$$E(z, \gamma, q) = \mathcal{O}(z-1); \quad \widetilde{E}(z, \gamma, 1) = \mathcal{O}((z-1)^3);$$

Thus $\tilde{\rho}$ is well-defined for $|z| > 1$, with $\tilde{\rho}(1, \gamma) = 0$, and so is R . The convergence rate $R(z, \gamma)$ is an analytic function of z for $|z| \geq 1$, which corresponds to $\eta \geq 0$. R satisfies a maximum principle for $\eta \geq 0$ and hence attains its maximum on the boundary $\eta = 0$. It is therefore sufficient to study the behavior of R for $\eta = 0$.

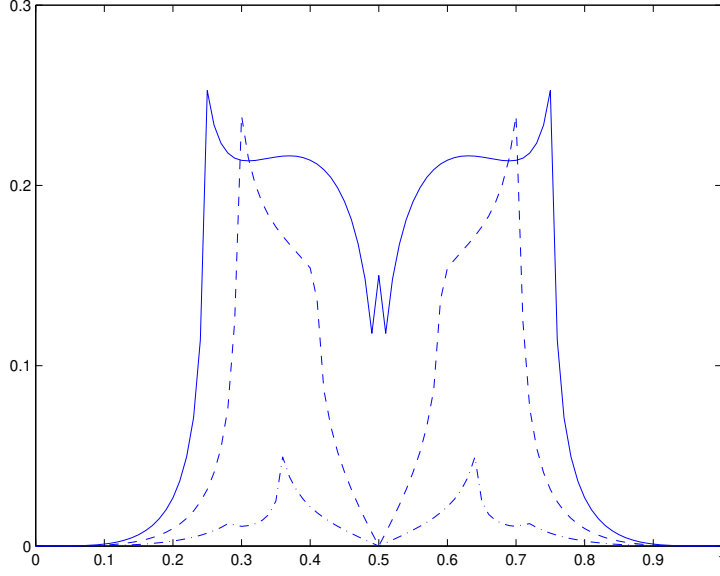


Figure 3: variation of $|R(e^{i\tau\Delta t}, \gamma)|$ as a function of $\tau\Delta t$ for various values of γ . $\gamma = 0.7$, solid, $\gamma = 0.8$, dash, $\gamma = 0.9$, dash-dot.

Lemma 10. *There exists a strictly positive constant K such that the convergence rate satisfies*

$$\sup_{|z|=1} |R(z, \gamma, c_1/c_2)| \leq K < 1.$$

and we conclude as in [6]

Theorem 11. *Let U_i^p be the iterates of algorithm (11,...,15) with zero initial values. Then we have*

$$\|U_i^p\|_{\Omega_i \times (0,T)} \leq (K)^{\lfloor \frac{p}{2} \rfloor} \max_{i=1,2} \|U_i^0\|_{\Omega_i}.$$

where $\|U_i\|_{\Omega_i \times (0,T)}^2 = \Delta t_i \Delta x_i \sum_n \sum_j |U_i(j, n)|^2$ and $\|U_i^0\|_{\Omega_i}^2 = \Delta x_i \sum_j |U_i^0(j)|^2$.

The same calculations, using the GKS theory as in [1] prove the

Theorem 12. *Let U_i be the limit of the iterates of algorithm (11,...,15). Then U_i is stable in the GKS sense.*

§4. Numerical results

We first compare our method to two other refinement methods. They do not involve domain decomposition, and have been analyzed in the case when the velocity is equal to 1 and one domain is a refinement of the neighbour.

4.1. Some comparisons in the case of two half-spaces

We restrict ourselves to the case where $c = 1$ and $(\Delta t_1, \Delta x_1) = (2\Delta t_2, 2\Delta x_2) = (2\Delta t, 2\Delta x)$. The first method, we call the I-method, consists in defining three transmission conditions as follows. $U_1(0, n)$ is obtained by the leapfrog scheme on the dotted stencil in Figure 4. Then continuity is enforced on the even points: $U_2(0, 2n) = U_1(0, n)$, and interpolation at odd points: $U_2(0, 2n+1) = \frac{1}{2}(U_1(0, n) + U_1(0, n+1))$. A complete analysis of stability for the Lax-Wendroff scheme can be found in [1].

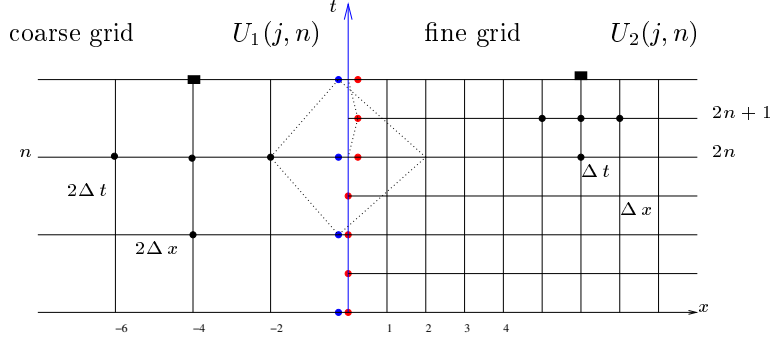


Figure 4: the I-method

In the second method, we call E-method developed in [2], three relations are written between U_1 and U_2 which preserve the energy in both domains. They are too complicated to be written here. The test case is the same as in [3]. $\Omega_1 = (0, 100)$ with transparent boundary condition on the left, and $\Omega_2 = (100, 110)$ with a transparent boundary condition on the right. The initial condition is drawn in Figure 5.

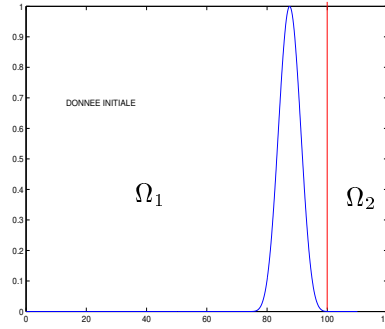


Figure 5: Initial value

The time T is equal to 200. The space mesh is fixed equal to $\Delta x = 0.5$, and we take various values of Δt .

We start with the case $\gamma_1 = \gamma_2 = 1$ in table 2. In this case a uniform mesh would produce an exact scheme as mentioned in the first paragraph. As noted by the authors in [2], the E-method produces wrong results. In the coarse domain, the scheme is exact. In the right domain, interpolation is better by a factor of 2.

	max error in Ω_1	max error in Ω_2
scheme error, fine grid	1.6098e-15	1.4988e-15
D.D-method	1.8319e-15	0.0089
I-method	1.9429e-15	0.0046
E-method	1.9971e-04	0.4949

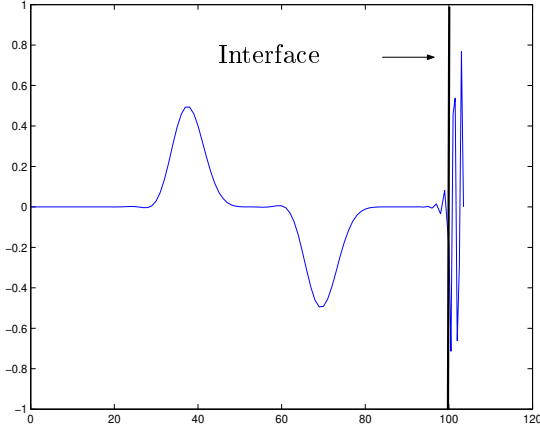
Table 2: error in $L^\infty(\Omega_i \times (0, T))$ for $\gamma_1 = \gamma_2 = 1$

We now reduce the coefficient γ , to $\gamma = 0.9091$ in table 3. The three methods are comparable on the left domain, in the right domain the interpolation method is the best, there is a factor 2 with the DD-method, and a factor 16 with the E-method.

	max error in Ω_1	max error in Ω_2
scheme error, fine grid	0.0022	0.0006
scheme error, coarse grid	0.009	0.003
D.D-method	0.0068	0.0064
I-method	0.0068	0.0027
E-method	0.0068	0.0437

Table 3: error in $L^\infty(\Omega_i \times (0, T))$ for $\gamma_1 = \gamma_2 = 0.9091$

We will not diminish γ further, since then the dispersion error would be too large. In [2] is shown an example with a Dirichlet boundary condition on the right. In that case the I-method is unstable. We reproduce their example in Figure 6. The velocity is constant equal to 1. The domains are $\Omega_1 = (0, 100)$, $\Omega_2 = (100, 103.5)$, $T = 1000$. The space meshes are the same in both domains equal to 1, $\gamma_1 = \gamma_2 = 0.913$.

Figure 6: Solution of the I-method in $\Omega_1 \cup \Omega_2$ for Dirichlet boundary condition after 400 timesteps

This phenomenon is also reported in [1] as a personal communication of Oliger. No detailed proof is available so far. Another case is where we modify Ω_2 and γ a little, taking $\Omega_2 = (100, 110)$, $\gamma = 0.9524$. The numerical data are the same as before. The scheme remains stable, and we have the errors in table 4.

	max error in Ω_1	max error in Ω_2
scheme error, fine grid	0.0265	0.0265
scheme error, coarse grid	0.1050	0.1050
D.D-method	0.0764	0.0693
I-method	0.0471	0.0425
E-method	0.0781	0.1062

Table 4: error in $L^\infty(\Omega_i \times (0, T))$ for Dirichlet boundary condition, $\gamma_1 = \gamma_2 = 0.9524$

Here the I-method is the best, followed by the present method, especially in the fine grid. In Figure 7 we draw the L^∞ error in time and space in both domains as a function of the mesh, for the same data. We see that the present method is of order 2.

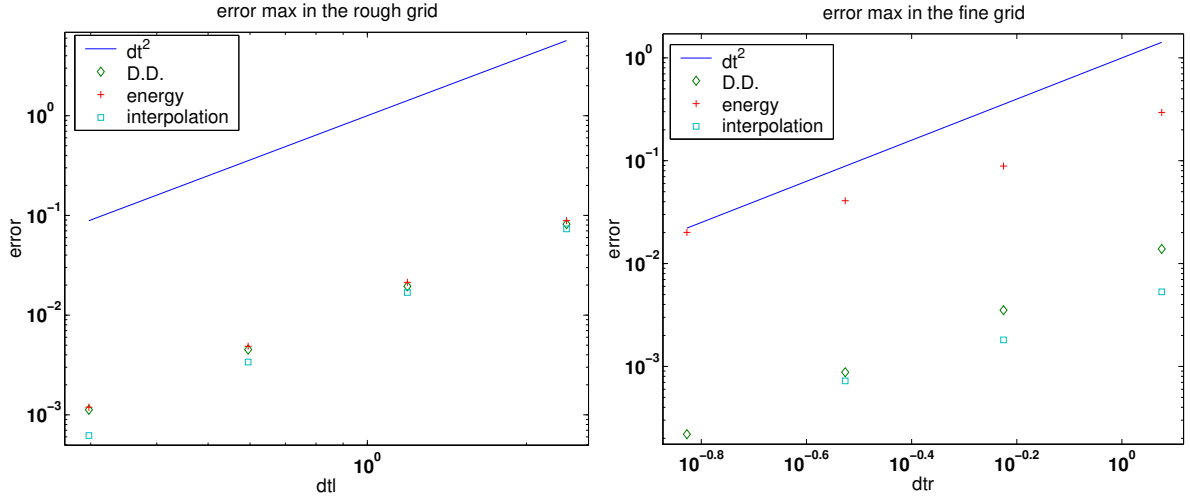


Figure 7: error in $L^\infty(\Omega_i \times (0, T))$ as a function of the time step, logarithmic scale. On the left the coarse grid, on the right the fine grid. Comparison between the three methods

4.2. A few more difficult cases

The domains are $\Omega_1 = (0, 1)$, $\Omega_2 = (1, 2)$, $T = 1$. The velocity are $c=1$ in Ω_1 , $c=1.7$ in Ω_2 . The initial data is as in Figure 5, supported in Ω_1 . We choose first 11 points in space in each domain, and the number of points in time such that $\gamma = 1$ in both domains, that is 11 points in time in Ω_1 and 18 in Ω_2 . Thereafter the meshes are divided by 2. In Figure 8 we draw the L^∞ error in time and space in both domains as a function of the mesh, and we see that the method is still of order 2.

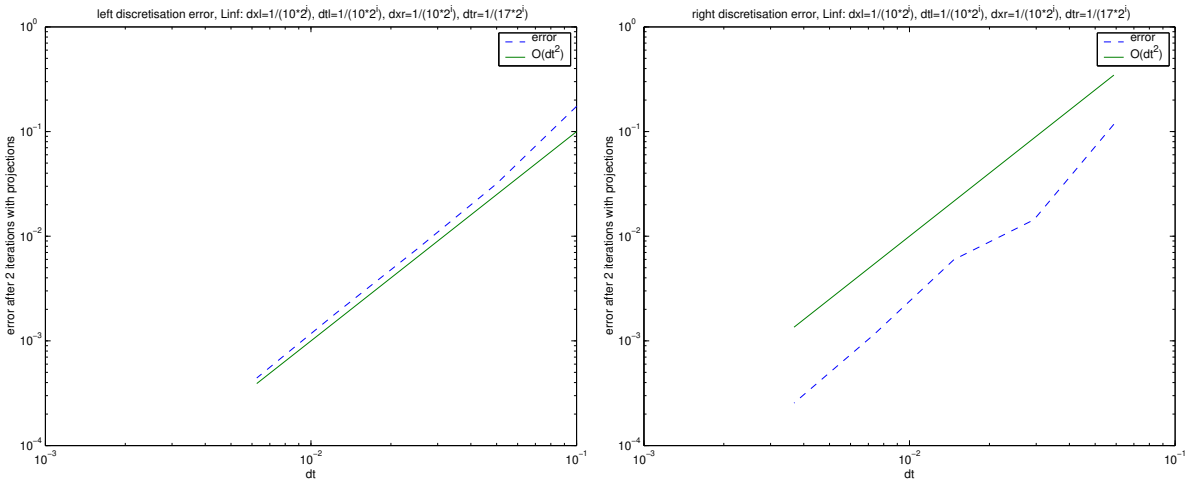


Figure 8: error in $L^\infty(\Omega_i \times (0, T))$ as a function of the time step, logarithmic scale. On the left the coarse grid, on the right the fine grid.

We consider now the space interval $[0, 6]$ divided in 6 layers $c_i \in \{1, 2/3, 1/2, 3/4, 4/5\}$. The six numerical domains are aligned with the discontinuities. We chose the same space mesh in the subdomains, $\Delta x_i = 1/50$, and the local time steps are such that γ_i is close to 1. The time interval is $[0, 1]$.

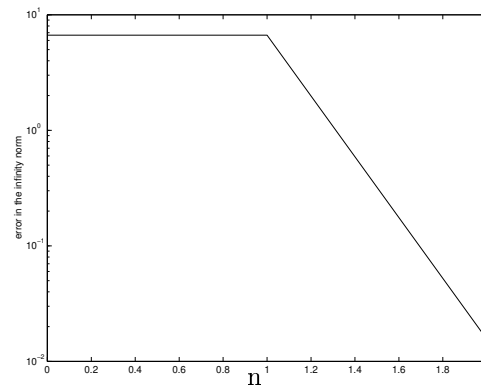


Figure 9: Convergence in 2 iterations with local transmission conditions on time interval $[0, 1]$

§5. Conclusions

We have proposed a new way of considering the mesh refinement problem. Applied to the leapfrog scheme, we have proven it to be stable, and given numerical evidence of an order 2 in time. The precise study of the accuracy will be done in a forthcoming paper. Its flexibility allows for the use of other schemes, like finite elements in time and space for instance, and other equations. It is robust and will extend to higher dimensions.

Acknowledgements

The author is grateful to M.J. Gander for providing the original scripts of the numerical method described first in [6].

References

- [1] BERGER, M. Stability of interfaces with mesh refinement. *Math. of Comp.* 45, 172 (1985), 301–318.
- [2] COLLINO, F., FOUQUET, T. AND JOLY, P. A conservative space-time mesh refinement method for the 1D Wave equation. Part I : construction. *Numer. Math.* 95, (2003), 197–221.
- [3] COLLINO, F., FOUQUET, T. AND JOLY, P. A conservative space-time mesh refinement method for the 1D Wave equation. Part II : analysis. *Numer. Math.* 95, (2003), 223–251.
- [4] GALLOUËT, T., HERBIN, R. AND VIGNAL, M.H. Error estimates for the approximate finite volume solution of convection diffusion equations with general boundary conditions. *SIAM J. Numer. Anal.* 37, 6(2000), 1935–1972.
- [5] GANDER, M. Analysis of parallel algorithms for time dependent partial differential equations. *Phd Thesis, Stanford, CA 94305, USA*, 1997.
- [6] GANDER, M., HALPERN, L. AND NATAF, F. Optimal Schwarz waveform relaxation for the one-dimensional wave equation. *SIAM J. Numer. Anal.* 42, 6(2004), 1935–1972.
- [7] STRIKWERDA, J.C. *finite difference schemes and partial differential equations*. Chapman and Hall, 1989.

Laurence Halpern
LAGA. Institut Galilée
Université Paris 13.
93430 Villetaneuse. France.
halpern@math.univ-paris13.fr