Coupling Parareal with Optimized Schwarz waveform relaxation for parabolic problems

Duc Quang Bui, Caroline Japhet, Yvon Maday, Pascal Omnes

COUPLING PARAREAL WITH OPTIMIZED SCHWARZ WAVEFORM RELAXATION FOR PARABOLIC PROBLEMS *

DUC QUANG BUI[‡], CAROLINE JAPHET[‡], YVON MADAY[§] AND PASCAL OMNES[‖‡]

Abstract. We propose and analyze a new parallel paradigm that uses both the time and the space directions. The original approach couples the Parareal algorithm with incomplete optimized Schwarz waveform relaxation (OSWR) iterations. The analysis of this coupled method is presented for a one-dimensional advection-reaction-diffusion equation. We also prove a general convergence result for this method via energy estimates. Numerical results for two-dimensional advection-diffusion problems and for a diffusion equation with strong heterogeneities are presented, to illustrate the performance of the coupled Parareal-OSWR algorithm.

Key words. optimized Schwarz waveform relaxation, Domain decomposition, Parareal in time algorithm, Robin transmission conditions, Convergence rates.

1. Introduction. The Parareal algorithm is a time-parallel method that was proposed by Lions, Maday, and Turinici [28] to solve evolutionary problems in parallel. The algorithm is constructed using two solvers: the coarse solver, which is fast but not very accurate, and the fine solver, which is slower but more accurate. The long time interval is divided into smaller windows, the fine solver is performed on each window, using some input initial conditions. The outputs are then corrected by the coarse solver and used as the inputs for the next iteration. In the most simplified convergence analysis, the initial value problem was considered to be an ODE, the fine solver was exact and the coarse solver was the Backward Euler method; in that analysis the outputs were computed exactly and so was the error. More complex analyses of the performance of Parareal were also achieved: for nonexact fine solvers for ODEs [5, 4], for linear ordinary and partial differential equations [18]; for the nonlinear case [11]; for improvement of the parallel efficiency [32].

In the construction of Parareal, one may choose suitable coarse and fine solvers to accelerate the process, for example by using an iterative method like, e.g., the Schwarz waveform relaxation (SWR) method [17]. As Parareal itself is also an iterative method, the overall process would then be composed of outer iterations (Parareal) and inner iterations corresponding to the coarse and fine solvers. In order to save CPU resources, we might think of stopping the inner iterative solvers after a small number of iterations, well before convergence, and hope that the overall convergence could be carried through Parareal iterations. This idea was first raised in [33] for general iterative methods; it was then developed for some iterative methods: for the Jacobi method (see [36]), for SWR methods (see [20, 15]). In this contribution, we are more interested in the latter, which will be recalled next.

SWR algorithms are based on a spatial domain decomposition. The spatial domain is decomposed into overlapping or nonoverlapping subdomains; then the original problem is transformed into multidomain space-time sub-problems. Starting from ini-

*This work was supported by the ANR project CINE-PARA under grant ANR-15-CE23-0019.

[‡]CNRS, UMR 7539, Laboratoire de Géométrie, Analyse et Applications, LAGA, Université Sorbonne Paris Nord, F-93430, Villetaneuse, France, (bui@math.univ-paris13.fr, japhet@math.univ-paris13.fr).

[§]Sorbonne Université, CNRS, Université de Paris, Laboratoire Jacques-Louis Lions (LJLL), F-75005 Paris, France, and Institut Universitaire de France, 75005 Paris, France (maday@ann.jussieu.fr).

[‖]CEA Saclay, DM2S-STMF, F-91191 Gif sur Yvette Cedex, France (pascal.omnes@cea.fr).

tial fluxes on the space-time interfaces, each subproblem can then be solved in parallel over the whole time range; then the data are exchanged through the interfaces to create better fluxes. The transmission conditions for the fluxes play an important role in the convergence process (especially for advection-reaction-diffusion problems) and several possibilities can be used, e.g., Dirichlet [19], Robin [34, 35], [12] or Ventcell [27, 21]. The last two types of conditions can be optimized [24, 26, 10], and in that case the corresponding SWR method is called optimized Schwarz waveform relaxation (OSWR). The OSWR method was introduced in [14], and analyzed and extended to linear advection-reaction-diffusion problems in [34, 35]. While in the classical Schwarz method, spatial fluxes are exchanged at each time step [10], the space-time fluxes in the SWR or OSWR methods leave a wide range of choices for the discretization method in the time direction in each sub-problem, which is quite useful in practice (see e.g. [21, 22, 23]).

We now explain the suitability of the SWR method. On the one hand, the analysis of this method, as well as its optimization, can be carried out directly at the (continuous) PDE level, which yields insight which is completely independent from the actual discretization that is used in practice. On the other hand, it can be run in parallel (one processor for each subdomain), and this allows a two-level parallelization process: one level in Parareal iterations, and the other in the SWR (or OSWR) iterations. In addition, as we do not run the solvers until convergence, we shall need to keep additional intermediate outputs from the solvers during Parareal iterations: these outputs are fluxes on the interface. Coupling Parareal with SWR with Dirichlet transmission condition was proposed in [20, 15], and in [38] for Dirichlet–Neumann/Neumann–Neumann waveform relaxation method.

The purpose of this paper is to propose and analyze a coupled method, called a Parareal-OSWR method [7, 8], that uses Parareal with incomplete OSWR iterations (with optimized transmission conditions) for the fine propagator. We are not aware of any convergence analysis of such method. While OSWR converges much faster than Dirichlet SWR, it also amplifies the difficulties in the convergence analysis, which cannot be performed using the same techniques as for Dirichlet SWR. We provide a convergence analysis, based on energy estimates, for one-dimensional parabolic equations and Robin transmission conditions. Numerical experiments are shown in that case, and also for two-dimensional parabolic equations, with Robin or Ventcel conditions, and in the case where OSWR is accelerated by GMRES (see e.g. [26, 9, 22, 2]).

Our paper is organized as follows. In Section 2, we introduce the model problem and state some stability and regularity properties of the solutions of the whole space-time domain problem and of the sub-space-time-domain problems. In Section 3 we recall the Parareal algorithm. In Section 4 we recall and extend some known results about the OSWR method, which will be used for the analysis of the coupled method. In Section 5, we introduce the coupled Parareal-OSWR algorithm and prove a general convergence result for the method via energy estimates in Section 6. Finally, in Section 8, one-dimensional (1D) and two-dimensional (2D) numerical results, comparing the different methods (OSWR, Parareal, and the coupled Parareal-OSWR algorithm), are discussed. In the 1D case, a comparison is also given with the PSWR method introduced in [38].

2. Model problem. Denote $\mathcal{L}u := \partial_t u - \nu\partial_{xx}u + a\partial_x u + bu$, $\Omega = \mathbb{R}$. For $T > 0$, we consider the following problem

$$\mathcal{L}(u) = f, \quad \text{in } \Omega \times (0,T), \tag{2.1a}$$
$$u(\cdot,0) = u_0, \quad \text{in } \Omega, \tag{2.1b}$$

where ν, a and b are constants, with $\nu > 0$ and $b \geq 0$. The source term f and the initial condition u_0 will be specified in Section 2.2.

2.1. Domain decomposition and notation. We consider a decomposition of the domain Ω into two nonoverlapping subdomains

$$\Omega_1 = (-\infty, 0), \quad \Omega_2 = (0, +\infty),$$

and introduce the Robin interface operator for $i = 1, 2$ as follows (see [30, 12]):

$$\mathcal{B}_1 = \nu\partial_x - \frac{a}{2} + \frac{p}{2}, \qquad \mathcal{B}_2 = -\nu\partial_x + \frac{a}{2} + \frac{p}{2}. \tag{2.2}$$

Then, problem (2.1) can be reformulated as the following equivalent multidomain problem [31], with $f_i = f_{|\Omega_i}$, $u_i = u_{|\Omega_i}$, and $u_{0,i} = u_{0|\Omega_i}$, $i = 1, 2$:

$$\begin{aligned} \mathcal{L}u_i &= f_i && \text{in } \Omega_i \times (0,T), \\ u_i(\cdot,0) &= u_{0,i} && \text{in } \Omega_i, \\ (\mathcal{B}_i u_i)(0,\cdot) &= \xi_i && \text{on } (0,T), \end{aligned} \qquad i = 1, 2, \tag{2.3}$$

with

$$\xi_i := (\mathcal{B}_i u_j)(0,\cdot) \quad \text{on } (0,T), \quad j = 3 - i, \ i = 1, 2. \tag{2.4}$$

In (2.3), the transmision condition involves the parameter p (through $\mathcal{B}_i$ defined in (2.2)), that is a free parameter chosen such that : a) a Robin subdomain problem of type (2.3) is well-posed, b) it leads to a fast converging algorithm (see Section 4). Both issues will be specified later.

In what follows we will use the notation $\boldsymbol{\xi} := (\xi_1, \xi_2)$ for the Robin data on $(0,T)$ associated with the solution u of (2.1).

2.2. Existence and regularity results. We introduce the following spaces corresponding to the subdomain problems,

$$\mathcal{X}_i = H^1(\Omega_i), \ i = 1, 2, \qquad \mathcal{Y} = H^{\frac{1}{4}}(0,T),$$

and the broken Sobolev space

$$\mathcal{X} := \{u \in L^2(\Omega) : u_{|\Omega_i} \in \mathcal{X}_i\},$$

equipped with the norm $\|u\|_{\mathcal{X}} = \left(\sum_i \|u_{|\Omega_i}\|^2_{\mathcal{X}_i}\right)^{\frac{1}{2}}$.

With the Robin transmission conditions, we will need more regularity in our analysis, in the anisotropic Sobolev spaces $H^{r,s}(\Omega \times (0,T)) = L^2(0,T;H^r(\Omega)) \cap H^s(0,T;L^2(\Omega))$ defined in [29].

We recall below some useful regularity properties from [30].

LEMMA 2.1. *(Regularity of problem (2.1)[1])*
If $u_0 \in H^1(\Omega)$ and $f \in L^2(0,T;L^2(\Omega))$, problem (2.1) has a unique solution u in $H^{2,1}(\Omega \times (0,T))$ and there exists a constant C independent of u_0 and f s.t.

$$\|u\|_{H^{2,1}(\Omega\times(0,T))} \leq C(\|u_0\|_{H^1(\Omega)} + \|f\|_{L^2(0,T;L^2(\Omega))}). \tag{2.5}$$

[1] See [30] Theorem 6.2 (extended to $\mathbb{R}$ in space), with $m = 1$ and $r = 0$.

LEMMA 2.2. *(Regularity of problem (2.3)[2])*
Let $i = 1$ or $i = 2$. If $u_{0,i} \in \mathcal{X}_i$, $f_i \in L^2(0,T;L^2(\Omega_i))$ and $\xi_i \in \mathcal{Y}$, problem (2.3) has a unique solution u_i in $H^{2,1}(\Omega_i \times (0,T))$ and there exists a constant C independent of u_0, f, and ξ_i s.t.

$$\|u_i\|_{H^{2,1}(\Omega_i\times(0,T))} \leq C(\|u_{0,i}\|_{\mathcal{X}_i} + \|f_i\|_{L^2(0,T;L^2(\Omega_i))} + \|\xi_i\|_{\mathcal{Y}}).$$

LEMMA 2.3. *(Trace theorem[3])*
If $u \in H^{2,1}(\Omega \times (0,T))$, then $u(\cdot,T) \in \mathcal{X}$, $(\mathcal{B}_i u)(0,\cdot) \in \mathcal{Y}$, $i = 1,2$, and there exists a constant C s.t.[4]

$$\|(\mathcal{B}_i u)(0,.)\|_{\mathcal{Y}} \leq C\|u\|_{H^{2,1}(\Omega\times(0,T))}, \quad i = 1,2.$$

Similar estimates hold by replacing Ω by Ω_i, u by u_i, and $\mathcal{X}$ by $\mathcal{X}_i$, for $i = 1,2$.

3. Parareal Method. The Parareal method introduced in [28] is a numerical method designed to solve evolution problems in parallel. It is based on a *decomposition in time* of $(0,T)$ into subintervals : $\overline{(0,T)} = \cup_{n=0}^{N-1}\overline{\mathcal{I}_n}$, with $\mathcal{I}_n = (T_n, T_{n+1})$, for $0 \leq n \leq N-1$, and $0 = T_0 < T_1 < \cdots < T_{N-1} < T_N = T$. Over each such interval generically noted as $\mathcal{I} := (t_0, t_1)$, it uses two propagation operators :

- $\mathcal{G}(\mathcal{I}, U_0)$ that provides a rough approximation of $u(\cdot, t_1)$, where u is the solution of (2.1), with initial condition $u(\cdot, t_0) = U_0$.
- $\mathcal{F}(\mathcal{I}, U_0)$ that provides a more accurate approximation of $u(\cdot, t_1)$.

For simplicity, we will consider a regular decomposition of $(0,T)$, i.e., such that $T_{n+1} - T_n = \Delta T$, $\forall n \in [\![0, N-1]\!]$. The plain Parareal algorithm [28] is as follows.

Algorithm 3.1 (Parareal)

Choose an initial data $(U_n^0)_{n\in[\![0,N]\!]}$ with $U_0^0 = u_0$ and U_n^0 an approximation of $u(\cdot, T_n)$, for example : $U_n^0 := \mathcal{G}(\mathcal{I}_{n-1}, U_{n-1}^0)$, for $n = 1, 2, \ldots, N$.

for $k = 0, 1, \ldots$ *(Parareal iterations)* **do**

Set $U_0^{k+1} = u_0$ and perform the correction iterations

$$U_{n+1}^{k+1} = \mathcal{G}(\mathcal{I}_n, U_n^{k+1}) + \mathcal{F}(\mathcal{I}_n, U_n^k) - \mathcal{G}(\mathcal{I}_n, U_n^k), \quad n = 0, 1, \ldots, N-1. \tag{3.1}$$

end for

We denote by u_n the solution of the (sequential) fine propagator at time T_n:

$$u_n = \mathcal{F}((0,T_n), u_0), \quad u_n = \mathcal{F}(\mathcal{I}_{n-1}, u_{n-1}), \quad \forall n \in [\![1, N]\!].$$

In practice, $\mathcal{F}$ will be close to exact, and thus, for the analysis presented here, we suppose that $\mathcal{F}$ is the exact propagator, i.e., $\mathcal{F}((t_0,t_1), \tilde{u}_0) = \hat{u}(t_1)$ where $\hat{u}$ is the solution of (2.1a) with initial condition $\tilde{u}_0$ at $t = t_0$. In particular we can identify u_n with $u(\cdot, T_n)$, where u is the solution of (2.1).

4. Optimized Schwarz Waveform Relaxation Method. The OSWR algorithm [13, 14, 35], for solving problem (2.1) is a space-time parallel method based on a *domain decomposition in space* only

$$\Omega_1 = (-\infty, 0), \quad \Omega_2 = (0, +\infty).$$

[2]See [30] Theorem 6.2 (extended to $\mathbb{R}$ in space), with $m = m_j = 1$ and $r = 0$.
[3]See [30] Theorem 2.1, with $r = 2$ and $s = 1$.
[4]Note that we have a better result : $u(\cdot, T) \in H^1(\Omega)$.

Let $\mathcal{I} = (0,T)$, $\mathcal{V}_i = L^2(\Omega_i \times (0,T))$, and $\mathbf{I}^+$ the set of intervals of $\mathbb{R}^+$. The method solves iteratively subproblems on $\Omega_1 \times \mathcal{I}$ and $\Omega_2 \times \mathcal{I}$, exchanging space-time boundary data[5] through the Robin operators $\mathcal{B}_1$ and $\mathcal{B}_2$ (defined in (2.2)), where the parameter p is chosen to optimize the convergence factor of the algorithm.

The method is thus defined using a subproblem solution operator and a transmission operator: for $i = 1, 2$:

- the solution operator $\mathcal{M}_i(\mathcal{I}, f, u_{0,i}, \xi_i)$, $i = 1, 2$, that maps the available Robin condition ξ_i, the initial condition $u_{0,i}$ and source term f to the subdomain solution u_i,

$$\mathcal{M}_i : \begin{array}{ccc} \mathbf{I}^+ \times \mathcal{V}_i \times H^1(\Omega_i) \times \mathcal{Y} & \to & H^{2,1}(\Omega_i \times \mathcal{I}) \\ (\mathcal{I}, f, u_{0,i}, \xi_i) & \to & u_i, \end{array} \tag{4.1}$$

  where u_i is the solution of the following Robin problem in $\Omega_i \times \mathcal{I}$:

$$\begin{array}{rll} \mathcal{L}u_i = & f & \text{in } \Omega_i \times \mathcal{I}, \\ u_i(\cdot, 0) = & u_{0,i} & \text{in } \Omega_i, \\ (\mathcal{B}_i u_i)(0,\cdot) = & \xi_i & \text{on } \mathcal{I}. \end{array} \qquad i = 1, 2, \tag{4.2}$$

- the transmission operator $\mathcal{B}_i$, $i = 1, 2$, that maps the available neighbor subdomain solution $u_j \in H^{2,1}(\Omega_j \times \mathcal{I})$, $j = 3 - i$, to a new Robin datum $\xi_i \in \mathcal{Y} : \xi_i = (\mathcal{B}_i u_j)(0,\cdot)$ on $\mathcal{I}$.

Using the definition of $\mathcal{M}_i$, problem (2.3)–(2.4) can be rewritten as

$$u_i = \mathcal{M}_i(\mathcal{I}, f, u_{0,i}, \xi_i), \ i = 1, 2, \tag{4.3a}$$

$$\xi_i = (\mathcal{B}_i u_j)(0,\cdot) \quad \text{on } \mathcal{I}, \quad j = 3 - i, \ i = 1, 2. \tag{4.3b}$$

The OSWR algorithm for solving problem (4.3) (or equivalently (2.1)) is as follows.

Algorithm 4.1 (OSWR)

Choose an initial Robin data $\boldsymbol{\xi}^0 = (\xi_1^0, \xi_2^0)$ on $\mathcal{I}$, for example $\xi_i^0 = (\mathcal{B}_i u_{0,i})(0,\cdot)$, on $(0,T)$, $i = 1, 2$.

for $\ell = 1, 2, \ldots$ *(OSWR iterations)* **do**

1. Solve the local space-time Robin problems by calculating

$$u_i^\ell = \mathcal{M}_i(\mathcal{I}, f, u_{0,i}, \xi_i^{\ell-1}), \qquad i = 1, 2. \tag{4.4}$$

2. Update the Robin interface term $\boldsymbol{\xi}^\ell = (\xi_1^\ell, \xi_2^\ell)$, with

$$\xi_i^\ell = (\mathcal{B}_i u_j^\ell)(0,\cdot) \quad \text{on } \mathcal{I}, \qquad j = 3 - i, \quad i = 1, 2. \tag{4.5}$$

end for

Remark 4.1. By definition of $\mathcal{M}_i$ and u_i^ℓ in (4.4), the interface condition is

$$(\mathcal{B}_i u_i^\ell)(0,\cdot) = \xi_i^{\ell-1}, \ i = 1, 2. \tag{4.6}$$

Then, from (4.5), and using that $\mathcal{B}_i = -\mathcal{B}_j + p$, we obtain, for $\ell \geq 1$,

$$\xi_i^\ell = -\xi_j^{\ell-1} + 2p u_j^\ell, \quad j = 3 - i, \ i = 1, 2. \tag{4.7}$$

[5] In the 1D case considered here, the interface is reduced to one point in space, thus the exchanged data depend on time only.

Let $L \in \mathbb{N}^*$. In what follows, we will denote in compact form

$$(u^L, \boldsymbol{\xi}^L) = \mathrm{OSWR}_L(\mathcal{I}, u_0, \boldsymbol{\xi}^0), \tag{4.8}$$

where $u^L \in L^2(\Omega \times \mathcal{I})$ with $u^L|_{\Omega_i} = u_i^L,\ i = 1, 2$, and $\boldsymbol{\xi}^L = (\xi_1^L, \xi_2^L)$ are the output after L iterations of algorithm (4.4)–(4.5) with initial condition u_0 and initial Robin datum $\boldsymbol{\xi}^0$ on $\mathcal{I}$ (the dependency with f is omitted in (4.8), to simplify the notation).

4.1. Stability and convergence. We suppose that $f \in L^2(0, T; L^2(\Omega))$. For simplicity, we will use the notation $\|\cdot\|$ for the L^2-norm in Ω or in $\Omega_i,\ i = 1, 2$, and $\|\cdot\|_{\mathcal{I}}$ for the $(L^2(\mathcal{I}))^2$-norm.
Let $(u_i^\ell, \xi_i^\ell),\ i = 1, 2$ be defined by (4.4)–(4.5). For the convergence analysis below, we introduce the following notation for the errors, for $i = 1, 2$ and $\ell \geq 1$:

$$\zeta_i^\ell := \xi_i^\ell - \xi_i, \quad \text{where } \xi_i \text{ is defined in (2.4)}, \quad \text{and we set } \boldsymbol{\zeta}^\ell := (\zeta_1^\ell, \zeta_2^\ell), \tag{4.9}$$

$$e_i^\ell := u_i^\ell - u, \quad \text{where } u \text{ is the solution of (2.1)}, \tag{4.10}$$

$$e^\ell := \text{ the function in } L^2(\Omega \times (0, T)) \text{ s.t. } e^\ell|_{\Omega_i} = e_i^\ell,\ i = 1, 2. \tag{4.11}$$

If $u_0 \in H^1(\Omega)$, a convergence analysis is done in [12], we recall it in Theorem 4.2-(i). However, in the context of the coupled Parareal-OSWR method in Section 5, incomplete iterations of the OSWR algorithm are performed at each Parareal iteration. This implies that the new initial condition for the OSWR algorithm, through Parareal iterations, will no longer be in $H^1(\Omega)$, but only in $\mathcal{X}$. Therefore we need the following extended result of Theorem 4.2-(ii) below, that will be used to prove the convergence of the coupled Parareal-OSWR method later. In that case, we suppose that the initial condition of Algorithm 4.1, denoted now by $\bar{u}_0$, verifies $\bar{u}_0 \in \mathcal{X}$, and we introduce the additional notation for the error at time $t = 0$:

$$e_0 := \bar{u}_0 - u(\cdot, 0), \qquad \text{and} \quad e_{0,i} := e_0|_{\Omega_i},\ i = 1, 2.$$

THEOREM 4.2. *Let $L \in \mathbb{N}^*$.*
(i) If $u_0 \in H^1(\Omega)$ and $\boldsymbol{\xi}^0 \in \mathcal{Y}^2$, then, Algorithm 4.1 is well defined and we have

$$\sum_{\ell=1}^{L} \left(\tfrac{1}{2}\|e^\ell(., T)\|^2 + \sum_i \|e_i^\ell\|^2_{L^2(0,T,H^1(\Omega_i))} \right) + \tfrac{1}{2p}\|\boldsymbol{\zeta}^L\|^2_{\mathcal{I}} = \tfrac{1}{2p}\|\boldsymbol{\zeta}^0\|^2_{\mathcal{I}}. \tag{4.12}$$

Hence, Algorithm 4.1 converges for $p > 0$ in $L^\infty(0, T; L^2(\Omega_1)) \cap L^2(0, T; H^1(\Omega_1)) \times L^\infty(0, T; L^2(\Omega_2)) \cap L^2(0, T; H^1(\Omega_2))$ to the solution u of (2.1).

(ii) If the initial condition of Algorithm 4.1 is $\bar{u}_0 \in \mathcal{X}$, and if $\boldsymbol{\xi}^0 \in \mathcal{Y}^2$, then, Algorithm 4.1 is well defined and we have

$$\sum_{\ell=1}^{L} \left(\tfrac{1}{2}\|e^\ell(., T)\|^2 + \sum_i \|e_i^\ell\|^2_{L^2(0,T,H^1(\Omega_i))} \right) + \tfrac{1}{2p}\|\boldsymbol{\zeta}^L\|^2_{\mathcal{I}} = \tfrac{L}{2}\|e_0\|^2 + \tfrac{1}{2p}\|\boldsymbol{\zeta}^0\|^2_{\mathcal{I}}. \tag{4.13}$$

Proof. The proof of (i) is done in [12, Theorem 5.15]. Note that the energy estimate (4.12) is a special case of (4.13) with $e_0 = 0$.

Proof of (ii). We proceed by induction. We have $u_{0,i} \in \mathcal{X}_i$ independently of ℓ. Moreover, let us suppose that $\xi_i^{\ell-1} \in \mathcal{Y}$ (this is true for $\ell = 1$). Then from Lemma 2.2 we have $u_i^\ell \in H^{2,1}(\Omega_i \times (0, T))$. Then from Lemma 2.3, or using (4.7) and the trace theorem, we have $\xi_i^\ell \in \mathcal{Y}$, thus Algorithm 4.1 is well defined.

The proof of the energy estimate (4.13) follows the same steps as in [12, Theorem 5.15] and is detailed in Section 9.

4.2. Optimized Robin parameters. In this section we give the methodology to calculate the Robin parameter p involved in the OSWR method. This parameter is chosen to optimize the convergence factor of the algorithm.
Recall that one challenge here is that in the context of the coupled Parareal-OSWR method introduced later, the initial condition of Algorithm 4.1, through Parareal iterations, is in $\mathcal{X}$. Thus we have to extend the calculation of the convergence factor of [11], Lemma 5.7 (that was done for $\zeta_i^0 \in {}_0H^{\frac{3}{4}}(0,T)$). This can be done by extending ζ_i^0 to a function $\tilde{\zeta}_i^0$ in $H^{\frac{1}{4}}(\mathbb{R})$, from which we obtain $\tilde{\zeta}_i^\ell \in H^{\frac{1}{4}}(\mathbb{R})$, $\ell \geq 1$ (by induction). Then the convergence factor is obtained by using Fourier transform in time.

By linearity of $\mathcal{M}_i$, from (4.3a) and (4.4), the error $e_i^\ell := u_i^\ell - u$, $i = 1,2$, at iteration ℓ of the OSWR method, satisfies $e_i^\ell = \mathcal{M}_i(\mathcal{I}, 0, 0, \zeta_i^{\ell-1})$, with $\zeta_i^{\ell-1} = \xi_i^{\ell-1} - \xi_i$. Equivalently, e_i^ℓ is solution of the following problem

$$\begin{array}{rcll} \mathcal{L}e_i^\ell = & 0 & \text{in } \Omega_i \times \mathcal{I}, & \\ e_i^\ell(\cdot,0) = & 0 & \text{in } \Omega_i, & i = 1,2, \\ (\mathcal{B}_i e_i^\ell)(0,\cdot) = & \zeta_i^{\ell-1} & \text{on } \mathcal{I}. & \end{array} \tag{4.14}$$

From (4.3b) and (4.5), we have $\zeta_i^\ell = \mathcal{B}_i e_j^\ell(0,\cdot)$, $i = 1,2$, and thus the transmission conditions on $\mathcal{I}$ in (4.14) also reads

$$(\mathcal{B}_i e_i^\ell)(0,\cdot) = (\mathcal{B}_i e_j^{\ell-1})(0,\cdot) \quad \text{on } \mathcal{I}, \quad j = 3-i, \quad i = 1,2. \tag{4.15}$$

In order to use the Fourier transform in time, we have to extend (4.14)–(4.15) to $\mathbb{R}$. We proceed by induction on ℓ :

• For $\ell = 1$, problem (4.14) has an initial Robin datum $\zeta_i^0 \in \mathcal{Y}$ that can be extended to $H^{\frac{1}{4}}(\mathbb{R})$[6] to obtain a function, denoted by $\tilde{\zeta}_i^0$, vanishing on $(-\infty, 0)$, and on $(T, +\infty)$, for $i = 1,2$. Then we can extend equations (4.14) to $\mathbb{R}$, and their solutions, denoted by $(\tilde{e}_1^1, \tilde{e}_2^1)$, vanish on $(-\infty, 0)$ and coincide with (e_1^1, e_2^1) on $(0,T)$.

• For $\ell \geq 1$, if $\ell = 1$ then $\tilde{\zeta}_i^{\ell-1}$ is defined above; else we define the Robin trace $\tilde{\zeta}_i^{\ell-1} := (\mathcal{B}_i \tilde{e}_j^{\ell-1})(0,\cdot)$ on $\{0\} \times \mathbb{R}$, that vanishes on $(-\infty, 0)$ and coincides with $(\mathcal{B}_i e_j^{\ell-1})(0,\cdot)$ on $\{0\} \times (0,T)$, for $i = 1,2$. We assume that $\tilde{\zeta}_i^{\ell-1} \in H^{\frac{1}{4}}(\mathbb{R})$, $i = 1,2$. Let us now prove that $\tilde{\zeta}_i^\ell$ belongs to $H^{\frac{1}{4}}(\mathbb{R})$ for $i = 1,2$.

The subdomain problems (4.14) are extended on $\Omega_i \times \mathbb{R}$ as follows:

$$\begin{array}{rcll} \mathcal{L}\tilde{e}_i^\ell = & 0 & \text{in } \Omega_i \times \mathbb{R}, & \\ \tilde{e}_i^\ell(\cdot,0) = & 0 & \text{in } \Omega_i, & i = 1,2, \\ (\mathcal{B}_i \tilde{e}_i^\ell)(0,\cdot) = & \tilde{\zeta}_i^{\ell-1} & \text{on } \mathbb{R}, & \end{array} \tag{4.16}$$

and their solution vanish on $(-\infty, 0)$ and coincide with (e_1^ℓ, e_2^ℓ) on $(0,T)$. In particular, for $\ell \geq 2$, by the definition of $\tilde{\zeta}_i^{\ell-1}$ above, (4.15) is extended on $\{0\} \times \mathbb{R}$ by

$$(\mathcal{B}_i \tilde{e}_i^\ell)(0,\cdot) = (\mathcal{B}_i \tilde{e}_j^{\ell-1})(0,\cdot) \quad \text{on } \mathbb{R}, \quad j = 3-i, \quad i = 1,2. \tag{4.17}$$

Moreover, by energy estimates, we have $\tilde{e}_i^\ell \in L^2(\mathbb{R}; H^1(\Omega_i))$ and, from the strong form of the equation satisfied by $\partial_t \tilde{e}_i^\ell$, we deduce that $\partial_t \tilde{e}_i^\ell \in L^2(\mathbb{R}; H^{-1}(\Omega_i))$, $i = 1,2$. Thus, in what follows, we can use Fourier transform in time. Then, we solve in each subdomain the ordinary differential equation[7]

$$i\omega \hat{e}_i^\ell - \nu \partial_{xx} \hat{e}_i^\ell + a \partial_x \hat{e}_i^\ell + b \hat{e}_i^\ell = 0, \qquad i = 1,2, \tag{4.18}$$

[6] See [29] Theorem 11.4, with $s = \frac{1}{4}$.

[7] Note that (4.18) is in the sense of distributions. Then (4.20) implies that (4.18) also holds in the classical sense.

with the characteristic roots

$$r^- = \frac{a-\sqrt{d}}{2\nu}, \qquad r^+ = \frac{a+\sqrt{d}}{2\nu}, \qquad d = a^2 + 4\nu(b+i\omega), \tag{4.19}$$

where $\sqrt{d}$ is the complex square-root with positive real part :

$$\sqrt{d} = \sqrt{\frac{\tilde{d}+a^2+4\nu b}{2}} + i\,\mathrm{sign}(\omega)\sqrt{\frac{\tilde{d}-a^2-4\nu b}{2}},$$

where $\tilde{d} := \sqrt{(a^2+4\nu b)^2+16\nu^2\omega^2}$. Thus, $\mathcal{R}e(r^+) > 0$ and $\mathcal{R}e(r^-) < 0$, and the solutions $\hat{e}_i^\ell \in L^2(\Omega_i),\ i = 1, 2$, are given by[8]

$$\hat{e}_1^\ell = \frac{2}{\sqrt{d}+p}\hat{\zeta}_1^{\ell-1}(\omega)e^{r^+x}, \qquad \hat{e}_2^\ell = \frac{2}{\sqrt{d}+p}\hat{\zeta}_2^{\ell-1}(\omega)e^{r^-x}, \qquad \ell \geq 1. \tag{4.20}$$

Then, replacing (4.20) in the transmission conditions (4.17) leads to

$$\forall \ell \geq 1, \quad \begin{pmatrix}\hat{\zeta}_1^\ell \\ \hat{\zeta}_2^\ell\end{pmatrix} = \begin{pmatrix}\dfrac{-\sqrt{d}+p}{\sqrt{d}+p}\end{pmatrix}\begin{pmatrix}\hat{\zeta}_2^{\ell-1} \\ \hat{\zeta}_1^{\ell-1}\end{pmatrix}. \tag{4.21}$$

Thus, from (4.21) we have, for all $\omega \in \mathbb{R}$, and for $\ell \geq 1$:

$$|\hat{\zeta}_i^\ell(\omega)| \leq |\hat{\zeta}_{3-i}^{\ell-1}(\omega)|, \qquad \text{for } i = 1, 2.$$

Using the induction hypothesis (i.e. $\tilde{\zeta}_i^{\ell-1} \in H^{\frac{1}{4}}(\mathbb{R})$, for $i = 1, 2$), the above inequality implies that $\tilde{\zeta}_i^\ell \in H^{\frac{1}{4}}(\mathbb{R})$, for $i = 1, 2$. Consequently, we get $\tilde{\boldsymbol{\zeta}}^\ell \in (H^{\frac{1}{4}}(\mathbb{R}))^2,\ \forall \ell \geq 1$.

Setting $\hat{\boldsymbol{\zeta}}^\ell := (\hat{\zeta}_1^\ell, \hat{\zeta}_2^\ell)$, from (4.21), we have

$$\forall \ell \geq 2, \quad \hat{\boldsymbol{\zeta}}^\ell = \rho_0(\omega,p)\,\hat{\boldsymbol{\zeta}}^{\ell-2}, \qquad \text{with} \quad \rho_0(\omega,p) := \left(\frac{-\sqrt{d}+p}{\sqrt{d}+p}\right)^2. \tag{4.22}$$

From (4.22), by induction on ℓ we obtain,

$$\hat{\boldsymbol{\zeta}}^{2\ell} = (\rho_0(\omega,p))^\ell\,\hat{\boldsymbol{\zeta}}^{(0)}, \qquad \forall \ell \geq 1. \tag{4.23}$$

From (4.23), the convergence factor of the algorithm is $\rho_0(\omega, p)$, defined in (4.22).

While we have $\max_{\omega\in\mathbb{R}} |\rho_0(\omega,p)| = \lim_{\omega\to\infty} |\rho_0(\omega,p)| = 1$, we can use the continuous convergence factor $\rho_0(\omega, p)$ to calculate an efficient Robin parameter for the discrete setting (see, e.g., [26, 27, 35, 10, 12]).

Indeed, in numerical computations, the frequency ω is bounded, i.e., we have $\omega_{\min} \leq \omega \leq \omega_{\max}$, where $\omega_{\max} = \frac{\pi}{\Delta t}$ is the largest discrete frequency supported by the numerical time grid, and $\omega_{\min} = \frac{\pi}{T}$ is smallest frequency relevant to the global time interval. Defining

$$\rho_c(p) := \max_{\frac{\pi}{T}\leq\omega\leq\frac{\pi}{\Delta t}} |\rho_0(p,\omega)|,$$

then the optimized Robin parameter p_c is chosen such that it verifies

$$\rho_c(p_c) = \min_{p>0} \rho_c(p). \tag{4.24}$$

In practice, the minimization problem (4.24) is solved using the `fminsearch` function in MATLAB.

[8]Note that here the term "∂_x" in $\mathcal{B}_i$ is multiplied by ν while this is not the case in [12]. Thus $\hat{e}_i^\ell$ is slightly different here from the one of [12].

5. Coupled Parareal-OSWR method. We set $(0,T) = \cup_{n=0}^{N-1}\mathcal{I}_n$ as in Section 3. Then the Parareal-OSWR algorithm is defined using the coarse propagator $\mathcal{G}$ of Section 3 and the incomplete[9] fine propagator OSWR_L defined by (4.8) as follows.

Algorithm 5.1 (Coupled Parareal-OSWR)

1. Choose an initial data $(U_n^0)_{n\in[\![0,N]\!]}$ with $U_0^0 = u_0$ and U_n^0 an approximation of $u(\cdot,T_n)$, for example $U_n^0 := \mathcal{G}(\mathcal{I}_{n-1}, U_{n-1}^0)$, for $n = 1,2,\ldots,N$.
2. Choose an initial Robin data $(\boldsymbol{\xi}_n^{0,0})_{n\in[\![0,N-1]\!]}$, with $\boldsymbol{\xi}_n^{0,0} := (\xi_{1,n}^{0,0}, \xi_{2,n}^{0,0})$ on $\mathcal{I}_n$, for example $\xi_{i,n}^{0,0} = (\mathcal{B}_i U_n^0)(0,\cdot),\ i = 1,2$.

for k=0,1,... *(Parareal iterations)* **do**

1. On each time interval $\mathcal{I}_n$, $n = 0,1,\ldots,N-1$:

$$\text{Calculate} \quad (u_n^{k,L}, \boldsymbol{\xi}_n^{k,L}) = \mathrm{OSWR}_L(\mathcal{I}_n, U_n^k, \boldsymbol{\xi}_n^{k,0}). \tag{5.1}$$

2. Set $U_0^{k+1} = u_0$ and do Parareal corrections:

$$U_{n+1}^{k+1} = u_n^{k,L}(\cdot,T_{n+1}) + \mathcal{G}(\mathcal{I}_n, U_n^{k+1}) - \mathcal{G}(\mathcal{I}_n, U_n^k). \tag{5.2}$$

$$\text{Update the interface term:} \quad \boldsymbol{\xi}_n^{k+1,0} = \boldsymbol{\xi}_n^{k,L}. \tag{5.3}$$

end for

Remark 5.1. In (5.1), if $L = \infty$, then $u_n^{k,L} \in H^1(\Omega)$, for all $n \geq 0,\ k \geq 0$. However, if $L < \infty$ and chosen small (e.g. $L = 2$), then $u_n^{k,L} \in \mathcal{X}$, for all $n \geq 0,\ k \geq 0$, but there is very few chance that $u_n^{k,L}$ be in $H^1(\Omega)$. Thus, using (5.2) and that $u_0 \in H^1(\Omega) \subset \mathcal{X}$, we will have $U_n^k \in \mathcal{X}$, for all $n \geq 0,\ k \geq 0$.

6. Convergence of the Parareal-OSWR algorithm. We will consider the convergence in the $L^2(\Omega)$-norm. As in Section 4, $\|\cdot\|$ will stand for the L^2-norm in Ω or in $\Omega_i,\ i = 1,2$, and $\|\cdot\|_{\mathcal{I}}$ the $(L^2(\mathcal{I}))^2$-norm. Let $\widetilde{\mathcal{G}}$ be the coarse propagator associated to the source term $f = 0$. We have the following result

THEOREM 6.1. *We suppose that there exists a constant γ_1 such that*

$$\|\widetilde{\mathcal{G}}(\mathcal{I}_n, U)\| \leq \gamma_1 \|U\|, \quad \forall U \in L^2(\Omega), \quad \forall n = 0,1,\ldots,N. \tag{6.1}$$

Then, when $k \to \infty$, U_n^k converges to $u(.,T_n)$ in $L^2(\Omega)$. Moreover, when $k \to \infty$, $u_{i,n}^{k,\ell}$ converges to $u|_{\Omega_i\times[T_n,T_{n+1}]}$ in $L^2(T_n,T_{n+1},H^1(\Omega))$ for all $\ell = 1,2,\ldots,L$.

In order to prove Theorem 6.1, we introduce the following notation.

Notation for error estimation. Let u be the solution of problem (2.1), and let $(u_n^{k,\ell}, \boldsymbol{\xi}_n^{k,\ell})_{1\leq\ell\leq L}$ be the sequence of iterates through the OSWR step (5.1). We define for $k = 0,1,\ldots$, and $n = 0,\ldots,N-1$, the following:

- $E_n^k := U_n^k - u_n,\ n = 0,\ldots,N$, where $u_n = u(.,T_n)$,
- $e_n^{k,\ell} := u_n^{k,\ell} - u$, the error in $L^2(\Omega\times\mathcal{I}_n)$ at each iteration ℓ inside step (5.1), with $e_{i,n}^{k,\ell} := e_n^{k,\ell}|_{\Omega_i},\ i = 1,2$, for $\ell = 1,\ldots,L$,
- $\boldsymbol{\zeta}_n^{k,\ell} := \boldsymbol{\xi}_n^{k,\ell} - \boldsymbol{\xi}_n$, for $\ell = 1,\ldots,L$, where $\boldsymbol{\xi}_n = ((\mathcal{B}_1 u)(0,\mathcal{I}_n), (\mathcal{B}_2 u)(0,\mathcal{I}_n))$.

Then, by linearity, the algorithm on the error reads as follows.

[9] In the sense that L will be smaller than the number of iterations required for convergence.

Algorithm 6.1 (Coupled Parareal-OSWR algorithm on the error)

1. Define initial data $(E_n^0)_{n\in[\![0,N]\!]}$ with $E_0^0 = 0$, $E_n^0 := \mathcal{G}(\mathcal{I}_{n-1}, U_{n-1}^0) - u_n$, where $U_0^0 = u_0$, $n = 1, 2, \ldots, N$.
2. Define initial Robin data $(\boldsymbol{\zeta}_n^{0,0})_{n\in[\![0,N]\!]}$, with $\boldsymbol{\zeta}_n^{0,0} := (\zeta_{1,n}^{0,0}, \zeta_{2,n}^{0,0})$ on $\mathcal{I}_n$, where $\zeta_{i,n}^{0,0} = (\mathcal{B}_i E_n^0)(0,\cdot)$, $i = 1, 2$.

for k=0,1,... *(Parareal iterations)* **do**

1. On each time interval $\mathcal{I}_n$, $n = 0, 1, \ldots, N-1$:

$$\text{Calculate} \quad (e_n^{k,L}, \boldsymbol{\zeta}_n^{k,L}) = \mathrm{OSWR}_L(\mathcal{I}_n, E_n^k, \boldsymbol{\zeta}_n^{k,0}). \tag{6.2}$$

2. Set $E_0^{k+1} = 0$ and do Parareal correction:

$$E_{n+1}^{k+1} = e_n^{k,L}(\cdot, T_{n+1}) + \widetilde{\mathcal{G}}(\mathcal{I}_n, E_n^{k+1} - E_n^k). \tag{6.3}$$

$$\text{Update the interface term:} \quad \boldsymbol{\zeta}_n^{k+1,0} = \boldsymbol{\zeta}_n^{k,L}. \tag{6.4}$$

end for

The proof of Theorem 2.1 relies on the OSWR estimate (4.13) with $e_0 = E_n^k$ (from incomplete iterations) and $\boldsymbol{\zeta}^\ell = \boldsymbol{\zeta}_n^{k,\ell}$. Then summing with respect to k, the terms with $\boldsymbol{\zeta}_n^{k,\ell}$ are treated using the update $\boldsymbol{\zeta}_n^{k+1,0} = \boldsymbol{\zeta}_n^{k,L}$ (see Lemma 6.2), and the terms with E_n^k are treated using hypothesis (6.1) and proving the new estimate of Lemma 6.3.

LEMMA 6.2. *For all $n \in [\![0, N-1]\!]$, we have*

$$\sum_{k=0}^{K} \|e_n^{k,L}(., T_{n+1})\|^2 \le L \sum_{k=0}^{K} \|E_n^k\|^2 + \tfrac{1}{p}\|\boldsymbol{\zeta}_n^{0,0}\|_{\mathcal{I}_n}^2. \tag{6.5}$$

Proof. Step (6.2) is the OSWR method on $\mathcal{I} := \mathcal{I}_n$, with initial condition $E_n^k \in \mathcal{X}$ (see Remark 5.1). Thus, from (4.13) with $e_0 := E_n^k$, $e_i^\ell := e_{i,n}^{k,\ell}$, $\boldsymbol{\zeta}^\ell := \boldsymbol{\zeta}_n^{k,\ell}$, we obtain

$$\sum_{\ell=1}^{L} \left(\tfrac{1}{2}\|e_n^{k,\ell}(., T_{n+1})\|^2 + \sum_i \|e_{i,n}^{k,\ell}\|^2_{L^2(T_n, T_{n+1}; H^1(\Omega_i))} \right) + \tfrac{1}{2p}\|\boldsymbol{\zeta}_n^{k,L}\|_{\mathcal{I}_n}^2 = \tfrac{L}{2}\|E_n^k\|^2 + \tfrac{1}{2p}\|\boldsymbol{\zeta}_n^{k,0}\|_{\mathcal{I}_n}^2.$$

Moreover, in our algorithm, from (6.4) we have $\boldsymbol{\zeta}_n^{k+1,0} = \boldsymbol{\zeta}_n^{k,L}$, thus

$$\sum_{\ell=1}^{L} \left(\tfrac{1}{2}\|e_n^{k,\ell}(., T_{n+1})\|^2 + \sum_i \|e_{i,n}^{k,\ell}\|^2_{L^2(T_n, T_{n+1}; H^1(\Omega_i))} \right) + \tfrac{1}{2p}\|\boldsymbol{\zeta}_n^{k+1,0}\|_{\mathcal{I}_n}^2 = \tfrac{L}{2}\|E_n^k\|^2 + \tfrac{1}{2p}\|\boldsymbol{\zeta}_n^{k,0}\|_{\mathcal{I}_n}^2.$$

Summing with respect to k, from 0 to K, we get a telescopic sum on the interface and therefore

$$\sum_{k=0}^{K}\sum_{\ell=1}^{L} \left(\tfrac{1}{2}\|e_n^{k,\ell}(., T_{n+1})\|^2 + \sum_i \|e_{i,n}^{k,\ell}\|^2_{L^2(T_n, T_{n+1}; H^1(\Omega_i))} \right) + \tfrac{1}{2p}\|\boldsymbol{\zeta}_n^{K+1,0}\|_{\mathcal{I}_n}^2 = \tfrac{L}{2}\sum_{k=0}^{K}\|E_n^k\|^2 + \tfrac{1}{2p}\|\boldsymbol{\zeta}_n^{0,0}\|_{\mathcal{I}_n}^2, \tag{6.6}$$

from which we obtain (6.5). ∎

LEMMA 6.3. *We suppose that $\widetilde{\mathcal{G}}$ satisfies (6.1). Then, for all $n \in [\![0, N-1]\!]$,*

$$\sum_{k=0}^{K} \|E_{n+1}^{k+1}\|^2 \leq 8\gamma_1^2 \sum_{k=0}^{K+1} \|E_n^k\|^2 + 2\sum_{k=0}^{K} \|e_n^{k,L}(.,T_{n+1})\|^2. \tag{6.7}$$

Proof. Using the triangle inequality in (6.3), and then (6.1), we get

$$\begin{aligned}
\|E_{n+1}^{k+1}\|^2 &\leq 2\|\widetilde{\mathcal{G}}(\mathcal{I}_n, E_n^{k+1} - E_n^k)\|^2 + 2\|e_n^{k,L}(.,T_{n+1})\|^2, \\
&\leq 2\gamma_1^2\|E_n^{k+1} - E_n^k\|^2 + 2\|e_n^{k,L}(.,T_{n+1})\|^2, \\
&\leq 4\gamma_1^2\left(\|E_n^{k+1}\|^2 + \|E_n^k\|^2\right) + 2\|e_n^{k,L}(.,T_{n+1})\|^2.
\end{aligned}$$

Then, summing with respect to k, from 0 to K, we have

$$\sum_{k=0}^{K} \|E_{n+1}^{k+1}\|^2 \leq 4\gamma_1^2 \sum_{k=0}^{K} \left(\|E_n^{k+1}\|^2 + \|E_n^k\|^2\right) + 2\sum_{k=0}^{K} \|e_n^{k,L}(.,T_{n+1})\|^2,$$

from which we deduce (6.7). ∎

With these lemmas, we can now prove Theorem 6.1

Proof of Theorem 6.1. From Lemmas 6.2 and 6.3, we get

$$\sum_{k=0}^{K} \|E_{n+1}^{k+1}\|^2 \leq 8\gamma_1^2 \sum_{k=0}^{K+1} \|E_n^k\|^2 + 2L\sum_{k=0}^{K} \|E_n^k\|^2 + \tfrac{2}{p}\|\boldsymbol{\zeta}_n^{0,0}\|_{\mathcal{I}_n}^2.$$

Setting $\gamma_2 = 8\gamma_1^2 + 2L$ and $R_n = \|E_{n+1}^0\|^2 + \frac{2}{p}\|\boldsymbol{\zeta}_n^{0,0}\|_{\mathcal{I}_n}^2$, we can rewrite this inequality as

$$\sum_{k=0}^{K+1} \|E_{n+1}^k\|^2 \leq \gamma_2 \sum_{k=0}^{K+1} \|E_n^k\|^2 + R_n, \quad n \in [\![0, N-1]\!].$$

From this inequality, by induction, we obtain

$$\sum_{k=0}^{K+1} \|E_{n+1}^k\|^2 \leq \gamma_2^{n+1} \sum_{k=0}^{K+1} \|E_0^k\|^2 + \sum_{j=0}^{n} \gamma_2^j R_{n-j}, \quad n \in [\![0, N-1]\!].$$

Using that $E_0^k = 0,\ \forall k \geq 0$, we finally obtain

$$\sum_{k=0}^{K+1} \|E_{n+1}^k\|^2 \leq \sum_{j=0}^{n} \gamma_2^j R_{n-j}, \quad n \in [\![0, N-1]\!]. \tag{6.8}$$

Since the right-hand side of (6.8) does not depend on K, this shows that, for a given $n \in [\![0, N]\!]$, the sum $\sum_{k=0}^{K} \|E_n^k\|^2$ is bounded with respect to K. Hence E_n^k converges to 0 in the $L^2(\Omega)$-norm when $k \to \infty$. Moreover, from inequality (6.6), we obtain, for all $\ell = 1, 2, \ldots, L$

$$\sum_{k=0}^{K} \|e_{i,n}^{k,\ell}\|_{L^2(T_n, T_{n+1}; H^1(\Omega_i))} \leq \frac{L}{2} \sum_{k=0}^{K} \|E_n^k\|^2 + \tfrac{1}{2p}\|\boldsymbol{\zeta}_n^{0,0}\|_{\mathcal{I}_n}^2.$$

As the sum in the right-hand side of the above inequality is bounded with respect to K, then the sum in the left is also bounded with respect to K. Hence, $e_{i,n}^{k,\ell}$ tends to 0 in $L^2(T_n, T_{n+1}; H^1(\Omega_i))$, i.e, $u_{i,n}^{k,\ell} \to u_{i,n}$ in $L^2(T_n, T_{n+1}; H^1(\Omega_i))$, for any ℓ. ∎

Remark 6.4. Note that Assumption 6.1 is very weak, as we do not require the Lipschitz constant to be strictly lower than 1.[10]

Remark 6.5. The proof of convergence of the nonoverlapping Parareal-OSWR algorithm in Theorem 6.1 (using energy estimates) is done with $U_n^k \in \mathcal{X}$ for all $n \geq 0$, $k \geq 0$ (see Remark (5.1)). This result is obtained without any correction on U_n^k so that it is more regular (i.e., in $H^1(\Omega)$). Thus, in practice we do not need a correction on U_n^k to have a convergent algorithm. Note that in the case of overlapping subdomains, a correction step will be needed; see [7].

Remark 6.6. In Algorithm 5.1, a possible choice for the initial Robin datum on $\mathcal{I}_n$ is to take $\xi_{i,n}^{0,0} = (\mathcal{B}_i U_n^0)(0),\ i = 1, 2$ (i.e., $\xi_{i,n}^{0,0}$ is constant on $\mathcal{I}_n$). A better choice, that improves the convergence of the Parareal-OSWR method, is to define V_n^0 as a linear interpolation between U_n^0 and U_{n+1}^0, and then to take $\xi_{i,n}^{0,0} = (\mathcal{B}_i V_n^0)(0),\ i = 1, 2$. Thus, in what follows the latter case will be considered.

7. Numerical results in 1D. In this section, we consider problem (2.1) with $\Omega =]0, 1[$ and Dirichlet boundary conditions at $x = 0$ and $x = 1$. A regular advection-diffusion example is shown, and then an advection-dominated one (where the Parareal method doesn't perform well). A cell-centered finite volume method is used, with a centered discretization for the advection term (for regular advection-diffusion) or an upwind discretization as in [6] (in the advection dominated case). For the time discretization, the backward Euler method is used, with time step ΔT for the coarse solver.

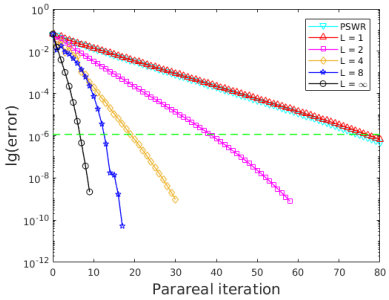
For the fine solver with $L = \infty$, as well as for the coarse solver, we use the OSWR method, with a stopping criterion such that the jump of the optimized transmission conditions, measured in the L^2-norm on the interface, has been reduced below 10^{-13}. Otherwise, we will consider L iterations for the fine solver, with different values of L. A comparison will be done with the PSWR method of [16], which corresponds to the case $L = 1$ coupled with an incomplete coarse solver based on only one iteration of the OSWR method.

We take $T = 1$, $a = 1$, and $b = 0$. The domain Ω is decomposed into two subdomains $\Omega_1 =]0, 1/2[$ and $\Omega_2 =]1/2, 1[$ and we choose $N = 10$ time subintervals. The error is measured in the $L^\infty(0, T; \mathcal{X})$-norm. Note that we obtain similar results in the $L^\infty(\Omega \times (0, T))$-norm. The cost of the coarse solver is negligible.

7.1. Regular Advection-Diffusion. We set $\nu = 1$ and choose the right-hand side f and the values of the boundary and initial conditions so that the exact solution is given by $u(x, t) = \exp(-t)\sin(\pi x)$. The mesh size and time steps (for the fine solver) are $\Delta x = 1, 25.10^{-3}, \Delta t = 1, 69.10^{-4}$ respectively.

In Figure 1, we plot the evolution of the relative error between the Parareal-OSWR (or PSWR) solution and the converged Parareal solution, as a function of the number of Parareal iterations.
The case $L = \infty$ (black circle curve) corresponds to $L = 48$. The horizontal dashed green line represents the relative scheme error divided by 10, and we consider the number of iterations such that the algorithm error is smaller than this value. This is obtained after 7 iterations for Parareal, and after 13, 19, 39, 77 iterations for Parareal-OSWR with $L = 8$, $L = 4$, $L = 2$, $L = 1$, respectively. These data are reported in Table 1.

[10] This assumption is for example satisfied for the implicit Euler scheme, and is deduced from the regularity of the elliptic problem: $\frac{U}{\Delta T} - \nu \partial_{xx} U + a \partial_x U + bU = \frac{U_0}{\Delta T}$.



L	PSWR	1	2	4	8	48 (∞)
k	76	77	39	19	13	7
$L * k$	77	77	78	76	104	336

TABLE 1
Number of Parareal iterations k and total number of OSWR iterations $L * k$, versus L

FIG. 1. *Relative error versus Parareal iterations*

We observe that the fastest case is $L = 4$ with 76 OSWR iterations globally, and it performs similarly as $L = 1$ or PSWR.

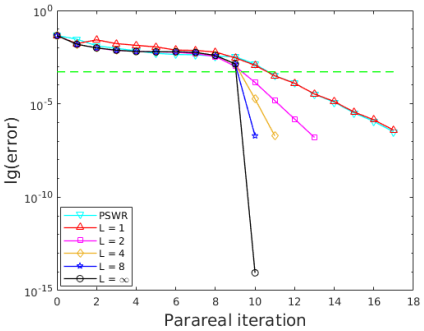
Table 2 shows the gain (in term of fine solver iterations), when the fine solvers are performed in parallel, of Parareal, coupled Parareal-OSWR, or PSWR methods compared to the pure OSWR algorithm (first column). We see that for $N = 10$ time windows, the gain factor is 1.43 for Parareal and 6.33 for Parareal-OSWR or PSWR.

Solver	OSWR	Parareal	Parareal-OSWR ($L = 4$)	PSWR
Iterations (n)	48	336	76	76
Loss factor ($\ell = n/48$)	–	7	1.58	1.58
Final gain factor (N/ℓ)	–	1.43	6.33	6.33

TABLE 2
Example in 1D (regular advection-diffusion): Gain factor for Parareal, coupled Parareal-OSWR, and PSWR methods compared to the OSWR solver, with $N = 10$ time windows

7.2. Advection dominated case. We set $\nu = 0.001$, $u_0(x) = x, \forall x \in [0, 1]$, $f = 0$, and $u(0, t) = 0$, $u(1, t) = 1, \forall t \in (0, T)$. The mesh size and time steps (for the fine solver) are $\Delta t = \Delta x = 10^{-3}$.

In Figure 2, we plot the evolution of the relative error between the Parareal-OSWR (or PSWR) solution and the converged Parareal solution, as a function of the number of Parareal iterations.



L	PSWR	1	2	4	8 ($\approx \infty$)
k	11	11	10	10	10
$L * k$	11	11	20	40	80

TABLE 3
Number of Parareal iterations k and total number of OSWR iterations $L * k$, versus L

FIG. 2. *Relative error versus Parareal iterations*

The classical Parareal algorithm, i.e. the case $L = \infty$ (black circle curve) does not work at all, since it fully needs ten iterations to converge. In consequence, the final gain factor of the best choice, shown in Table 4, decreases compared to the regular advection-diffusion case, so do all other choices. Figure 2 and Table 3 show that PSWR and Parareal-OSWR with $L = 1$ are the best choice, and are a good alternative method to classical Parareal, when the latter doesn't work. Indeed, we gain a factor 5.45 for Parareal-OSWR or PSWR.

Solver	OSWR	Parareal	Parareal-OSWR ($L = 1$)	PSWR
Iterations (n)	6	80	11	11
Loss factor ($\ell = n/6$)	–	13.3	1.83	1.83
Final gain factor (N/ℓ)	–	0.75	5.45	5.45

TABLE 4

Example in 1D (advection dominated): Gain factor for Parareal, coupled Parareal-OSWR, and PSWR methods compared to the OSWR solver, with $N = 10$ time windows

8. Numerical results in 2D. In this section, we give some numerical illustrations of the performances of the coupled Parareal-OSWR method (Algorithm 5.1) with or without Krylov acceleration (see below), in two space dimensions. In Sections 8.1 and 8.2, these results are shown in the context of the NICEM method [25], for the space discretization, that enables the use of more efficient transmission conditions in the OSWR method (i.e., Ventcel conditions) with general domain decomposition and meshes. A regular advection-diffusion case is first considered, with a comparison between Robin and Ventcel conditions.[11] Then, an example with boundary layer and vorticies is given, on conforming and nonconforming meshes, and on long and short time intervals (the latter being the worst case for Parareal). In section 8.3 we show an example in an industrial context [3], using mixed finite elements and Robin transmission conditions. In all cases the backward Euler scheme in time is used, with time step ΔT for the coarse solver.

The multidomain problem (4.3) can actually be reformulated as an interface problem (see [9, 22]) that can be solved by various iterative methods, such as block-Jacobi (which corresponds to the OSWR algorithm) or GMRES. The latter being faster, it will be used and called "OSWRG" in what follows, and the corresponding coupled method is called Parareal-OSWRG. In that case L will designate the number of GMRES iteration, in the fine solver. A comparison between Parareal-OSWR and Parareal-OSWRG will be done in Section 8.1.

For the coarse solver, as well as for the fine solver with $L = \infty$, we use the OSWRG method, with a stopping criterion such that the jump of the optimized transmission conditions, measured in the L^2-norm on the interface, has been reduced below 10^{-13}. Otherwise, we will consider L iterations for the fine solver, with different values of L. In the results below, the cost of the coarse solver is negligible.

Note that in what follows the error is measured in the $L^\infty(0, T; \mathcal{X})$-norm, and that we obtain similar results in the $L^\infty(\Omega \times (0, T))$-norm.

[11] These optimized parameters are variable and calculated on each edge of the interfaces.

8.1. A rotating velocity. We set $\Omega =]0,1[\times]0,1[$, $T = 21$, and consider the 2D problem:

$$\partial_t u + \nabla \cdot (\boldsymbol{a} u) - \nu \Delta u = f \quad \text{in } \Omega \times (0,T), \tag{8.1a}$$
$$u = u_D \quad \text{on } \partial\Omega \times (0,T), \tag{8.1b}$$
$$u(\cdot,0) = u_0 \quad \text{in } \Omega, \tag{8.1c}$$

with a rotating velocity field $\boldsymbol{a} = (a_x, a_y)$, where $a_x = -\sin\left(\pi(y-\frac{1}{2})\right)\cos\left(\pi(x-\frac{1}{2})\right)$ and $a_y = \cos\left(\pi(y-\frac{1}{2})\right)\sin\left(\pi(x-\frac{1}{2})\right)$; see Figure 3 on the left. We choose f and the values of the boundary and initial conditions so that the exact solution is given by

$$u(x,y,t) = \cos(\pi x)\sin(\pi y)\cos(\frac{2\pi t}{11}), \quad \forall (x,y) \in \Omega,\ \forall t \in (0,T). \tag{8.2}$$

The number of time windows for the Parareal iterations is $N = 21$. The time steps of the coarse and fine solvers are $\Delta t_C = 1$ and $\Delta t_F = 0.0156$, respectively.

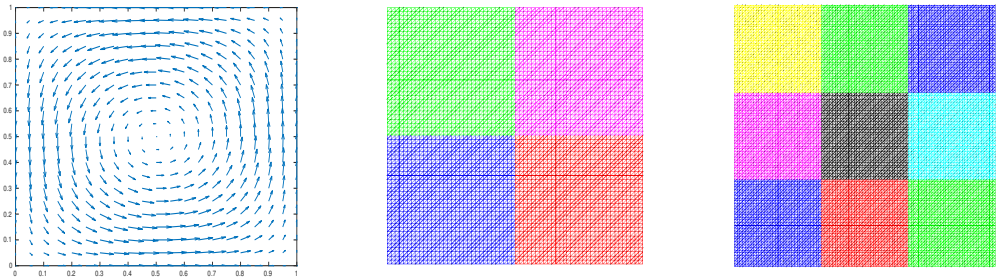


FIG. 3. *Example 1: Rotating velocity (left), decomposition of Ω into four subdomains (middle), and decomposition of Ω into nine subdomains (right)*

In what follows we denote by e the relative scheme error, in $L^\infty(0,T;\mathcal{X})$-norm, between the converged Parareal solution and the solution u of problem (8.1), given in (8.2). The term "OSWRG iterations" will designate the iterations of the fine solver. We will give two cases for the diffusion and the space domain decomposition.

Case 1. We take $\nu = 0.05$ and a decomposition of Ω into four subdomains as in Figure 3 in the middle. The number of triangles in the whole domain $\bar{\Omega}$ is 8192 and the mesh size in each subdomain is 0.0221.

In Figure 4 (left), we plot the evolution of the relative error between the Parareal-OSWRG solution (with Ventcel conditions) and the converged Parareal solution, as a function of the number of Parareal iterations, for different values of L. The case $L = \infty$ (black circle curve) corresponds to $L = 10$ (average value). The horizontal dashed green line represents $0.1e$ and we consider the number of iterations such that the algorithm error is smaller than this value. This is obtained after 3 iterations for Parareal or Parareal-OSWRG with $L = 8$, and after 5, 8, 18 iterations for Parareal-OSWRG with $L = 4$, $L = 2$, $L = 1$, respectively. These data are reported in Table 5 (left). We observe that the fastest case is $L = 2$ with 16 OSWRG iterations globally.

Table 6 shows the gain (in term of fine solver iterations), when the fine solvers are performed in parallel, of the full Parareal ($L = 10$) or the coupled Parareal-OSWRG

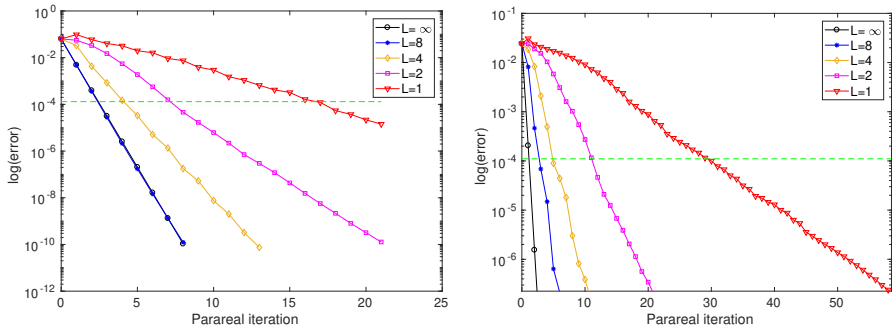


FIG. 4. *Example 1: Relative error versus Parareal iterations. Case $\nu = 0.05$ with four subdomains (left) and case $\nu = 0.1$ with nine subdomains (right)*

L	1	2	4	8	10 ($\approx \infty$)
k	17	8	5	3	3
$L * k$	17	16	20	24	30

L	1	2	4	8	24 ($\approx \infty$)
k	30	12	5	3	2
$L * k$	30	24	20	24	48

TABLE 5

Example 1: Number of Parareal iterations k and total number of OSWRG iterations $L * k$, versus L, Case $\nu = 0.05$ with four subdomains (left) and case $\nu = 0.1$ with nine subdomains (right)

Solver	OSWRG	Parareal	Parareal-OSWRG ($L = 2$)
Iterations (n)	16	30	16
Loss factor ($\ell = n/16$)	–	1.87	1
Final gain factor (N/ℓ)	–	11.2	21

TABLE 6

Example 1 (case 1): Gain factor for Parareal and coupled Parareal-OSWRG methods compared to the OSWRG solver for solving problem (8.1), with $N = 21$ time windows

methods compared to the OSWRG algorithm for solving problem (8.1). The first column corresponds to the number of OSWRG iterations to reach $0.1e$ (where e is the relative scheme error), when the OSWRG method is used to solve (8.1). We observe that for $N = 21$ time windows, we gain a factor 11.2 for Parareal and a factor 21 for Parareal-OSWRG which is the expected parallel efficiency.

In Figure 5, we consider four cases for the fine solver: OSWRG with Ventcel (top left), OSWR with Ventcel (top right), OSWRG with Robin (bottom left), and OSWR with Robin (bottom right). The best choice for the latter is $L = 4$, and $L = 2$ else.

The gain factor, when the fine solvers are performed in parallel, compared to the corresponding pure OSWR (or OSWRG) algorithm (with Robin or Ventcel) is as follows: 21 for Parareal-OSWRG with Ventcel (i.e., the expected parallel efficiency), 22.05 for Parareal-OSWRG with Robin, 23.86 for Parareal-OSWR with Ventcel, and 25.66 for Parareal-OSWR with Robin (i.e., the gain factor is a little bit better when using OSWR instead of OSWRG).

In what follows (case 2 and Section 8.2), we will consider the fastest method, that is the Parareal-OSWRG algorithm with Ventcel conditions.

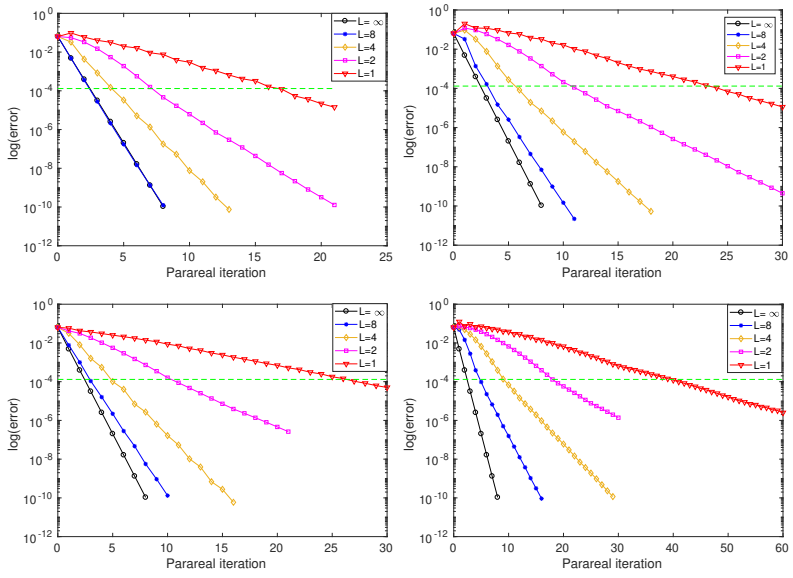


FIG. 5. *Example 1 (case 1): Relative error versus Parareal iterations. Parareal-OSWRG with Ventcel (top left), Parareal-OSWR with Ventcel (top right), Parareal-OSWRG with Robin (bottom left), and Parareal-OSWR with Robin (bottom right)*

Case 2. We take $\nu = 0.1$ and a decomposition of Ω into nine subdomains as in Figure 3 (right). The number of triangles in the whole domain $\bar{\Omega}$ is 18432 and the mesh size in each subdomain is 0.0147.

In Figure 4 (right), the Parareal algorithm (black circle curve) corresponds to $L = 24$. The horizontal dashed green line represents $0.1e$, and is reached after 2 iterations for Parareal, and after 3, 5, 12, 30 iterations for Parareal-OSWRG with $L = 8$, $L = 4$, $L = 2$, $L = 1$, respectively. These data are shown in Table 5 (right) and we see that the fastest case is $L = 4$, with a total of 20 OSWRG iterations.

Table 7 shows the gain, when the fine solvers are performed in parallel, of the Parareal or the coupled Parareal-OSWRG methods compared to the pure OSWRG algorithm. In the first column we give the number of OSWRG iterations to reach $0.1e$, when it is used as a solver for problem (8.1). We observe that for $N = 21$ time windows, we gain a factor 10.94 for Parareal and a factor 26.25 for Parareal-OSWRG which represents an efficiency strictly greater than 1, since the number of processors is 21. This is notably better than the expected parallel efficiency.

Solver	OSWRG	Parareal	Parareal-OSWRG ($L = 4$)
Iterations (n)	25	48	20
Loss factor ($\ell = n/25$)	–	1.92	0.8
Final gain factor (N/ℓ)	–	10.94	26.25

TABLE 7

Example 1 (case 2): Gain factor for Parareal and coupled Parareal-OSWRG methods compared to the OSWRG solver for solving problem (8.1), with $N = 21$ time windows

8.2. A boundary layer case with vorticies. We consider problem (8.1) with $f = 0$, $u_0(x, y) = 1 - x$, $u_D(x, y) = 1$ on $\{x = 0\}$, and $u_D(x, y) = 0$ elsewhere, and the following velocity field (see [37, 34]): $\boldsymbol{a} = 0.32\pi\big(\sin(4\pi x)\sin(4\pi y), \cos(4\pi x)\cos(4\pi y)\big)$; see Figure 6 (left). The diffusion coefficient is $\nu = 0.01$ and the final time is $T = 51$. The number of time windows for the Parareal iterations is $N = 51$. The time step of the coarse solver is $\Delta t_C = 1$.

In what follows we consider a decomposition of Ω into nine subdomains and consider a uniform mesh (case 1) and then a nonconforming mesh adapted to the physics (case 2). The computed Parareal-OSWRG solution at final time $t = T$ (with the nonconforming mesh) is shown in Figure 6 (right).

Case 1. We consider a uniform mesh as in Figure 3 (right), with a mesh size in each subdomain equal to 0.0147. The time step of the fine solver is $\Delta t_F = 0.0156$.

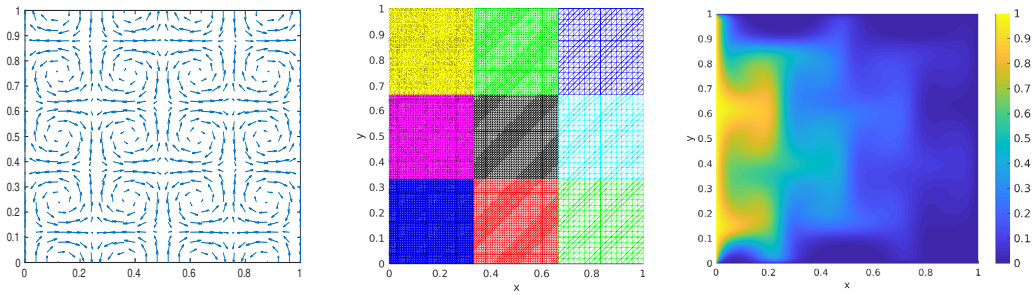


FIG. 6. *Example 2: Velocity field (left), nonconforming meshes (middle), and computed Parareal-OSWRG solution at final time (right)*

In Figure 7 (left), we plot the evolution of the relative error in $L^\infty(0, T; \mathcal{X})$-norm, between the Parareal-OSWRG solution and the converged Parareal solution, as a function of the number of Parareal iterations, for different values of L. The Parareal algorithm corresponds to $L = 10$ and we will compare the other cases of L to this case. We do not know the solution of problem (8.1) and thus the relative scheme error e, however we expect that e is between 10^{-1} and 10^{-3}, and thus $0.1e$ between 10^{-2} and 10^{-4}. Consequently, in Figure 7 (left), the horizontal dashed green lines represent three possible values for $0.1e$, and Table 8 shows the number of Parareal and total OSWRG iterations to reach these different values, for $L = 1, 2, 4, 8, 10$. We observe that the fastest case is $L = 2$.

We see in Figure 7 two convergence regimes, on the one hand for the case $L = \infty$ where the iterative method is purely the parareal method, and on the other hand for $L = 1$ where the method is essentially of the Schwarz type.

More precisely, for finite L (=1, 2, 4, 8) we notice that the first iterations converge at the parareal convergence rate, to take into account the temporal component of the error; then the algorithm takes into account the spatial component of the error with convergence curves that looks like those of the pure Schwarz type iterations (see [10]), since the coupled iterations are essentially Schwarz type iterations, this happening all the faster as L is large.

The OSWRG algorithm used as a solver for problem (8.1) will need 22, 36 and 49 iterations to get a relative error in $L^\infty(0, T; \mathcal{X})$-norm (between the iterate solution and the converged OSWRG solution) smaller than 10^{-2}, 10^{-3} and 10^{-4}, respectively. Table 9 shows the gain, when the fine solvers are performed in parallel, of the Parareal

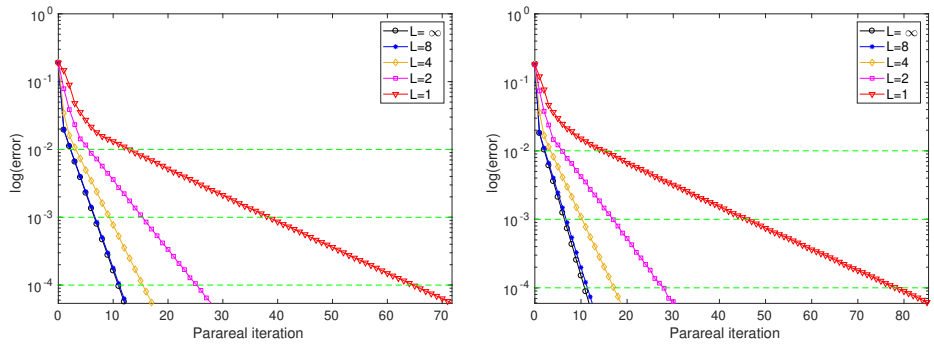


FIG. 7. *Example 2: Relative error versus Parareal iterations, with a zoom on the first iterations. Left : Case 1 (conforming meshes). Right : Case 2 (nonconforming meshes)*

or the coupled Parareal-OSWRG methods compared to the OSWRG algorithm. In this table we take the values obtained for $e = 10^{-2}$ (note that these values correspond also to those obtained with a mean value of the values for $e = 10^{-1}$, $e = 10^{-2}$ and $e = 10^{-3}$). We observe that for $N = 51$ time windows, we gain approximately a factor 26.29 for Parareal and a factor 57.38 for Parareal-OSWRG which is slightly better than the expected parallel efficiency.

L	1	2	4	8	10 ($\approx \infty$)
k	13	6	4	3	3
$L * k$	13	12	16	24	30

L	1	2	4	8	10 ($\approx \infty$)
k	39	16	10	7	7
$L * k$	39	32	40	56	70

L	1	2	4	8	10 ($\approx \infty$)
k	65	26	16	11	11
$L * k$	65	52	64	88	110

TABLE 8

Example 2 (case 1): Number of Parareal iterations k and total number of OSWRG iterations $L * k$, versus L to reach 3 different values: 10^{-2} (top left), 10^{-3} (top right) and 10^{-4} (bottom)

Solver	OSWRG	Parareal	Parareal-OSWRG ($L = 2$)
iterations (n)	36	70	32
loss factor ($\ell = n/36$)	–	1.94	0.89
final gain factor (N/ℓ)	–	26.29	57.38

TABLE 9

Example 2 (case 1): Gain factor for Parareal and coupled Parareal-OSWRG methods compared to the OSWRG solver for solving problem (8.1), with $N = 51$ time windows

Case 2. We consider nonconforming meshes, refined in the region of the boudary layer (see Figure 6 (middle)), with a mesh size equal to 0.0065, 0.0131, and 0.0295

for the subdomains with a boundary along $\{x = 0\}$, those with boundaries along $\{x = \frac{1}{3}\}$ and $\{x = \frac{2}{3}\}$, and those with a boundary along $\{x = 1\}$, respectively. The time step of the fine solver is $\Delta t_F = 0.01$. In Figure 7 (right), we plot the evolution of the relative error in $L^\infty(0, T; \mathcal{X})$-norm, between the Parareal-OSWRG solution and the converged Parareal solution, as a function of the number of Parareal iterations, for different values of L. The Parareal algorithm ($L = \infty$) corresponds to the case $L = 12$. As for case 1, the horizontal dashed green lines represent three possible values of 10% of the error, and Table 10 shows the number of Parareal and total OSWRG iterations to reach these different values, for $L = 1, 2, 4, 8, 12$. We observe that $L = 2$ is the case with the fewest iterations. We also observe on this figure two convergence regimes as for Case 1.

The OSWRG algorithm used as a solver for problem (8.1) will need 24, 40 and 56 iterations to get a relative error (between the iterative solution and the converged OSWRG solution) smaller than 10^{-2}, 10^{-3} and 10^{-4}, respectively. Table 11 shows the gain, when the fine solvers are performed in parallel, of the Parareal or the coupled Parareal-OSWRG methods compared to the OSWRG algorithm. In this table we take the values obtained for $e = 10^{-2}$ (note that these values correspond also to those obtained with a mean value of the values for $e = 10^{-1}$, $e = 10^{-2}$ and $e = 10^{-3}$). We observe that for $N = 51$ time windows, the results with a conforming mesh (Table 9) or with nonconforming meshes (Table 11) are very close and slightly better for nonconforming meshes adapted to the physics. Indeed, the gain is approximately a factor 24.28 for Parareal and a factor 60 for Parareal-OSWRG which is significantly better than the expected parallel efficiency.

L	1	2	4	8	12 ($\approx \infty$)
k	16	6	4	3	3
$L * k$	16	12	16	24	36

L	1	2	4	8	12 ($\approx \infty$)
k	47	17	11	7	
$L * k$	47	34	44	56	84

L	1	2	4	8	12 ($\approx \infty$)
k	79	28	17	11	
$L * k$	79	56	68	88	132

TABLE 10

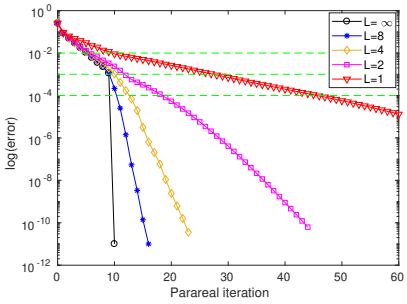
Example 2 (case 2): Number of Parareal iterations k and total number of OSWRG iterations $L * k$, versus L to reach 3 different values: 10^{-2} (top left), 10^{-3} (top right) and 10^{-4} (bottom)

Solver	OSWRG	Parareal	Parareal-OSWRG ($L = 2$)
Iterations (n)	40	84	34
Loss factor ($\ell = n/40$)	–	2.1	0.85
Final gain factor (N/ℓ)	–	24.28	60

TABLE 11

Example 2 (case 2): Gain factor for Parareal and coupled Parareal-OSWRG methods compared to the OSWRG solver for solving problem (8.1), with $N = 51$ time windows

Case 3. We consider the example of case 1, now with $T = 1$ and $N = 10$, i.e. an advection dominated case on short time interval for which the Parareal algorithm converges slowly.



L	1	2	4	8	35 ($\approx \infty$)
k	30	12	11	10	10
$L * k$	30	24	44	80	350

TABLE 12
Number of Parareal iterations k and total number of OSWRG iterations $L * k$, versus L

FIG. 8. *Relative error versus Parareal iterations*

In Figure 8, we plot the evolution of the relative error in $L^\infty(0,T;\mathcal{X})$-norm, between the Parareal-OSWRG solution and the converged Parareal solution, as a function of the number of Parareal iterations, for different values of L. The Parareal algorithm ($L = \infty$) corresponds to the case $L = 35$. As for case 1, the horizontal dashed green lines represent three possible values of 10% of the error, and Table 12 shows the number of Parareal and total OSWRG iterations to reach 10^{-3} for the values $L = 1, 2, 4, 8, 35$. As in Section 7.2, we observe that the pure Parareal algorithm doesn't perform well since it needs full 10 iterations to converge. Parareal-OSWRG with $L = 2$ is the best choice, and a good alternative method to the Parareal method. Indeed, as shown on Table 13, we gain a factor 29.58 for Parareal-OSWRG compared to the pure OSWRG algorithm, and a factor 14.58 compared to the pure Parareal algorithm.

Solver	OSWRG	Parareal	Parareal-OSWRG ($L = 4$)
Iterations (n)	71	350	24
Loss factor ($\ell = n/71$)	–	4.93	0.34
Final gain factor (N/ℓ)	–	2.03	29.58

TABLE 13
Example 2 (case 3): Gain factor for Parareal and coupled Parareal-OSWRG methods compared to the OSWRG solver for solving problem (8.1), with $N = 10$ time windows, to reach 10^{-3}

Remark 8.1. Note that in the results above, the number of OSWRG iterations to reach full convergence, inside each time window in the Parareal iterations (35), is smaller than the one needed to reach full convergence on $(0,T)$ (71), probably because the time windows are smaller. This explains the gain of a factor 2.03 when using Parareal, compared to the pure OSWRG algorithm.

Remark 8.2. The 2D simulations of this section don't show a comparison with the PSWR algorithm of [15]. However we have observed that the case $L = 1$ is not the optimal choice of L. Doing incomplete iterations for the coarse solver as well, will very likely lead to similar or slower convergence.

8.3. Example in an industrial context. We consider a model problem given by ANDRA, the French National Agency for Radioactive Waste Management (see [22, 1, 3]), that simulates the transport of a contaminant in and around a nuclear waste repository site. The simulation domain is depicted in Figure 9 (left) (not at scale). The

nuclear waste is stored in the repository (yellow), which is a 2950m by 10m rectangle located in the center of a clay domain of 3950m by 140m. In this example, we are concerned with the following time-dependent diffusion problem in mixed formulation:

$$\boldsymbol{\sigma} = -\boldsymbol{S}\nabla u \quad \text{in} \quad \Omega \times (0,T), \tag{8.3a}$$

$$\phi \frac{\partial u}{\partial t} + \nabla \cdot \boldsymbol{\sigma} = f \qquad \text{in} \quad \Omega \times (0,T), \tag{8.3b}$$

where $\Omega = [0, 3950] \times [0, 140]$ is decomposed into nine subdomains with Ω_5 the nuclear waste repository; see Figure 9. Here u is the (dimensionless) concentration of the contaminant, ϕ is the porosity, and $\boldsymbol{S}$ is the time-independent diffusion tensor, given in Table 14 (where $\mathbb{I}$ is the identity matrix).

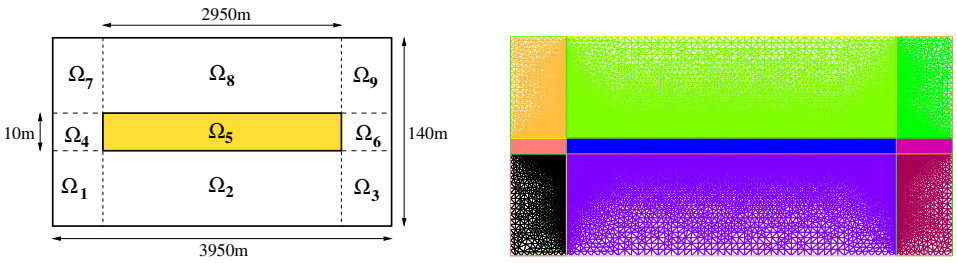


FIG. 9. *Geometry of the nuclear waste repository (yellow) and the clay layer around it, with its decomposition into nine subdomains (left), and mesh used in and around the repository (right)*

	ϕ	$\boldsymbol{S}$	f
Ω_5	0.2	$2 \times 10^{-9}\mathbb{I}$ m^2/s	$\begin{cases} 10^{-5} \text{ years}^{-1} & \text{if } t \leq 10^5 \text{ years}, \\ 0 & \text{if } t > 10^5 \text{ years} \end{cases}$
Ω_i $(i \neq 5)$	0.05	$5 \times 10^{-12}\mathbb{I}$ m^2/s	0

TABLE 14
Porosity ϕ, diffusion tensor $\boldsymbol{S}$, and source term f

The initial condition is $u_0 = 0$, and we set homogeneous Dirichlet conditions on the top and the bottom of Ω, and homogeneous Neumann conditions on the other sides of $\partial\Omega$. We are interested in the long-term behavior of the repository, over one million years, so that we set $T = 10^6$ years. A dimensionless form of this problem is given in [3]. The number of triangles in the mesh of $\bar{\Omega}$ is 58710 (see Figure 9 (right)) and the number of DoF is 148134. We use the Parareal-OSWRG method with Robin transmission conditions to solve this problem.

In Figure 10, we plot the relative error in $L^2(\Omega \times (0,T))$-norm, between the Parareal-OSWRG solution and the converged Parareal solution, versus the number of Parareal iterations, for different values of L. The Parareal algorithm ($L = \infty$) corresponds to $L = 10$.

In [1], an a posteriori stopping criterion is given for this test case and the dashed green line shown in Figure 10 corresponds to this criterion. This line is reached after 3 iterations for Parareal and Parareal-OSWRG with $L = 4$, and after 13 iterations for Parareal-OSWRG with $L = 2$.[12] These data are reported in Table 15 (for the methods

[12]The case $L = 1$ is not shown, as the error is still far above the green line after 30 iterations.

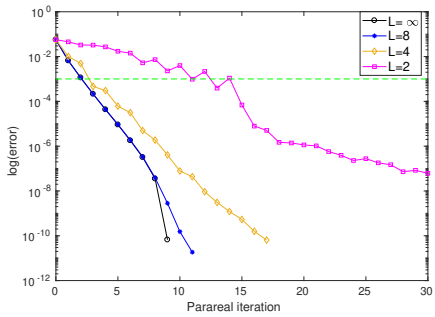


FIG. 10. *Relative error versus Parareal iterations*

L	2	4	8	12 ($\approx \infty$)
k	13	3	3	3
$L * k$	26	12	24	36

TABLE 15
Number of Parareal iterations k and total number of OSWRG iterations $L * k$, versus L

that converge in at most 30 iterations) and we observe that the fastest case is $L = 4$ with a total of 12 OSWRG iterations.

Table 16 shows the gain when the fine solvers are performed in parallel, of the Parareal or the coupled Parareal-OSWRG methods compared to the OSWRG method. In the first column we give the number of OSWRG iterations given in [1] from the a posteriori stopping criterion, when the OSWRG method is used for solving problem (8.3). We see that for $N = 10$ time windows, we gain a factor 3.06 for Parareal and a factor 9.17 for Parareal-OSWRG which is almost the expected parallel efficiency.

Solver	OSWRG	Parareal	Parareal-OSWRG ($L = 4$)
iterations (n)	11	36	12
loss factor ($\ell = n/11$)	–	3.27	1.09
final gain factor (N/ℓ)	–	3.06	9.17

TABLE 16
Example 3: Gain factor for Parareal and coupled Parareal-OSWRG methods compared to the OSWRG solver for solving problem (8.3), with $N = 10$ time windows

Remark 8.3. Note that increasing the contrast in the diffusion tensor $\boldsymbol{S}$ by taking $\boldsymbol{S} = 2 \times 10^{-9}$ in Ω_5 and $\boldsymbol{S} = 5 \times 10^{-13}$ in the other subdomains leads to slightly better results : The best choice is still $L = 4$, and for $N = 10$ time windows we gain a factor 2.67 for Parareal, and a factor 10 for Parareal-OSWRG, which is the expected parallel efficiency.

Remark 8.4. Note that to determine a suitable L in general, which may vary through Parareal iterations, one could design a posteriori error estimates that combine contributions from the scheme error and from the current nonconverged OSWR iteration. This would give a suitable L and an efficient stopping criterion for the Parareal-OSWR iterations, which would enhance the application of the method.

9. Proof of estimate (4.13) in Theorem 4.2 .

Proof. For $i = 1, 2$, let (u_i^ℓ, ξ_i^ℓ) be defined by (4.4)–(4.5) with initial condition $\bar{u}_0$. We set $\zeta_i^\ell := \xi_i^\ell - \xi_i$, where ξ_i is defined in (2.4), and $e_i^\ell := u_i^\ell - u|_{\Omega_i}$, where u is the

solution of (2.1). With (ζ_1^0, ζ_2^0) given, the error e_i^ℓ satisfies for $\ell = 1, 2, \ldots, L$:

$$\mathcal{L} e_i^\ell = 0 \qquad \text{in } \Omega_i \times \mathcal{I}, \tag{9.1a}$$

$$e_i^\ell(\cdot, 0) = e_{0,i} \quad \text{in } \Omega_i, \tag{9.1b}$$

$$(\mathcal{B}_i e_i^\ell)(0, \cdot) = \zeta_i^{\ell-1} \quad \text{on } \mathcal{I}, \tag{9.1c}$$

where

$$\zeta_i^\ell = \mathcal{B}_i e_j^\ell(0, \cdot), \ \ell \geq 1. \tag{9.2}$$

Then the proof follows the same steps as in [12, Theorem 5.15].

Multiplying equation (9.1a) by e_i^ℓ and integrating in space on Ω_i, we obtain

$$\tfrac{1}{2}\tfrac{d}{dt}\|e_1^\ell\|^2 + \nu\|\partial_x e_1^\ell\|^2 + b\|e_1^\ell\|^2 - \big(\nu\partial_x e_1^\ell - \frac{a}{2}e_1^\ell\big)(0,.)e_1^\ell(0,.) = 0,$$
$$\tfrac{1}{2}\tfrac{d}{dt}\|e_2^\ell\|^2 + \nu\|\partial_x e_2^\ell\|^2 + b\|e_2^\ell\|^2 + \big(\nu\partial_x e_2^\ell - \frac{a}{2}e_2^\ell\big)(0,.)e_2^\ell(0,.) = 0.$$

Introducing the Robin interface operators $\mathcal{B}_i,\ i = 1, 2$, defined in (2.2), and rewriting the terms on the interface in the form

$$\big(\nu\partial_x e_i^\ell - \frac{a}{2}e_i^\ell\big)e_i^\ell = \tfrac{1}{2p}\big((\mathcal{B}_1 e_i^\ell)^2 - (\mathcal{B}_2 e_i^\ell)^2\big), \qquad i = 1, 2,$$

we obtain the energy estimates for $i = 1, 2$

$$\tfrac{1}{2}\tfrac{d}{dt}\|e_i^\ell\|^2 + \nu\|\partial_x e_i^\ell\|^2 + b\|e_i^\ell\|^2 + \tfrac{1}{2p}((\mathcal{B}_j e_i^\ell)(0,.))^2 = \tfrac{1}{2p}((\mathcal{B}_i e_i^\ell)(0,.))^2, \quad j = 3 - i.$$

Replacing $(\mathcal{B}_i e_i^\ell)(0,.) = \zeta_i^{\ell-1}$, $(\mathcal{B}_j e_i^\ell)(0,.) = \zeta_j^\ell$ (from (9.1c) and (9.2)), and summing the above expression over i, we get for all $t \in \mathcal{I}$,

$$\sum_i \big(\tfrac{1}{2}\tfrac{d}{dt}\|e_i^\ell\|^2 + \nu\|\partial_x e_i^\ell\|^2 + b\|e_i^\ell\|^2\big) + \tfrac{1}{2p}\sum_i (\zeta_i^\ell)^2 = \tfrac{1}{2p}\sum_i (\zeta_i^{\ell-1})^2.$$

Then, integrating on $\mathcal{I}$, and considering $L^2(0, T; H^1(\Omega_i))$ endowed with the norm $\|u\|^2_{L^2(0,T;H^1(\Omega_i))} := \int_{\mathcal{I}} \big(\nu\|\partial_x u\|^2 + b\|u\|^2\big)\, dt$, we obtain

$$\tfrac{1}{2}\|e^\ell(., T)\|^2 + \sum_i \|e_i^\ell\|^2_{L^2(0,T,H^1(\Omega_i))} + \tfrac{1}{2p}\|\boldsymbol{\zeta}^\ell\|^2_{\mathcal{I}} = \tfrac{1}{2}\sum_i \|e_i^\ell(., 0)\|^2 + \tfrac{1}{2p}\|\boldsymbol{\zeta}^{\ell-1}\|^2_{\mathcal{I}}. \quad \square$$

Summing over ℓ from 1 to L, we get a telescopic sum on the interface and thus we get (4.13).

Acknowledgment. The authors warmly thank the referees for their comments that helped us to improve the results of this paper.

REFERENCES

[1] S. Ali Hassan, *A Posteriori Error Estimates and Stopping Criteria for Solvers using Domain Decomposition Method and with Local Time Stepping*, PhD thesis, Université Pierre et Marie Curie, Sorbonne Université, 2017.

[2] S. Ali Hassan, C. Japhet, M. Kern, and M. Vohralík, *A posteriori stopping criteria for optimized Schwarz domain decomposition algorithms in mixed formulations*, Comput. Methods Appl. Math., 18 (2018), pp. 495–519.

[3] S. Ali Hassan, C. Japhet, and M. Vohralík, *A posteriori stopping criteria for space-time domain decomposition for the heat equation in mixed formulations*, Electron. Trans. Numer. Anal., 49 (2018), pp. 151–181.

[4] L. Baffico, S. Bernard, Y. Maday, G. Turinici, and G. Zérah, *Parallel-in-time molecular-dynamics simulations*, Phys. Rev. E, 66 (2002), p. 057701.
[5] G. Bal and Y. Maday, *A "Parareal" time discretization for non-linear PDE's with application to the pricing of an american put*, in Recent Developments in Domain Decomposition Methods, L. F. Pavarino and A. Toselli, eds., Springer, Berlin, Heidelberg, 2002, pp. 189–202.
[6] P.-M. Berthe, *Méthodes de décomposition de domaine de type relaxation d'ondes optimisée pour l'équation de convection-diffusion instationnaire discrétisée par volumes finis*, PhD thesis, Thèse de doctorat en Mathématiques appliquées, Université Paris XIII, 2013. 2013PA132055.
[7] D. Q. Bui, *New Space-time Domain Decomposition Algorithms Combined with the Parareal Algorithm*, PhD thesis, Thèse de doctorat, Mathématiques appliquées, Université Sorbonne Paris Nord, 2021.
[8] D. Q. Bui, C. Japhet, Y. Maday, and P. Omnes, *Coupling Parareal with Optimized Schwarz Waveform Relaxation for Parabolic Problems.* HAL Preprint 03167571, 2021.
[9] V. Dolean, P. Jolivet, and F. Nataf, *An Introduction to Domain Decomposition Methods*, Algorithms, Theory, and Parallel Implementation, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 2015.
[10] M. J. Gander, *Optimized Schwarz methods*, SIAM J. Numer. Anal., 44 (2006), pp. 699–731.
[11] M. J. Gander and E. Hairer, *Nonlinear convergence analysis for the Parareal algorithm*, in Domain Decomposition Methods in Science and Engineering XVII, U. Langer, M. Discacciati, D. E. Keyes, O. B. Widlund, and W. Zulehner, eds., Springer, Berlin, Heidelberg, 2008, pp. 45–56.
[12] M. J. Gander and L. Halpern, *Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems*, SIAM J. Numer. Anal., 45 (2007), pp. 666–697.
[13] M. J. Gander, L. Halpern, and F. Nataf, *Optimal convergence for overlapping and non-overlapping Schwarz waveform relaxation*, in Proceedings of the 11th International Conference on Domain Decomposition Methods, C.-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, eds., DDM.org, Augsburg, 1999, pp. 27–36.
[14] ———, *Optimal Schwarz waveform relaxation for the one dimensional wave equation*, SIAM J. Numer. Anal., 41 (2003), pp. 1643–1681.
[15] M. J. Gander, Y. Jiang, and B. Song, *A superlinear convergence estimate for the Parareal Schwarz waveform relaxation algorithm*, SIAM J. Sci. Comput., 41 (2019), pp. A1148–A1169.
[16] M. J. Gander, Y.-L. Jiang, and R.-J. Li, *Parareal Schwarz waveform relaxation methods*, in Domain Decomposition Methods in Science and Engineering XX, R. Bank, M. Holst, O. Widlund, and J. Xu, eds., Springer, Berlin, Heidelberg, 2013, pp. 451–458.
[17] M. J. Gander and A. M. Stuart, *Space-time continuous analysis of waveform relaxation for the heat equation*, SIAM J. Sci. Comput., 19 (1998), pp. 2014–2031.
[18] M. J. Gander and S. Vandewalle, *Analysis of the Parareal time-parallel time-integration method*, SIAM J. Sci. Comput., 29 (2007), pp. 556–578.
[19] E. Giladi and H. Keller, *Space-time domain decomposition for parabolic problems*, Numer. Math., 93 (2002), pp. 279–313.
[20] R. Guetat, *Méthode de parallélisation en temps : application aux méthodes de décomposition de domaine*, PhD thesis, Thèse de doctorat dirigée par Yvon Maday, Mathématiques appliquées, Université Pierre et Marie Curie et Ecole polytechnique de Tunisie, 2011.
[21] L. Halpern, C. Japhet, and J. Szeftel, *Optimized Schwarz waveform relaxation and discontinuous Galerkin time stepping for heterogeneous problems*, SIAM J. Numer. Anal., 50 (2012), pp. 2588–2611.
[22] T.-T.-P. Hoang, J. Jaffré, C. Japhet, M. Kern, and J. E. Roberts, *Space-time domain decomposition methods for diffusion problems in mixed formulations*, SIAM J. Numer. Anal., 51 (2013), pp. 3532–3559.
[23] T.-T.-P. Hoang, Japhet, M. Kern, and J. E. Roberts, *Space-time domain decomposition for reduced fracture models in mixed formulation*, SIAM J. Numer. Anal., 54 (2016), pp. 288–316.
[24] C. Japhet, *Optimized Krylov-Ventcell Method. Application to convection-diffusion problems*, in Domain Decomposition Methods in Science and Engineering IX (3-8 June 1996), P. Bjørstad, M. Espedal, and D. Keyes, eds., John Wiley & Sons Ltd, 1998, pp. 382–389.
[25] C. Japhet, Y. Maday, and F. Nataf, *A new interface cement equilibrated mortar method with Ventcel conditions*, in Domain Decomposition Methods in Science and Engineering XXI, J. Erhel, M. Gander, L. Halpern, G. Pichot, T. Sassi, and O. Widlund, eds., Lect. Notes Comput. Sci. Eng. 98, Springer, 2014, pp. 329–336.

[26] C. Japhet and F. Nataf, *The best interface conditions for domain decomposition methods: absorbing boundary conditions*, in Absorbing Boundaries and Layers, Domain Decomposition Methods, Nova Sci. Publ., Huntington, NY, 2001, pp. 348–373.

[27] C. Japhet, F. Nataf, and F. Rogier, *The Optimized Order 2 method: Application to convectiondiffusion problems*, Future Gener. Comput. Syst., 18 (2001), pp. 17–30.

[28] J.-L. Lions, Y. Maday, and G. Turinici, *A "parareal" in time discretization of PDE's*, Comptes Rendus de l'Académie des Sciences - Series I - Mathematics, 332 (2001), pp. 661–668.

[29] J.-L. Lions and E. Magenes, *Non-homogeneous Boundary Value Problems and Applications. Vol. I*, Springer-Verlag, New York, 1972. Translated from the French by P. Kenneth, Die Grundlehren der mathematischen Wissenschaften, Band 181.

[30] ———, *Non-homogeneous Boundary Value Problems and Applications. Vol. II*, Springer-Verlag, New York, 1972. Translated from the French by P. Kenneth, Die Grundlehren der mathematischen Wissenschaften, Band 182.

[31] P.-L. Lions, *On the Schwarz alternating method. III: A variant for nonoverlapping subdomains*, in Proceedings of the Third International Symposium on Domain Decomposition Methods for Partial Differential Equations, Houston, TX, 1989, T. F. Chan, R. Glowinski, J. Périaux, and O. Widlund, eds., Philadelphia, PA, SIAM, 1990, pp. 202–223.

[32] Y. Maday and O. Mula, *An adaptive Parareal algorithm*, J. Comput. Appll. Math., 377 (2020), p. 112915.

[33] Y. Maday and G. Turinici, *The parareal in time iterative solver: a further direction to parallel implementation*, in Domain Decomposition Methods in Science and Engineering, T. J. Barth, M. Griebel, D. E. Keyes, R. M. Nieminen, D. Roose, T. Schlick, R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Widlund, and J. Xu, eds., Lect. Notes Comput. Sci. Eng. 40, Springer, Berlin, Heidelberg, 2005, pp. 441–448.

[34] V. Martin, *Méthodes de décomposition de domaine de type relaxation d'ondes pour des équations de l'océanographie*, PhD thesis, Thèse de doctorat, Mathématiques appliquées, Université Paris 13, 2003.

[35] ———, *An optimized Schwarz waveform relaxation method for the unsteady convection diffusion equation in two dimensions*, Appl. Numer. Math., 52 (2005), pp. 401–428.

[36] O. Mula Hernandez, *Quelques contributions vers la simulation parallèle de la cinétique neutronique et la prise en compte de données observées en temps réel*, PhD thesis, Thèse de doctorat dirigée par Yvon Maday, Mathématiques appliquées, Université Pierre et Marie Curie, 2014.

[37] P. Smolarkiewicz, *The multi-dimensional crowley advection scheme*, Mon. Weather Rev., 110 (1982), pp. 1968–1983.

[38] B. Song, Y. Jiang, and X. Wang, *Analysis of two new Parareal algorithms based on the Dirichlet-Neumann/Neumann-Neumann waveform relaxation method for the heat equation*, Numer. Algor., 86 (2021), pp. 1685–1703.