DDM - NSWE and Serre equations

Joao Guilherme CALDAS STEINSTRAESSER

November 14, 2016

1 The coupling method

We will explain in this section the scheme we propose for coupling the NSWE and the Serre equations. Without loss of generality, we will consider the domain $\Omega = \mathbb{R}$, divided in two subdomains Ω_1 and Ω_2, with $\Gamma = \Omega_1 \cap \Omega_2$, and we describe the conditions imposed on the interface Γ. We suppose that the Serre equations describe the wave propagation in Ω_1, while the NSWE describe it in Ω_2.

1.1 The models

The NSWE and the Serre equations are given respectively by

$$\begin{cases} h_t + (hu)_x = 0 \\ u_t + uu_x + gh_x = 0 \end{cases} \tag{1}$$

$$\begin{cases} h_t + (hu)_x = 0 \\ u_t + uu_x + gh_x - \frac{1}{3h}\left(h^3\left(u_{xt} + uu_{xx} - (u_x)^2\right)\right)_x = 0 \end{cases} \tag{2}$$

The main idea for the construction of our scheme is that the Serre equations (2) are solved using a splitting method, which separates the advection terms from the dispersive ones. Therefore, the resolution of the Serre equations consists in solving, in each time step:

$$\begin{cases} \tilde{h}_t + \left(\tilde{h}\tilde{u}\right)_x = 0 \\ \tilde{u}_t + \tilde{u}\tilde{u}_x + g\tilde{h}_x = 0, \ \ t \in [t_n, t_{n+1}], \ \ (\tilde{h}, \tilde{u})(x, t_n) = (h, u)(x, t_n) \end{cases} \tag{3}$$

$$\begin{cases} \overline{h}_t = 0 \\ \overline{u}_t - \frac{1}{3\overline{h}}\left(\overline{h}^3\left(\overline{u}_{xt} + \overline{u}\overline{u}_{xx} - (\overline{u}_x)^2\right)\right)_x = 0, \ \ t \in [t_n, t_{n+1}], \ \ (\overline{h}, \overline{u})(x, t_n) = (\tilde{h}, \tilde{u})(x, t_{n+1}) \end{cases} \tag{4}$$

$$\begin{cases} (h, u)(x, t_{n+1}) = (\overline{h}, \overline{u})(x, t_{n+1}) \end{cases} \tag{5}$$

If we denote the two systems by the operators $T_a^{\Delta t}$ and $T_d^{\Delta t}$, respectively, where the superscript indicates that the operator is performed over a time step Δt, the resolution of the Serre equations can be written as

$$(h,u)(x,t_{n+1}) = T_d^{\Delta t}\left(T_a^{\Delta t}\left((h,u)(x,t_n)\right)\right) \tag{6}$$

We remark that the first step of the splitted Serre equations (equation 3) corresponds to the NSWE (equation 1).

Different numerical schemes are proposed for solving each one of these equations, taking advantage of their properties. The advection equation (3) (and 1), which can be written in a conservative form, is solved using a finite volume method, with an approximate Riemann solver and fourth-order MUSCL interpolation on each cell interface. The dispersive equation (4) is solved using an explicit finite difference scheme with fourth-order spatial discretization. We refer to the previous reports for details about these implementations.

1.2 The coupling

Our coupling scheme takes advantage of the fact that the first step of the splitted Serre equations is identical to the NSWE. Indeed we propose, in each time step :

1. A Domain Decomposition Method between NSWE in Ω_1 and NSWE in Ω_2 (coupling between (1) and (3));
2. A "Domain Decomposition Method" between the dispersive equation (4) in Ω_1 and a zero solution in Ω_2.

The quotes in the second step refer to the fact that a DDM between an equation and a zero solution is, in fact, a Transparent Boundary Condition problem.

1.2.1 The interface conditions

Appropriate interface conditions must be imposed in order to the DDM problem give "correct" solutions regarding the solution given by the monodomain problem. Before detailing the conditions imposed in each step of the coupling, we describe the spatial discretization around the interface.

We define Ω_1 and Ω_2 such that they have exactly one common point, located on the interface Γ. Moreover, as the fourth-order finite volume scheme requires three ghost cells in each boundary, we extend the subdomains by three cells in each side, giving the discretization showed in the figure 1 :

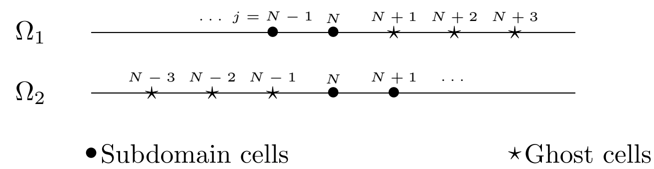


Figure 1: Scheme indicating the spatial discretization of the subdomains around the interface

We remark that the solution is computed only in the "subdomain cells".

1.2.2 Notation

The discrete solutions of the problem will be denoted as

$$\begin{array}{lll} u_{j,A}^{i,n}, & u_{j,D}^{i,n} & u_j^{i,n} \\ h_{j,A}^{i,n}; & h_{j,D}^{i,n} & h_j^{i,n} \end{array}$$

where i refers to the subdomain, n to the time step, j to the spatial point and A (respectively D) to the solution of the advection (dispersive) step. When this last subscript is not used, we refer to the final solution of the time step.

1.2.3 DDM between NSWE and NSWE

The imposition of the interface conditions for coupling the two NSWE models is made simply by copying the respective solutions to the ghost cells in the beginning of each time step, *i.e.*, considering the discretization showed in the figure 1 :

$$\begin{array}{ll} u_{j,A}^{1,n} = u_j^{2,n-1}, & j \in \{N+1, N+2, N+3\} \\ u_{j,A}^{2,n} = u_j^{1,n-1}, & j \in \{N-3, N-2, N-1\} \end{array} \tag{7}$$

and similarly to h.

Such interface conditions are the characteristic conditions, which recovers exactly the monodomain finite volume discretization of the problem, *i.e.*, the appropriate interface conditions are naturally imposed in the DDM. Therefore, **with no iteration, the DDM solution of the coupling NSWE-NSWE is exactly the solution of the NSWE in the monodomain**. We also notice that, as the stencils of the common point u^N becomes identical in both subdomains, we always have $u_{N,A}^{1,n} = u_{N,A}^{2,n}$.

1.2.4 TBC problem for the dispersive equation

The appropriate Transparent Boundary Condition that should be imposed for the dispersive equation are not yet very clear. After trying some alternatives and comparing qualitatively the results and the stability provided by them, we propose a simple TBC which consists in imposing the solution of the advection step in the two points closest to the interface :

$$u_{j,D}^{1,n} = u_{j,A}^{1,n}, \qquad j = N-1, N \tag{8}$$

We recall that in the dispersive part of the Serre equation, only the solution u is modified, because h is constant in time (as shown in the equation 4).

We also remark that **the computation of the solution of the dispersive step does not require iterations.** Indeed, usual DDMs lead to iterative processes because the solution of the neighbour subdomain changes in each iteration (so the right-hand side of the problem changes), but, in this case, the neighbour solution is always zero.

2 Numerical tests

There is no analytical solution to the problem we solve here (coupling between NSWE and Serre equations); therefore, we will perform a qualitative validation of the numerical solution, analysing if it is "reasonable" and comparing it to solution of the NSWE model applied to all the model (which can be computed simply by omitting the dispersive step of the method).

We first make a test with a horizontal bottom, with the couplings NSWE-Serre and Serre-NSWE. After, looking forward to more realistic applications, we test the method on a variable bottom, simulating the near shore relief.

2.1 Tests with a horizontal bottom

We divide the domain Ω in two subdomains of same size, Ω_1 and Ω_2. We use as initial solution a solitary cnoidal solution with parameters ($a_0 = 0.3$; $a_1 = 0.02$) such that it separates in two waves travelling in opposite senses under the NSWE model. Therefore, in the coupling system, we will have one wave in the NSWE domain, and another one in the Serre domain, so we expect to observe the differences provided by each model. The choice of the parameters was also made in order to postpone the shock formation in the NSWE model (as our finite volume scheme does not use limiters, discontinuous solutions provoke instabilities).

We applied the interface boundary conditions described in (7) and (9). Concerning the external boundary conditions, we assured that the domain is large enough so the solution doesn't reach the boundaries inside the time of simulation, and we applied open boundary conditions for the NSWE and Dirichlet homogeneous boundary conditions for the dispersive step.

The figures 2 and 3 show the solution in the final instant of simulation. The two figures differ from the subdomains where each model is applied.

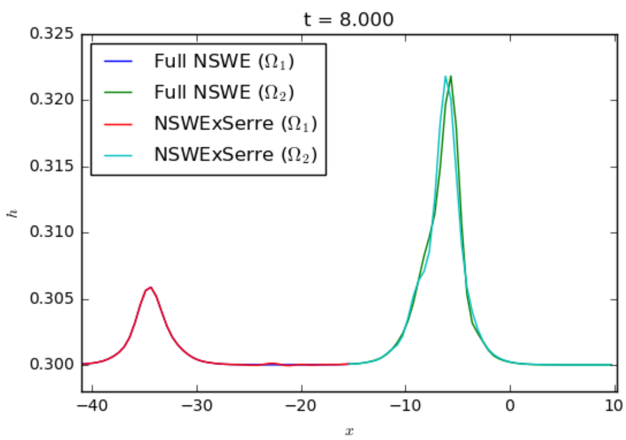


Figure 2: Coupling between NSWE (in the left subdomain) and Serre (in the right subdomain), and comparison with NSWE in the whole domain

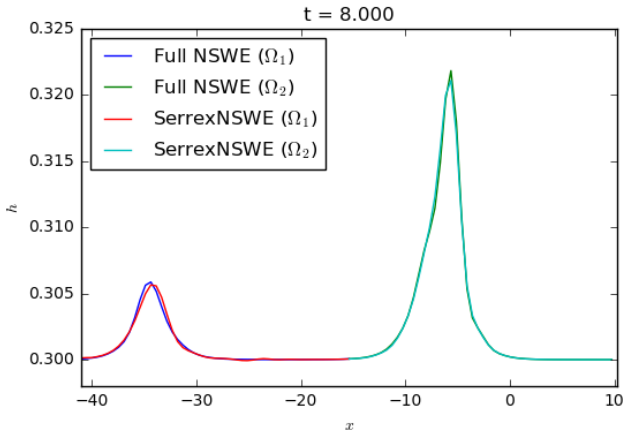


Figure 3: Coupling between Serre (in the left subdomain) and NSWE (in the right subdomain), and comparison with NSWE in the whole domain

Both figures show clearly the difference between the two models. The NSWE provokes a shock formation, *i.e.*, a deformation of the wave, while the Serre equations preserve both the form and the amplitude of the wave (because the cnoidal solitary wave is a solution of the Serre equations).

2.2 Tests with a variable bottom

The idea here is to simulate the approximation of the wave to the coast : as the depth is no longer constant, the wave propagation is described by different models.

We simulate a wave travelling to the right, with the domain Ω divided in three subdomains of same size: in the extremal subdomains, Ω_1 and Ω_3, we apply the NSWE model; in the center subdomain, Ω_2, we apply the Serre model. The variation of the bottom occurs mainly in Ω_2.

In our initial tests, the bottom consisted in a constant slope between two horizontal platforms. Nevertheless, we observed instabilities of the numerical schemes due to the discontinuities of the derivative of the bottom. Some instabilities were also observed in the case where the bottom isn't horizontal near the boundaries. Therefore, to avoid these problems, we construct the bottom with an arctangent function, approximately horizontal in the boundaries. The figure 4 shows the bottom and the initial solution profiles :

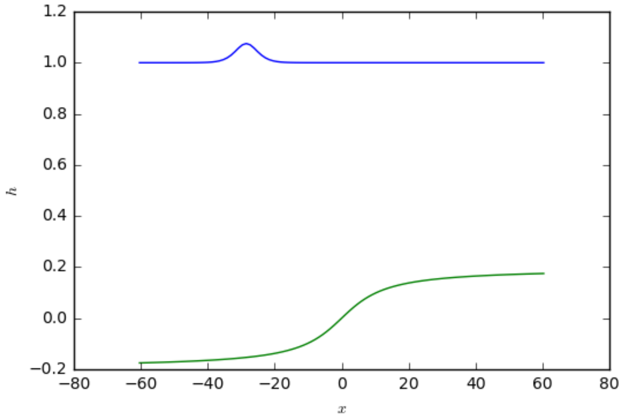


Figure 4: Profiles of the bottom and the initial solution

2.2.1 Interface boundary conditions

The interface boundary conditions applied in this problem are essentially the same as described above (equations 7 and 9), with only a small modification in the TBCs for the Serre subdomain : as the dispersive part of the Serre equations requires three boundary conditions, we consider two TBCs for its left boundary and one for the right, but constructed by the same principle (imposition of the solution of the advection step).

Therefore, if we denote by the index N the common point between Ω_1 and Ω_2, and by $2N$ the common point between Ω_2 and Ω_3, the interface boundary conditions four our coupling are :

1. Advection step :

$$\begin{aligned}
u_{j,A}^{1,n} &= u_j^{2,n-1}, \qquad j \in \{N+1, N+2, N+3\} \\
u_{j,A}^{2,n} &= u_j^{1,n-1}, \qquad j \in \{N-3, N-2, N-1\} \\
u_{j,A}^{2,n} &= u_j^{3,n-1}, \qquad j \in \{2N+1, 2N+2, 2N+3\} \\
u_{j,A}^{3,n} &= u_j^{2,n-1}, \qquad j \in \{2N-3, 2N-2, 2N-1\}
\end{aligned}$$

2. Dispersive step :

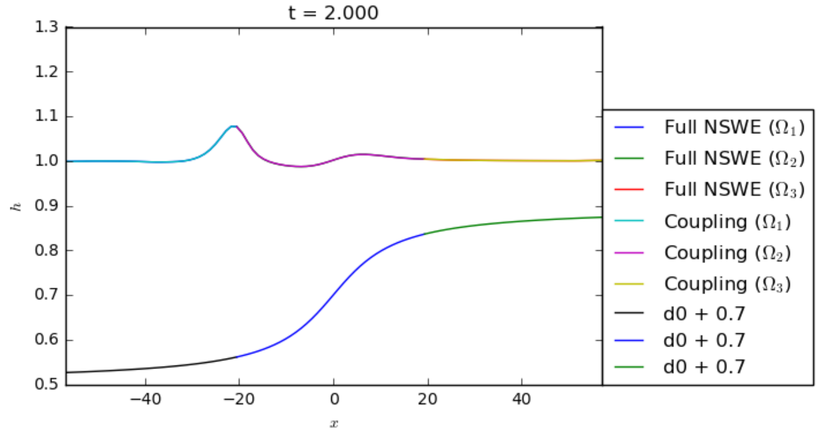
$$\begin{aligned}
u_{j,D}^{2,n} = u_{j,A}^{2,n}, \qquad j = N, N+1 \\
u_{2N,D}^{2,n} = u_{2N,A}^{2,n}
\end{aligned} \tag{9}$$

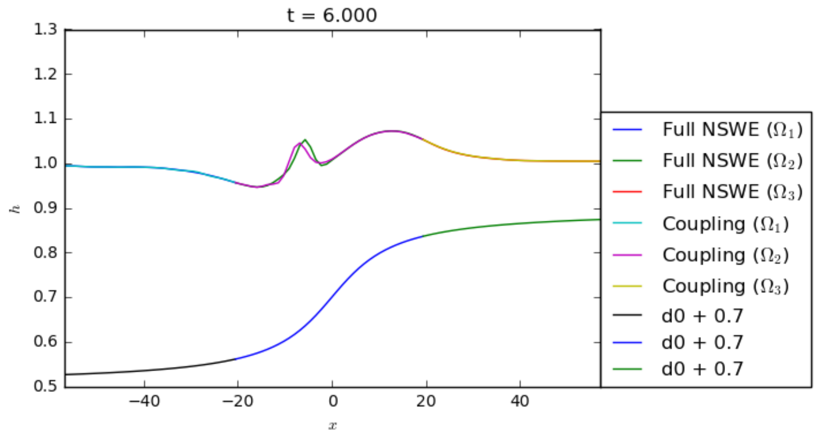
Concerning the external boundaries of Ω, we always consider for the right boundary the standard "open boundary conditions" for the finite volume scheme, and we make tests with different conditions for the left boundary. We recall that open boundary conditions are imposed by simply reflecting the solution (h and u) around the boundary to the ghost cells :

$$u_{3N+j,A}^{3,n} = u_{3N+1-j}^{3,n-1}, \qquad j \in \{1, 2, 3\} \tag{10}$$

2.2.2 Test with open left boundary conditions

In this first test, we also impose to the left boundary the standard open boundary conditions, similarly to (10). We use as initial solution a solitary cnoidal wave with parameters $(a_0, a_1) = (1.0, 0.075)$ and we perform the tests until the shock formation. As done previously, we compare the solution of the coupling with the solution provided by the NSWE applied to the entire domain. The solution in some instants is presented in the figure 5, where the bottom was translated in order to appear in the plots.





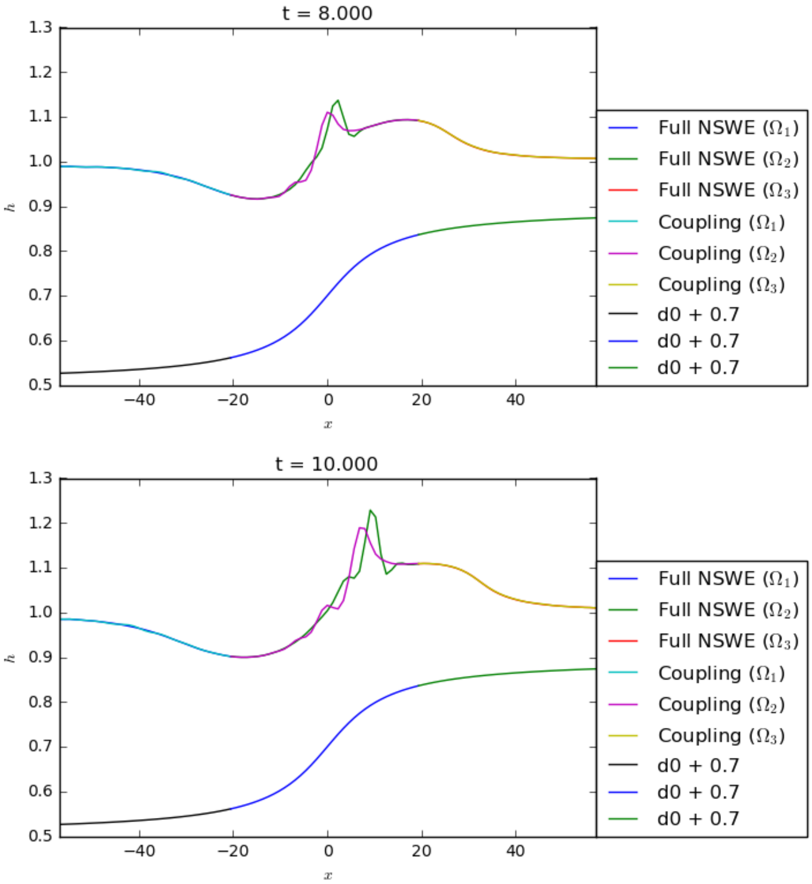


Figure 5: Result of the coupling and of the NSWE in the entire domain, for open boundary conditions on the left boundary. The bottom was translated to the appear in the plots

2.2.3 Tests with a constant height imposed on the left boundary

...