Frozen Gaussian Approximation based Domain decomposition methods for the linear Schrödinger equation beyond the semi-classical regime

E. Lorin\textsuperscript{a,c}, X. Yang\textsuperscript{b}, X. Antoine\textsuperscript{d}

\textsuperscript{a}Centre de Recherces Mathématiques, Université de Montréal, Montréal, Canada, H3T 1J4
\textsuperscript{b}University of California at Santa Barbara, Department of Mathematics, USA
\textsuperscript{c}School of Mathematics and Statistics, Carleton University, Ottawa, Canada, K1S 5B6
\textsuperscript{d}Institut Elie Cartan de Lorraine, Université de Lorraine, Inria Nancy-Grand Est, F-54506 Vandoeuvre-lès-Nancy Cedex, France

Abstract

The paper is devoted to develop efficient domain decomposition methods for the linear Schrödinger equation beyond the semiclassical regime, which does not carry a small enough rescaled Planck constant for asymptotic methods (e.g. geometric optics) producing good accuracy, but is still computationally expensive if direct methods (e.g. finite difference) are applied. This belongs to the category of computing middle-frequency wave propagation, where neither asymptotic nor direct methods can be directly used with both efficiency and accuracy. Motivated by recent works of the authors on absorbing boundary conditions [X. Antoine et al, J. Comput. Phys., 277 (2014), 268–304] and [X. Yang and J. Zhang, SIAM J. Numer. Anal., 52 (2014), 808–831], we introduce semiclassical Schwarz Waveform Relaxation methods (SSWR), which are seamless integrations of semiclassical approximation to Schwarz Waveform Relaxation methods. Two versions are proposed based on Herman-Kluk propagation and geometric optics respectively, and we prove the convergence and provide numerical evidence of efficiency and accuracy of these methods.

Keywords: Semiclassical approximation, absorbing boundary condition, domain decomposition, linear Schrödinger equation, pseudodifferential operators.

1. Introduction

We are interested in computing the time-dependent linear Schrödinger Equation (TDSE) in the middle-frequency regime:

\[
\begin{align*}
\imath \varepsilon \frac{\partial \psi^\varepsilon}{\partial t} &= -\frac{\varepsilon^2}{2} \Delta \psi^\varepsilon(t, x) + V(x, t)\psi^\varepsilon(t, x), \quad x \in \mathbb{R}^d, \ t \in (0, T],
\end{align*}
\]

with \( \psi^\varepsilon(0, x) = \psi^\varepsilon_0(x) \in L^2(\mathbb{R}^d, \mathbb{C}) \) as the initial wave function, \( \varepsilon \in \mathbb{R}^*_+ \) as the rescaled Planck constant, \( V \) as the potential function and \( \imath = \sqrt{-1} \). The middle-frequency regime is calibrated as \( \varepsilon_0 \leq \varepsilon \ll 1 \) whereas \( \varepsilon \ll \varepsilon_0 \) is the high-frequency regime.

Direct numerical simulation of (1) is prohibitively expensive due to the small parameter \( \varepsilon \ll 1 \), for example a mesh size of \( \text{O}(\varepsilon) \) is required when using the time-splitting spectral method [14], and the mesh size (and the time step as well) becomes even worse, since they need to be as small as \( \text{o}(\varepsilon) \), if finite difference methods are used [35, 36]. In the meantime, \( \varepsilon \geq \varepsilon_0 \) is not small enough for semiclassical methods (e.g. geometric optics, Gaussian beam methods, and frozen Gaussian approximation) to provide good accuracy [19, 27, 26, 32]. This motivates us to derive, analyze and present preliminary numerical performance of semiclassical Schwarz Waveform Relaxation (SSWR) Domain Decomposition Methods (DDM) for the computation of (1) in the middle-frequency regime.

Schwarz Waveform Relaxation (SWR) methods are iterative fixed-point methods aiming to reconstruct global solution from locally computed ones. Important efforts have been put on the development of Optimized
Schwarz Waveform Relaxation (OSWR) methods for linear and nonlinear Schrödinger equations, diffusion-advection-reaction equations, and Maxwell’s equations [23, 20, 17, 16, 11, 10]. The principle of OSWR is to derive, for large frequencies, transmission conditions from Transparent (TBCs) or high-order Absorbing Boundary Conditions (ABCs). The latter allow for an almost null spurious reflection of waves at boundaries of a computational domain. Such conditions were developed for many kinds of wave equations. We refer here to some of analytical and numerical works, mainly based on pseudodifferential operators (although other robust techniques exist such as perfectly matched layers (PMLs) [15]): wave equations [18], diffusion equation [22, 21], linear [4, 7, 8, 6, 1, 31], and nonlinear Schrödinger equations [38, 9, 5, 3, 28, 39], Maxwell [2], Klein-Gordon and Dirac equations [12]. From the DDM point of view, and most particularly for OSWR-type methods, the use of TBC-based transmission conditions leads to fast converging algorithms (in a few fixed-point iterations). Therefore, most of the works dedicated to the development and discretization of TBCs benefit to the design of efficient and robust OSWR methods. In addition, the precise formal analysis of the rate of convergence of OSWR is also possible through the use of pseudodifferential operator techniques; see [10, 23] for instance. Although OSWR methods are in principle very attractive, they may nevertheless suffer from computational complexity issues. Indeed, TBCs for the TDSE are often defined through nonlocal (fractional) pseudodifferential operators which are computationally nontrivial and expensive to approximate, most particularly if one wants to preserve accuracy and stability properties of the interior schemes. More simple SWR techniques exist, such as the Classical Schwarz Waveform Relaxation (CSWR) method, which is based on Dirichlet boundary conditions. Even if these methods are simple, they suffer from a slow convergence (and sometimes do not even converge), as discussed in [23, 10]. More elaborated SWR methods such as the ones based on optimized Robin boundary conditions seem to be a good compromise (relatively fast convergence and small computational complexity), [23].

In this paper, we develop semiclassical SWR methods for (1) in the middle-frequency regime. The originality primarily comes from the use of semiclassical solutions to the Schrödinger equation, more specifically the Frozen Gaussian Approximation (FGA) which provides a very accurate but relatively “inexpensive” ansatz in the semi-classical regime. This method was originally developed by Herman-Kluk (HK) [24] and was later mathematically analyzed in [37]. More recently, the HK formalism was used and analyzed to derive fast numerical solvers in the semi-classical regime for different classes of partial differential equations: scalar wave equations [32, 30], linear hyperbolic systems of conservation laws [33, 34], and ABCs for the linear Schrödinger equation in the semiclassical regime [40].

When the exact solution to the Schrödinger equation is known at subdomain interfaces, it is possible to derive extremely efficient DDMs. Although in general it is not possible to determine such solutions, some accurate approximations can be provided through FGA. In addition, as we will show, the FGA is well-adapted to parallel computing. The main purpose of this paper is to derive a SWR DDM by using the semi-classical approximation for the Schrödinger equation in the middle-frequency regime, where a pure FGA or Geometric Optics Approximation (GOA) would be more appropriate than solving a full Schrödinger equation. The general strategy can then be summarized as follows: the Schrödinger equation is solved by using a SWR domain decomposition method with FGA (or GOA)-based transmission conditions. Analysis and numerical simulations will be presented to show the main features of these original methods.

The paper is organized as follows. In Section 2, we recall some basic facts about FGA for solving the Schrödinger equation in the high-frequency regime. Then, we derive a frequency-based DDM in the purely semi-classical regime. An analysis is provided to show the scalability of this DDM. Section 3 is dedicated to the derivation of a basic FGA-based DDM in the middle-frequency regime. Some connections between FGA and the TBCs for the Schrödinger equation are also discussed. Although basic, these first approaches have interesting features which will be used in Section 4, where the so-called FGA Waveform Relaxation methods are presented and analyzed. Alternative techniques based on Geometric Optics is discussed in Section 5. Numerical experiments are provided in Section 6 to illustrate some of the ideas developed before as well as to show the efficiency of the proposed methods. We conclude in Section 7.
2. Domain decomposition in the phase space for the FGA formalism in the high-frequency regime

As a warmup, we first study the domain decomposition method for solving the Schrödinger equation (1) in the high-frequency regime \((\varepsilon \leq \varepsilon_0)\) based on FGA, and then generalize it to middle-frequency regime later.

Let us introduce two open sets \(\Omega^\pm_\eta\) such that \(\mathbb{R}^d = \Omega^+ \cup \Omega^-\). We denote by \(\phi^\pm_\varepsilon\) the solution to a time-dependent Schrödinger equation in \(\Omega^\pm_\eta\). Recall that solving the Schrödinger equation by domain decomposition in the quantum regime, see [11] for instance, often requires a Schwarz algorithm at the domain interface, which can be for instance derived from Dirichlet (CSWR) or transparent boundary conditions (OSWR). More specifically, for any Schwarz iteration \(k \geq 1\), the equation in \(\Omega^\pm_\eta\) reads

\[
\begin{cases}
\varepsilon \partial_t \phi^{\varepsilon,k}_{\pm} = H_\varepsilon \phi^{\varepsilon,k}_{\pm} \text{ in } \Omega^\pm_\eta \times (0,T), \\
\mathcal{B}^\varepsilon_{\pm} \phi^{\varepsilon,k}_{\pm} = \mathcal{B}^\varepsilon_{\pm} \phi^{\varepsilon,k-1}_{\pm} \text{ on } \Gamma^\pm_\eta \times (0,T),
\end{cases}
\]

with \(H_\varepsilon := -\varepsilon^2 \Delta/2 + V(x,t)\) and where \(\phi^{\varepsilon,0}\) are two given functions and \(\Gamma^\pm_\eta = \partial \Omega^\pm_\eta\). In the CSWR case, \(\mathcal{B}^\varepsilon_{\pm}\) is simply the identity operator and, for OSWR, \(\mathcal{B}^\varepsilon_{\pm}\) is a nonlocal Dirichlet-to-Neumann-like (DtN) pseudodifferential operator. We refer to [11, 20, 23] for further reading.

We propose here a different approach by decomposition of the phase space rather than the real space. This allows for avoiding Schwarz’ algorithms. FGA may appear as computational complex since it involves some additional dimensions. However, when the initial data is localized in the phase space, the computational complexity (integral in \((p,q)\)) is drastically reduced; see [40, 34]. In addition, we will show that domain decomposition algorithms with ideal speed-up can be easily derived.

In \(\mathbb{R}^d\), the FGA reads [40], for \(x \in \mathbb{R}^d\),

\[
\phi^{\varepsilon}_{\text{FGA}}(t,x) = \frac{1}{(2\pi \varepsilon)^{d/2}} \int_{\mathbb{R}^{2d}} a(t,q,p) w^\varepsilon(q,p) e^{\frac{i}{\varepsilon} \left(S(t,q,p) + P(t,q,p) \cdot (x - Q(t,q,p))\right)}
\]

\[
\times e^{-\frac{i}{\varepsilon} \left|p \cdot (y-q) - \frac{1}{\varepsilon} |y-q|^2 \phi^{\varepsilon}_0(y)\right| dy},
\]

with

\[
w^\varepsilon(q,p) = \int_{\mathbb{R}^d} e^{-\frac{i}{\varepsilon} p \cdot (y-q) - \frac{1}{\varepsilon} |y-q|^2} \phi^{\varepsilon}_0(y) dy.
\]

We now define \(\phi^{\varepsilon}_{\text{FGA,\pm}}\) by

\[
\phi^{\varepsilon}_{\text{FGA,\pm}}(t,x) = \frac{1}{(2\pi \varepsilon)^{d/2}} \int_{\mathbb{R}^d \times \Omega^\pm_\eta} a(t,q,p) w^\varepsilon(q,p) e^{\frac{i}{\varepsilon} \left(S(t,q,p) + P(t,q,p) \cdot (x - Q(t,q,p))\right)}
\]

\[
\times e^{-\frac{i}{\varepsilon} |x - Q(t,q,p)|^2} dq dp,
\]

with \(\mathbb{R}^d = \Omega^+_\eta \cup \Omega^-_\eta\) and \(\Omega^+_\eta \cap \Omega^-_\eta = \emptyset\). Clearly, \(\phi^{\varepsilon}_{\text{FGA,\pm}}\) can be computed independently at any time. We detail below the overall computational complexity.

One-domain approach (sequential computing). From a numerical point of view and following the notations of [40], with \((j,k,l) \in D_x \times D_q \times \tilde{D}_p\), the FGA reads at time \(t^n\) and \(x^j\) as

\[
\phi^{\varepsilon}_{\text{FGA}}(t^n, x^j) = \sum_{(k,l) \in D_q \times \tilde{D}_p} \left(\frac{a^{k,l}(t^n)}{(2\pi \varepsilon)^{d/2}} \int_{\mathbb{R}^{2d}} w^\varepsilon(k^j,p^k) e^{\frac{i}{\varepsilon} S(t^n,q^k,p^l, x^j) Q^k(t^n) - \frac{1}{\varepsilon} |x^j - Q^k(t^n)|^2} dq dp \right) |\delta q| |\delta p|.
\]

\(^1\)The notation \(\tilde{D}_p\) is used to precise that the integrand in the FGA is localized with respect to \(p\) [40, 34].
where \((Q^{k,1}(t^n), P^{k,1}(t^n))\) and \(a^{k,1}(t^n)\) are updated by using a fourth order Runge-Kutta (RK4) scheme, and \(r_\theta\) is the local truncation function with radius \(\theta\). We can estimate the overall computational cost to determine \(\phi^\epsilon_{\text{FGA}}(t^n, x^j)\) as follows. For \((k,l) \in D_q \times \tilde{D}_p\), we need

- \(N^{w,1}_{w}\) operations to compute \(w^\epsilon(q^k, p^{k,1})\).
- \(N^{d,1}_q\) RK4 operations to solve \(dQ^{k,1}/dt = P^{k,1}\) up to time \(t^n\).
- \(N^{p,1}_p\) RK4 operations to solve \(dP^{k,1}/dt = -\nabla_Q V(Q^{k,1})\) up to time \(t^n\).
- \(N^{n,1}_n\) RK4 operations to solve

\[
\frac{d a^{k,1}_x}{dt} = \frac{1}{2} a^{k,1}_x \text{Tr} \left( (Z^{-1})^{k,1} (\nabla_x P^{k,1} - i \nabla_x Q^{k,1} \nabla_x Q^{V^{k,1}}) \right),
\]

with

\[
\partial_z = \partial_q - i \nabla_p, \quad Z = \partial_z(Q + i P).
\]

- \(c N_q \tilde{N}_p\) operations to compute \(\phi^\epsilon_{\text{FGA}}(t^n, x^j)\) from (3), where \(c\) is a fixed integer, \(N_q\) is the cardinal of \(D_q\), and \(\tilde{N}_p\) is the cardinal of \(\tilde{D}_p\).

As a consequence the total number of operations to compute the FGA up to time \(t^n\), and in \(x^j\) is

\[
\text{Op} = \text{Op}(\phi^\epsilon_{\text{FGA}}(t^n, x^j)) \approx \sum_{(k,l) \in D_q \times \tilde{D}_p} \left( N^{w,1}_w + N^{n,1}_n + N^{p,1}_p + N^{n,1}_n \right) + c N_q \tilde{N}_p.
\]

**Two-domain approach (parallel computing).** Using the above notations, we have

\[
\phi^\epsilon_{\text{FGA}, \pm}(t^n, x^j) = \sum_{(k,l) \in D^\pm_q \times \tilde{D}_p} \left( a^{k,1}(t^n) r_\theta(Q^k, P^{k,1}) e^{\pm \left( \frac{1}{2} (z^k, x^j) + \frac{1}{4} \left| (z^k - Q^{k,1}(t^n)) \right|^2 \right) \| \delta_q \| \| \delta_p \right)
\]

(6)

where for \((k,l) \in D^\pm_q \times \tilde{D}_p\), we define \(N^{w,1}_w, N^{n,1}_n, N^{p,1}_p,\) and \(N^{n,1}_n\) as above. As a consequence, the total number of operations to update the FGA up to time \(t^n\) and in \(x^j\) is

\[
\text{Op} = \text{Op}^+ + \text{Op}^-,
\]

where

\[
\text{Op}^\pm = \text{Op}(\phi^\epsilon_{\text{FGA}, \pm}(t^n, x^j)) \approx \sum_{(k,l) \in D^\pm_q \times \tilde{D}_p} \left( N^{w,1}_w + N^{n,1}_n + N^{p,1}_p + N^{n,1}_n \right) + c N^\pm_q \tilde{N}_p,
\]

where \(N^\pm\) is the cardinal of \(D^\pm\). We observe that \(\text{Op}^+ \approx \text{Op}^-\) and then \(\text{Op}^+ \approx \text{Op}/2\). We then expect a quasi-ideal speed-up in the case of a parallel implementation of the FGA method. More generally, a parallel implementation of the FGA method on \(m\) nodes is expected to lead to a number of operations per node of the order of \(\text{Op}/m\), where \(\text{Op}\) is the total number of operations. Numerical examples for this method will be given in Section 6. We show in Section 4 that this DDM can also be useful to efficiently implement in a parallel framework the so-called FGA Schwarz Waveform Relaxation algorithm (23).
3. Basic FGA-based domain decomposition algorithm in the middle-frequency regime

In this and next sections, we are interested in the derivation of an accurate approximate SWR algorithm by using FGA formalism in the middle-frequency regime. We start our discussions with some important analogies about Transparent Boundary Conditions (TBCs).

3.1. Analogy between TBCs in the FGA formalism and pseudodifferential formalism

We denote by \( \Omega \) an open bounded domain of \( \mathbb{R}^d \). In this section, we show analogies between TBCs on \( \partial \Omega \) in the FGA formalism in the semi-classical regime \([37, 24]\) and TBCs derived from microlocal analysis beyond the semi-classical regime \([1, 6]\), which will justify the SWR methods presented below. We introduce a set \( \Omega^\varepsilon \) containing \( \Omega \), defined as \( \Omega^\varepsilon = \bigcup_{x \in \Omega} B(x, \sqrt{\varepsilon}) \), where \( B \) a ball of center \( x \) and radius \( \sqrt{\varepsilon} \). We recall that the FGA reads

\[
\psi_{\text{FGA}}^\varepsilon(t, x) = \frac{1}{(2\pi\varepsilon)^{3d/2}} \int_{\mathbb{R}^{3d}} a(t, q, p) e^{i\frac{\varepsilon}{2} \Phi(t, x, y, q, p)} \phi_0^\varepsilon(y) dy dp dq,
\]

by setting

\[
\Phi(t, x, y, q, p) = S(t, q, p) + \frac{i}{2} |x - Q(t, q, p)|^2 + P(t, q, p) \cdot (x - Q(t, q, p))
\]

\[
+ \frac{i}{2} |y - q|^2 - p \cdot (y - q).
\]

The Hamiltonian is given by

\[
H(t, Q, P) = \frac{|P|^2}{2} + V(Q, t),
\]

and in this formalism, the Hamiltonian flow is

\[
\begin{cases}
\frac{dQ}{dt} = P, \\
\frac{dP}{dt} = -\nabla_Q V(Q, t),
\end{cases}
\]

with initial data

\[
\begin{cases}
Q(0, q, p) = q, \\
P(0, q, p) = p.
\end{cases}
\]

The classical action function satisfies

\[
\frac{dS}{dt} = \frac{|P|^2}{2} - V(Q, t).
\]

In [40], outgoing wave conditions at time \( t \), for a particle \( B(Q_0(t, q, p), \sqrt{\varepsilon}) \in \Omega^\varepsilon - \Omega \) is determined according to the condition:

\[
\frac{dQ_0}{dt} \cdot n_0 > 0,
\]

where \( n_0 \) is the outward normal vector of \( \partial \Omega \) at \( Q(t, q, p) \). In that case, the contribution of particle \( Q_0 \) is removed from the wavefunction.
Let us now denote by $p_\varepsilon$ the classical symbol of the Schrödinger operator

$$P_\varepsilon = -i\varepsilon \partial_t - \frac{\varepsilon^2}{2} \Delta + V(x, t),$$

that is

$$p_\varepsilon(x, t, \xi, \omega) = \frac{\varepsilon^2}{2}|\xi|^2 + \varepsilon \omega + V(x, t).$$

We denote by $T^*(\partial\Omega)$ the cotangent bundle at the boundary. The ray propagates in the hyperbolic zone defined by

$$\mathcal{H} = \{(x_0, t_0, \xi_0, \omega_0) \in T^*(\partial\Omega) : \varepsilon \omega_0 + \frac{\varepsilon^2}{2}|\xi_0|^2/2 + V < 0\}.$$

The outgoing bicharacteristic strips (from which the TBCs (2.3) are derived) at the boundary are selected from the following Hamiltonian equations

$$\begin{align*}
\frac{dx}{ds}(s) &= \nabla_\xi p_\varepsilon(x(s), t(s), \xi(s), \omega(s)) = \varepsilon^2 \xi(s), \\
\frac{dt}{ds}(s) &= \nabla_\omega p_\varepsilon(x(s), t(s), \xi(s), \omega(s)) = \varepsilon, \\
\frac{d\xi}{ds}(s) &= -\nabla_x p_\varepsilon(x(s), t(s), \xi(s), \omega(s)) = -\nabla_x V, \\
\frac{d\omega}{ds}(s) &= -\nabla_t p_\varepsilon(x(s), t(s), \xi(s), \omega(s)) = 0,
\end{align*}$$

(8)

based on the condition at $x_0 \in \partial\Omega$, with outward normal vector $n_0$ at $\partial\Omega$

$$\frac{dx}{ds}(s) \cdot n_0 > 0.$$ (9)

Note that $\omega(s) = \omega_0$ and $t(s) = \varepsilon s + t_0$. In addition, let us recall that for

$$p_\varepsilon(x_0, t_0, \xi_0, \omega_0) = 0,$$

along the null bicharacteristic strips [25], we get

$$\frac{dp_\varepsilon}{ds}(x(s), t(s), \xi(s), \omega(s)) = 0,$$

From there, the Schrödinger operator is factorized in the hyperbolic region of the cotangent bundle $T^*(\partial\Omega)$ as the composition of an outgoing and an incoming wave operators to get the TBC (see Eq. 10). An analogy can now be deduced between (7) and (9). Indeed, from

$$P = \frac{dQ}{dt} = \frac{dQ}{ds} \frac{ds}{dt} = \frac{1}{\varepsilon} \frac{dQ}{ds},$$

and

$$-\nabla_Q V(Q, t) = \frac{dP}{dt} = \frac{dP}{ds} \frac{ds}{dt} = \frac{1}{\varepsilon} \frac{dP}{ds},$$
we deduce

\[
\begin{align*}
\frac{dQ}{ds} &= \varepsilon P; \\
\frac{dP}{ds} &= -\varepsilon \nabla Q V(Q, t).
\end{align*}
\]

We then have the following analogy

\[
\begin{align*}
x(s) &\leftrightarrow Q(s); \\
\xi(s) &\leftrightarrow \frac{1}{\varepsilon} P(s).
\end{align*}
\]

From a practical viewpoint, the OSWR algorithm is derived from TBCs based on DtN operators through the factorization. For time-independent potential \( V \) we have the exact factorization

\[
i \varepsilon \partial_t + \frac{\varepsilon^2}{2} \partial_x^2 - V = \left( \frac{\varepsilon}{\sqrt{2}} \partial_x + \sqrt{V - i \varepsilon \partial_t} \right) \left( \frac{\varepsilon}{\sqrt{2}} \partial_x - \sqrt{V - i \varepsilon \partial_t} \right).
\]

Operators \( \left( \frac{\varepsilon}{\sqrt{2}} \partial_x \pm \sqrt{V - i \varepsilon \partial_t} \right) \) incoming/outgoing wave operator (at domain boundary) and are used to define TBC. OSWR algorithm will then be here (\( V = V(x) \)) in the form

\[
\begin{align*}
i \varepsilon \partial_t \phi_{\pm}^\varepsilon &= H_\varepsilon \phi_{\pm}^\varepsilon \text{ in } (0,T) \times \Omega^\pm; \\
\left( \frac{\varepsilon}{\sqrt{2}} \partial_x \pm \sqrt{V - i \varepsilon \partial_t} \right) \phi_{\pm}^\varepsilon &= \left( \frac{\varepsilon}{\sqrt{2}} \partial_x \pm \sqrt{V - i \varepsilon \partial_t} \right) \phi_{\pm}^{\varepsilon,k-1} \text{ on } (0,T) \times \Gamma^\pm.
\end{align*}
\]

Now, \( \phi_{\pm}^\varepsilon_F(t^n, \cdot) \) defined above are precisely constructed as approximate solution, for \( t > t^n \), to

\[
\begin{align*}
\left( \frac{\varepsilon}{\sqrt{2}} \partial_x \pm \sqrt{V - i \varepsilon \partial_t} \right) \phi_{\pm}(t, x) &= 0,
\end{align*}
\]

with \( \phi_{\pm}(t^n, \cdot) = \phi_{\pm}^\varepsilon_F (t^n, \cdot) \) and \( \Omega_\eta^+ \cup \Omega_\eta^- \) and \( \Omega = \Omega_\eta^+ \cup \Omega_\eta^- \). As a consequence, the plan is to developing DDM mimicking OSWR algorithm, involving FGA-based conditions rather than TBC.

3.2. A first FGA-based domain decomposition algorithm in the middle-frequency regime

Let us again introduce the two subdomains \( \Omega^\pm_\eta \) such that \( \mathbb{R}^d = \Omega^+_\eta \cup \Omega^-_\eta \) and we denote by \( \phi_{\pm}^\varepsilon_F \) the solution in \( \Omega^\pm_\eta \). We propose now an original domain decomposition method for solving (1). More specifically, we have the equations in \( \Omega^\pm_\eta \) (2) for any Schwarz iteration \( k \geq 1 \). The OSWR approach (11) is known to be efficient in terms of Schwarz iterations \([11, 23]\). In particular, it can be proven that, in the 1d potential-free case, the exact convergence is reached in only two iterations \([23]\). However, for higher dimensions and in the presence of a varying potential (linear or nonlinear), the convergence is slower (but still clearly better than for the CSWR DDM). Several reasons explain this claim. First, the simple 1d OSWR DDM with constant potential is based on a TBC which cannot be built for more complex situations, except for some very specific ones. As a consequence, the exact operator \( B^\varepsilon_\pm \) needs to be approximated, resulting in ABCs, by assuming generally that we are working in the high-frequency regime. This last assumption allows for the derivation of accurate approximate pseudodifferential operators \( B^{\varepsilon,h}_\pm \). Next, the operators \( B^{\varepsilon,h}_\pm \) are usually (since they are DtN-like pseudodifferential operators) nonlocal in time or/and space. This
implies i) data storage issues (storage of the trace of the solution at the domain boundary, at anytime), ii) computational issues due to stability constraints and computational complexity. In addition, for smooth domain boundary, the operators $B_{T}$ are built through local coordinates change, making their derivation and implementation even more difficult. All these issues are extensively discussed in the literature (see for instance the review paper [1]) and motivate the use of alternative techniques such as the one presented below.

To derive ABCs and consequently transmission conditions, we first emphasis the fact that although $\varepsilon$ is a small parameter, we solve the quantum time-dependent Schrödinger equation. However, the transmission conditions will be derived assuming semi-classical conditions at the subdomains interfaces. Let us recall that the transmission conditions do not need to be exact but i) should be accurate enough to get a fast convergence of the Schwarz DDM and ii) should not be too computationally expensive or, at least, efficiently scalable. More specifically, we introduce a Fourier Integral Operator $I^{\varepsilon}$ and we use now the notations of [37] which are here more convenient for deriving the DDM. Let us define:

$$I^{\varepsilon}(\kappa; a) \cdot \phi(x) = \frac{1}{(2\pi\varepsilon)^{d/2}} \int_{\mathbb{R}^{d}} a(t, q, p)e^{\frac{i}{\varepsilon}\phi^{\varepsilon}(x, y, q, p)}\phi(y)dydpdq,$$

with the transformation

$$\kappa^{\varepsilon}(q, p) = (Q^{\varepsilon}, P^{\varepsilon})$$

and the phase function

$$\Phi^{\varepsilon}(x, y, q, p) = S^{\varepsilon}(q, p) + \frac{3}{2}|x - Q^{\varepsilon}|^2 + P^{\varepsilon} \cdot (x - Q^{\varepsilon}) + \frac{1}{2}|y - q|^2 - p \cdot (y - q),$$

where $S^{\varepsilon}$ is the classical action. The classical Hamiltonian flow is here defined by

$$\begin{cases}
\frac{dQ^{\varepsilon}}{dt} = P^{\varepsilon}, \\
\frac{dP^{\varepsilon}}{dt} = -\nabla_{Q}V(Q^{\varepsilon}).
\end{cases}$$

The phase evolves thanks to

$$\frac{dS}{dt} = \frac{|P|^2}{2} - V(Q, t)$$

and the amplitude as

$$\frac{da}{dt}(t, q, p) = \frac{1}{2}a(t, q, p)\text{Tr}\left(Z^{-1}(F^{\varepsilon}(q, p))\frac{dZ}{dt}(F^{\varepsilon}(q, p))\right).$$

where $Z$ is defined in (4) and $\kappa^{\varepsilon}(q, p)$ is a $C^{1}$-family of canonical transformations of the classical phase space $T^{*}(\mathbb{R}^{d})$, [37]. For $x_{\eta}^{\pm} \in \partial \Omega^{\pm}_{\eta}$ with outward normal vector $n_{\eta}^{\pm}$ and for $\phi^{\pm, \varepsilon}$, we define in $\Omega^{\pm}_{\eta}$

$$I^{\varepsilon}_{\delta}(\kappa; a) \cdot \phi(x^{\pm}) = \frac{1}{(2\pi\varepsilon)^{d/2}} \int_{D^{\pm}_{\delta}} a(t, q, p)e^{\frac{i}{\varepsilon}\phi^{\varepsilon}(x, y, q, p)}\phi(y)dydpdq,$$

where the domain $D^{\pm, \varepsilon}_{\delta}$ is

$$D^{\pm, \varepsilon}_{\delta} = \{(y, q, p) \in \Omega^{\pm}_{\eta} \times \mathbb{R}^{2d} : P^{\varepsilon}(q, p) \cdot n_{\eta}^{\pm} > 0 \quad \& \quad |x^{\pm} - Q^{\varepsilon}(q, p)| \leq \delta\}.$$
As discussed above, there is an analogy between the motion of particles $Q^{\kappa'}$ and the rays trajectories in the fully quantum picture. The operator $\mathcal{I}^{\pm,\nu}(\kappa'; a)$ is connected to $\mathcal{B}_\nu$. The method requires the knowledge of $P^{\kappa'}$, $Q^{\kappa'}$ and $a$ at anytime and for all $(p, q)$, which is here naturally a drawback. Note however that the computation of $P^{\kappa'}$, $Q^{\kappa'}$ and $a$ in parallel is trivial and discussed above. Now the first FGA-based SWR algorithm reads as follows

$$\left\{ \begin{array}{l} i\varepsilon \partial_t \phi_{\pm}^{\varepsilon,k}(t, \cdot) = H_\varepsilon \phi_{\pm}^{\varepsilon,k}(t, \cdot) \text{ in } \{ t > t^n \} \times \Omega_{\eta}^\pm, \\
\phi_{\pm}^{\varepsilon,k}(t, \cdot) = \mathcal{I}_{\delta}^{\pm}(\kappa^t; a) \cdot \phi_{\pm}^{\varepsilon,k-1}(t^n, \cdot) \text{ on } \{ t > t^n \} \times \Gamma_{\eta}^\pm. \end{array} \right.$$

(12)

We need to be a bit more specific regarding the explicit expression of the transmission condition at time say $t > t^n$. We assume that, at Schwarz iteration $k$ and time $t^n$, the solution $\phi_{\pm}^k(t^n)$ is known (in the following we will omit the index $\varepsilon$) in $\Omega_{\eta}^\pm$, as well as $S^{\kappa'}$, $Q^{\kappa'}(q, p)$, $P^{\kappa'}(q, p)$, $a(t, q, p)$, for all $(q, p) \in \Omega_{\eta} \times \Omega_P$. For $x_{\eta}^\pm \in \partial \Omega_{\eta}^\pm$ and for $t > t^n$, we define

$$\mathcal{I}_{\delta}^{\pm}(\kappa^t; a) \cdot \phi_{\pm}^k(t^n, x_{\eta}^\pm) = \frac{1}{(2\pi \varepsilon)^{3d/2}} \int_{D_{\eta}^{\pm}} a(t, q, p) w_{k,\pm}(q, p, t^n)$$

$$\times e^{i \frac{\varepsilon}{2} (S^{\kappa'}(q, p) + P^{\kappa'}(x_{\eta}^\pm - Q^{\kappa'}))} e^{-\frac{\varepsilon}{2} |x_{\eta}^\pm - Q^{\kappa'}|^2} dqdq,$$

where

$$w_{k,\pm}(q, p, t^n) = \int_{\Omega_{\eta}^{\pm}} \exp \left( - \frac{i}{\varepsilon} \cdot (p(y - q) - |x_{\eta}^\pm - q|^2) \phi_{\pm}^k(t^n, y) dy. \right.$$

From a numerical viewpoint, the transmission conditions can easily be discretized as follows. Let us assume that, for the iteration $k$ and the time $t^n$, the solutions $\phi_{\pm}^k(t^n)$ are known in $\Omega_{\eta}^\pm$, as well as $Q^{\kappa^{n+1}, k, 1}$, $Q^{\kappa^{n+1}, k, 1}$, $a^{k,1}(t^{n+1})$, where $(k, l) \in D_{\eta} \times D_P$ denote the discretization indices in $(q, p) \in \Omega_{\eta} \times \Omega_P$ for some $t^{n+1} > t^n$. Spatial indices for $\Omega_{\eta}^\pm$ are $j \in D_{\eta}^\pm$. We make the hypothesis that $x_{\eta}^\pm = x_{\eta}^j$, for some indices $j_{\eta}^\pm \in D_{\eta}^\pm$. Then, we have the approximation

$$\mathcal{I}_{\delta}^{\pm}(\kappa^{n+1}; a) \cdot \phi_{\pm}^k(t^n, x_{\eta}^\pm) \approx \frac{1}{(2\pi \varepsilon)^{3d/2}} \sum_{j \in D_{\eta}^\pm} \alpha_j(qk^1, pk^1) e^{i \frac{\varepsilon}{2} \left( S^{\kappa^{n+1}, k, 1} + P^{\kappa^{n+1}, k, 1} - (x_{\eta}^\pm - Q^{\kappa^{n+1}, k, 1}) \right)}$$

$$\times e^{-\frac{\varepsilon}{2} |x_{\eta}^\pm - Q^{\kappa^{n+1}, k, 1}|^2} |d\eta| |dp|,$$

where

$$\alpha_j(qk^1, pk^1) = |\delta y| e^{-\frac{i}{2} \cdot (y_{\eta} - qk^1) - \frac{\varepsilon}{2} |y_{\eta} - qk^1|^2}.$$

Formally the transmission conditions read

$$\phi_{\pm}^{\varepsilon,k}(t^{n+1}, x_{\eta}^\pm) = \sum_j \beta_j^{\mp}(t^n) \phi_{\pm}^{k-1}(t^n).$$
for some coefficient $\beta^\pm_j$ defined as follows

$$
\beta^\pm_j(t^n) = \frac{1}{(2\pi \varepsilon)^{d/2}} \sum_{(k,l) \in D_n \times D_p} \alpha_j (q^{k,l}, p^{k,l}) e^{i (s_k^{n+1} + p_k^{n+1} \cdot x_k - Q^{n+1} \cdot l)} - \frac{1}{2} |x_k - Q^{n+1} \cdot l|^2 |\delta q| |\delta p|
$$

Remark that the transmission condition does not modify the condition number of the matrix constructed from the interior scheme which is a priori not true for OSWR. In order to have an overall efficient algorithm, the computation of coefficients $\beta^\pm_j(t^n)$ should be very fast. However, the overall computational cost of the above approach is expected to be high due to reconstruction of FGA at each time iteration. Although it could be a nice method to explore, we will study less computationally complex methods, but still based on FGA.

3.3. FGA/Dirichlet-based domain decomposition algorithm

In the following, we use the notation $P_\varepsilon := \iota \varepsilon \partial_t + \varepsilon^2 \Delta/2 - V$ in $\mathbb{R}^d$, where $\varepsilon$ is a fixed small parameter and $\Psi_\varepsilon$ denotes the solution to $P_\varepsilon \cdot \Psi_\varepsilon = 0$ with $\Psi_\varepsilon(0, \cdot) = \phi_0(\cdot)$. We propose a series of lemmas leading to a convergent Schwarz waveform relaxation algorithm in Section 4. We denote again by $\phi^\pm_\varepsilon$ the solution in $\Omega^\pm_\eta$, where $\Omega^+_\eta \cup \Omega^-_\eta = \mathbb{R}^d$ and $\eta$ is a fixed small positive parameter. We introduce $\Gamma^\pm_\eta = \partial \Omega^\pm_\eta$, which are assumed to be smooth. For $d = 1$, we choose $\Omega^-_\eta = (-\infty, \eta/2)$ and $\Omega^+_\eta = (-\eta/2, +\infty)$.

Lemma 3.1. Let us consider an initial data $\phi_0$, and the following Schwarz method, for $k \geq 1$,

$$
\begin{cases}
    P_\varepsilon \cdot \phi^\pm_k(t, \cdot) = 0, & \text{on } (0, T) \times \Omega^\pm_\eta, \\
    \phi^\pm_k(0, \cdot) = \phi_0(\cdot), & \text{on } \Omega^\pm_\eta, \\
    \phi^\pm_k(t, \cdot) = \Psi_\varepsilon(t, \cdot), & \text{on } (0, T) \times \Gamma^\pm_\eta.
\end{cases}
$$

Then, for any $k \geq 1$, $\phi^\pm_k \equiv \Psi_{e|\Omega^\pm_\eta}$. In other words, the Schwarz algorithm is convergent in one iteration.

Proof. We first notice that the compatibility condition is trivially satisfied

$$
\phi^\pm_k(0, x^\pm_\eta) = \Psi_\varepsilon(0, x^\pm_\eta) = \phi_0(x^\pm_\eta), \quad \forall x^\pm_\eta \in \Gamma^\pm_\eta.
$$

We set

$$
e^\pm_k := \Psi_{e|\Omega^\pm_\eta} - \phi^\pm_k.
$$

Naturally we have

$$
\begin{cases}
    P_\varepsilon \cdot e^\pm_k(t, \cdot) = 0, & \text{on } (0, T) \times \Omega^\pm_\eta, \\
    e^\pm_k(0, \cdot) = 0, & \text{on } \Omega^\pm_\eta, \\
    e^\pm_k(t, \cdot) = 0, & \text{on } (0, T) \times \Gamma^\pm_\eta.
\end{cases}
$$

We then easily deduce that for any $k \geq 1$, $e^\pm_k \equiv 0$, that is $\phi^\pm_k \equiv \Psi_{e|\Omega^\pm_\eta}$. $\square$

We also have the following result.

Lemma 3.2. Assume that $V$ is a subquadratic potential in $C^\infty(\mathbb{R}^d, \mathbb{R})$ and $\phi_0 \in L^2(\mathbb{R}^d, \mathbb{C})$. Then for any $\phi_0$, the Schwarz algorithm, for $k \geq 1$,

$$
\begin{cases}
    P_\varepsilon \cdot \phi^\pm_k(t, \cdot) = 0, & \text{on } (0, T) \times \Omega^\pm_\eta, \\
    \phi^\pm_k(0, \cdot) = \phi_0(\cdot), & \text{on } \Omega^\pm_\eta, \\
    \phi^\pm_k(t, \cdot) = I^\varepsilon(a\cdot) \cdot \phi_0(t, \cdot), & \text{on } (0, T) \times \Gamma^\pm_\eta.
\end{cases}
$$

10
is not convergent.

**Proof.** Denoting by $\mathcal{I}^\varepsilon(\kappa^T; a) \cdot \phi_0$ the FGA of $\Psi_e$, it is proven in [37] that at anytime $T > 0$

$$\sup_{t \in (0, T)} \| \Psi_e(t, \cdot) - \mathcal{I}^\varepsilon(\kappa^T; a) \cdot \phi_0\|_{L^2} \leq C(T)\varepsilon\|\phi_0\|_{L^2}.$$

In fact, this result can even be improved with an estimate in $\varepsilon^N$, considering an order $N$ FGA. Thus, at anytime $t \in (0, T)$ and for any $x^\pm_n \in \Gamma^\pm_n$, we have

$$\Psi_e(t, x^\pm_n) - \mathcal{I}^\varepsilon(\kappa^T; a) \cdot \phi_0(x^\pm_n) = C(t, x^\pm_n)\varepsilon\eta,$$

where $C(t, x^\pm_n)$ is a positive real number. Denoting again $e^k = \Psi_e|_{\Omega^k} - \phi^k$, we have

$$\begin{cases}
  P_\varepsilon \cdot e^k(t, \cdot) = 0, & \text{on } (0, T) \times \Omega^\pm_n, \\
  e^k(0, \cdot) = 0, & \text{on } \Omega^\pm_n, \\
  e^k(t, \cdot) = C(t, \cdot)\varepsilon\eta, & \text{on } (0, T) \times \Gamma^\pm_n.
\end{cases}$$

As a consequence, the error is a priori $k$-independent and the algorithm is then non-convergent, although for small $\varepsilon$ and $\eta$, the error is expected to be small. This method will be numerically tested in Section 6. □

4. Herman-Kluk Schwarz Waveform Relaxation (HKSWR) algorithm in the middle-frequency regime

In order to derive a convergent Schwarz method, we need to introduce a $k$-dependence in the RHS of the transmission condition. In addition, as the FGA accurately computes the amplitude of the wavefunction on coarse grids but not the phase, we then propose to use the modulus of FGA, but not its phase. We will refer as Herman-Kluk Schwarz Waveform Relaxation methods (HKSWR).

For $\gamma \in [0, 1]$, we now introduction an important function, $F_\gamma$, defined on $F_\gamma : \mathbb{C} \times \mathbb{C} \rightarrow \mathbb{R}$, by all $(z_1, z_2) \in \mathbb{C}^2$

$$F_\gamma(z_1, z_2) = \begin{cases}
  z_2 \left| \frac{z_1}{z_2} \right|^\gamma, & \text{if } z_2 \neq 0, \\
  0, & \text{if } z_2 = 0
\end{cases} \quad (15)$$

Note that for any $(z_1, z_2) \in \mathbb{C} \times \mathbb{C}^*$: $F_1(z_1, z_2) = |z_1| \exp(i \arg(z_2))$.

4.1. Preliminary remarks

We start this section by the presentation of preliminary remarks that leads us to a convergent version of the SWR in the following subsection. For $k \geq 1$, for any initial data, we consider the following iterative scheme

$$\begin{cases}
  P_\varepsilon \cdot \phi^k_{\pm}(t, \cdot) = 0, & \text{on } (0, T) \times \Omega^\pm_n, \\
  \phi^k_{\pm}(0, \cdot) = \phi_0(\cdot), & \text{on } \Omega^\pm_n, \\
  \phi^k_{\pm}(t, \cdot) = F_1(\mathcal{I}^\varepsilon(\kappa^T; a) \cdot \phi_0(\cdot), \phi^{k-1}_{\pm}(t, \cdot)), & \text{on } (0, T) \times \Gamma^\pm_n.
\end{cases} \quad (16)$$
Then, writing the error equations, ones gets
\[
\begin{align*}
P_\varepsilon \cdot e_{\pm}^k (t, \cdot) &= 0, & \text{on } (0, T) \times \Omega^\pm, \\
e_{\pm}^k (0, \cdot) &= 0, & \text{on } \Omega^\pm, \\
e_{\pm}^k (t, \cdot) &= \Psi_{e|\Omega^\pm} (t, \cdot) - \mathcal{F}_1 \left( \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (\cdot), \phi_{k}^{k-1} (t, \cdot) \right), & \text{on } (0, T) \times \Gamma^\pm.
\end{align*}
\]

Now, since for \( x^\pm_\eta \in \Gamma^\pm_\eta \): \( \Psi_{e} (t, x^\pm_\eta) = \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta) + R(t, x^\pm_\eta) \) for some regular function \( R \), we deduce that, when defined:
\[
\begin{align*}
e_{\pm}^k (t, x^\pm_\eta) &= \left| \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta) \right| \left[ \exp \left( i \arg \left( \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta) \right) \right) \\
&\quad - \exp \left( i \arg \left( \phi_{k}^{k-1} (t, x^\pm_\eta) \right) \right) \right] + R(t, x^\pm_\eta) \\
&= \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta) \left[ 1 - \exp \left( i \arg \left( \frac{\phi_{k}^{k-1} (t, x^\pm_\eta)}{\mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta)} \right) \right) \right] + R(t, x^\pm_\eta).
\end{align*}
\]

By construction, we can state that, for any \( x^\pm_\eta \in \Gamma^\pm_\eta \),
\[
\begin{align*}
\left| \phi_{k}^\pm (t, x^\pm_\eta) \right| &= \left| \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta) \right|, \\
\arg \left( \phi_{k}^\pm (t, x^\pm_\eta) \right) &= \arg \left( \phi_{k}^{k-1} (t, x^\pm_\eta) \right).
\end{align*}
\]

For fixed \( \varepsilon \), the Schwarz algorithm (16) is then not convergent as
\[
\left| \Psi_{e} (t, x^\pm_\eta) \right| = \left| \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta) + R(t, x^\pm_\eta) \right| \neq \left| \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (x^\pm_\eta) \right|.
\]

By construction, we could try to locate the interface in a region where \( R(t, x^\pm_\eta) \) is small, but unfortunately, this is a priori difficult to do. Algorithm (16) is then not convergent and new algorithms have to be derived.

### 4.2. Convergent HKSWR algorithm

For \( \varepsilon \) fixed, we introduce a function \( \gamma_\varepsilon : \mathbb{N}^* \rightarrow [0, 1] \), which will be precised below. At iteration \( k \), we consider the following algorithm:
\[
\begin{align*}
P_\varepsilon \cdot \phi_{k}^\pm &= 0, & \text{on } (0, T) \times \Omega^\pm, \\
\phi_{k}^\pm (0, \cdot) &= \phi_0 (\cdot), & \text{on } \Omega^\pm, \\
\phi_{k}^\pm (t, \cdot) &= \left| \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (\cdot) \right|^{\gamma_\varepsilon(k)} \phi_{k}^{k-1} (t, \cdot), & \text{on } (0, T) \times \Gamma^\pm.
\end{align*}
\]

We notice that (17), is defined for nonzero \( \phi_{k}^{k-1} (t, \cdot) \). In order to get a rigorous definition, we make use of Function \( \mathcal{F}_\gamma \) defined in (15), and (17) is rewritten:
\[
\begin{align*}
P_\varepsilon \cdot \phi_{k}^\pm &= 0, & \text{on } (0, T) \times \Omega^\pm, \\
\phi_{k}^\pm (0, \cdot) &= \phi_0 (\cdot), & \text{on } \Omega^\pm, \\
\phi_{k}^\pm (t, \cdot) &= \mathcal{F}_{\gamma (k)} \left( \mathcal{I}^\varepsilon (\kappa^i; a) \cdot \phi_0 (\cdot), \phi_{k}^{k-1} (t, \cdot) \right), & \text{on } (0, T) \times \Gamma^\pm.
\end{align*}
\]
We now expect that i) the contribution from the FGA will lead to a small error after a few Schwarz iterations and ii) the contribution $\phi_{k+}(t, x_{\eta}^\pm)$ will ensure the convergence of the overall algorithm. The smaller $\varepsilon$, the better is the FGA, we then expect that the reconstructed solution will be close to the reference solution after a few Schwarz iterations. To ensure the convergence of the algorithm, we then switch to the CSWR algorithm. More specifically, this leads to construct $\gamma_\varepsilon$ rapidly convergent to 0 as a function of $k$. For large $\varepsilon$, the FGA is not a priori a suitable approximation of the solution and therefore using the present method is irrelevant. We now state

**Lemma 4.1.** Assuming that for all positive $\varepsilon$, $\gamma_\varepsilon(k) \to k \to \infty 0$ then (18) is convergent.

**Proof.** The proof is decomposed into three steps.

**Step 1.** We first analyze the following algorithm

\[
\begin{cases}
P_e \cdot \phi_{k+}^e(t, \cdot) = 0, & \text{on } (0, T) \times \Omega_\eta^\pm, \\
\phi_{k+}^e(0, \cdot) = \phi_0(\cdot), & \text{on } \Omega_\eta^\pm, \\
\phi_{k+}^e(t, \cdot) = \mathcal{F}_{\gamma_\varepsilon}(\Psi_e(t, \cdot), \phi^e_{k-1}(t, \cdot)), & \text{on } (0, T) \times \Gamma_\eta^\pm.
\end{cases}
\]

(19)

Then, for nonzero $\phi^e_{k-1}(t, \cdot)$ at $\Gamma_\eta^\pm$, one gets

\[
\begin{cases}
P_e \cdot e_{k+}^e(t, \cdot) = 0, & \text{on } (0, T) \times \Omega_\eta^\pm, \\
e_{k+}^e(0, \cdot) = 0, & \text{on } \Omega_\eta^\pm, \\
e_{k+}^e(t, \cdot) = \Psi_e(t, \cdot) \left(1 - \left| \frac{\Psi_e(t, \cdot)}{\phi^e_{k-1}(t, \cdot)} \right| \gamma_\varepsilon(k) \right) + \left| \frac{\Psi_e(t, \cdot)}{\phi^e_{k-1}(t, \cdot)} \right| \gamma_\varepsilon(k) e_{k-1}^e(t, \cdot), & \text{on } (0, T) \times \Gamma_\eta^\pm.
\end{cases}
\]

(20)

It is easy to show that for $\lim_{k \to \infty} \gamma_\varepsilon(k) = 0$, if $e_{k+}^e$ is convergent, then its limit is 0.

**Step 2.** We now consider, for $x_{\eta}^\pm \in \Gamma_\eta^\pm$,

$\phi_{k+}(t, x_{\eta}^\pm) = \mathcal{F}_{\gamma_\varepsilon}(\mathcal{I}^\tau(\kappa^\tau; a) \cdot \phi_0(x_{\eta}^\pm), \phi^e_{k-1}(t, x_{\eta}^\pm))$,

where $\Psi_e(t, x_{\eta}^\pm) = \mathcal{I}^\tau(\kappa^\tau; a) \cdot \phi_0(x_{\eta}^\pm) + R(t, x_{\eta}^\pm)$ for some regular function $R$. Recall first that the CSWR algorithm, corresponding to $\gamma_\varepsilon(k) = 0$ for all $k$

\[
\begin{cases}
P_e \cdot \phi_{k+}(t, \cdot) = 0, & \text{on } (0, T) \times \Omega_\eta^\pm, \\
\phi_{k+}(0, \cdot) = \phi_0(\cdot), & \text{on } \Omega_\eta^\pm, \\
\phi_{k+}(t, \cdot) = \phi^e_{k-1}(t, \cdot), & \text{on } (0, T) \times \Gamma_\eta^\pm.
\end{cases}
\]

(21)

is convergent. In 1D and for $x_{\eta}^\pm = \pm \eta/2$, we can even estimate the convergence rate [10]

$C_\eta = \sup_{\tau \in \mathcal{H}_\tau} |L_\eta(\tau)|$,

for $|\tau| \gg 1$, where $\mathcal{H}_\tau$ is the hyperbolic zone and

\[
L_\eta(\tau) \approx \exp \left( -\eta \sqrt{-2\tau} - \frac{1}{\sqrt{-2\tau}} \int_{-\eta/2}^{\eta/2} V(y) dy \right).
\]

(22)
In the case \( \gamma_\varepsilon(k) = 1 \) for all \( k \), we get Algorithm (13), which is shown to be non-converging, but nevertheless leads to a bounded (and small) error, \( e_k^\varepsilon \) after a few iterations.

**Step 3.** A sufficient condition to ensure convergence is to construct \( \gamma_\varepsilon \) such that i) \( \lim_{k \to \infty} \gamma_\varepsilon(k) = 0 \) or ii) such that for some \( k_0 \), \( \gamma_\varepsilon(k) = 0 \), for all \( k > k_0 \). Indeed, in that case, (18) degenerates into CSWR, which is known to be convergent, with a contraction factor given by (22).

**Remark 4.1.** For \( k \) small, it is natural to choose \( \gamma_\varepsilon(k) \) close to 1. In that case, for \( \varepsilon \) small enough, according to Lemma 3.2, within the first Schwarz iterations, the reconstructed solution will be close (but not equal) to the exact one. For larger \( k \), due to Lemmas 3.2 and 4.1, it is natural to take \( \gamma_\varepsilon \) close to 0 to ensure the convergence.

We now derive a DDM, which will refer to as Herman-Kluk Schwarz Waveform Relaxation (HKSWR) method. Consider

\[ P_\varepsilon \cdot \psi^\varepsilon(t, x) = 0, \ x \in \mathbb{R}^d, \ t \in (0, T), \]

we define a decreasing function \( \gamma_\varepsilon : \mathbb{N}^* \mapsto [0, 1] \) such that \( \gamma_\varepsilon(1) = 1 \), whose features will be detailed in the following subsection, and consider the following Schwarz algorithm

\[
\begin{align*}
P_\varepsilon \cdot \phi^k_\pm(t, \cdot) &= 0, \quad \text{on} \quad (0, T) \times \Omega^\pm_\eta, \\
\phi^k_\pm(0, \cdot) &= \phi_0(\cdot), \quad \text{on} \quad \Omega^\pm_\eta, \\
\phi^k_\pm(t, \cdot) &= \mathcal{F}_{\gamma_\varepsilon(k)}(\mathcal{I}_\varepsilon^\varepsilon(k^\varepsilon; a) \cdot \phi_{0}(\cdot), \phi_{k-1}(t, \cdot)), \quad \text{on} \quad (0, T) \times \Gamma^\pm_\eta,
\end{align*}
\]

where \( \varepsilon \) is a small parameter, which in principle satisfies \( \varepsilon \leq \eta \).

Notice that \( \varepsilon \) is used as a free parameter in the transmission conditions. In practice, \( \varepsilon \) does not need to be equal to \( \varepsilon \) since the convergence is finally ensured by the CSWR method. However, the more accurate the FGA, the better (say for \( \varepsilon \approx \eta \ll 1 \)). The ideal choice for \( \varepsilon \), in combination with the function \( \gamma_\varepsilon \), is then the one that lead to the fastest convergence of the DD algorithm. This last point is discussed below.

**Remark 4.2.** In the HKSWR algorithm, the transmission conditions could, in principle, have been chosen of the form

\[ \phi^k_\pm(t, \cdot) = \left[ \mathcal{I}_\varepsilon^\varepsilon(k^\varepsilon; a) \cdot \phi_0(\cdot) \right]^{\gamma_\varepsilon(k)} \left[ \phi^\varepsilon_{k-1}(t, \cdot) \right]^{1-\gamma_\varepsilon(k)}. \]

However, although FGA accurately computes the phase of the solution to (1) for small \( \varepsilon \), the interest of HKSWR comes from the fact that the amplitude of the FGA can be efficiently computed on coarse grids (in space and frequency) compared to the interior scheme grid. In that case, at high-frequency and for coarse grids, we consider the amplitude of the FGA rather than its phase, the latter requiring a very fine grid for an discrete accurate representation of \( \exp \left( \frac{i}{\varepsilon} \left( S + P \cdot (x - Q) \right) \right) \).

### 4.3. Derivation of the \( \gamma_\varepsilon \)-function

This subsection is devoted to the construction of the function \( \gamma_\varepsilon \) allowing the coupling between the pure FGA and Dirichlet transmission conditions. We begin our discussion with some important remarks. We first recall that, as discussed above, the algorithm based on pure FGA conditions (14) is not convergent and as a consequence \( \gamma_\varepsilon \) should not be taken identically equal to 1. Now, from the analysis of the CSWR method [23], it is possible to estimate its rate of convergence thanks to a fixed point contraction factor. For a constant potential \( V \), we have:

\[ C_{\varepsilon, \eta} = \sup_{r \in \mathcal{H}} \left| \exp \left( -\eta \sqrt{-2\tau} - \eta V / \sqrt{-2\tau} \right) \right| \approx \exp \left( -\eta \sqrt{2} / \varepsilon - \eta V \sqrt{\varepsilon} / 2 \right), \]

where \( \tau \) is
the covariable of $t$ and $\eta$ the size of the overlapping region (in the one dimensional case). The error estimate at Schwarz iteration $k$ is of the form $C_{t,\eta}^{2k}E_0$, where $E_0$ denotes the (known) initial error. As a consequence, we can estimate the number of CSWR iterations $k_{\varepsilon,\eta}$ to reach a given error threshold $\varepsilon_T$

$$k_{\varepsilon,\eta} \approx \frac{\log (\varepsilon_T / E_0)}{2 \log C_{t,\eta}} = \frac{\sqrt{\varepsilon}}{\eta \sqrt{2(2 + \varepsilon V)}} \log \left( \frac{E_0}{\varepsilon_T} \right).$$

As a part of this discussion, we now compare the rate of convergence for different HKSWR approaches: i) $\gamma_{\varepsilon}(k) = 0$, $\forall k \geq 1$ (CSWR), ii) $\gamma_{\varepsilon}(k) = (-\alpha_{\varepsilon}(k-1)^2)$, iii) $\gamma_{\varepsilon}(k) = 1$, $\forall k \geq 1$ (pure FGA), iv) $\gamma_{\varepsilon}(k) = 1$, $\forall k \leq k_0$ & $\gamma_{\varepsilon}(k) = 0$, $\forall k > k_0$. The corresponding functions $\gamma_{\varepsilon}$ are plotted on Fig. 1. We compare the rate of convergence of the HKSWR algorithm for the different functions $\gamma_{\varepsilon}$ defined above (Fig. 1). This comparison will help us to validate the above analysis and to carefully select $\gamma_{\varepsilon}$. The numerical framework is the same as Test I from Subsection 6.2.1, with in addition

- Test 1: $\varepsilon = 2^{-3}, \tilde{\varepsilon} = 2^{-6}, \alpha_{\varepsilon} = 5 \times 10^{-2}$ and $k_0 = 1$ and 5. The other numerical data are the same as Subsection 6.2.1.

- Test 2: $\varepsilon = 2^{-5}, \tilde{\varepsilon} = 2^{-10}, \alpha_{\varepsilon} = 10^{-2}$, and $k_0 = 1$ and 10. The other numerical data are the same as Subsection 6.2.2.

The numerical results suggest that the higher the frequency (the smaller $\varepsilon$), the smaller $\alpha_{\varepsilon}$ should be chosen (see Figs. 2). This also illustrates the fact that the higher the frequency, the more precise the FGA, and the closer to 1, $\gamma_{\varepsilon}$ should be taken. When $\varepsilon \to 0$, the FGA tends to the exact solution to (1) and as proven in Lemma 13, only 1 iteration is necessary to converge. In other words, for fixed $k$, when $\varepsilon \to 0$, $\alpha_{\varepsilon}$ goes to 0 and $\gamma_{\varepsilon}$ tends to 1. Inversely, the larger $\varepsilon$, the rougher the FGA, the closer to 0, $\gamma_{\varepsilon}$ should be. Naturally, at low-frequency, the HKSWR method is useless, and it degenerates in CSWR. Note also that, as expected, the full FGA approach ($\gamma_{\varepsilon}(k) = 1$ for all $k$) is shown to be numerically non-convergent.

To summarize, $\gamma_{\varepsilon} : \mathbb{N}^* \to [0, 1]$ is constructed according the following rules: i) $\gamma_{\varepsilon}$ is decreasing, ii) $\gamma_{\varepsilon}(1) = 1$, iii) and for fixed $\varepsilon$, $\lim_{k \to \infty} \gamma_{\varepsilon}(k) = 0$ or such that for some $k_0$, $\gamma_{\varepsilon}(k) = 0$, for all $k \geq k_0$, ensuring the convergence of the HKSWR method. A natural choice is typically as follows:

$$\gamma_{\varepsilon}(k) = \exp \left(-\alpha_{\varepsilon}(k-1)^2\right),$$

![Figure 1: Various functions $\gamma_{\varepsilon}$.](image-url)
where $\alpha_{\varepsilon} \in [0, 1)$ is an increasing function of $\varepsilon$ and goes to 0 when $\varepsilon$ tends toward 0.

### 4.4. Numerical analysis

This section is devoted to the numerical analysis of the domain decomposition method described in Section 4.2, for the time-dependent Schrödinger equation at high-frequency in quantum regime. The chosen scheme for the Schrödinger equation is a Crank-Nicolson scheme which is known to be unconditionally stable on uniform grids and Dirichlet boundary conditions. On bounded domains, transparent (and high-order absorbing) boundary conditions are derived from DtN-like or NtD-like operators [6], [1]. From these TBCs and ABCs, it is possible to derive transmission conditions for DDM, [17], [16], [20], [23], [11]. Unfortunately, except in simple situations [4] the complexity of the (nonlocal pseudodifferential absorbing or transmission) conditions leads to computational issues, in particular linear stability and data storage issues (storage of the solution at the boundary at any time). Algorithm (23) is based on FGA, and then requires the numerical solution to ODE and numerical integration. Stable and accurate solvers can easily be derived, as the transmission conditions (23) preserve the unconditional stability of the Crank-Nicolson scheme on uniform grids. This is a simple consequence of the Dirichlet-based transmission conditions. As discussed in Section 2, parallelizing FGA is trivial and can be theoretically realized with perfect speed-up. Note in addition that this choice of boundary/ transmission conditions does not deteriorate the conditioning of the sparse matrices coming typically from the semi-implicit scheme. This is not necessarily true for pseudodifferential boundary/ transmission conditions [28]. As a consequence, using Algorithm (23) and a Crank-Nicolson scheme, for parallel DD computing is theoretically very efficient, despite the computational cost of FGA (efficiently parallelizable) and easy to derive and implement on any unstructured domain in any dimension.

We conclude this discussion by an important remark regarding the parallel computation of the HKSWR algorithm.

**Remark 4.3.** As discussed in Section (2), FGA parallel computation can be very efficiently implemented in the semi-classical regime, by decomposing in frequency the FGA integrals. This feature can still be used in the HKSWR framework for the middle/high frequency regime, in order to make the HKSWR algorithm fast, despite the cost of the FGA computation. This will require two levels of parallelization, one in space (interior scheme) and one in frequency (transmission conditions). To analyze the gain, we propose to work in the same framework as in Section 2. We have shown in Section 2, that the computational complexity to determine a FGA at a given point $x^{j}$, $\text{Op}(\hat{\phi}_{\text{FGA}}(t^{n}, x^{j}))$ was given by (5). On a two frequency subdomains $D_{a}^{+} \times D_{p}$, this
can be performed in roughly \( \text{Op}(\varphi_{\text{FGA}}(t^n, x^j)) / 2 \). We now denote by \( D^\pm \) two real space domains used in the HKSWR, with \( N^\pm \) degrees of freedom. Each time iteration requires about \( \text{O}(N^\pm) \alpha^\pm \) operations, for some \( \alpha^\pm > 1 \). Assuming that the overlapping region \( \Gamma \) involves \( M \) nodes, a parallel implementation of FGA on two subdomains requires \( \sum_{x^j \in \Gamma} \text{Op}(\varphi_{\text{FGA}}(t^n, x^j)) / 2 \). As a consequence, unlike the OSWR algorithm, the transmission condition computation based for HKSWR is scalable and does not require the storage of the solution at the domain boundary for all time. Said differently, for fixed \( M \), the nonlocal transmission conditions based on TBCs (OSWR) requires a number of operations, independent of the number of processors \( p \), except if elaborated parallelism is developed, while the number of operations for computing the HKSWR transmission conditions, is trivially almost perfectly scalable. The consequence is that the computation of FGA-based transmission condition, decreases linearly with the number of processors.

5. Geometric Optics Schwarz Waveform Relaxation (GOSWR)

Although FGA allows for an accurate approximation of the solution \( \psi^\varepsilon \) solution to Equation (1), i.e. a fast convergence of the HKSWR for small \( \varepsilon \), we can however argue that it suffers from a large computational complexity. As a consequence, the overall complexity of the DDM can certainly be improved by using a rougher but also cheaper approximation to \( \psi^\varepsilon \). It is however important to recall at this stage that, for \( \varepsilon \ll 1 \), which corresponds to the semi-classical regime, a full FGA approach for solving (1), and for which DDM is straightforward in frequency. The expectation here is, beyond the semi-classical regime, to make decrease the convergence rate and accelerating each Schwarz iteration computation.

5.1. Geometric Optics Schwarz Waveform Relaxation (GOSWR) algorithm

A natural approach in this goal is to use a Geometric Optics Approximation (GOA) rather than a FGA. We recall that

\[
\psi^\varepsilon \sim_{\varepsilon \to 0} \psi^{\varepsilon}_{\text{GOA}} = a^\varepsilon e^{i S(t, x) / \varepsilon},
\]

where one gets the asymptotic expansion

\[
a^\varepsilon \sim a_0 + \varepsilon a_1 + \varepsilon^2 a_2 + \cdots.
\]

The phase function \( S \) satisfies a Hamilton-Jacobi Equation (HJE)

\[
\partial_t S + \frac{1}{2} |\nabla S|^2 + V = 0,
\]

and the leading order term of \( a \), denoted by \( a_0 \), satisfies the transport equation

\[
\partial_t a_0 + \nabla S \cdot \nabla a_0 + \frac{a_0}{2} \Delta S = 0.
\]

Higher order terms \( (n \leq N) \) can easily be iteratively computed (if necessary) from

\[
\partial_t a_n + \nabla S \cdot \nabla a_n + \frac{1}{2} a_n \Delta S = \Delta a_{n-1},
\]

and an approximate GOA \( \psi^{\varepsilon, N}_{\text{GOA}} \) can be constructed as \( e^{i S(t, x) / \varepsilon} \sum_{l=0}^{N} a_l(t, x) \). Note that this superposition of WKB solutions is only valid when the solutions to HJE are smooth. Beyond caustic appearance, new frequency scales appear and multi-valued solutions should be obtained. However, from a practical point of view, we do not expect any difficulty since GOA will only be applied on a sufficiently short time (typically \( \Delta t \)) avoiding then caustics.
The so-called Geometric Optics Schwarz Waveform Relaxation (GOSWR) scheme consists now in solving

\[
\begin{align*}
P_z \cdot \phi^k_\pm &= 0, & \text{on } (0, T) \times \Omega_t^\pm, \\
\phi^k_\pm(0, \cdot) &= \phi_0(\cdot), & \text{on } \Omega_t^\pm, \\
\phi^k_\pm(t, \cdot) &= \mathcal{F}_{\gamma_r(k)}(\psi_{a,N}^{\pm}(t, \cdot), \phi^{k-1}_\pm(t, \cdot)), & \text{on } (0, T) \times \Gamma_t^\pm,
\end{align*}
\]

which is similar to the algorithm (23), except that FGA is now replaced by GOA. Concretely, exact the GOA can be constructed by using the method of characteristics. For a non-zero potential, a numerical solver which is similar to the algorithm (23), except that FGA is now replaced by GOA. Concretely, exact the

\[
\phi^k_\pm(0, \cdot) = \phi_0(\cdot),
\]

which is used for computing for GOA can be performed with an ideal speed-up.

We consider a one-dimensional problem (d = 1). We determine \(\phi_{\text{FGA}}\) at time \(T = 1\), for \(\Delta t = 10^{-5}\), \(\varepsilon = 2^{-9}\) and the spatial domain of computation (−40, 40). The FGA is computed following the procedure described above with RK4 scheme for simulating the particle motions. Details can also be found in [40]. We represent in Fig. 3, \(\phi_{\text{FGA}}\), \(\phi_{\text{FGA,+}}\), \(\phi_{\text{FGA,-}}\) at final time \(T = 1\), corresponding respectively to the total FGA, the contribution from the first subdomain in \(q (\Omega_q^+\) and \(\Omega_q^-\) and which are respectively defined in (3) and (6). Naturally we have \(\phi_{\text{FGA}} = \phi_{\text{FGA,+}} + \phi_{\text{FGA,-}}\). The complexity to compute \(\phi_{\text{FGA}}\) on 1 domain is as expected roughly twice the one for computing \(\phi_{\pm,\text{FGA}}\) on 2 disjoint subdomains. The computational times are summarized in Table 1, and confirm that parallel computing for FGA can be performed with an ideal speed-up.

6. Numerical tests

This section is devoted to the numerical illustration of the DDM built in the previous sections, for both the high-frequency and middle-frequency regimes.

6.1. High-frequency regime

We first validate the domain decomposition approach presented in Section 2 in the semi-classical regime. We consider a one-dimensional problem (d = 1). Tests in higher dimension are actually not required since the parallelism structure is similar to the one-dimensional case. We choose the following initial data and potential

\[
\phi_0(x) = \exp \left(-25x^2 + 1(x^2 + 3x)/\varepsilon\right), \quad V(x) = \exp(-0.2x^2).
\]

We determine \(\phi_{\text{FGA}}\) at time \(T = 1\), for \(\Delta t = 10^{-5}\), \(\varepsilon = 2^{-9}\) and the spatial domain of computation (−40, 40). The FGA is computed following the procedure described above with RK4 scheme for simulating the particle motions. Details can also be found in [40]. We represent in Fig. 3, \(\phi_{\text{FGA}}\), \(\phi_{\text{FGA,+}}\), \(\phi_{\text{FGA,-}}\) at final time \(T = 1\), corresponding respectively to the total FGA, the contribution from the first subdomain in \(q (\Omega_q^+\) and \(\Omega_q^-\) and which are respectively defined in (3) and (6). Naturally we have \(\phi_{\text{FGA}} = \phi_{\text{FGA,+}} + \phi_{\text{FGA,-}}\). The complexity to compute \(\phi_{\text{FGA}}\) on 1 domain is as expected roughly twice the one for computing \(\phi_{\pm,\text{FGA}}\) on 2 disjoint subdomains. The computational times are summarized in Table 1, and confirm that parallel computing for FGA can be performed with an ideal speed-up.
Table 1: CPU time comparison

<table>
<thead>
<tr>
<th>FGA domain</th>
<th>CPU time (in sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contribution Domain 1 ($\phi_{\text{FGA},+}$)</td>
<td>34.1</td>
</tr>
<tr>
<td>Contribution Domain 2 ($\phi_{\text{FGA},-}$)</td>
<td>35.2</td>
</tr>
<tr>
<td>One unique domain ($\phi_{\text{FGA}}$)</td>
<td>70.4</td>
</tr>
</tbody>
</table>

6.2. Middle-frequency regime

This subsection is devoted to numerical tests for the high-frequency regime as described in Section 4. In the following, we compare the CSWR (21), OSWR (11), HKSWR (23) and GOSWR methods (24). For solving the particle and phase evolutions, RK4 is used in HKSWR and Heun’s scheme in GOSWR.

6.2.1. Test I

We consider the one-dimensional Schrödinger equation ($d = 1$)

$$i\varepsilon \frac{\partial \psi}{\partial t} = -\frac{\varepsilon^2}{2} \Delta \psi(t, x) + V(x)\psi(t, x), \quad x \in \Omega = (-2, 2), \quad t \in (0, T),$$

with $\varepsilon = 2^{-3}$. We fix the initial data and potential to

$$\psi_0(x) = \exp \left( -25x^2 + i(x^2 + 3x)/\varepsilon \right), \quad V(x) = 10 \exp(-0.2x^2).$$

The Schrödinger equation is discretized with a second order finite difference scheme on $\Omega^\mp_\eta$. The parameters for the simulation are as follow. Respective space and time steps are $\Delta x = 2^{-5}$, $\Delta t = 5 \times 10^{-3}$. The final physical time is $T = 0.1$. The overlapping region is reduced to $\eta = \Delta x$, and more specifically $\Omega^-_\eta = (-2, \Delta x)$ and $\Omega^+_\eta = (0, 2)$. The numerical simulations are performed by choosing $\bar{\varepsilon} = 2^{-6}$ and $\gamma_\varepsilon(k) = \exp \left( -(k - 1)^2/20 \right)$, where $k$ refers to as the Schwarz iteration.
In Figs. 4, we report, with respect to the Schwarz iterations, i) the $\ell^2$-norm in time of $\phi_+(t, \Delta x) - \phi_-(t, \Delta x)$ and ii) the $\ell^2$-error at time $T = 0.1$ in space, between the reconstructed solution and a solution of reference. We compare the convergence rate of HKSWR (23) with the simple CSWR algorithm (21) as well as an OSWR as described in [11], based on second-order pseudodifferential absorbing boundary conditions. More specifically: “Order 2 OSWR” refers to as transmission conditions based on a DtN operator including effects of the potential, which writes at (0 $\times \Gamma_+^-$)
\[
(\pm \sqrt{\varepsilon} \partial_x + B^{(2)}_x)\phi_\pm^k = (\pm \sqrt{\varepsilon} \partial_x + B^{(2)}_x)\phi_\pm^{k-1},
\]
where $B^{(2)}_x := e^{i\pi/4} \partial_x^{1/2} - tV'(x) - iV'(x)I_t/4$. The operator $\partial_x^{1/2}$ is the half-order derivative operator while $I_t$ denotes the integral operator. Details can be found in [11].

If $\varepsilon$ is not small enough as in this example, it is not surprising to observe that transmission conditions based on TBCs or high-order ABCs allow for a faster convergence than HKSWR and GOSWR. Indeed, the FGA does not provide a very accurate approximation of the Schrödinger equation. As a consequence, although it is improved compared to CSWR, the HKSWR and GOSWR algorithms still have a slower convergence rate than OSWR. The expectation is however that for a smaller valued of $\varepsilon$, (23) and (24) lead to interesting gains in terms of efficiency.

6.2.2. Test II

This test is similar to the one presented in the numerical example II, except that $\varepsilon$ is now taken smaller: $\varepsilon = 2^{-5}$, (resp. $\varepsilon = 2^{-6}$). The Schrödinger equation is discretized with finite differences on $\Omega_{+}^\pm$ for the following simulation data. Respective space and time steps are $\Delta x = 2^{-7}$ (resp. $\Delta x = 2^{-9}$), $\Delta t = 5 \times 10^{-4}$ (resp. $\Delta t = 10^{-4}$). The final time of computation is $T = 0.1$. The overlapping region is reduced to $\eta = \Delta x$, and more specifically $\Omega^-_{+} = (-2, \Delta x)$ and $\Omega^+_{+} = (0, 2)$. Numerical tests are performed by choosing $\varepsilon = 2^{-10}$ and $\gamma_\varepsilon(k) = \exp\left((- (k - 1)^2)/100\right)$, where $k$ denotes the Schwarz iteration. Figures 5 report, with respect to the Schwarz iteration $k$, (resp. Figs. 7) i) the $\ell^2$-norm in time of $\phi_+(t, \Delta x) - \phi_-(t, \Delta x)$, and ii) the $\ell^2$-error in space of the reconstructed solution with the solution of reference computed at final time $T = 0.1$. We also represent the solution of reference and the reconstructed one on Figure 6, for $\varepsilon = 2^{-7}$. We observe that the smaller $\varepsilon$, the slower the convergence of the FGA-like DDM (4), (5), (7). These results also suggest that GOA is a good comprise between computational complexity and rate of convergence. As expected, the convergence rate is indeed not as fast as FGA, but it requires a lower computational complexity and is faster than CSWR.
We conclude this subsection by a remark regarding the convergence rate for different values of \( \varepsilon \). It may look surprising that for instance the convergence of the CSWR algorithm seems slower for a higher frequency (see \( \varepsilon = 2^{-6} \) vs. \( 2^{-3} \)). It even seems to be in contradiction with the estimate of the convergence factor. This is simply due to the fact that the smaller \( \varepsilon \), the smaller the chosen space step, and in all the simulations, the overlapping region was taken equal to one space step.

7. Conclusion

This paper was devoted to the derivation of domain decomposition methods for the time-dependent Schrödinger equation in the middle/high-frequency regime by using Schwarz waveform relaxation methods with FGA-based Dirichlet transmission conditions. The idea is to simultaneously benefit from the simplicity of the CSWR algorithm and the accuracy of FGA. Convergence of the so-called Herman-Kluk Schwarz Waveform Relaxation algorithm (HKSWR) is indeed a direct consequence of the CSWR convergence, but with acceleration provided by FGA. Some numerical evidences were proposed showing the efficiency of the method. Table 2 roughly summarizes and compares the behavior of each presented method depending on the regime. Among the criteria from Table 2, let us precise that \textit{MultiD easyness} refers to the easyness to implement, in particular in parallel, the corresponding DDM in more than one dimension. Although this criterion is quite subjective, we can surely ensure that the complexity of implementation of FGA-based DDM
(FGA, FGA-based, HKSWR) or CSWR is independent of the spatial dimension, which is not true for DtN-based transmission conditions (OSWR). \textit{Comput. Comp.} refers to as the computational cost to implement the transmission condition, by comparison with simple Dirichlet-based transmission conditions (CSWR). The

(FGA, FGA-based, HKSWR) or CSWR is independent of the spatial dimension, which is not true for DtN-based transmission conditions (OSWR). \textit{Comput. Comp.} refers to as the computational cost to implement the transmission condition, by comparison with simple Dirichlet-based transmission conditions (CSWR). The

HKSWR and GOSWR algorithms are planed to be implemented, in higher dimension, and tested on more realistic problems, in particular related to laser-molecule interactions which involve high frequency generation by electronic recombination [11], [29], [13]. We conclude by mentioning that the HKSWR method presented here can in principle be extended to hyperbolic systems of conservation laws and wave equations using the corresponding FGA [32, 34].

\section*{Acknowledgements.}

E. Lorin authors thanks NSERC for the financial support via the Discovery Grant program. X. Yang was partially supported by the NSF grants DMS-1418936, and DMS-1107291: NSF Research Network in Mathematical Sciences “Kinetic description of emerging challenges in multiscale problems of natural science”. X. Antoine thanks the support of the french ANR grants ”Bond” (ANR-13-BS01-0009-01).

\section*{References}


