Journal of Computational Physics 446 (2021) 110655

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

An asymptotic Green's function method for the wave equation

Jay Mayfield, Yijin Gao, Songting Luo *

Department of Mathematics, Iowa State University, Ames, IA 50011, USA

ARTICLE INFO

Article history:
Available online 25 August 2021

Keywords:
Asymptotic Green's function method
Huygens' principle
WKBJ approximations
Lowrank approximation
Fast Fourier transform

ABSTRACT

We present an effective asymptotic Green's function method for propagating the waves through the linear scalar wave equation. The wave is first split into its forward-propagating and backward-propagating parts. Following that, the method, which combines the Huygens' principle and the geometrical optics approximations, is designed to propagate the forward-propagating and backward-propagating waves, where an integral with Green's functions that is based on the Huygens' principle is used to propagate the waves, and the Green's functions are approximated by the geometrical optics approximations. Upon obtaining analytic approximations for the phase and amplitude in the geometrical optics approximations through short-time Taylor series expansions, a short-time propagator for the waves is derived and the resulting integral can be evaluated efficiently by fast Fourier transform after appropriate lowrank approximations. The short-time propagator can be applied repeatedly to propagate the waves for a long time. In order to restrict the computation onto a bounded domain of interest, the perfectly matched layer technique with complex coordinate stretching transformation is incorporated. The method is first-order accurate, and has complexity $O(t_\epsilon N \log N)$ per time step with N the number of spatial points and t_ϵ the low rank for a prescribed accuracy requirement $\epsilon > 0$. Numerical experiments are presented to demonstrate the proposed method.

© 2021 Elsevier Inc. All rights reserved.

1. Introduction

We consider numerical simulation of wave propagation governed by the linear scalar wave equation,

$$\begin{aligned} &\frac{\partial^2 u(\boldsymbol{x}, t)}{\partial t^2} = c^2(\boldsymbol{x})\nabla^2 u(\boldsymbol{x}, t),\ (\boldsymbol{x}, t) \in \mathbb{R}^d \times (0, \infty), \\ &u(\boldsymbol{x}, 0) = u_0(\boldsymbol{x}),\ \frac{\partial u(\boldsymbol{x}, 0)}{\partial t} = 0,\ \boldsymbol{x} \in \mathbb{R}^d, \end{aligned} \tag{1}$$

for $d = 1, 2, 3$. Here, $\nabla^2 \equiv \nabla \cdot \nabla \equiv \partial_{x_1}^2 + \partial_{x_2}^2 + \partial_{x_3}^2$ is the Laplacian operator at $\boldsymbol{x} \equiv (x_1, x_2, x_3)$, $c(\boldsymbol{x})$ is the wave velocity in the medium, $u(\boldsymbol{x}, t)$ is the wavefield displacement, and $u_0(\boldsymbol{x})$ is the initial wavefield. The problem is multiscale if one compares the wavelength and the overall size of the domain, which makes it equivalent to simulating high frequency wave propagation in an appropriate domain of interest. It is well known that the computational cost to maintain uniform accuracy of the numerical solutions to equation (1) grows algebraically with frequency for conventional numerical methods based on direct discretizations of the equation on meshes, such as the finite-difference and finite-element methods [2,3,18], and the

* Corresponding author.
E-mail addresses: jmayfiel@iastate.edu (J. Mayfield), yjgao@iastate.edu (Y. Gao), luos@iastate.edu (S. Luo).

https://doi.org/10.1016/j.jcp.2021.110655
0021-9991/© 2021 Elsevier Inc. All rights reserved.

Courant-Friedrichs-Lewy (CFL) condition, $c_{\max}\Delta t/\Delta \boldsymbol{x} \leq 1$, needs to be satisfied if an explicit time stepping approach is used, where Δt and $\Delta \boldsymbol{x}$ represent the time step size and spatial mesh size, respectively, and $c_{\max}$ is the maximum speed. If an implicit time stepping approach is used, it often requires to solve a large linear system at each time step. Consequently, direct numerical simulation with such methods is no longer feasible for many practical applications, especially for three-dimensional problems. However, due to its wide applications in various areas such as acoustics, seismology, geophysics, electromagnetism, etc., effective numerical methods for solving equation (1) have been proposed and developed, aiming to balance efficiency and accuracy.

In [20,37,39], a wave extrapolation method using lowrank symbolic approximation was proposed to simulate the seismic wave propagation governed by equation (1). This method splits the wave into its forward-propagating and backward-propagating parts. Following that, an efficient time propagator in the form of a pseudo-differential operator was designed to propagate the forward-propagating and backward-propagating waves, along with lowrank approximations [19,22,23] and fast Fourier transforms (FFT) [12]. The equation (1) can be expressed as a first-order system,

$$\partial_t \begin{bmatrix} u \\ u_t \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -c^2(-\nabla^2) & 0 \end{bmatrix} \begin{bmatrix} u \\ u_t \end{bmatrix} \equiv \mathbf{A} \begin{bmatrix} u \\ u_t \end{bmatrix}, \tag{2}$$

with $u_t \equiv \partial u/\partial t$. The matrix operator $\mathbf{A}$ has an eigenvalue decomposition $\mathbf{A} = \mathbf{R}\boldsymbol{\Lambda}\mathbf{R}^{-1}$ given as,

$$\mathbf{R} = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ \iota\Psi & -\iota\Psi \end{bmatrix}, \ \boldsymbol{\Lambda} = \begin{bmatrix} \iota\Psi & 0 \\ 0 & -\iota\Psi \end{bmatrix}, \ \mathbf{R}^{-1} = \begin{bmatrix} 1 & -\iota\Psi^{-1} \\ 1 & \iota\Psi^{-1} \end{bmatrix}, \tag{3}$$

where Ψ is a pseudo-differential operator in the spatial domain which is defined as $\Psi = c\sqrt{-\nabla^2}$. One can simplify the equation by defining,

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \mathbf{R}^{-1} \begin{bmatrix} u \\ u_t \end{bmatrix} = \begin{bmatrix} u - \iota\Psi^{-1}u_t \\ u + \iota\Psi^{-1}u_t \end{bmatrix}, \tag{4}$$

where u_1 and u_2 are the forward-propagating and backward-propagating waves, respectively. This yields

$$\begin{cases} \partial_t u_1 = \iota\Psi u_1, \\ \partial_t u_2 = -\iota\Psi u_2, \end{cases} \tag{5}$$

which implies

$$\begin{cases} u_1(\boldsymbol{x},\, t_0 + \Delta t) = e^{\iota\Psi\Delta t}u_1(\boldsymbol{x},\, t_0), \\ u_2(\boldsymbol{x},\, t_0 + \Delta t) = e^{-\iota\Psi\Delta t}u_2(\boldsymbol{x},\, t_0), \end{cases} \tag{6}$$

from any starting time t_0 for a short period of time Δt.

If the wave velocity c is constant, then equation (6) is given as,

$$\begin{aligned} u_{1,2}(\boldsymbol{x},\, t) &= \int_{\mathbb{R}^3} e^{\iota(\boldsymbol{k}\cdot\boldsymbol{x} \pm c|\boldsymbol{k}|\Delta t)}\hat{u}_{1,2}(\boldsymbol{k},\, t_0)d\boldsymbol{k} \\ &= \int_{\mathbb{R}^3} \left(\int_{\mathbb{R}^3} \frac{1}{(2\pi)^3} e^{\iota(\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{x}_0) \pm c|\boldsymbol{k}|\Delta t)} d\boldsymbol{k} \right) u_{1,2}(\boldsymbol{x}_0,\, t_0)d\boldsymbol{x}_0, \end{aligned} \tag{7}$$

with $t = t_0 + \Delta t$, where $\hat{u}_{1,2}(\boldsymbol{k},\, t) \equiv \mathcal{F}\left\{u_{1,2}(\boldsymbol{x},\, t)\right\}$ is the spatial Fourier transform of $u_{1,2}(\boldsymbol{x},\, t)$, i.e.,

$$\hat{u}_{1,2}(\boldsymbol{k},\, t) \equiv \mathcal{F}\left\{u_{1,2}(\boldsymbol{x},\, t)\right\} \equiv \frac{1}{(2\pi)^3}\int_{\mathbb{R}^3} e^{-\iota\boldsymbol{k}\cdot\boldsymbol{x}}u_{1,2}(\boldsymbol{x},\, t)d\boldsymbol{x}.$$

The formulation is extended to problems with variable velocity function $c(\boldsymbol{x})$ in a similar way [20,37,39], i.e., the forward-propagating and backward-propagating waves are propagated as,

$$u_{1,2}(\boldsymbol{x},\, t) = \int_{\mathbb{R}^3} \left(\int_{\mathbb{R}^3} \frac{1}{(2\pi)^3} e^{\iota(\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{x}_0) \pm c(\boldsymbol{x})|\boldsymbol{k}|\Delta t} d\boldsymbol{k} \right) u_{1,2}(\boldsymbol{x}_0,\, t_0)d\boldsymbol{x}_0. \tag{8}$$

The integral in (7) and (8) can be evaluated by FFT after appropriate lowrank approximations that can be constructed through a randomized pivoted QR factorization [19,20,37], which results in $O(N \log N)$ complexity per time step with N the number of spatial points used in the simulation and the pre-factor depending on the low rank.

Motivated by the wave extrapolation method developed in [20,37,39], we propose to propagate the wave with appropriate time propagators that combine the Huygens' principle [4,35,38] and the geometrical optics approximations [7,9,18,24,30,31], i.e., from equations (5) and (6), for $t > t_0$,

$$u_{1,2}(\boldsymbol{x}, t) = \int_{\mathbb{R}^3} G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) u_{1,2}(\boldsymbol{x}_0, t_0) d\boldsymbol{x}_0, \tag{9}$$

where, for a generic source point $(\boldsymbol{x}_0, t_0) \in \mathbb{R}^3 \times (0, \infty)$, $G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0)$ is the Green's function. The Green's function generally needs to be approximated such that the integral in (9) can be evaluated. For example, in equations (7) and (8), G is approximated as,

$$G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) \approx \int_{\mathbb{R}^3} \frac{1}{(2\pi)^3} e^{\iota(\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{x}_0) \pm c(\boldsymbol{x})|\boldsymbol{k}|\Delta t)} d\boldsymbol{k}.$$

In this work, we will approximate the Green's function by requiring it to satisfy,

$$\begin{aligned} &\partial_t^2 G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) - c^2(\boldsymbol{x}) \nabla^2 G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) = 0, \quad (\boldsymbol{x}, t) \in \mathbb{R}^3 \times (t_0, \infty), \\ &\lim_{t \to t_0^+} G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) = \delta(\boldsymbol{x} - \boldsymbol{x}_0), \quad \boldsymbol{x} \in \mathbb{R}^3, \\ &G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) = 0, \quad (\boldsymbol{x}, t) \in \mathbb{R}^3 \times (-\infty, t_0), \end{aligned} \tag{10}$$

and utilizing the geometrical optics (GO) approximations, specifically the Wentzel-Kramers-Brillouin-Jeffreys (WKBJ) approximations [7,9,18,24,30,31]. In the WKBJ approximation, the phase and amplitude of the Green's function can be determined through an eikonal equation and a recurrent system of transport equations, respectively, and the solutions to the eikonal equation and the transport equations can be approximated analytically as in [20,25,27,29] for a short period of time using short-time Taylor series expansions. With the WKBJ approximation and the analytic approximations of the phase and amplitude, the integral (9) can be approximated and evaluated efficiently for a short period of time with FFT after appropriate lowrank approximations, which provides an efficient short-time propagator for the wave propagation. And the short-time propagator can be repeatedly used to propagate the waves for a long time [20,25,27].

The idea of combining the Huygens' principle and the geometrical optics approximations for designing efficient propagators for high frequency wave propagation has been utilized for solving the point-source high frequency Helmholtz equation [28,34], the point-source high frequency electromagnetic vector wave equations [26,33], and the time-dependent Schrödinger equations in the semi-classical regime [25,27,21]. In these works, the integral with Green's functions based on the Huygens' principle provides the mechanism for propagating the wave and the geometrical optics approximations provide necessary information of the Green's functions such that the resulting integral is evaluable by FFT [12] or the butterfly algorithm [8] after approximate algebraic manipulation and/or lowrank approximations [19,22,23].

Here, we will combine the ideas in [20,37,39] and [25,27] to develop an efficient time propagator for the wave equation (1), in consideration of both efficiency and accuracy. In order to conduct numerical calculations on a bounded domain of interest, the problem will be restricted to a bounded domain by applying the perfectly matched layer (PML) method [5]. The PML method is an effective approach of absorbing boundary conditions (ABCs) [11,17,32]. It surrounds the bounded domain of interest by an absorbing layer in a way that the waves coming into this layer will be damped to an extent such that the information of the waves outside of the domain of computation becomes unimportant, and the boundary of the absorbing layer and the interface between the absorbing layer and the domain of interest will not cause reflections. Following the PML method that is often implemented by a complex coordinate stretching transform [10], the WKBJ approximation, the analytic approximations of the phase and amplitudes, and the lowrank approximation through a randomized pivoted QR factorization [19] are combined to derive an efficient short-time propagator for propagating the forward-propagating and backward-propagating waves. The proposed asymptotic Green's function method has complexity $O(N \log N)$ per time step as the wave extrapolation method in [20,37,39], with the pre-factor dependent of the low rank for a prescribed accuracy requirement. The method is first-order accurate since second-order accurate short-time Taylor expansions will be used to approximate the phase and amplitude of the Green's function in the one-term WKBJ approximation. More importantly, compared with the wave extrapolation method in [20,37,39], the proposed approach has the potential to improve the accuracy of the amplitude of the waves, which will be demonstrated with numerical experiments.

The rest of the paper is organized as follows. In Section 2, we will present the asymptotic Green's function method for propagating the forward-propagating and backward-propagating waves, with detailed formulations and numerical implementations, including the PMLs. In Section 3, numerical experiments will be performed to demonstrate the proposed method. And conclusive remarks will be given in Section 4.

2. Asymptotic Green's function method

In this section, we shall present the short-time propagator for propagating the waves through a combination of Huygens' principle and WKBJ approximations, which yields the asymptotic Green's function method.

2.1. WKBJ approximation based short-time propagator

The WKBJ approximation for the Green's function $G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0)$ [18,24,25,30,31], for a generic source $(\boldsymbol{x}_0, t_0)$, is given as

$$G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) \approx \frac{1}{(2\pi)^3} \int_{\mathbb{R}^3} e^{-\iota \boldsymbol{x}_0 \cdot \boldsymbol{k}} G_{WKBJ}(\boldsymbol{x}, t; t_0, \boldsymbol{k}) d\boldsymbol{k}, \ t > t_0, \tag{11}$$

with

$$G_{WKBJ}(\boldsymbol{x}, t; t_0, \boldsymbol{k}) = e^{\iota |\boldsymbol{k}| \tau(\boldsymbol{x}, t; t_0, \boldsymbol{k})} \sum_{m=0}^{\infty} A_m(\boldsymbol{x}, t; t_0, \boldsymbol{k}) (\iota |\boldsymbol{k}|)^{-m}, \tag{12}$$

where τ is the phase, $\{A_m\}_{m=0}^{\infty}$ are the amplitude terms, and G_{WKBJ} satisfies

$$\begin{aligned} &\partial_t^2 G_{WKBJ} - c^2(\boldsymbol{x}) \nabla^2 G_{WKBJ} = 0, \quad (\boldsymbol{x}, t) \in \mathbb{R}^3 \times (t_0, \infty), \\ &\lim_{t \to t_0^+} G_{WKBJ}(\boldsymbol{x}, t; t_0, \boldsymbol{k}) = e^{\iota \boldsymbol{x} \cdot \boldsymbol{k}}, \quad \boldsymbol{x} \in \mathbb{R}^3. \end{aligned} \tag{13}$$

Remark 1. If the initial wave $u_0(\boldsymbol{x})$ is given as $u_0(\boldsymbol{x}) = A(\boldsymbol{x}) \exp(\iota \omega \tau(\boldsymbol{x}))$, where $A(\boldsymbol{x})$ and $\tau(\boldsymbol{x})$ stand for the amplitude and phase, respectively, and ω is the high frequency, the WKBJ approximation (12) is recast as,

$$G_{WKBJ}(\boldsymbol{x}, t; t_0, \boldsymbol{k}) = e^{\iota \omega \tau(\boldsymbol{x}, t; t_0, \boldsymbol{k})} \sum_{m=0}^{\infty} A_m(\boldsymbol{x}, t; t_0, \boldsymbol{k}) (\iota \omega)^{-m}, \tag{14}$$

and it satisfies,

$$\begin{aligned} &\partial_t^2 G_{WKBJ} - c^2(\boldsymbol{x}) \nabla^2 G_{WKBJ} = 0, \quad (\boldsymbol{x}, t) \in \mathbb{R}^3 \times (t_0, \infty), \\ &\lim_{t \to t_0^+} G_{WKBJ}(\boldsymbol{x}, t; t_0, \boldsymbol{k}) = e^{\iota \omega \boldsymbol{x} \cdot \boldsymbol{k}}, \quad \boldsymbol{x} \in \mathbb{R}^3. \end{aligned} \tag{15}$$

Then the Green's function is approximated as,

$$G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_0) \approx \frac{\omega^3}{(2\pi)^3} \int_{\mathbb{R}^3} e^{-\iota \omega \boldsymbol{x}_0 \cdot \boldsymbol{k}} G_{WKBJ}(\boldsymbol{x}, t; t_0, \boldsymbol{k}) d\boldsymbol{k}, \ t > t_0. \tag{16}$$

Since the WKBJ approximation of the Green's function G is generally valid only for a short period of time while the phase and amplitude terms are smooth [13,14], we can propagate the waves by formula (9) for a short period of time, i.e., for $t \in [t_0, t_0 + \Delta t]$ with Δt sufficiently small [20,25,27,37,39]. This process can be repeated such that the waves can be propagated for a long time [25,27], i.e.,

$$u(\boldsymbol{x}, t) = \int_{\mathbb{R}^3} G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_n) u(\boldsymbol{x}_0, t_n) d\boldsymbol{x}_0, \ t_n < t \le t_{n+1}, \tag{17}$$

with $t_n = n\Delta t$ for $n = 0, 1, \ldots$. Here and after, the subscripts of $u_{1,2}$ are omitted for notational simplicity.

With the WKBJ approximation of the Green's functions, we have a short-time propagator for the waves, i.e.,

$$\begin{aligned} u(\boldsymbol{x}, t) \approx &\int_{\mathbb{R}^3} \frac{1}{(2\pi)^3} \int_{\mathbb{R}^3} e^{-\iota \boldsymbol{x}_0 \cdot \boldsymbol{k}} e^{\iota |\boldsymbol{k}| \tau(\boldsymbol{x}, t; t_n, \boldsymbol{k})} \\ &\times \sum_{m=0}^{\infty} A_m(\boldsymbol{x}, t; t_n, \boldsymbol{k}) (\iota |\boldsymbol{k}|)^{-m} u(\boldsymbol{x}_0, t_n) d\boldsymbol{k} \, d\boldsymbol{x}_0 \\ = &\sum_{m=0}^{\infty} \int_{\mathbb{R}^3} e^{\iota |\boldsymbol{k}| \tau(\boldsymbol{x}, t; t_n, \boldsymbol{k})} A_m(\boldsymbol{x}, t; t_n, \boldsymbol{k}) (\iota |\boldsymbol{k}|)^{-m} \hat{u}(\boldsymbol{k}, t_n) d\boldsymbol{k}, \end{aligned} \tag{18}$$

for $t_n < t \le t_{n+1}$ $(n = 0, 1, \ldots)$.

2.2. Phase and amplitude in WKBJ approximation

In order to utilize formulas (17) and (18) to propagate the waves, we need to compute the phase τ and the amplitude terms $\{A_m\}_{m=0}^{\infty}$. The phase and amplitude terms can be determined though an eikonal equation and a recurrent system of transport equations, respectively, which can be derived by substituting the WKBJ ansatz (12) into (13) and collecting the terms in the same order of $|\boldsymbol{k}|$ [18,24,25,30,31].

The eikonal equation is given as,

$$\begin{aligned}&(\tau_t(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}))^2 - c^2(\boldsymbol{x})|\nabla\tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})|^2 = 0, \quad t > t_n,\\ &\tau(\boldsymbol{x},\, t_n;\, t_n,\, \boldsymbol{k}) = \frac{\boldsymbol{x}\cdot\boldsymbol{k}}{|\boldsymbol{k}|},\end{aligned} \tag{19}$$

which yields,

$$\begin{aligned}&\tau_t(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}) \mp c(\boldsymbol{x})|\nabla\tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})| = 0, \quad t > t_n,\\ &\tau(\boldsymbol{x},\, t_n;\, t_n,\, \boldsymbol{k}) = \frac{\boldsymbol{x}\cdot\boldsymbol{k}}{|\boldsymbol{k}|},\end{aligned} \tag{20}$$

where the minus sign corresponds to the phase for the forward-propagating wave, and the positive sign corresponds to the phase for the backward-propagating wave.

The recurrent system of transport equations for $\{A_m(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})\}_{m=0}^{\infty}$ is given as: for $m \geq -1$, for the forward-propagating wave, we have,

$$\begin{aligned}&(A_{m+1})_t - c(\boldsymbol{x})\frac{\nabla\tau\cdot\nabla A_{m+1}}{|\nabla\tau|} - \frac{c^2(\boldsymbol{x})\Delta\tau - \tau_{tt}}{2c(\boldsymbol{x})|\nabla\tau|}A_{m+1} - \frac{c^2(\boldsymbol{x})\Delta A_m - (A_m)_{tt}}{2c(\boldsymbol{x})|\nabla\tau|} = 0, \quad t > t_n,\\ &A_{m+1}(\boldsymbol{x},\, t_n;\, t_n,\, \boldsymbol{k}) = \delta(m+1),\end{aligned} \tag{21}$$

and for the backward-propagating waves, we have,

$$\begin{aligned}&(A_{m+1})_t + c(\boldsymbol{x})\frac{\nabla\tau\cdot\nabla A_{m+1}}{|\nabla\tau|} + \frac{c^2(\boldsymbol{x})\Delta\tau - \tau_{tt}}{2c(\boldsymbol{x})|\nabla\tau|}A_{m+1} + \frac{c^2(\boldsymbol{x})\Delta A_m - (A_m)_{tt}}{2c(\boldsymbol{x})|\nabla\tau|} = 0, \quad t > t_n,\\ &A_{m+1}(\boldsymbol{x},\, t_n;\, t_n,\, \boldsymbol{k}) = \delta(m+1),\end{aligned} \tag{22}$$

where $A_{-1} \equiv 0$, and $\delta(x)$ is related to the Kroneker delta function $\delta_{x,0}$ that is 1 if $x = 0$ and 0 if $x \neq 0$.

2.3. Analytic solutions to eikonal and transport equations

Solving equations (20), (21) and (22) with a mesh-based numerical method will incur a heavy computational burden, since they must be computed at every time step for all $\boldsymbol{k} \in \mathbb{R}^3$. Instead, we will employ the analytic approaches based on short-time Taylor series expansions to obtain analytic approximations for the phase and amplitudes [20,25,29]. Assume we want to compute the phase and amplitudes at $t = t_n + \Delta t$, we will expand them as Taylor series at t_n, i.e.,

$$\begin{aligned}&\tau(\boldsymbol{x},\, t;\, t_n;\, \boldsymbol{k}) = \sum_{\ell=0}^{\infty}\tau_\ell(\boldsymbol{x};\, t_n;\, \boldsymbol{k})(\Delta t)^\ell,\\ &A_m(\boldsymbol{x},\, t;\, t_n;\, \boldsymbol{k}) = \sum_{\ell=0}^{\infty}A_{m\ell}(\boldsymbol{x};\, t_n;\, \boldsymbol{k})(\Delta t)^\ell,\ m \geq 0,\end{aligned} \tag{23}$$

with the expansion terms, $\{\tau_\ell\}_{\ell=0}^{\infty}$ and $\{A_{m\ell}\}_{\ell=0}^{\infty}$, to be determined.

Substituting the Taylor series of τ into the eikonal equation (20) and collecting terms in the same order of Δt, we can determine $\{\tau_\ell\}$ term by term, i.e.,

$$\begin{aligned}&\tau_0 = \frac{\boldsymbol{x}\cdot\boldsymbol{k}}{|\boldsymbol{k}|},\\ &\tau_1 = \pm c(\boldsymbol{x}),\end{aligned}$$

and τ_ℓ ($\ell \geq 2$) can be determined similarly.

Substituting the Taylor series of A_0 into the transport equations (21) and (22) and collecting terms in the same order of Δt, we can determine $\{A_{0\ell}\}$ term by term, i.e.,

$$\begin{aligned}&A_{00} = 1,\\ &A_{01} = \mp\frac{1}{2}\frac{\nabla c(\boldsymbol{x})\cdot\boldsymbol{k}}{|\boldsymbol{k}|},\end{aligned}$$

and $A_{0\ell}$ $(\ell \geq 2)$ can be determined similarly.

Similarly, $A_{m\ell}$ $(\ell \geq 0,\ m \geq 1)$ can be determined by collecting terms in the same order of Δt after being substituted into the transport equations (21) and (22).

2.4. A short-time propagator

With the short-time Taylor series expansions, we can obtain analytic approximations for the phase and amplitudes up to any order of accuracy in Δt, following which we can approximate the Green's functions within the WKBJ approximation [25, 27]. We will use second-order accurate Taylor expansions for τ and A_0 in the one-term WKBJ approximation to approximate the Green's function.

The second-order accurate Taylor expansions for τ and A_0 are given as

$$\begin{aligned} \tau &= \tau_0 + \tau_1 \Delta t + O(\Delta t^2), \\ A_0 &= A_{00} + A_{01}\Delta t + O(\Delta t^2), \end{aligned} \tag{24}$$

i.e.,

$$\begin{aligned} \tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}) &= \frac{\boldsymbol{x}\cdot\boldsymbol{k}}{|\boldsymbol{k}|} \pm c(\boldsymbol{x})\Delta t + O(\Delta t^2), \\ A_0(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}) &= 1 \mp \frac{\Delta t}{2}\frac{\nabla c(\boldsymbol{x})\cdot\boldsymbol{k}}{|\boldsymbol{k}|} + O(\Delta t^2). \end{aligned} \tag{25}$$

Then the first-order short-time propagator for the waves is given as,

$$\begin{aligned} u(\boldsymbol{x},\, t) \approx & \int_{\mathbb{R}^3} e^{\iota(\boldsymbol{x}\cdot\boldsymbol{k} \pm c(\boldsymbol{x})|\boldsymbol{k}|\Delta t)}\hat{u}(\boldsymbol{k},\, t_n)d\boldsymbol{k} \\ & \mp \frac{\Delta t}{2}\sum_{i=1}^{3} c_{x_i}\int_{\mathbb{R}^3}\frac{k_i}{|\boldsymbol{k}|}e^{\iota(\boldsymbol{x}\cdot\boldsymbol{k} \pm c(\boldsymbol{x})|\boldsymbol{k}|\Delta t)}\hat{u}(\boldsymbol{k},\, t_n)d\boldsymbol{k},\ t_n < t \leq t_{n+1}, \end{aligned} \tag{26}$$

with $\boldsymbol{k} \equiv (k_1, k_2, k_3)$. If c is constant, it reduces to equation (7), i.e.,

$$u(\boldsymbol{x},\, t) \approx \mathcal{F}^{-1}\left\{e^{\pm\iota c|\boldsymbol{k}|\Delta t}\mathcal{F}\{u(\boldsymbol{x},\, t_n)\}\right\},\ t_n < t \leq t_{n+1}, \tag{27}$$

where $\mathcal{F}^{-1}$ is the inverse Fourier transform defined as,

$$\mathcal{F}^{-1}\left\{\hat{u}(\boldsymbol{k},\, t)\right\} = \int_{\mathbb{R}^3} e^{\iota\boldsymbol{x}\cdot\boldsymbol{k}}\hat{u}(\boldsymbol{k},\, t)d\boldsymbol{k}.$$

The integrals in the first-order short-time propagator (26) must be approximated and evaluated, which will have complexity $O(N^2)$ if they are computed directly. Instead, we can explore the lowrank structures embedded in these integrals and construct lowrank approximations as in [19,20,27,37] such that they can be computed with FFT and relevant matrix-matrix multiplications with complexity $O(t_\epsilon N)$, where t_ϵ is the low rank for a prescribed accuracy requirement $\epsilon > 0$.

Remark 2. The short-time propagator in [20,37,39] is closely related to (26) in that it involves only the first term on the right hand side of (26). In other words, in the WKBJ approximation, the amplitude is approximated as $A_0 \approx 1$ in [20,37,39].

Remark 3. Higher order short-time propagators can be derived similarly as (26) by using more terms in the WKBJ approximation and higher order Taylor expansions for the phase and amplitude terms. For example, for a second-order short-time propagator, the two-term WKBJ approximation is needed to approximate the Green's function and third-order accurate Taylor expansions are required to approximate the phase and amplitude terms, i.e.,

$$\begin{aligned} u(\boldsymbol{x},\, t \equiv t_n + \Delta t) \approx & \int_{\mathbb{R}^3} e^{\iota|\boldsymbol{k}|\tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})}\hat{u}(\boldsymbol{k},\, t_n)d\boldsymbol{k} \\ & + \int_{\mathbb{R}^3} e^{\iota|\boldsymbol{k}|\tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})}A_0(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})\hat{u}(\boldsymbol{k},\, t_n)d\boldsymbol{k} \\ & + \int_{\mathbb{R}^3} e^{\iota|\boldsymbol{k}|\tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})}\frac{1}{\iota|\boldsymbol{k}|}A_1(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})\hat{u}(\boldsymbol{k},\, t_n)d\boldsymbol{k}, \end{aligned}$$

and

$$\tau \approx \tau_0 + \tau_1 \Delta t + \tau_2 \Delta t^2 + O(\Delta t^3),$$
$$A_0 \approx A_{00} + A_{01} \Delta t + A_{02} \Delta t^2 + O(\Delta t^3),$$
$$A_1 \approx A_{10} + A_{11} \Delta t + A_{12} \Delta t^2 + O(\Delta t^3),$$

with $\tau_0, \tau_1, \tau_2, A_{00}, A_{01}, A_{02}, A_{10}, A_{11}$ and A_{12} derived analytically in the Taylor expansions.

2.5. Short-time propagator with perfectly matched layers

In order to apply the formula (26) or (27) as the short-time propagator for propagating the waves numerically, we need to limit our computation to a bounded domain of interest, unless the waves are compactly supported. For truncating the infinite domain into a bounded subdomain, we will use the perfectly matched layers (PML) technique that is implemented by complex coordinate stretching transform [5,10].

Without loss of generality, we define the computational domain as $\Omega \equiv [x_{1,\min}, x_{1,\max}] \times [x_{2,\min}, x_{2,\max}] \times [x_{3,\min}, x_{3,\max}]$. We adopt the absorbing profile function [1,5,10,40],

$$\sigma_i(x_i) = \begin{cases} C_i \exp\left(\frac{\exp(-1/p)}{p-1}\right), \ p = \frac{x_i - x_{i,\min}}{\eta_i}, & x_i \in [x_{i,\min}, x_{i,\min} + \eta_i], \\ 0, & x_i \in [x_{i,\min} + \eta_i, x_{i,\max} - \eta_i] \\ C_i \exp\left(\frac{\exp(-1/p)}{p-1}\right), \ p = -\frac{x_i - x_{i,\max}}{\eta_i}, & x_i \in [x_{i,\max} - \eta_i, x_{i,\max}], \end{cases} \tag{28}$$

and

$$s_i(x_i) = \frac{1}{1 \mp \imath \sigma_i(x_i)}, \tag{29}$$

for $i = 1, 2, 3$, where η_i is the width of the PML layer in the x_i-direction, C_i is constant, and the $\mp$ correspond to the forward-propagating and backward-propagating waves, respectively. The PML technique replaces the partial derivatives ∂_{x_i} in equation (1) with $s_i(x_i)\partial_{x_i}$, which alters equation (1) to become

$$\partial_t^2 u(\boldsymbol{x}, t) = c^2(\boldsymbol{x}) \sum_{i=1}^{3} s_i(x_i) \partial_{x_i} \left(s_i(x_i) \partial_{x_i} u(\boldsymbol{x}, t) \right), \quad (\boldsymbol{x}, t) \in \mathbb{R}^3 \times (0, \infty). \tag{30}$$

With the substitution,

$$u(\boldsymbol{x}, t) = \frac{\psi(\boldsymbol{x}, t)}{\sqrt{s_1(x_1) s_2(x_2) s_3(x_3)}}, \tag{31}$$

we have the modified wave equation with PMLs given as

$$\partial_t^2 \psi(\boldsymbol{x}, t) = c^2(\boldsymbol{x}) \sum_{i=1}^{3} s_i^2(x_i) \partial_{x_i}^2 \psi(\boldsymbol{x}, t) + w(\boldsymbol{x}, t) \psi(\boldsymbol{x}, t), \tag{32}$$

where,

$$w(\boldsymbol{x}, t) = c^2(\boldsymbol{x}) \left(\sum_{i=1}^{3} \frac{(s'(x_i))^2}{4} - \frac{s_i(x_i) s''(x_i)}{2} \right). \tag{33}$$

The time propagator for the forward- and backward-propagating parts of ψ, based on Huygens' principle, is given similarly as (17), where the Green's function $G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_n)$ for a generic source $(\boldsymbol{x}_0, t_n)$ is chosen to satisfy

$$\begin{aligned} &\partial_t^2 G - c^2(\boldsymbol{x}) \sum_{i=1}^{3} s_i^2(x_i) \partial_{x_i}^2 G - wG = 0, \quad (\boldsymbol{x}, t) \in \mathbb{R}^3 \times (t_n, \infty), \\ &\lim_{t \to t_n^+} G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_n) = \delta(\boldsymbol{x} - \boldsymbol{x}_0), \quad \boldsymbol{x} \in \mathbb{R}^3, \\ &G(\boldsymbol{x}, t; \boldsymbol{x}_0, t_n) = 0, \quad (\boldsymbol{x}, t) \in \mathbb{R}^3 \times (-\infty, t_n). \end{aligned} \tag{34}$$

Similarly, we will approximate the Green's function with the WKBJ approximations as in (11), which results in a short-time propagator that can be repeatedly applied to propagate the waves for a long time, i.e.,

$$\psi(\boldsymbol{x},\, t) = \int_{\Omega} G(\boldsymbol{x},\, t;\, \boldsymbol{x}_0,\, t_n)\psi(\boldsymbol{x}_0,\, t_n)d\boldsymbol{x}_0, \quad t_n < t \le t_{n+1}, \tag{35}$$

with $t_n = n\Delta t$ for $n = 0,\, 1,\, \ldots$.

The modified eikonal equation for the phase τ is given as,

$$\begin{aligned} &\partial_t \tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}) \mp c(\boldsymbol{x}) \sqrt{\sum_{i=1}^{3} s_i^2(x_i)(\tau_{x_i}(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}))^2} = 0, \quad t > t_n, \\ &\tau(\boldsymbol{x},\, t_n;\, t_n,\, \boldsymbol{k}) = \frac{\boldsymbol{x}\cdot\boldsymbol{k}}{|\boldsymbol{k}|}. \end{aligned} \tag{36}$$

The recurrent system of modified transport equations for the amplitudes $\{A_m(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k})\}_{m=0}^{\infty}$ is given as,

$$\begin{aligned} &2\tau_t (A_{m+1})_t - 2c^2(\boldsymbol{x}) \sum_{i=1}^{3} s_i^2(x_i)\tau_{x_i}(A_{m+1})_{x_i} - \left(c^2(\boldsymbol{x}) \sum_{i=1}^{3} s_i^2(x_i)\tau_{x_i x_i} - \tau_{tt} \right) A_{m+1} \\ &- c^2(\boldsymbol{x}) \sum_{i=1}^{3} s_i^2(x_i)(A_m)_{x_i x_i} + (A_m)_{tt} - w(\boldsymbol{x},\, t)A_m = 0, \quad t > t_n, \\ &A_{m+1}(\boldsymbol{x},\, t_n;\, t_n,\, \boldsymbol{k}) = \delta(m+1), \end{aligned} \tag{37}$$

for $m \ge -1$ and $A_{-1} \equiv 0$.

We will again approximate the solutions to equations (36) and (37) analytically through Taylor series expansions as in Section 2.3, and obtain second-order accurate Taylor expansions as in (24), i.e.,

$$\begin{aligned} &\tau(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}) = \frac{\boldsymbol{x}\cdot\boldsymbol{k}}{|\boldsymbol{k}|} \pm \frac{\Delta t}{|\boldsymbol{k}|} c(\boldsymbol{x}) \sqrt{\sum_{i=1}^{3} s_i^2(x_i)k_i^2} + O(\Delta t)^2, \\ &A_0(\boldsymbol{x},\, t;\, t_n,\, \boldsymbol{k}) = 1 - \Delta t \frac{|\boldsymbol{k}|}{2} \frac{\sum_{i=1}^{3} s_i(x_i)k_i(\tau_1)_{x_i}}{\sum_{i=1}^{3} s_i^2(x_i)k_i^2} + O(\Delta t^2). \end{aligned} \tag{38}$$

Then, we have the first-order short-time propagator, from (35), given as,

$$\begin{aligned} \psi(\boldsymbol{x},\, t) \approx &\int_{\Omega} e^{\iota \boldsymbol{x}\cdot\boldsymbol{k}} E^{\pm}(\boldsymbol{x},\, \boldsymbol{k})\hat{\psi}(\boldsymbol{k},\, t_n)d\boldsymbol{k} \\ &\mp \frac{\Delta t}{2} \sum_{i=1}^{3} c_{x_i}(\boldsymbol{x}) s_i(x_i) \int_{\Omega} e^{\iota \boldsymbol{x}\cdot\boldsymbol{k}} F_{1,i}^{\pm}(\boldsymbol{x},\, \boldsymbol{k})\hat{\psi}(\boldsymbol{k},\, t_n)d\boldsymbol{k} \\ &\mp \frac{\Delta t}{2} c(\boldsymbol{x}) \sum_{i=1}^{3} s_i^2(x_i) s_i'(x_i) \int_{\Omega} e^{\iota \boldsymbol{x}\cdot\boldsymbol{k}} F_{2,i}^{\pm}(\boldsymbol{x},\, \boldsymbol{k})\hat{\psi}(\boldsymbol{k},\, t_n)d\boldsymbol{k}, \end{aligned} \tag{39}$$

for $t_{n+1} \ge t > t_n$ $(n = 0,\, 1,\, \ldots)$, where,

$$\begin{aligned} &E^{\pm}(\boldsymbol{x},\, \boldsymbol{k}) = \exp\left(\pm \iota \Delta t c(\boldsymbol{x}) \sqrt{\sum_{i=1}^{3} s_i^2(x_i)k_i^2} \right), \\ &F_{1,i}^{\pm}(\boldsymbol{x},\, \boldsymbol{k}) = \frac{k_i}{\sqrt{\sum_{j=1}^{3} s_j^2(x_j)k_j^2}} E^{\pm}(\boldsymbol{x},\, \boldsymbol{k}), \\ &F_{2,i}^{\pm}(\boldsymbol{x},\, \boldsymbol{k}) = \frac{k_i^3}{(\sum_{j=1}^{3} s_j^2(x_j)k_j^2)^{3/2}} E^{\pm}(\boldsymbol{x},\, \boldsymbol{k}), \end{aligned} \tag{40}$$

for $i = 1,\, 2,\, 3$.

We will use formula (39) as the propagator for propagating the forward-propagating and backward-propagating waves.

2.6. Numerical implementations

We assume the domain of computation Ω is covered by a uniform mesh $\tilde{\boldsymbol{x}} \equiv \{\boldsymbol{x}_{lmn} \equiv (x_{1,l}, x_{2,m}, x_{3,n})\}$ with

$$\{x_{1,l} = x_{1,\min} + (l-1)\Delta x_1\}_{l=1}^{N_1+1},$$
$$\{x_{2,m} = x_{2,\min} + (m-1)\Delta x_2\}_{m=1}^{N_2+1},$$
$$\{x_{3,n} = x_{3,\min} + (n-1)\Delta x_3\}_{n=1}^{N_3+1},$$

and $\Delta x_i = (x_{i,\max} - x_{i,\min})/N_i$ for $i = 1, 2, 3$. For the phase domain, we assume it is covered by a uniform mesh $\tilde{\boldsymbol{k}} \equiv \{\boldsymbol{k}_{pqr} \equiv (k_{1,p}, k_{2,q}, k_{3,r})\}$ with

$$\{k_{1,p} = (p-1)\Delta k_1\}_{p=-N_1/2+1}^{N_1/2},$$
$$\{k_{2,q} = (q-1)\Delta k_2\}_{q=-N_2/2+1}^{N_2/2},$$
$$\{k_{3,r} = (r-1)\Delta k_3\}_{r=-N_3/2+1}^{N_3/2},$$

and $\Delta k_i = 2\pi/(x_{i,\max} - x_{i,\min})$ for $i = 1, 2, 3$. And for notational simplicity, we denote $\Delta \boldsymbol{x} \equiv \Delta x_1 \Delta x_2 \Delta x_3$, $\Delta \boldsymbol{k} \equiv \Delta k_1 \Delta k_2 \Delta k_3$, $\psi_{lmn}(t) \equiv \psi(\boldsymbol{x}_{lmn}, t)$ and $\hat{\psi}_{pqr}(t) \equiv \hat{\psi}(\boldsymbol{k}_{pqr}, t)$.

The Fourier transform is discretized as

$$\hat{\psi}_{pqr}(t_n) \approx \frac{\Delta \boldsymbol{x}}{(2\pi)^3} \sum_{l,m,n} \exp\left(-\iota \boldsymbol{x}_{lmn} \cdot \boldsymbol{k}_{pqr}\right) \psi_{lmn}(t_n), \tag{41}$$

which can be computed efficiently by FFT.

The first-order short-time propagator (39) is discretized as,

$$\begin{aligned}
\psi_{lmn}(t) \approx\ & \Delta \boldsymbol{k} \sum_{p,q,r} \exp\left(\iota \boldsymbol{x}_{lmn} \cdot \boldsymbol{k}_{pqr}\right) E^{\pm}(\boldsymbol{x}_{lmn}, \boldsymbol{k}_{pqr}) \hat{\psi}_{pqr}(t_n) \\
& \mp \frac{\Delta t \Delta \boldsymbol{k}}{2} \sum_{i=1}^{3} c_{x_i}(\boldsymbol{x}_{lmn}) s_i \sum_{p,q,r} \exp\left(\iota \boldsymbol{x}_{lmn} \cdot \boldsymbol{k}_{pqr}\right) F_{i,1}^{\pm}(\boldsymbol{x}_{lmn}, \boldsymbol{k}_{pqr}) \hat{\psi}_{pqr}(t_n) \\
& \mp \frac{\Delta t \Delta \boldsymbol{k}}{2} c(\boldsymbol{x}_{lmn}) \sum_{i=1}^{3} s_i^2 s_i' \sum_{p,q,r} \exp\left(\iota \boldsymbol{x}_{lmn} \cdot \boldsymbol{k}_{pqr}\right) F_{i,2}^{\pm}(\boldsymbol{x}_{lmn}, \boldsymbol{k}_{pqr}) \hat{\psi}_{pqr}(t_n).
\end{aligned} \tag{42}$$

Direct evaluation of the summations in (42) for computing $\{\psi_{lmn}(t)\}$ has complexity $O(N^2)$ with $N = \prod_{i=1}^{3}(N_i + 1)$, which is generally too high for practical applications. To alleviate the computational cost, we will explore the lowrank structures [19,22,23] and construct the lowrank approximations for $E^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$, $F_{i,1}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$ $(i = 1, 2, 3)$, and $F_{i,2}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$ $(i = 1, 2, 3)$.

2.6.1. Lowrank approximations

For constructing the lowrank approximations, we will employ the randomized pivoted QR factorization approach introduced in [19]. For a generic matrix $A(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$, the desired lowrank approximation is given as

$$A \approx A(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}}^{dom}) D A(\tilde{\boldsymbol{x}}^{dom}, \tilde{\boldsymbol{k}}), \tag{43}$$

where $\tilde{\boldsymbol{x}}^{dom} \subset \tilde{\boldsymbol{x}}$ denotes the set of principal rows of A, $\tilde{\boldsymbol{k}}^{dom} \subset \tilde{\boldsymbol{k}}$ denotes the set of principal columns of A, and D is an appropriate matrix. The size of $\tilde{\boldsymbol{x}}^{dom}$ and $\tilde{\boldsymbol{k}}^{dom}$ is $O(t_\epsilon)$ with t_ϵ the low rank for a prescribed accuracy requirement $\infty \gg \epsilon > 0$. And D is of size $O(t_\epsilon) \times O(t_\epsilon)$. The randomized pivoted QR factorization approach generates the lowrank approximation in the following way [19],

1. *Determine principal columns* $\tilde{\boldsymbol{k}}^{dom}$. We randomly select a subset of the rows of A, denoted as Θ_1 with size $|\Theta_1| = O(t_\epsilon)$, and compute the pivoted QR factorization,

   $$A(\Theta_1, \Lambda_1) = Q_1 R_1,$$

   where Λ_1 is a permutation of the columns of A such that the diagonal of R_1 is decreasing in magnitude. We then determine the entries of the diagonal of R_1 which are greater than ϵ, and set $\tilde{\boldsymbol{k}}^{dom}$ to be the columns of A that are associated with these entries.

2. *Determine principal rows* $\tilde{\boldsymbol{x}}^{dom}$. We randomly select a subset of the columns of A, denoted as Θ_2 with size $|\Theta_2| = O(t_\epsilon)$, and compute the pivoted QR factorization of A^*,

$$A(\Lambda_2, \Theta_2) = R_2^* Q_2^*,$$

where Λ_2 is a permutation of the columns of A^* such that the diagonal of R_2 is decreasing in magnitude. We then find the entries of the diagonal of R_2 which are greater than ϵ, and set $\tilde{\boldsymbol{x}}^{dom}$ to be the columns of A^* that are associated with these entries.

3. *Compute matrix D*. We randomly select sets of rows and columns as Γ_1 and Γ_2 respectively, with sizes $|\Gamma_1| = O(t_\epsilon)$ and $|\Gamma_2| = O(t_\epsilon)$, and compute D as

$$D = (A(\Gamma_1, \tilde{\boldsymbol{k}}^{dom}))^\dagger A(\Gamma_1, \Gamma_2)(A(\tilde{\boldsymbol{x}}^{dom}, \Gamma_2))^\dagger, \tag{44}$$

where $()^\dagger$ refers to the pseudo-inverse.

The sizes $|\Gamma_1|$ and $|\Gamma_2|$ are often chosen to be larger than $|\Theta_1|$ and $|\Theta_2|$ [19].

We will apply this approach to generate lowrank approximations for $E^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$, $F_{i,1}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$, and $F_{i,2}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$ $(i = 1, 2, 3)$, which are denoted as

$$\begin{aligned}
E^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}}) &\approx E^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}}^{dom}) D^E E^{\pm}(\tilde{\boldsymbol{x}}^{dom}, \tilde{\boldsymbol{k}}),\\
F_{i,1}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}}) &\approx F_{i,1}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}}^{dom}) D^{F_{i,1}} F_{i,1}^{\pm}(\tilde{\boldsymbol{x}}^{dom}, \tilde{\boldsymbol{k}}),\\
F_{i,2}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}}) &\approx F_{i,2}^{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}}^{dom}) D^{F_{i,2}} F_{i,2}^{\pm}(\tilde{\boldsymbol{x}}^{dom}, \tilde{\boldsymbol{k}}).
\end{aligned} \tag{45}$$

Here, $\tilde{\boldsymbol{k}}^{dom}$ and $\tilde{\boldsymbol{x}}^{dom}$ are generally different for $E^{\pm}$, $F_{i,1}^{\pm}$ and $F_{i,2}^{\pm}$; we use same notations only for notional simplicity.

With the lowrank approximations, the formula (42) is approximated as

$$\begin{aligned}
\psi(\tilde{\boldsymbol{x}}, t) \approx\ & \Delta \boldsymbol{k} \sum_{j=1}^{|\tilde{\boldsymbol{k}}^{dom}|} E^{\pm}(\tilde{\boldsymbol{x}}, \boldsymbol{k}_j^{dom}) \sum_{l=1}^{|\tilde{\boldsymbol{x}}^{dom}|} d_{jl}^E \boldsymbol{K}_l^E(\tilde{\boldsymbol{x}})\\
& \mp \frac{\Delta t \Delta \boldsymbol{k}}{2} \sum_{i=1}^{3} c_{x_i}(\tilde{\boldsymbol{x}}) s_i \sum_{j=1}^{|\tilde{\boldsymbol{k}}^{dom}|} F_{i,1}^{\pm}(\tilde{\boldsymbol{x}}, \boldsymbol{k}_j^{dom}) \sum_{l=1}^{|\tilde{\boldsymbol{x}}^{dom}|} d_{jl}^{F_{i,1}} \boldsymbol{K}_l^{F_{i,1}}(\tilde{\boldsymbol{x}})\\
& \mp \frac{\Delta t \Delta \boldsymbol{k}}{2} c(\tilde{\boldsymbol{x}}) \sum_{i=1}^{3} s_i^2 s_i' \sum_{j=1}^{|\tilde{\boldsymbol{k}}^{dom}|} F_{i,2}^{\pm}(\tilde{\boldsymbol{x}}, \boldsymbol{k}_j^{dom}) \sum_{l=1}^{|\tilde{\boldsymbol{x}}^{dom}|} d_{jl}^{F_{i,2}} \boldsymbol{K}_l^{F_{i,2}}(\tilde{\boldsymbol{x}}),
\end{aligned} \tag{46}$$

where,

$$\tilde{\boldsymbol{x}}^{dom} \equiv \{\boldsymbol{x}_l^{dom}\}, \tilde{\boldsymbol{k}}^{dom} \equiv \{\boldsymbol{k}_j^{dom}\}, D^E \equiv [d_{jl}^E],\ D^{F_{i,1}} \equiv [d_{jl}^{F_{i,1}}],\ D^{F_{i,2}} \equiv [d_{jl}^{F_{i,2}}],$$

and

$$\begin{aligned}
\boldsymbol{K}_l^E(\tilde{\boldsymbol{x}}) &\equiv \sum_{p,q,r} \exp\left(\iota \tilde{\boldsymbol{x}} \cdot \boldsymbol{k}_{pqr}\right) E^{\pm}(\boldsymbol{x}_l^{dom}, \boldsymbol{k}_{pqr}) \hat{\psi}_{pqr}(t_n),\\
\boldsymbol{K}_l^{F_{i,1}}(\tilde{\boldsymbol{x}}) &\equiv \sum_{p,q,r} \exp\left(\iota \tilde{\boldsymbol{x}} \cdot \boldsymbol{k}_{pqr}\right) F_{i,1}^{\pm}(\boldsymbol{x}_l^{dom}, \boldsymbol{k}_{pqr}) \hat{\psi}_{pqr}(t_n),\\
\boldsymbol{K}_l^{F_{i,2}}(\tilde{\boldsymbol{x}}) &\equiv \sum_{p,q,r} \exp\left(\iota \tilde{\boldsymbol{x}} \cdot \boldsymbol{k}_{pqr}\right) F_{i,2}^{\pm}(\boldsymbol{x}_l^{dom}, \boldsymbol{k}_{pqr}) \hat{\psi}_{pqr}(t_n).
\end{aligned} \tag{47}$$

Note that $\boldsymbol{K}_l^E(\tilde{\boldsymbol{x}})$, $\boldsymbol{K}_l^{F_{i,1}}(\tilde{\boldsymbol{x}})$ and $\boldsymbol{K}_l^{F_{i,2}}(\tilde{\boldsymbol{x}})$ can be computed efficiently by FFT.

2.6.2. The algorithm: asymptotic Green's function method

With the short-time propagator and lowrank approximations, i.e., (46) and (47), we summarize the asymptotic Green's function method for computing the waves of equation (32).

Table 1
1D example: order of accuracy at final time $T = 0.5$ for **Method I**. Accuracy is measured in both 2− and ∞−norms. $\Delta t = \Delta x/2$, $C = 0.5$ for the PML, $|\Theta_1| = |\Theta_2| = 20$, $|\Gamma_1| = |\Gamma_2| = 30$, $\epsilon = 10^{-8}$. u_N denotes the numerical solution with the mesh consisting of $N + 1$ spatial points.

N	$\|u_{2N} - u_N\|_\infty$	$\log_2\left(\frac{\|u_{2N}-u_N\|_\infty}{\|u_{4N}-u_{2N}\|_\infty}\right)$	$\|u_{2N} - u_N\|_2$	$\log_2\left(\frac{\|u_{2N}-u_N\|_2}{\|u_{4N}-u_{2N}\|_2}\right)$
		$c(x) = \exp(-x^2)$		
2^7	3.2215e-03	–	1.9322e-03	–
2^8	1.6405e-03	0.9736	9.7325e-04	0.9894
2^9	8.2835e-04	0.9859	4.8740e-04	0.9977
2^{10}	4.1757e-04	0.9882	2.4601e-04	0.9864
2^{11}	2.0643e-04	1.0164	1.2141e-04	1.0188
		$c(x) = 0.2 + \sin^2(x - 1)$		
2^7	4.5555e-03	–	3.2237e-03	–
2^8	2.2835e-03	0.9964	1.6115e-03	1.0003
2^9	1.1375e-03	1.0054	8.0573e-04	1.0001
2^{10}	5.7007e-04	0.9967	4.0367e-04	0.9971
2^{11}	2.8319e-04	1.0094	2.0161e-04	1.0016

Algorithm 1 *(Asymptotic Green's function method).*

1. Apply the randomized pivoted QR factorization for generating the lowrank approximations in (45).
2. For $n = 0, 1, 2, \ldots$ with $t_n = n\Delta t$,
   - Apply FFT to compute $\hat{\psi}(\tilde{\boldsymbol{k}}, t_n)$ through formula (41);
   - Apply formulas (46) and (47) with FFT to compute $\psi(\tilde{\boldsymbol{x}}, t_n + \Delta t)$.

Algorithm 1 is expected to be first-order accurate since the second-order accurate Taylor expansions and one-term WKBJ approximation are used. The complexity of Step 1 is $O(t_\epsilon^2 N)$, and the complexity of Step 2 is $O(t_\epsilon N \log N)$ per time step. Numerical experiments will be presented to demonstrate the method.

3. Numerical experiments

In this section, we present one-dimensional (1D), two-dimensional (2D), and three-dimensional (3D) numerical experiments to demonstrate the proposed method. We will refer to Algorithm 1 as "**Method I**", and the method in [20,37] as "**Method II**". For the 1D experiments, we will further compare the numerical solutions with those computed by the θ-method [15]. For all the experiments, we present results for the full wave $u = (u_1 + u_2)/2$. Wherever applicable, the units used in the numerical experiments are kilometers for length and seconds for time.

3.1. 1D examples

We present several synthetic 1D experiments to check the accuracy and efficiency of **Method I**.

We consider two cases of wave velocity, $c(x) = \exp(-x^2)$ and $c(x) = 0.2 + \sin^2(x - 1)$. The domain of interest is $[-2, 2]$, and the computational domain is $\Omega = [-4, 4]$ with $\eta = 2$ for the width of the PML. For all the experiments, the initial condition is,

$$u(x, 0) = \exp\left(-x^2/0.08\right).$$

Table 1 shows the accuracy in the domain of interest of **Method I**, from which we observe that **Method I** is first-order accurate in Δt. Table 2 shows the CPU times per time step for both **Method I** and **Method II**, from which we observe that their complexity is $O(N \log N)$ at each time step. In general, **Method I** consumes slightly more CPU time than **Method II**, which is expected since **Method I** uses an extra term in the Taylor expansion to improve the accuracy of the amplitude. Table 3 demonstrates the number of principal rows and columns in the lowrank approximations, where we only present results for the matrix $E^{\pm}$, and others are similar. From this table, we observe that the low ranks depend monotonically on $\Delta t/\Delta x$. Fig. 1 shows the numerical plots of both **Method I** and **Method II**, with comparisons to the θ-method, from which we observe that **Method I** has the potential to improve the accuracy of the amplitude of the wave, compared with **Method II**.

3.2. 2D examples

We present 2D examples to demonstrate the proposed method.

We first check the accuracy of **Method I** using two synthetic velocity models. The domain of interest is $[-2, 2]^2$, and the computational domain is $\Omega = [-4, 4]^2$ with $\eta_1 = \eta_2 = 2$ for the PMLs. The initial condition is given as

Table 2
Averaged CPU times per time step for **Method I** and **Method II**, with the same parameters as in Table 1.

N	**Method I**	**Method II**
$c(x) = \exp(-x^2)$		
2^7	1.8244e-03	8.6678e-04
2^8	2.5113e-03	1.0788e-03
2^9	5.3742e-03	1.9003e-03
2^{10}	5.1456e-03	4.0153e-03
2^{11}	9.2970e-03	7.9436e-03
$c(x) = 0.2 + \sin^2(x-1)$		
2^7	1.4511e-03	8.5581e-04
2^8	2.3951e-03	1.0809e-03
2^9	3.4161e-03	2.6050e-03
2^{10}	6.4289e-03	5.4605e-03
2^{11}	7.2329e-03	7.9694e-03

Table 3
1D example: number of principal rows/columns of $E_{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$, $N = 2^{10}$, $|\Theta_1| = |\Theta_2| = 20$, $|\Gamma_1| = |\Gamma_1| = 30$.

$c(x) = \exp(-x^2)$					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	5 5	4 4	4 4	3 3
	$C = 0.5$	5 5	5 4	4 4	3 3
	$C = 0.25$	5 5	4 4	4 4	3 3
$\epsilon = 10^{-7}$	$C = 1$	6 6	5 5	5 5	4 4
	$C = 0.5$	6 6	5 5	5 5	4 4
	$C = 0.25$	6 6	5 5	5 5	4 4
$\epsilon = 10^{-9}$	$C = 1$	7 7	6 6	5 5	5 5
	$C = 0.5$	7 7	6 5	6 5	5 5
	$C = 0.25$	7 7	6 6	5 6	5 5
$c(x) = 0.2 + \sin^2(x-1)$					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	6 5	5 5	4 4	4 4
	$C = 0.5$	5 5	4 5	4 4	4 4
	$C = 0.25$	5 5	4 5	4 4	4 3
$\epsilon = 10^{-7}$	$C = 1$	7 7	6 6	5 5	5 5
	$C = 0.5$	7 7	6 6	5 5	4 4
	$C = 0.25$	6 6	5 5	5 5	4 4
$\epsilon = 10^{-9}$	$C = 1$	8 8	7 7	6 6	5 5
	$C = 0.5$	8 8	7 7	6 6	5 5
	$C = 0.25$	7 8	6 7	6 6	5 5

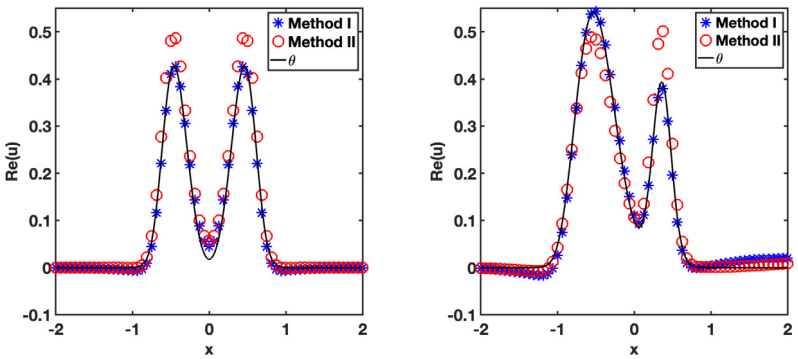


Fig. 1. 1D example: numerical plots of real part of u at final time $T = 0.5$. Left: $c(x) = \exp(-x^2)$, Right: $c(x) = 0.2 + \sin^2(x-1)$. Reference solution is computed using the θ-method.

Table 4
2D example: order of accuracy at final time $T = 0.5$ for **Method I**. Accuracy is measured in both $2-$ and $\infty-$norms. $\Delta t = \Delta x/2$, $C_1 = C_2 = 0.5$ for the PML, $|\Theta_1| = |\Theta_2| = 30$, $|\Gamma_1| = |\Gamma_1| = 50$, $\epsilon = 10^{-8}$. u_N denotes the numerical solution with the mesh consisting of $(N_1 + 1)(N_2 + 1)$ spatial points.

$N_1 \times N_2$	$\|u_{2N} - u_N\|_\infty$	$\log_2\left(\frac{\|u_{2N}-u_N\|_\infty}{\|u_{4N}-u_{2N}\|_\infty}\right)$	$\|u_{2N} - u_N\|_2$	$\log_2\left(\frac{\|u_{2N}-u_N\|_2}{\|u_{4N}-u_{2N}\|_2}\right)$
		$c(x_1, x_2) = \exp(-x_1^2 - x_2^2)$		
$2^4 \times 2^4$	2.4424e-01	–	1.6519e-01	–
$2^5 \times 2^5$	4.3133e-02	2.5014	3.8505e-02	2.1010
$2^6 \times 2^6$	8.4637e-03	2.3494	7.0094e-03	2.4577
$2^7 \times 2^7$	3.4038e-03	1.3142	3.1938e-03	1.1340
$2^8 \times 2^8$	1.5120e-03	1.1706	1.5546e-03	1.0387
		$c(x_1, x_2) = 0.2 + \sin^2(0.5x_1 + 0.5x_2)$		
$2^4 \times 2^4$	2.7480e-01	–	1.5620e-01	–
$2^5 \times 2^5$	1.2442e-02	4.4651	4.3246e-03	5.1747
$2^6 \times 2^6$	1.1829e-03	3.3948	8.4538e-04	2.3549
$2^7 \times 2^7$	5.9155e-04	0.9997	4.1251e-04	1.0352
$2^8 \times 2^8$	2.8610e-04	1.0480	1.9840e-04	1.0560

Table 5
Averaged CPU times per time step for **Method I** and **Method II**, with the same parameters as in Table 4.

$N_1 \times N_2$	**Method I**	**Method II**
	$c(x_1, x_2) = \exp(-x_1^2 - x_2^2)$	
$2^4 \times 2^4$	2.0130e-02	3.7437e-03
$2^5 \times 2^5$	4.1706e-02	9.7394e-03
$2^6 \times 2^6$	9.1768e-02	1.9895e-02
$2^7 \times 2^7$	3.7928e-01	6.3670e-02
$2^8 \times 2^8$	1.6749e+00	2.7196e-01
	$c(x_1, x_2) = 0.2 + \sin^2(0.5x_1 + 0.5x_2)$	
$2^4 \times 2^4$	2.5407e-02	7.1136e-03
$2^5 \times 2^5$	6.4611e-02	1.6993e-02
$2^6 \times 2^6$	1.4040e-01	4.6523e-02
$2^7 \times 2^7$	6.8776e-01	1.2895e-01
$2^8 \times 2^8$	2.6871e+00	6.3098e-01

$$u(x_1, x_2, 0) = \exp\left(-(x_1^2 + x_2^2)/0.08\right).$$

Table 4 shows the accuracy of **Method I** with the two choices of wave velocity. Table 5 shows the CPU times per time step for both **Method I** and **Method II**. From Table 4, we observe that **Method I** is first-order accurate in Δt, and from Table 5, we observe that both **Method I** and **Method II** have complexity $O(N \log N)$ at each time step. In general, **Method I** consumes more CPU time than **Method II** as expected, since **Method I** uses extra terms to improve the accuracy of the amplitude. Table 6 shows the number of principal rows and columns for the matrix $E^{\pm}$, from which we observe that the low ranks depend monotonically on $\Delta t/\{\Delta x_1, \Delta x_2\}$.

We further demonstrate the proposed method using the benchmark Marmousi model [6] and BPTTI 2007 quasi-P velocity model [16] from the geophysics community. Fig. 2 and Fig. 3 show the numerical simulation for the original Marmousi model and the smoothed Marmousi model, where a sampled window is adopted as the wave velocity. Table 7 shows the number of principal rows and columns for the matrix $E^{\pm}$, from which we observe that the low ranks depend monotonically on $\Delta t/\{\Delta x_1, \Delta x_2\}$. Fig. 4 and Fig. 5 show the numerical simulation for the BPTTI 2007 quasi-P velocity model, where two sampled windows of this model are adopted as the wave velocity. Table 8 shows the number of principal rows and columns for the matrix $E^{\pm}$, from which we observe that the low ranks depend monotonically on $\Delta t/\{\Delta x_1, \Delta x_2\}$.

From Figs. 2, 3, 4 and 5, we observe that the numerical solutions of **Method I** and **Method II** generally match well, which verifies the accuracy of **Method I**.

3.3. 3D examples

We present several examples to demonstrate the applicability of Algorithm 1 for 3D problems.

We first use two synthetic velocity models. The domain of interest is $[-2, 2]^3$, and the computational domain is $\Omega = [-4, 4]^3$, with $\eta_1 = \eta_2 = \eta_3 = 2$ for the PMLs. The initial condition is given as

$$u(x_1, x_2, x_3, 0) = \exp\left(-(x_1^2 + x_2^2 + x_3^2)/0.08\right).$$

Table 6
2D example: number of principal rows/columns of $E_{\pm}(\boldsymbol{x}, \boldsymbol{k})$, $N_i = 32$, $C_i = C$, for $i = 1, 2$. $|\Theta_1| = |\Theta_2| = 40$, $|\Gamma_1| = |\Gamma_2| = 50$.

$c(x_1, x_2) = \exp(-x_1^2 - x_2^2)$					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	7 7	7 4	6 5	5 4
	$C = 0.5$	7 6	7 4	5 5	5 4
	$C = 0.25$	6 4	6 5	5 4	4 4
$\epsilon = 10^{-7}$	$C = 1$	11 9	11 8	9 8	8 7
	$C = 0.5$	10 9	8 8	8 7	7 6
	$C = 0.25$	10 8	8 5	7 7	7 6
$\epsilon = 10^{-9}$	$C = 1$	15 12	14 12	13 10	12 8
	$C = 0.5$	14 12	13 10	11 10	10 9
	$C = 0.25$	12 9	12 9	11 9	9 7

$c(x_1, x_2) = 0.2 + \sin^2(0.5x_1 + 0.5x_2)$					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	22 21	19 16	15 14	12 13
	$C = 0.5$	18 19	15 14	11 11	11 10
	$C = 0.25$	15 16	12 12	10 10	8 8
$\epsilon = 10^{-7}$	$C = 1$	32 30	27 26	22 21	19 17
	$C = 0.5$	26 26	22 21	18 17	15 15
	$C = 0.25$	22 22	19 18	16 16	13 12
$\epsilon = 10^{-9}$	$C = 1$	39 35	34 30	30 27	26 25
	$C = 0.5$	33 32	31 28	25 24	21 21
	$C = 0.25$	29 29	25 24	22 20	18 17

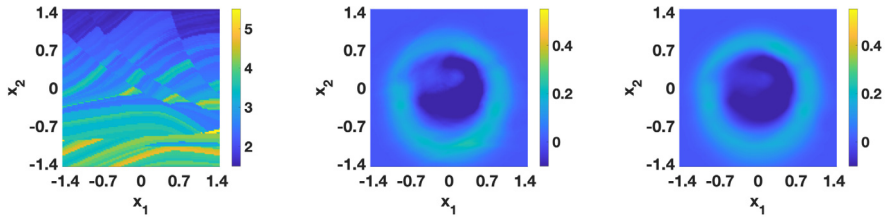


Fig. 2. 2D example with original Marmousi velocity model: real part of the wave at final time $T = 0.3$. $\Delta x_i = 0.024$. $\Delta t = \Delta x_i/16$, $C_i = 0.5$, $\eta_i = 1.5$, for $i = 1, 2$. $|\Theta_1| = |\Theta_2| = 40$, $|\Gamma_1| = |\Gamma_2| = 70$, $\epsilon = 10^{-10}$. From Left to Right: sampled window of the original Marmousi velocity model (data is re-centered for simplicity); wave by **Method I**; and wave by **Method II**, respectively. (For interpretation of the colors in the figures, the reader is referred to the web version of this article.)

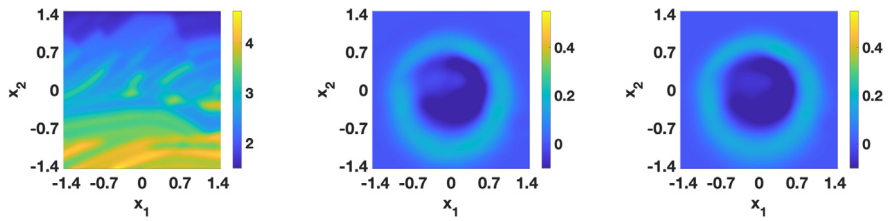


Fig. 3. 2D example with smoothed Marmousi velocity model: real part of the wave at final time $T = 0.3$. $\Delta x_i = 0.024$. $\Delta t = \Delta x_i/16$, $C_i = 0.5$, $\eta_i = 1.5$, for $i = 1, 2$. $|\Theta_1| = |\Theta_2| = 40$, $|\Gamma_1| = |\Gamma_2| = 70$, $\epsilon = 10^{-10}$. From Left to Right: a smoothed velocity from Fig. 2(a); wave by **Method I**; and wave by **Method II**, respectively.

Table 9 shows the numbers of principal rows and columns for the matrix $E^{\pm}$ for the two choices of velocity model. From this table, we also observe that the low ranks depend monotonically on $\Delta t/\{\Delta x_1, \Delta x_2, \Delta x_3\}$.

We further apply Algorithm 1 to a synthetic model constructed with the smoothed Marmousi model by choosing a window of the smoothed Marmousi velocity model in 2D and constantly extrapolating it along the x_3-axis, which is shown in Fig. 6. Fig. 7 shows the numerical simulation by both **Method I** and **Method II**.

We finally apply Algorithm 1 to a sampled region of the 3D SEG-EAGE salt and overthrust benchmark velocity model [36], shown in Fig. 8. The numerical simulation by both **Method I** and **Method II** is given in Fig. 9.

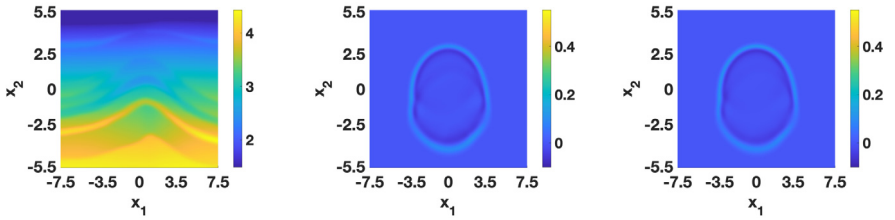


Fig. 4. 2D example with BPTTI 2007 quasi-P velocity model (Window I): real part of wave at final time $T = 1.2$. $\Delta x_i = 0.05$. $\Delta t = \Delta x_i/16$, $C_i = 0.5$, $\eta_i = 3$, for $i = 1, 2$. $|\Theta_1| = |\Theta_2| = 40$, $|\Gamma_1| = |\Gamma_2| = 70$, $\epsilon = 10^{-10}$. From Left to Right: a sampled window of BPTTI 2007 quasi-P velocity model (data is re-centered for simplicity); wavefield by **Method I**; and wave by **Method II**, respectively.

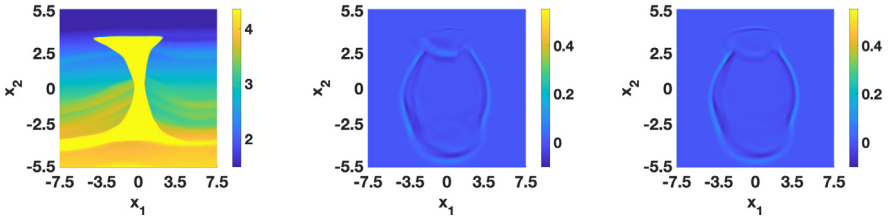


Fig. 5. 2D example with BPTTI 2007 quasi-P velocity model (Window II): real part of wave at final time $T = 1.2$. $\Delta x_i = 0.05$. $\Delta t = \Delta x_i/16$, $C_i = 0.5$, $\eta_i = 3$, for $i = 1, 2$. $|\Theta_1| = |\Theta_2| = 40$, $|\Gamma_1| = |\Gamma_2| = 70$, $\epsilon = 10^{-10}$. From Left to Right: a sampled window of BPTTI 2007 quasi-P velocity model (data is re-centered for simplicity); wavefield by **Method I**; and wave by **Method II**, respectively.

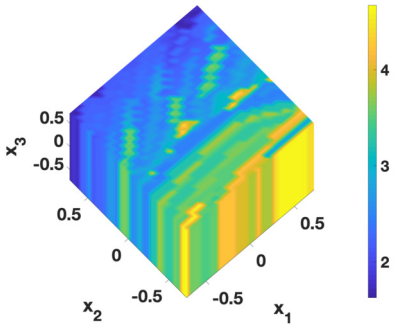


Fig. 6. Marmousi smooth velocity model extended to 3D.

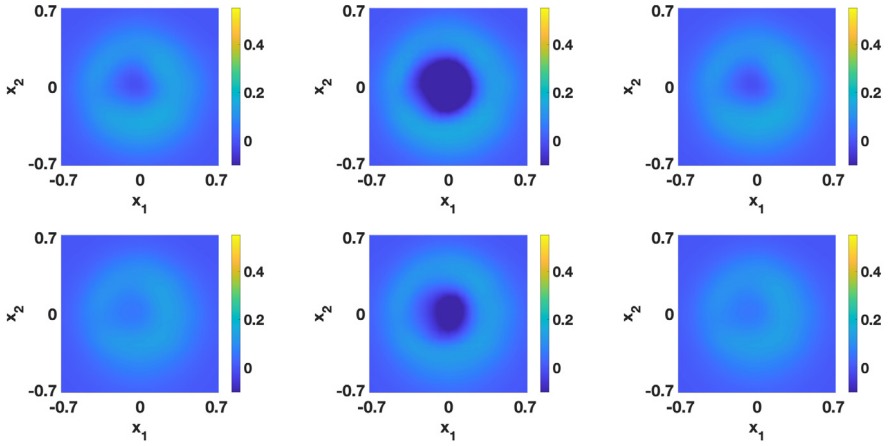


Fig. 7. 3D example: synthetic model constructed from the smoothed Marmousi model. $\Delta x_i = 0.0720$. $\Delta t = \Delta x_i/16$. $C_i = 0.5$, $\eta_i = 1.5$, for $i = 1, 2, 3$. $|\Theta_1| = |\Theta_2| = 50$, $|\Gamma_1| = |\Gamma_2| = 80$, $\epsilon = 10^{-5}$. From Left to Right: real part of the wave at $x_3 = -0.4220, 0, 0.4220$, respectively, propagated to final time $T = 0.1$. Top: **Method I**; Bottom: **Method II**.

Table 7
2D example with Marmousi velocity model: number of principal rows/columns of $E_{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$, $|\Theta_1| = |\Theta_2| = 80$, $|\Gamma_1| = |\Gamma_2| = 100$. $C_i = C$, for $i = 1, 2$.

Marmousi model (original)					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	43 40	32 27	25 22	19 18
	$C = 0.5$	35 33	27 24	20 19	16 15
	$C = 0.25$	30 27	21 20	16 16	14 13
$\epsilon = 10^{-7}$	$C = 1$	60 50	47 43	36 34	29 27
	$C = 0.5$	48 44	37 34	29 27	24 23
	$C = 0.25$	41 36	31 28	24 23	21 18
$\epsilon = 10^{-9}$	$C = 1$	73 65	62 51	49 43	41 38
	$C = 0.5$	62 55	50 47	40 37	33 30
	$C = 0.25$	54 49	41 38	33 30	27 25
Marmousi model (smoothed)					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	41 39	31 28	23 22	19 18
	$C = 0.5$	34 32	25 24	19 19	15 15
	$C = 0.25$	27 25	20 20	15 15	12 12
$\epsilon = 10^{-7}$	$C = 1$	59 51	44 41	35 33	28 28
	$C = 0.5$	47 46	36 34	28 27	23 22
	$C = 0.25$	40 38	29 29	22 22	19 17
$\epsilon = 10^{-9}$	$C = 1$	76 64	60 51	48 46	39 37
	$C = 0.5$	60 57	48 43	39 37	32 30
	$C = 0.25$	51 47	38 38	31 30	27 25

Table 8
2D example with BPTTI 2007 quasi-P velocity model: number of principal rows/columns of $E_{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$, $|\Theta_1| = |\Theta_2| = 80$, $|\Gamma_1| = |\Gamma_2| = 100$. $C_i = C$, for $i = 1, 2$.

BPTTI 2007 (Window I)					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	41 34	31 27	23 22	19 18
	$C = 0.5$	32 30	23 23	19 18	16 14
	$C = 0.25$	28 25	19 19	15 16	13 12
$\epsilon = 10^{-7}$	$C = 1$	58 41	44 40	36 31	29 27
	$C = 0.5$	47 36	36 33	28 26	23 22
	$C = 0.25$	38 30	29 28	22 22	19 17
$\epsilon = 10^{-9}$	$C = 1$	75 52	58 43	49 39	40 36
	$C = 0.5$	61 45	47 41	39 32	32 29
	$C = 0.25$	51 38	38 36	30 30	25 24
BPTTI 2007 (Window II)					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	40 34	30 27	23 22	19 18
	$C = 0.5$	33 29	24 23	18 18	16 15
	$C = 0.25$	27 24	19 19	15 16	11 12
$\epsilon = 10^{-7}$	$C = 1$	58 45	44 37	35 29	28 26
	$C = 0.5$	45 40	36 32	27 25	22 21
	$C = 0.25$	38 34	28 28	22 22	18 17
$\epsilon = 10^{-9}$	$C = 1$	72 44	61 42	46 39	40 37
	$C = 0.5$	60 49	48 42	38 33	32 29
	$C = 0.25$	51 46	39 36	31 30	26 24

From Figs. 7 and 9, we observe that the numerical solutions of **Method I** and **Method II** generally match well, which verifies the accuracy of **Method I**.

3.4. Discussion of numerical experiments

From the numerical experiments, we observe that

Table 9
3D example: number of principal rows/columns of $E_{\pm}(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{k}})$. $N_i = 32$, $C_i = C$, for $i = 1, 2$. For the exponential velocity, $|\Theta_1| = |\Theta_2| = 40$, $|\Gamma_1| = |\Gamma_2| = 50$. For the sinusoidal velocity, $|\Theta_1| = |\Theta_2| = 60$, $|\Gamma_1| = |\Gamma_2| = 75$.

$c(x_1, x_2, x_3) = \exp(-x_1^2 - x_2^2 - x_3^3)$					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	10 6	10 5	8 5	7 5
	$C = 0.5$	9 5	8 6	7 5	6 5
	$C = 0.25$	8 6	7 5	6 5	6 4

$c(x_1, x_2, x_3) = 0.2 + \sin^2(0.5x_1 + 0.5x_2 + 0.5x_3)$					
		$\Delta t = \Delta x/2$	$\Delta t = \Delta x/4$	$\Delta t = \Delta x/8$	$\Delta t = \Delta x/16$
$\epsilon = 10^{-5}$	$C = 1$	57 53	51 47	44 38	38 32
	$C = 0.5$	47 42	40 36	30 26	25 24
	$C = 0.25$	36 37	29 28	23 21	19 18

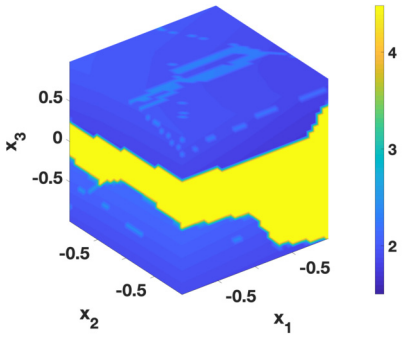


Fig. 8. A sampled region of the 3D SEG-EAGE salt and overthrust model [36]. The velocity is re-scaled and re-centered to domain $[-1, 1]^3$ for simplicity.

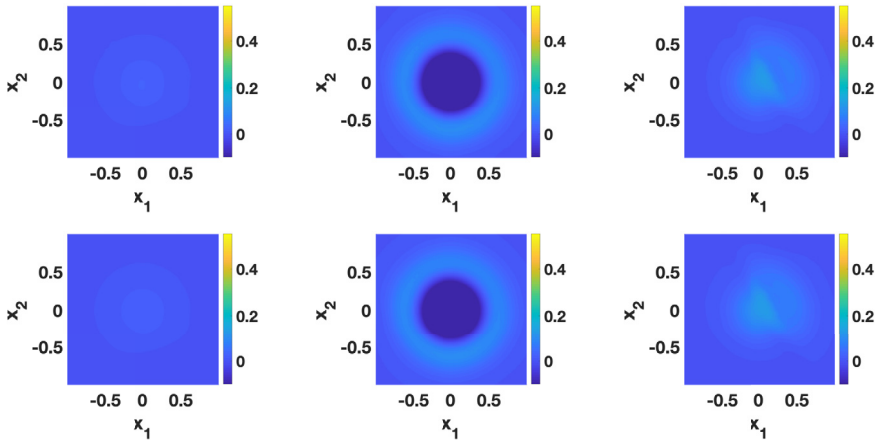


Fig. 9. 3D example: 3D SEG-EAGE salt velocity model [36]. $\Delta x_i = 0.06$. $\Delta t = \Delta x_i/16$. $C_i = 0.5$, $\eta_i = 0.5$, for $i = 1, 2, 3$. $|\Theta_1| = |\Theta_2| = 50$, $|\Gamma_1| = |\Gamma_2| = 80$, $\epsilon = 10^{-5}$. From Left to Right: real part of the wave at $x_3 = -0.69, -0.09, 0.69$, respectively, propagated to final time $T = 0.1$. Top: **Method I**; Bottom: **Method II**.

- The proposed method is first-order accurate in Δt, provided that the error from the lowrank approximations is not dominant.
- In the proposed method, we can choose $\Delta t = O(\{\Delta x_1, \Delta x_2, \Delta x_3\})$.
- The complexity per time step of the proposed method is $O(t_\epsilon N \log N)$ with N the number of spatial points and t_ϵ the low rank corresponding to a prescribed accuracy requirement $\epsilon > 0$. For given ϵ, t_ϵ depends on $\Delta t/\{\Delta x_1, \Delta x_2, \Delta x_3\}$ monotonically.
- We remark that the efficiency of the proposed method can be improved if parallel implementation is applied in that it can be parallelized with respect to the points in $\tilde{\boldsymbol{x}}^{dom}$.

4. Conclusions

We present an asymptotic Green's function method for propagating the waves through the linear scalar wave equation. The wave is split into its forward-propagating and backward-propagating parts. After that, the method which combines Huygens' principle, that provides the mechanism for propagating the wave through an integral with Green's functions, with WKBJ approximations, that provide short-time approximations for the Green's functions, is designed to propagate the forward-propagating and backward-propagating waves. The phase and amplitude in the WKBJ approximation for the Green's function are approximated analytically with second-order accurate Taylor expansions as solutions to the eikonal equation and a recurrent system of transport equations, respectively, such that the resulting integral can be approximated and evaluated by fast Fourier transform after appropriate lowrank approximations generated by a randomized pivoted QR factorization. The method is first-order accurate, and has complexity $O(t_\epsilon N \log N)$ with N the number of simulation points and t_ϵ the lowrank for a prescribed accuracy requirement $\epsilon > 0$. Numerical experiments are presented to demonstrate the methods. The method will be extended to simulate the wave propagation with a body force, and further to the seismic wave propagation through elastic wave equations in a future work.

CRediT authorship contribution statement

Jay Mayfield: Conceptualization, Formal analysis, Investigation, Methodology, Visualization, Writing – original draft, Writing – review & editing. **Yijin Gao:** Conceptualization, Methodology, Writing – review & editing. **Songting Luo:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Supervision, Visualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors thank Dr. Sergey Fomel for sharing the insights of the wave extrapolation method. The authors thank the anonymous reviewers for valuable comments that improve this work. This work was partially supported by NSF DMS 1418908 and 1719907, and Simons Foundation 714376.

References

[1] X. Antoine, A. Arnold, C. Besse, M. Ehrhardt, A. Schaedle, A review of transparent and artificial boundary conditions techniques for linear and nonlinear Schrödinger equations, Commun. Comput. Phys. 4 (2008) 729–796.
[2] I. Babuska, F. Ihlenburg, T. Strouboulis, S.K. Gangaraj, A posteriori error estimation for finite element solutions of Helmholtz' equation. Part I: the quality of local indicators and estimators, Int. J. Numer. Methods Eng. 40 (18) (1997) 3443–3462.
[3] I.M. Babuska, S.A. Sauter, Is the pollution effect of the fem avoidable for the Helmholtz equation considering high wave numbers?, SIAM Rev. 42 (3) (2000) 451–484.
[4] B. Baker, E.T. Copson, The Mathematical Theory of Huygens' Principle, AMS Chelsea Publishing, 1987.
[5] J.-P. Berenger, A perfectly matched layer for the absorption of electromagnetic waves, J. Comput. Phys. 114 (2) (1994) 185–200.
[6] A. Bourgeois, M. Bourget, P. Lailly, M. Poulet, P. Ricarte, R. Versteeg, Marmousi, model and data, in: The Marmousi Experience, 1991, pp. 5–16.
[7] M. Brack, R.K. Bhaduri, Semiclassical Physics, Addison Wesley, 1997.
[8] E. Canés, L. Demanet, L. Ying, A fast butterfly algorithm for the computation of Fourier integral operators, Multiscale Model. Simul. 7 (4) (2009) 1727–1750.
[9] P. Chand, J. Hoekstra, A review of the semi-classical WKB approximation and its usefulness in the study of quantum systems, in: Proc. of IEEE Semiconductor Advances for Future Electronics, 2001, pp. 13–19.
[10] W.C. Chew, J.M. Jin, E. Michielssen, Complex coordinate stretching as a generalized absorbing boundary condition, Microw. Opt. Technol. Lett. 15 (6) (1997) 363–369.
[11] R. Clayton, B. Engquist, Absorbing boundary conditions for acoustic and elastic wave equations, Bull. Seismol. Soc. Am. 67 (6) (1977) 1529–1540.
[12] J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex Fourier series, Math. Comput. 19 (1965) 297–301.
[13] M.G. Crandall, L.C. Evans, P.L. Lions, Some properties of viscosity solutions of Hamilton-Jacobi equations, Trans. Am. Math. Soc. 282 (2) (1984) 487–502.
[14] M.G. Crandall, P.-L. Lions, Viscosity solutions of Hamilton-Jacobi equations, Trans. Am. Math. Soc. 277 (1) (1983) 1–42.
[15] J. Crank, P. Nicolson, A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type, Math. Proc. Camb. Philos. Soc. 43 (1) (1947) 50–67.
[16] EAGE, 2007 BP Anisotropic Velocity Benchmark, https://wiki.seg.org/wiki/2007_BP_Anisotropic_Velocity_Benchmark, 2007.
[17] B. Engquist, A. Majda, Absorbing boundary conditions for numerical simulation of waves, Proc. Natl. Acad. Sci. 74 (5) (1977) 1765–1766.
[18] B. Engquist, O. Runborg, Computational high frequency wave propagation, Acta Numer. 12 (2003) 181–266.
[19] B. Engquist, L. Ying, A fast directional algorithm for high frequency acoustic scattering in two dimensions, Commun. Math. Sci. 7 (2) (2009) 327–345.
[20] S. Fomel, L. Ying, X. Song, Seismic wave extrapolation using lowrank symbol approximation, Geophys. Prospect. 61 (3) (2013) 526–536.
[21] Y. Gao, J. Mayfield, S. Luo, A second-order fast Huygens sweeping method for time-dependent Schrödinger equations with perfectly matched layers, J. Sci. Comput. 88 (2021), https://doi.org/10.1007/s10915-021-01560-6.
[22] S.A. Goreinov, N.L. Zamarashkin, E.E. Tyrtyshnikov, Pseudo-skeleton approximations by matrices of maximal volume, Math. Notes 62 (4) (1997) 515–519.
[23] S.A. Goreinov, E.E. Tyrtyshnikov, N.L. Zamarashkin, A theory of pseudoskeleton approximations, Linear Algebra Appl. 261 (1) (1997) 1–21.

[24] J.B. Keller, Semiclassical mechanics, SIAM Rev. 27 (1985) 485–504.
[25] S. Leung, J. Qian, S. Serna, Fast Huygens sweeping methods for Schrödinger equations in the semi-classical regime, Methods Appl. Anal. 21 (1) (2014) 31–66.
[26] W. Lu, J. Qian, R. Burridge, Extending Babich's ansatz for point-source Maxwell's equations using Hadamard's method, Multiscale Model. Simul. 16 (2) (2018) 727–751.
[27] S. Luo, Fast Huygens sweeping methods for time-dependent Schrödinger equation with perfectly matched layers, SIAM J. Sci. Comput. 41 (2) (2019) A877–A899.
[28] S. Luo, J. Qian, R. Burridge, Fast Huygens sweeping methods for Helmholtz equations in inhomogeneous media in the high frequency regime, J. Comput. Phys. 270 (2014) 378–401.
[29] S. Luo, J. Qian, R. Burridge, High-order factorization based high-order hybrid fast sweeping methods for point-source eikonal equations, SIAM J. Numer. Anal. 52 (1) (2014) 23–44.
[30] A. Martinez, An Introduction to Semiclassical and Microlocal Analysis, Springer, New York, 2002.
[31] V.P. Maslov, M.V. Fedoriuk, Semi-classical Approximation in Quantum Mechanics, D. Reidel Publishing Company, 1981.
[32] G. Mur, Absorbing boundary conditions for the finite-difference approximation of the time-domain electromagnetic-field equations, IEEE Trans. Electromagn. Compat. EMC-23 (4) (1981) 377–382.
[33] J. Qian, W. Lu, L. Yuan, S. Luo, R. Burridge, Eulerian geometrical optics and fast Huygens sweeping methods for three-dimensional time-harmonic high-frequency Maxwell's equations in inhomogeneous media, Multiscale Model. Simul. 14 (2) (2016) 595–636.
[34] J. Qian, S. Luo, R. Burridge, Fast Huygens' sweeping methods for multiarrival Green's functions of Helmholtz equations in the high-frequency regime, Geophysics 80 (2) (2015) T91–T100.
[35] L.S. Schulman, Techniques and Applications of Path Integration, Wiley, New York, 1981.
[36] SEG/EAGE, SEG/EAGE Salt and Overthrust Models, https://wiki.seg.org/wiki/SEG/EAGE_Salt_and_Overthrust_Models, 1997.
[37] J. Sun, S. Fomel, Y. Sripanich, P. Fowler, Recursive integral time extrapolation of elastic waves using low-rank symbol approximation, Geophys. J. Int. 211 (3) (2017) 1478–1493.
[38] A. Zagoskin, Quantum Theory of Many-Body Systems: Techniques and Applications, 2nd ed., Springer Publishing Company, Incorporated, 2014.
[39] Y. Zhang, G. Zhang, One-step extrapolation method for reverse time migration, Geophysics 74 (4) (2009) A29–A33.
[40] C. Zheng, A perfectly matched layer approach to the nonlinear Schrödinger wave equations, J. Comput. Phys. 227 (1) (2007) 537–556.