Title: Optimized Schwarz Waveform Relaxation method for the linearized KdV equation

Corresponding Author: Mr. Joao Guilherme Caldas Steinstraesser,

Corresponding Author's Institution: MERIC - Marine Energy Research & Innovation Center

First Author: Joao Guilherme Caldas Steinstraesser

Order of Authors: Joao Guilherme Caldas Steinstraesser; Rodrigo Cienfuegos; José Daniel Galaz Mora; Antoine Rousseau

Abstract: We propose an optimized Schwarz Waveform Relaxation method for solving the linearized KdV equation with only the dispersive term, using simple interface operators based on the exact transparent boundary conditions for this equation. An optimization process is performed for obtaining the approximation that provides the method with the fastest convergence to the solution of the monodomain problem.

# Optimized Schwarz Waveform Relaxation method for the linearized KdV equation

Joao Guilherme Caldas Steinstraesser[a,*], Rodrigo Cienfuegos[b], José Daniel Galaz Mora[b], Antoine Rousseau[c]

[a] *MERIC, Marine Energy Research & Innovation Center, Avda. Apoquindo 2827, Santiago, Chile*
[b]*Departamento de Ingeniería Hidráulica y Ambiental, Pontificia Universidad Católica de Chile, Av. Vicuña Mackenna 4680 - Macul, Santiago, Chile*
[c]*Inria and Inria Chile, Avda. Apoquindo 2827, Santiago, Chile*

**Abstract**

We propose an optimized Schwarz Waveform Relaxation method for solving the linearized KdV equation with only the dispersive term, using simple interface operators based on the exact transparent boundary conditions for this equation. An optimization process is performed for obtaining the approximation that provides the method with the fastest convergence to the solution of the monodomain problem.

*Keywords:* domain decomposition method, optimized Schwarz Waveform Relaxation method, transparent boundary conditions, KdV equation

## 1. Introduction

The Korteweg - de Vries (KdV) equation, derived by [9] in 1895, models the propagation of waves with small amplitude and large wavelength, taking in account nonlinear and dispersive effects. In terms of dimensionless but unscaled variables, it can be written as [2]

$$u_t + u_x + uu_x + u_{xxx} = 0$$

As done in [12] (and in [3] as a special case of their work), we will focus in this paper on the linearized KdV equation without the advective term :

$$u_t + u_{xxx} = 0 \tag{1}$$

to which we will refer as *dispersion equation.*

---

*Corresponding author
   Email addresses:* `joao.caldas@meric.cl` (Joao Guilherme Caldas Steinstraesser), `racienfu@ing.puc.cl` (Rodrigo Cienfuegos), `jdgalaz@uc.cl` (José Daniel Galaz Mora), `antoine.rousseau@inria.fr` (Antoine Rousseau)

The work developed here is inspired from [12] and [3]. Nevertheless, our objectives are different from theirs. In this paper we propose an optimized Schwarz Waveform Relaxation method (SWR) for solving the dispersion equation (1) in a bounded domain, *i.e.*, we will decompose the computational domain in subdomains and solve the time-dependent problem in each one of them. Our work focuses on the formulation of appropriate and optimized conditions on the interface between the subdomains, in order to minimize the error due to the domain decomposition method (DDM) and to accelerate the convergence of the method.

The interface boundary conditions (IBCs) proposed here are based on the exact Transparent Boundary Conditions (TBCs) for the equation (1), derived by [12] and [3]. The TBCs make the approximate solution in the computational domain coincide with the solution of the whole domain, but its exact computation is not doable in general [1]. [12] and [3] propose numerical approximations for these conditions, seeking to reduce the error created by the introduction of artificial boundaries.

In the work presented here, we do not propose approximate transparent boundary conditions for reducing the error related to the finitude of the computational domain. In fact, we intend to reduce the error created by the decomposition of the domain and the introduction of an artificial interface boundary condition, in the context of a DDM. In other words, we study the effectiveness of the boundary conditions as IBCs, not as TBCs. As a consequence, our work shall not use the same reference solution as the one used by [12] and [3]: for validating their approaches, they compare their approximate solution with the exact solution in the whole domain. On the other hand, our reference solution will be the approximate solution computed on the computational monodomain.

This paper is organized in the following way: in Section 2, we recall the exact TBCs derived by [12] for (1) and propose IBCs based on them, leading to very simple mixed-type conditions (avoiding, for example, integrations in time) depending on two coefficients. With some numerical experiments, we show that these conditions give reasonable results when used as TBCs (although not as well as the approaches of [12] and [3]), motivating us to use them in the sequel of our work. In Section 3, we describe the domain decomposition method used here and we construct it using our operators as interface boundary conditions (IBCs). Small modifications are proposed for these IBCs such that the solution of the DDM problem converges exactly to the reference solution (the solution of the monodomain problem). Finally, we perform a large set of numerical tests in order to optimize the IBCs, in the sense that we search the coefficients that provide the fastest convergence for the DDM iterative process.

## 2. Interface boundary condition operators based on the exact TBCs for the dispersion equation

### 2.1. The exact TBCs for the continuous equation

In [3], transparent boundary conditions (TBCs) are derived for the one-dimensional continuous linearized KdV equation (or Airy equation):

$$u_t + U_1 u_x + U_2 u_{xxx} = h(t, x), \quad t \in \mathbb{R}^+, \quad x \in \mathbb{R} \tag{2}$$

where $U_1 \in \mathbb{R}$, $U_2 \in \mathbb{R}_*^+$ and $h$ is a source term, assumed to be compactly supported in a finite computational domain $[a, b]$, $a < b$.

For the homogeneous initial boundary value problem

$$\begin{cases} u_t + U_1 u_x + U_2 u_{xxx} = 0, \quad t \in \mathbb{R}^+, \quad x \in [a, b] \\ u(0, x) = u_0(x), \quad x \in [a, b] \\ +\text{boundary conditions} \end{cases}$$

the TBCs are given by [3, equations (2.17) -(2.18)]

$$u(t, a) - U_2 \mathcal{L}^{-1} \left( \frac{\lambda_1(s)^2}{s} \right) * u_x(t, a) - U_2 \mathcal{L}^{-1} \left( \frac{\lambda_1(s)}{s} \right) * u_{xx}(t, a) = 0$$

$$u(t, b) - \mathcal{L}^{-1} \left( \frac{1}{\lambda_1(s)^2} \right) * u_{xx}(t, b) = 0 \tag{3}$$

$$u_x(t, b) - \mathcal{L}^{-1} \left( \frac{1}{\lambda_1(s)} \right) * u_{xx}(t, b) = 0$$

where $\mathcal{L}^{-1}$ denotes the inverse Laplace transform, $*$ the convolution operator, $s \in \mathbb{C}$, $Re(s) > 0$, is the Laplace frequency and $\lambda_1$ is, among the three roots of the cubic characteristic equation obtained when solving (2) in the Laplace space and in the complementary set of $[a, b]$, the only one with negative real part.

In this paper, we will focus on the special case $U_1 = 0, U_2 = 1$, which results on the dispersion equation (1). In this case, accordingly to [12], the only root with negative real part is

$$\lambda(s) = \lambda_1(s) = -\sqrt[3]{s} \tag{4}$$

### 2.2. Construction of operators based on the exact TBCs

The computation of the TBCs (3) is not simple due to the inverse Laplace transform, which makes these conditions nonlocal in time. Therefore, we will propose approximations of the root (4) that avoid integrations in time, making the operators considerably simpler.

Obviously, as we can see through the results shown in this section, when playing the role of transparent boundary conditions, these operators are not as accurate as the approximate TBCs proposed by [3] (who derives TBCs for the discrete linearized KdV equation). Nevertheless, the objectives of our work and the work of [3] are very different: while they seek to minimize the error of the computed solution (compared to the analytical one) due to the boundary conditions, we want here to apply our operators as interface boundary conditions (IBCs) in a domain decomposition method (DDM). Therefore, our objective lays on the convergence of the DDM to the solution of the same problem in the monodomain, independently of the errors on the external boundaries.

We will use the constant polynomial $P_0(s) = c$ for approximating $\lambda^2/s$. Moreover, as a consequence of (4), we can approximate the other operands of the inverse Laplace transforms in (3) only in function of $c$ :

$$\frac{\lambda^2}{s} = c, \qquad \frac{\lambda}{s} = -c^2, \qquad \frac{1}{\lambda(s)^2} = c^2, \qquad \frac{1}{\lambda(s)} = -c \qquad (5)$$

Replacing (5) in (3), using some well-know properties of the Laplace Transform (linearity and convolution) and considering possibly different polynomial approximations for the left and the right boundaries (respectively with the coefficients $c_L$ and $c_R$), we get the approximate transparent boundary conditions

$$\Theta_1^{c_L}(u, x) = u(t, x) - c_L u_x(t, x) + c_L^2 u_{xx}(t, x) = 0$$
$$\Theta_2^{c_R}(u, x) = u(t, x) - c_R^2 u_{xx}(t, x) = 0 \qquad (6)$$
$$\Theta_3^{c_R}(u, x) = u_x(t, x) + c_R u_{xx}(t, x) = 0$$

We notice that the approximation (6) has the same form as the exact TBCs for the equation (1) presented in [12] and [3], being the constants $c_L, c_R$ an approximation for fractional integral operators.

We also remark that (6) are mixed-type boundary conditions (up to the second derivative of the solution), which we will apply as interface boundary conditions in a domain decomposition method and we will seek to optimize in order to accelerate the convergence of this method. The idea of using optimized boundary conditions in DDMs was already explored in [7], in the context of the Schrödinger equation.

Considering a discrete domain with mesh size $\Delta x$ and points $x_0, ..., x_N$ and using some finite difference approximations, the operators (6) are discretized as

$$u_0 - c_L \frac{u_1 - u_0}{\Delta x} + c_L^2 \frac{u_0 - 2u_1 + u_2}{\Delta x^2} = 0$$
$$u_N - c_R^2 \frac{u_N - 2u_{N-1} + u_{N-2}}{\Delta x^2} = 0 \qquad (7)$$
$$\frac{u_N - u_{N-1}}{\Delta x} + c_R \frac{u_N - 2u_{N-1} + u_{N-2}}{\Delta x^2} = 0$$

In order to illustrate the results provided by these approximations, we briefly present some numerical tests with the same problem solved by [12] and [3], given by (8a)-(8c) and for which the exact solution is given by (9):

$$\begin{cases} u_t + u_{xxx} = 0, \quad x \in \mathbb{R} & (8a) \\ u(0, x) = e^{-x^2}, \quad x \in \mathbb{R} & (8b) \\ u \to 0, \quad |x| \to \infty & (8c) \end{cases}$$

$$u_{exact}(t, x) = \frac{1}{\sqrt[3]{3t}} Ai\left(\frac{x}{\sqrt[3]{3t}}\right) * e^{-x^2} \qquad (9)$$

where $Ai$ is the Airy function.

The numerical solution was computed with an implicit finite difference scheme, with second order discretizations for the spatial derivative. As done by [12] and [3], we solved the problem in the spatial domain $[-6, -6]$, for $0 \leq t \leq T_{max}$, with $T_{max} = 4$. The mesh size is $\Delta x = 12/500 = 0.024$ and the time step is $\Delta t = 4/2560 = 0.0015625$. We computed, as in [3], the following errors, respectively in each time step and in all the time interval :

$$e^n = \frac{\left\| u^n_{exact} - u^n_{computed} \right\|_2}{\| u^n_{exact} \|_2} \qquad e_{L2} = \sqrt{\Delta t \sum_{n=1}^{T_{max}} (e^n)^2}$$

In order to verify the influence of $c_L$ and $c_R$ on the computed solutions (and possibly identify a range of values that better approximate the TBCs), we made several tests with all the possible pairs $c_L, c_R \in \{-10, -1, -0.1, 0, 0.1, 1, 10\}^2$. The results were classified accordingly to their errors $e_{L2}$. Figure 1 shows, for some instants, a comparison between the best, the worst and the exact solution. For naming the worst result, we did not consider the ones in which the numerical solution diverged (following the arbitrary criteria $e_{L2} > 10$). Finally, Table 1 presents the ten tests that presented the smallest $e_{L2}$.
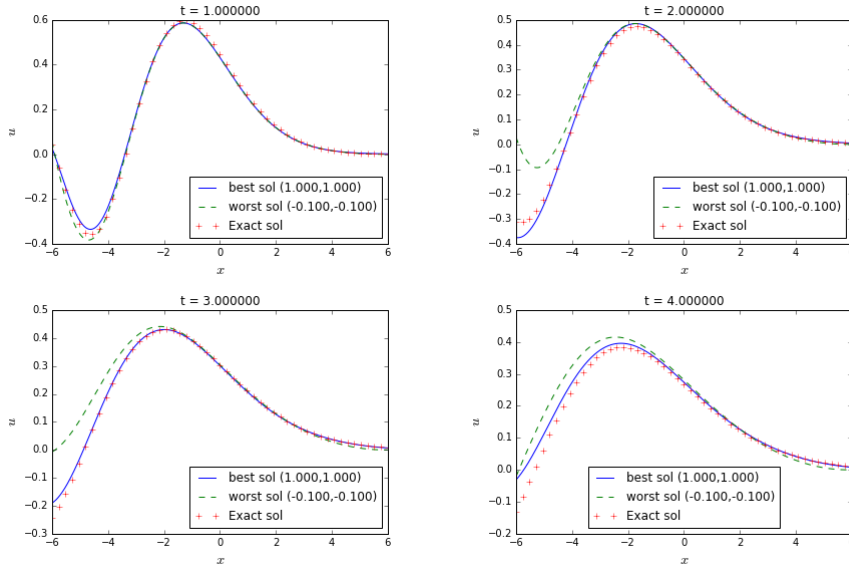


Figure 1: Best and worst solution compared with analytical solution, for the constant polynomial approximation

| $c_L$ | $c_R$ | $e_{L2}$ |
|-------|-------|----------|
| 1.0 | 1.0 | 0.1075 |
| 1.0 | 10.0 | 0.1099 |
| 1.0 | 0.1 | 0.1109 |
| 1.0 | 0.0 | 0.1116 |
| 1.0 | -10.0 | 0.1117 |
| 1.0 | -0.1 | 0.1123 |
| 1.0 | -1. | 0.1138 |
| 10.0 | 1.0 | 0.3447 |
| 10.0 | 0.1 | 0.3451 |
| 10.0 | 0.0 | 0.3452 |

Table 1: Best results (smallest $e_{L2}$) for the constant polynomial approximation

## 3. Application to a domain decomposition method

The operators (7) derived in the previous section will be applied as interface boundary conditions (IBC) in a domain decomposition method (DDM). Firstly, following [8], we will briefly describe the DDM that we will consider here, and after we will describe and test the incorporation of the proposed IBCs.

### 3.1. The Schwarz Method

Domain Decomposition Methods allow to decompose a domain $\Omega$ in multiple subdomains $\Omega_i$ (that can possibly overlap) and solve the problem in each one of them. Therefore, one must find functions that satisfies the PDE in each subdomain and that match on the interfaces.

The first DDM developed was the Schwarz method [8, 5], which consists on an iterative method: in the case of a evolution problem, the solution $u_i^{n,\infty}$, in each time step $t_n$ and each subdomain $\Omega_i$, is computed as the convergence of the solution obtained in each iteration, $u_i^{n,k}$, $k \geq 0$.

We will consider here the Additive (or parallel) Schwarz method (ASM). In this method, the interface boundary conditions are always constructed using the solution $u_j^{n,k-1}$, $j \neq i$ of the previous iteration in the neighbor subdomains. Therefore, in each interface between the subdomains $\Omega_i$ and $\Omega_j$, the boundary condition for the problem in $\Omega_i$ is

$$\mathcal{B}_i(u_i^{n,k+1}) = \mathcal{B}_i(u_j^{n,k}) \tag{10}$$

The ASM is a modification, proposed by [10], of the original (Alternating or Multiplicative) Schwarz Method, in which the IBCs are constructed using always the most updated solution of the neighbor domains. This modification originates an inherently parallel algorithm, which one naturally implements with parallel computing. The advantages obtained with the parallelism become more evident when the number of subdomains increases [10].

In (10), $\mathcal{B}_i$ denotes the operator of the IBC. This operator allows the construction of of more general Schwarz methods: in the original one, the IBC's are Dirichlet conditions (*i.e.*, $\mathcal{B}_i(u) = u$ ) [8, 11].

Without loss of generality, in the following we will consider a domain $\Omega$ decomposed in two non-overlapping subdomains, $\Omega_1$ and $\Omega_2$, with $\Gamma = \Omega_1 \bigcap \Omega_2$.

When implementing a Schwarz methods, one must define appropriate operators $\mathcal{B}_i$ such that:

- There is a unique solution $u_i$ in each subdomain $\Omega_i$;

- The solution $u_i$ in each subdomain $\Omega_i$ converges to $u|_{\Omega_i}$, *i.e.*, the solution $u$, restricted to $\Omega_i$, of the problem in the monodomain $\Omega$;

Moreover, one wants the method to show a fast convergence.

In fact, accordingly to [8], the optimal additive Schwarz method for solving the problem

$$\begin{cases} \mathcal{A}(u) = f \ \ \text{in} \ \ \Omega \\ u = 0 \ \ \text{on} \ \ \partial\Omega \end{cases}$$

where $\mathcal{A}$ is a partial differential operator, is the one which uses as interface boundary conditions the exact transparent boundary conditions, given by

$$B_i(u) = \frac{\partial}{\partial n_i} u + D2N(u)$$

where $\partial n_i$ is the outward normal to $\Omega_i$ on $\Gamma$, and the D2N (Dirichlet to Neumann) operator is defined by

$$D2N : \alpha(x) \mapsto \left. \frac{\partial}{\partial n_i^c} v \right|_\Gamma$$

with $\alpha$ defined on $\Gamma$. $v$ is solution of the following problem, solved in the complementary set of $\Omega_i$, denoted by $\Omega_i^c$

$$\begin{cases} \mathcal{A}(v) = f \ \ \text{in} \ \ \Omega_i^c \\ v = 0 \ \ \text{on} \ \ \partial\Omega_i \backslash \Gamma \\ v = \alpha \ \ \text{on} \ \ \Gamma \end{cases}$$

The ASM using such exact TBCs is optimal in the sense that it converges in two iterations, and no other ASM can converge faster [8]. Nevertheless, these TBC, in general, are not simple to compute both analtically and numerically. More specifically, they are nonlocal in time, so they must be approximated for an efficient numerical implementation [1]. These facts motivate us to implement the operators (7) as interface boundary conditions for the ASM: they were derived based on the exact TBCs for the equation (1), but, on the other hand, they are very simple to compute.

### 3.2. ASM with the proposed IBCs : an optimized Schwarz Waveform Relaxation method

The resolution of the dispersion equation (1) with the Additive Schwarz method, using the IBCs (7), is written as

$$
\begin{cases}
(u_1^{n,k+1})_t + (u_1^{n,k+1})_{xxx} = 0, & x \in \Omega_1, \quad t \geq t_0 \\
u_1^{n,0} = u_1^{n-1,\infty}, & x \in \Omega_1 \\
\Upsilon_1^{c_L}(u_1^{n+1,k+1}, -L) = 0, \\
\Theta_2^{c_R}(u_1^{n+1,k+1}, 0) = \Theta_2^{c_R}(u_2^{n,k}, 0), \\
\Theta_3^{c_R}(u_1^{n+1,k+1}, 0) = \Theta_3^{c_R}(u_2^{n,k}, 0)
\end{cases}
\tag{11}
$$

$$
\begin{cases}
(u_2^{n,k+1})_t + (u_2^{n,k+1})_{xxx} = 0, & x \in \Omega_2, \quad t \geq t_0 \\
u_2^{n,0} = u_2^{n-1,\infty}, & x \in \Omega_2 \\
\Theta_1^{c_L}(u_2^{n+1,k+1}, 0) = \Theta_1^{c_L}(u_1^{n,k}, 0) \\
\Upsilon_2^{c_R}(u_2^{n+1,k+1}, L) = 0 \\
\Upsilon_3^{c_R}(u_2^{n+1,k+1}, L) = 0
\end{cases}
\tag{12}
$$

where $\Upsilon_i$, $i = 1, 2, 3$, are the external boundary conditions (*i.e.*, defined on $\partial \Omega_i \backslash \Gamma$). As it solves a time-dependant problem in each subdomain, the method given by is (11)-(12) is called Schwarz Waveform Relaxation method [4].

Considering that we want to analyze and minimize the error due to the application of a domain decomposition method, the reference solution $u^{ref}$ in our study will be the solution of the monodomain problem

$$
\begin{cases}
u_t + u_{xxx} = 0, & x \in \Omega, \quad t \in [t_0, t_0 + \Delta t] \\
u(t_0, x) = u^{exact}(t_0, x), & x \in \Omega \\
\Upsilon_1(u, -L) = 0, & t \in [t_0, t_0 + \Delta t] \\
\Upsilon_2(u, L) = 0, & t \in [t_0, t_0 + \Delta t] \\
\Upsilon_3(u, L) = 0, & t \in [t_0, t_0 + \Delta t]
\end{cases}
\tag{13}
$$

We notice that we will always compare the solutions computed along only one time step. This is necessary for the separated study of the DDM (without influence, for example, of the error accumulated along the time steps, due to the temporal discretization).

The external BCs $\Upsilon_i$, $i = 1, 2, 3$ are independent of the interface BCs. Here, we will consider $\Upsilon_1 = \Theta_1^{c_L=1.0}$, $\Upsilon_2 = \Theta_2^{c_R=0.0}$ and $\Upsilon_3 = \Theta_3^{c_R=0.0}$, which gives

$$
\Upsilon_1(u, x) = u - u_x + u_{xx} = 0
$$
$$
\Upsilon_2(u, x) = u = 0
$$
$$
\Upsilon_3(u, x) = u_x = 0
$$

This choice was made based on the easy implementation and the good results provided by the coefficients $c_L = 1.0$ and $c_R = 0.0$ in approximating the analytical solution in $\Omega$ (as shown in Table 1). Nevertheless, it does not have much importance in the study that we will done here, as we want to study exclusively

the behavior of the DDM. The only restriction for an appropriate study is that the external BCs for computing $u_{ref}$ must be the same $\Upsilon_i$, $i = 1, 2, 3$, used for each subdomain in the DDM, as we did in (11)-(12) and (13).

A simple analysis (for example in the Laplace domain) shows that the monodomain and DDM problems (13) and (11)-(12) have an unique solution.

*Remarks on the notation.* As the following study will be made considering the execution of the method over only one time step, we can suppress the index denoting the instant $t_n$ and use a clearer notation for the solution: $u_j^i$, where $i$ indicates the subdomain $\Omega_i$ (or, in the case of the reference solution, $i = ref$, and in the convergence of the method, $i = *$) and $j$ indicates the spatial discrete position. In the cases where the iterative process is taken into account, we will add the superscript $k$ to indicate the iteration.

Concerning the spatial discretization, the monodomain $\Omega$ will be divided in $2N + 1$ homogeneously distributed points, numbered from 0 to $2N$. In all the analytical description, we will consider that the two subdomains $\Omega_1$ and $\Omega_2$ have the same number of points, respectively $x_0, ..., x_N$ and $x_N, ..., x_{2N}$. The interface point $x_N$ is common to the two domains, having different computed solutions $u_N^1$ and $u_N^2$ in each one of them. Evidently, we expect, at the convergence of the SWR, that $u_N^1 = u_N^2 = u_N^*$

*3.3. Discretization of the problem*

As done in the initial numerical tests in the section 2, an implicit Finite Difference scheme will be used here. For the interior points of each one of the domains, we will consider a second order spatial discretization of the equation (1):

$$\frac{u_j^i - \alpha_j^i}{\Delta t} + \frac{-\frac{1}{2}u_{j-2}^i + u_{j-1}^i - u_{j+1}^i + \frac{1}{2}u_{j+2}^i}{\Delta x^3} = 0 \qquad (14)$$

which is valid for $j = 2, ..., N - 2$ in the case $i = 1$; for $j = N + 2, ..., 2N - 2$ in the case $i = 2$; and for $j = 2, ..., 2N - 2$ in the case $i = ref$. In the above expression, $\alpha_j^i$ is a given data (for example, the converged solution in the previous time step).

For the points near the boundaries, we use second order uncentered discretizations or the appropriate boundary condition. Considering that one boundary condition is written for the left boundary and two for the right one, we have to impose an uncentered discretization only for the second leftmost point of the domain. For example, for the point $x_1$ :

$$\frac{u_1^2 - \alpha_1^2}{\Delta t} + \frac{-\frac{5}{2}u_1^2 + 9u_2^2 - 12u_3^2 + 7\frac{1}{2}u_4^2 - \frac{3}{2}u_5^2}{\Delta x^3} = 0$$

and similarly to the other points near the boundaries.

In the resolution of the problem in $\Omega_1$, two interface boundary conditions are imposed (corresponding to $\Theta_2$ and $\Theta_3$) to the discrete equations for the points $x_{N-1}$ and $x_N$. On the other hand, in the resolution of the problem in $\Omega_2$, only

one interface boundary condition is used (corresponding to $\Theta_1$), being imposed to the point $x_N$.

*Remark : modification of the reference solution.* Even if the DDM with the proposed interface boundary conditions is compatible with the monodomain problem (which we will see that is not the case), the solution of the DDM does not converge exactly to $u^{ref}$, for a reason that does not depend on the expression of the IBCs, but on the fact that for each domain we write two boundary conditions in the right boundary and only one on the left boundary. We are using a second order centered discretization for the third spatial derivative (which uses a stencil of two points in each side of the central point), implying that we must write an uncentered discretization for the point $x_{N+1}$ when solving the problem in $\Omega_2$. Therefore, this point does not satisfy the same discrete equation as in the reference problem. In order to avoid this incompatibility and allow us to study the behavior of the DDM, we will modify the discretization for the point $u_{N+1}$ in the monodomain problem, using the same second-order uncentered expression :

$$\frac{u_{N+1}^2 - \alpha_{N+1}^2}{\Delta t} + \frac{-\frac{5}{2}u_{N+1}^2 + 9u_{N+2}^2 - 12u_{N+3}^2 + 7\frac{1}{2}u_{N+4}^2 - \frac{3}{2}u_{N+1}^2}{\Delta x^3} = 0$$

Figure 2 resumes the discretizations imposed to each point in the monodomain and the DDM problems, as described above:
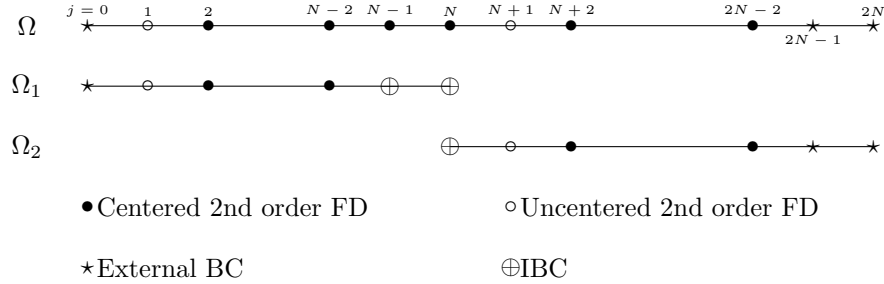


Figure 2: Scheme indicating the discretization imposed to each point in the monodomain and the DDM problems

*3.4. Corrections for the approximate IBCs*

When using approximate IBCs in the SWR, one should guarantee that the converged solutions $u^*$ satisfy the same equation as the solution $u_{ref}$ of the monodomain problem. Nevertheless, one can easily see that, in the convergence, the solution $u^*$ does not satisfy the discrete equation (14) on the points where the IBCs are imposed (the poins $x_{N-1}, x_N \in \Omega_1$ and $x_N \in \Omega_2$).

As pointed out by [6], a finite difference discretization of the IBCs requires a special treatment to be consistent with the monodomain discretization. Therefore, we will formulate modified IBCs for the optimized SWR in order to avoid this problem:

$$\Theta_1^{c_L}(u_2^{n+1,k+1}) + \theta_1 = \Theta_1^{c_L}(u_1^{n,k}) + \theta_1'$$
$$\Theta_2^{c_R}(u_1^{n+1,k+1}) + \theta_2 = \Theta_2^{c_R}(u_2^{n,k}) + \theta_2' \qquad (15)$$
$$\Theta_3^{c_R}(u_1^{n+1,k+1}) + \theta_3 = \Theta_3^{c_R}(u_2^{n,k}) + \theta_3'$$

with $\theta_i, \theta_i'$ given by

$$\theta_1 = \Delta x c_L \frac{u_{N+1}^2 - 2u_N^2 + u_{N-1}^1}{\Delta x^2} + c_L^2 \frac{\Delta x}{\Delta t}\left(u_N^2 - \alpha_N^2\right)$$
$$\theta_1' = -c_L^2 \frac{\Delta x}{\Delta t}\left(u_N^1 - \alpha_N^1\right)$$

$$\theta_2 = \frac{\Delta x}{\Delta t} c_R^2 (u_N^1 - \alpha_N^1)$$
$$\theta_2' = -\frac{\Delta x}{\Delta t} c_R^2 (u_N^2 - \alpha_N^2)$$

$$\theta_3 = 2\frac{\Delta x}{\Delta t}\left[-\Delta x(u_{N-1}^1 - \alpha_{N-1}^1) - c_R(u_N^1 - \alpha_N^1)\right] + \Delta x \frac{u_{N-3}^1 - 2u_{N-2}^1 + u_{N-1}^1}{\Delta x^2}$$
$$\theta_3' = 0$$

It is straightforward to verify that the DDM problem with these modifications in the IBCs insure that the converged solution $u^*$ satisfies, in every point, the same discrete equations as the solution $u^{ref}$ of the monodomain problem (13).

In addition, we notice that all the modification terms $\theta_i, \theta_i'$, $i = 1, 2, 3$, are of order $O(\Delta x)$ (they are composed of discrete versions of time derivatives and second spatial derivatives multiplied by $\Delta x$). It is essential to insure that these terms are small, for the consistency with the approximate IBCs $\Theta_i$ to be fulfilled.

### 3.5. Optimization of the IBCs (speed of convergence)

Our objective now is to optimize the IBCs in the sense of minimizing the number of iterations of our method until the convergence. We will make a very large set of tests in order to find the coefficients $c_L$ and $c_R$ ( which define the operators based a constant polynomial approximation for the TBCs) that provide the fastest convergence. To start with, we will make this study with fixed time step and space step, in order to analyze exclusively the influence of the coefficient.

As we are interested in the speed with which the solution of the DDM method converges to the reference solution, the criteria of convergence used is

$$e^{\Omega,k} \le \varepsilon$$

with $\varepsilon = 10^{-9}$ and

$$e^{\Omega,k} = ||u_N^{ref} - u_N^k||_2 = \sqrt{\Delta x \left[ \sum_{j=0}^{N} (u_j^{ref} - u_j^{1,k})^2 + \sum_{j=N}^{2N} (u_j^{ref} - u_j^{2,k})^2 \right]}$$

In order to simplify the tests and avoid expensive computations, we will always consider $c_L = c_R = c$ in this optimization. The range of tested coefficients is $[-10.0, 20.0]$ (chosen after initial tests to identify a proper interval), with a step equal to 0.1 between them (or even smaller, up to 0.005, in the regions near the optimal coefficients), and the maximal number of iterations is set to 100.

### 3.5.1. Test varying the initial data and the interface position

As said above, in the first set of tests we will consider a fixed time step $\Delta t = 20/2560 = 0.0078125$ and a fixed mesh size $\Delta x = 12/500 = 0.024$. Moreover, we will consider two subsets of tests, that will allow us to study the speed of convergence with different initial conditions and different sizes of the subdomains:

1. Tests varying the initial time step $t_0$, with the interface in the center of the monodomain $\Omega = [-6, 6]$;
2. Tests varying the position of the interface ($x_{interface} = -L + \alpha 2L$, where $L = 6$ and $0 < \alpha < 1$), for a fixed initial time $t_0 = 0.78125$.

In all the cases, the reference solution $u^{ref}$ will be the solution of the monodomain problem (13).

The results are summarized in Figure 3, with the number of iterations plotted as function of the coefficient $c$ (for the positive coefficients). We can see a very similar behavior of all the curves, with two minima whose position do not depend on $t_0$ and $\alpha$ (approximately, $c = 0.20$ and $c = 4.5$). For $c < 0$, the curves are very similar, with two minima located at $c = -0.10$ and $c = -1.35$, approximately. Moreover, the minima closest to zero ($c = -0.10$ and $c = 0.20$) are both associated with very discontinuous peaks, while the other two minima are associated with smoother curves. A detail of the curves around each positive minima are shown in Figures 3c - 3d and 3e - 3f. Finally, we remark that, for some curves, the minimal number of iterations is associated with the coefficients closest to zero, and, for other ones, to the other minimum, but the minimal number of iterations are very similar (between 5 and 7).
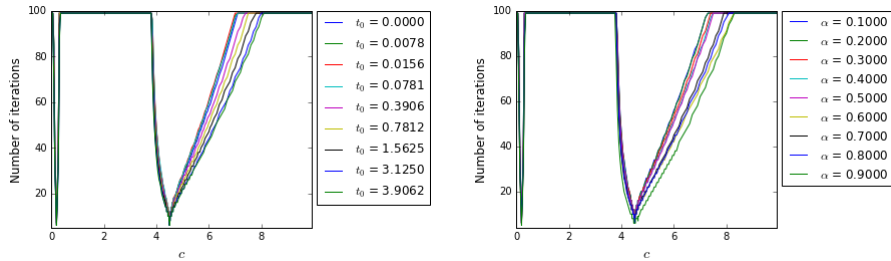


(a) General view (for a fixed interface and different values of $t_0$)



(b) General view (for a fixed $t_0$ and different positions of the interface)
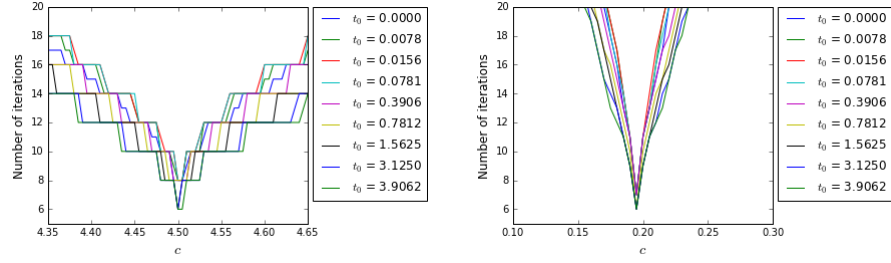
(c) Detail around one of the optimal coefficients (for a fixed interface and different values of $t_0$)

(d) Detail around the other optimal positive coefficient (for a fixed interface and different values of $t_0$)
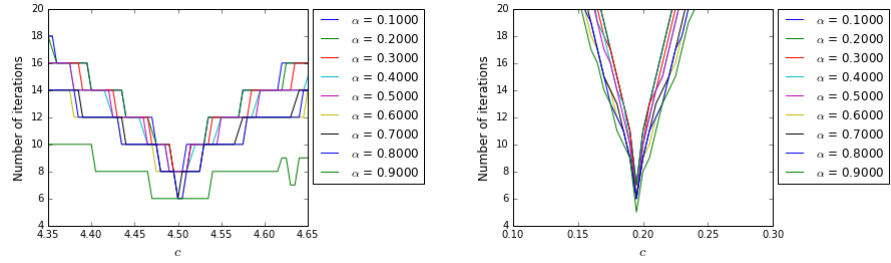


(e) Detail around one of the optimal coefficients (for a fixed $t_0$ and different positions of the interface)

(f) Detail around the other optimal positive coefficient (for a fixed $t_0$ and different positions of the interface)

Figure 3: Number of iterations until the convergence as function of the coefficient of the TBC, in the case of positive coefficients

Figure 4 shows the evolution of the error, as function of the iterations, for the five positive coefficients $c$ that gave the fastest convergences, for a fixed initial instant and a fixed position of the interface. For other values of $t_0$ and $\alpha$ this graph is similar, concerning the number of iterations and the fact that the convergence is more regular for the coefficients closest to zero, compared to the other optimal coefficients.
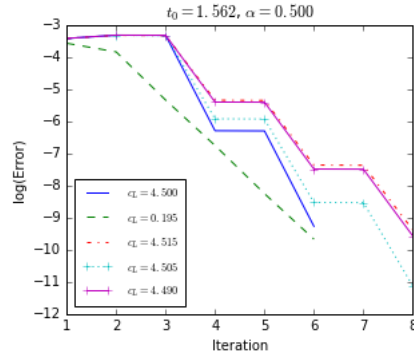


Figure 4: Error evolution with the iterations for the fastest results

13

*3.5.2. Tests varying $\Delta t$ and $\Delta x$*

After verifying that the method behaves similarly for every initial condition (*i.e.*, every $t_0$) and every position of the interface, we will now keep these parameters fixed ($t_0 = 0$ and $\alpha = 0.5$) and make new tests with different values of $\Delta t$ (with fixed $\Delta x = 12/250$) and different values of $\Delta x$ (with fixed $\Delta t = 0.02$).

The number of iterations as functions of the coefficient, for some of the tests, are shown in Figure 5, in the case of positive coefficients. The results for negative coefficients are similar.

Figure 6 presents the optimal positive coefficient for each $\Delta t$ or $\Delta x$ (for one fixed value for the other coefficient). Considering the observation we did before about the similar results (*i.e.* the number of iterations until the convergence) for the four optimal coefficients, we only took into account, for the construction of this curve, the positive minimum farther from zero: it was done because, as shown in Figure 5, these minima have a strong dependency on $\Delta t$ or $\Delta x$, and we will seek to study this relation.


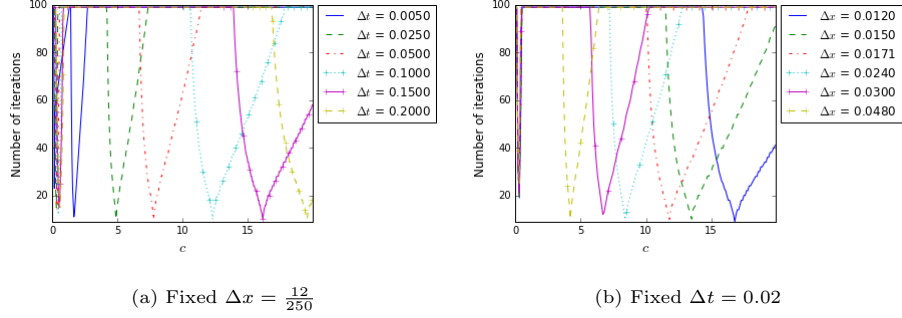
(a) Fixed $\Delta x = \frac{12}{250}$　　　　　　　　(b) Fixed $\Delta t = 0.02$

Figure 5: Number of iterations until the convergence as function of the coefficient of the TBC (for positive coefficients)



(a) Fixed $\Delta x = \frac{12}{250}$　　　　　　　　(b) Fixed $\Delta t = 0.02$
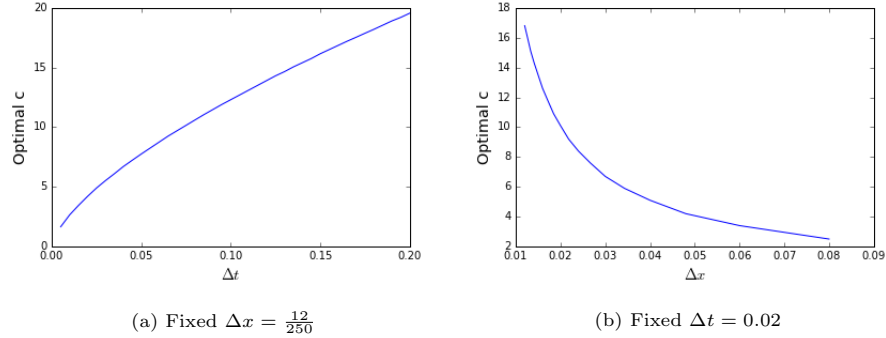
Figure 6: Optimal coefficients as function of the time step and the space step

Figure 6 suggests a dependence of the optimal coefficient on $(\Delta t)^\nu$ and $(\Delta x)^\eta$, with $0 \leq \nu \leq 1$ and $\eta < 0$. In fact, performing some regressions with $\Delta t$ or $\Delta x$ fixed, we could conclude that $\nu = \frac{2}{3}$ and $\eta = -1$ provide really well-fitted regression curves (with the coefficients of determination $R^2$ bigger than 0.99), both for the negative and the positive coefficients (although each

14

one of these cases correspond to different curves). Therefore, we will seek to model a function

$$c_{opt}(\Delta t, \Delta x) = \kappa + \alpha(\Delta t)^{\frac{2}{3}} + \beta \frac{1}{\Delta x} + \gamma \frac{(\Delta t)^{\frac{2}{3}}}{\Delta x}$$

A regression using the corners of the rectangle $[0.001, 0.1] \times [12/100, 12/1000]$ and fifteen inner points gives the surfaces

$$c_{opt}^{+}(\Delta t, \Delta x) = 0.0775 - 0.3353(\Delta t)^{\frac{2}{3}} - 0.0012 \frac{1}{\Delta x} + 2.7407 \frac{(\Delta t)^{\frac{2}{3}}}{\Delta x} \qquad (16)$$

$$c_{opt}^{-}(\Delta t, \Delta x) = -0.0583 - 1.5024(\Delta t)^{\frac{2}{3}} - 0.0006 \frac{1}{\Delta x} - 0.7287 \frac{(\Delta t)^{\frac{2}{3}}}{\Delta x} \qquad (17)$$

respectively for the positive and the negative optimal coefficients. The coefficients of determination of each regression are $R^{2,+} = 0.9999894$ are $R^{2,-} = 0.9998993$, showing an excellent representation.

In order to validate the expressions (16) and (17), we used them to compute the optimal coefficients for several points $(\Delta t, \Delta x)$, with $\Delta t \in [0.0005, 0.3]$ and $\Delta x \in [12/5000, 12/50]$. For almost all the points in the considered domain, the computed optimal coefficient provides a fast convergence to the monodomain solution, with less than 20 iterations, what is also observed in the case of the negative coefficients. The numbers of iterations observed are not always the smallest ones that we could find (cf. Figures 3 to 5), because the expressions (16) and (17) are regressions constructed from optimal coefficients obtained among a discrete set of possible values. Nevertheless, they give a very good approximation for the optimal $c$ for each $(\Delta t, \Delta x)$, and one could search around a small region around the computed $c_{opt}$ to obtain an even faster convergence.

### 3.6. Partial conclusion

The results presented in this section show that the domain decomposition method proposed here, consisting in an optimized SWR with our proposed interface conditions, is able to provide a fast convergence toward the solution of the monodomain problem. Furthermore, using the corrected IBCs (15), this convergence is exact. Therefore, we reached our goals of solving the dispersion equation in a finite domain divided in two subdomains.

Moreover, the results of the optimization tests are very satisfying regarding a more general application of our method. Firstly, for fixed spatial and temporal discretizations, we obtained optimal coefficients for the method independently of the initial solution and the size of the subdomains (*i.e.*, independently of the initial instant and the position of the interface). Secondly, we obtained good regression expressions for the optimal coefficient as function of $\Delta t$ and $\Delta x$, which could allow the application of the model, with fast convergence, in other computational frameworks.

## 4. Conclusion and outlook

We presented and implemented in this paper an optimized Schwarz Waveform Relaxation method for the resolution of an one dimensional dispersive evolution equation, using as interface conditions between the subdomains some operators constructed based on the exact transparent boundary conditions for this equation. Although not as accurate (in the role of TBCs) as the ones proposed in the works we are based on (providing better TBCs was not our objective here), these approximate conditions stand out for its simple form and implementation and the fast convergence that they provide for the Schwarz method. Moreover, we also proposed small corrections to them, which insure that the solution of the DDM problem converges exactly to the solution of the monodomain problem. Finally, we verified that the speed of convergence depends on the time step, the mesh size and the (only) coefficient for constructing the approximate interface conditions; thus, via an optimization process, we obtained and validated regression expressions that provide the optimal coefficient (*i.e.*, the one that provides the fastest convergence) in function of $\Delta t$ and $\Delta x$.

Natural continuations of the work presented here would be the study of the method considering simulations with many time steps and its extension to other problems, for example the linearized KdV equation, which adds an advective term on the equation solved here, as well as other models of wave propagation.

## Acknowledgement

## References

[1] X. Antoine, A. Arnold, C. Besse, M. Ehrhardt, C. Schädle, A review of Transparent Boundary Conditions or linear and nonlinear Schrödinger equations, Communications in Compuational Physics 4 (2008) 729–796.

[2] T.B. Benjamin, J.L. Bona, J.J. Mahony, Model equations for long waves in nonlinear dispersive systems, Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences 272 (1972) 47–78.

[3] C. Besse, M. Ehrhardt, I. Lacroix-Violet, Discrete Artificial Boundary Conditions for the Korteweg-de Vries Equation, 2015. Working paper or preprint.

[4] M.J. Gander, Overlapping Schwarz Waveform Relaxation for parabolic problems, Contemporary Mathematics 218 (1998) 425–431.

[5] M.J. Gander, Schwarz methods over the course of time., ETNA. Electronic Transactions on Numerical Analysis [electronic only] 31 (2008) 228–255.

[6] M.J. Gander, L. Halpern, F. Nataf, Internal report no 469 - Optimal Schwarz waveform relaxation for the one dimensional wave equation, Technical Report, École Polytéchnique - Centre de Mathématiques Appliquées, 2001.

[7] L. Halpern, J. Szeftel, Optimized and quasi-optimal schwarz waveform relaxation for the one dimensional schrödinger equation, in: U. Langer, M. Discacciati, D.E. Keyes, O.B. Widlund, W. Zulehner (Eds.), Domain Decomposition Methods in Science and Engineering XVII, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 221–228.

[8] C. Japhet, F. Nataf, The best interface conditions for Domain Decompostion methods : Absorbing Boundary Conditions, 2003. https://www.ljll.math.upmc.fr/ nataf/chapitre.pdf.

[9] D.J. Korteweg, G. de Vries, On the change of form of long waves advancing in a rectangular canal and on a new type of long stationary waves, Philosophical Magazine 5 (1895) 422–443.

[10] P.L. Lions, On the Schwarz alternating method.I, in: R.Glowinski, G.Golub, G.Meurant, J.Périaux (Eds.), Proceedings of the First International Symposium on Domain Decomposition Methods for Partial Differential Equations, SIAM, 1988, pp. 1–42.

[11] P.L. Lions, On the Schwarz alternating method III: a variant for nonoverlapping sub-domains, in: T. Chan, R. Glowinski, J. Périaux, O. Widlund (Eds.), Proceedings of the Third International Conference on Domain Decomposition Methods, p. 202–223.

[12] C. Zheng, X. Wen, H. Han, Numerical solution to a linearized kdv equation on unbounded domain, Numerical Methods for Partial Differential Equations 24 (2008) 383–399.

**LaTeX Source Files**