Modeling the atmospheric long-range electromagnetic weves propagation in 3D using the wavelet transform

Thomas Bonnafont

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *26/11/2020* **par :**

Thomas BONNAFONT

Modélisation de la Propagation Atmosphèrique d'Ondes Electromagnétiques sur de Longues Distances en 3D à Partir de la Transformée en Ondelettes

Modeling the Atmopsheric Long-range Electromagnetic Waves Propagation in 3D Using the Wavelet Transform

---

JURY

Funda Akleman	Professeur	Rapporteur
Alexandre Chabory	Professeur	Directeur
Rémi Douvenot	Professeur	Co-directeur
Vincent Fabbro	Ingénieur HDR	Invité
Ramakrishna Janaswamy	Professeur	Rapporteur
Ali Khenchaf	Professeur	Président
Philippe Pouliguen	Ingénieur HDR	Examinateur

École doctorale et spécialité :
GEET : Électromagnétisme et Systèmes Haute Fréquence

Unité de Recherche :
École Nationale de l'Aviation Civile - TELECOM-EMA

Directeur(s) de Thèse :
Alexandre CHABORY et *Rémi DOUVENOT*

Rapporteurs :
Funda AKLEMAN et *Ramakrishna JANASWAMY*

Les meilleurs rêves, lorsqu'ils se réalisent, en engendrent de nouveaux que notre imagination n'osait pas approcher auparavant.

– Kylian Jornet, Summit of my Life

Remerciements

Je tiens tout d'abord à remercier la Direction Générale de l'Armement (DGA) ainsi que l'École Nationale de l'Aviation Civile (ENAC) pour le financement de ces travaux de doctorat. Je tiens également à remercier M. Christophe Macabiau (responsable de TELECOM) et M. Alexandre Chabory (responsable de EMA) pour l'accueil au sein de l'équipe TELECOM, et plus particulièrement au sein de l'axe EMA.

Je tiens également à remercier chaleureusement Mme. Funda Akleman et M. Ramakrishna Janaswamy pour avoir pris le temps de relire et commenter mon manuscrit malgré le contexte difficile. Je remercie également M. Ali Khenchaf (président), M. Philippe Pouliguen (encadrant DGA) et M. Vincent Fabbro (invité) d'avoir accepté de faire partie de mon jury de thèse (à distance).

Je tiens à exprimer ma sincère gratitude envers mes deux directeurs de thèse pour m'avoir proposé ce sujet de thèse et fait confiance durant ces trois années. Tout d'abord Alexandre Chabory (Professeur à l'ENAC) pour m'avoir poussé à abandonner (un peu) les mathématiques appliquées pour faire de l'électromagnétisme. Je le remercie également pour son suivi et la relecture de mes différents travaux au cours des trois années. Et peut-être qu'un jour on sera en conférence en même temps ! Ensuite, Rémi Douvenot (Professeur à l'ENAC) pour son suivi plus quotidien et pour le temps passé à répondre à mes nombreuses questions sur les ondelettes... et sur le livre de M. Mallat. En particulier sur les résultats "triviaux" qui ont engagé de longues discussions avant et pendant le confinement. Je le remercie également pour la relecture attentive de mes travaux.

Je tiens également à remercier mes différents collègues du bâtiment F puis de ces (magnifiques) bungalows (sans fuites) pour l'agréable ambiance de travail qui a régné durant ces trois années.

Tout d'abord les permanents : Christophe Mo. (comme pour Rémi D., les pantacourts et chemisettes sont toujours dans le top des vêtements à éviter), Hélène (j'attends toujours l'invitation au loto du club de rugby de ton village !), Anaïs (à jamais les premiers à 13 défaites en LDC), Paul (même si tu fais du rougail quand je ne suis pas présent), Antoine (qui comme Thierry Roland, dit toujours ce qu'il pense !), le nouveau permanent Jérémy (pour ses connaissances cinématographiques), Anne-Christine (pour ses anecdotes qui font vivre les pauses-café).

Je remercie également les post-doctorants/doctorants que j'ai pu croiser (à la pause-café ou au Biergarten) durant ces trois années et qui ont contribué à la bonne ambiance au sein du laboratoire et à ce que cette thèse se déroule dans un contexte agréable: Anne-Marie (et ses gâteaux du lundi matin qui vont me manquer), Thomas V. (j'espère que la fermeture des bars ne t'a pas trop déprimé et t'a permis de peaufiner ta bière !), Maxandre

(qui a sans doute pu profiter du confinement pour regarder le train norvégien en direct sur YouTube), Florian (qui a dû bien fêter les deux ans du mouvement), Rémi C. (pour sa présence assidue au labo et ses connaissances sur les normes d'insonorisation), David (Voy a hablar espanol contigo cuando vuelva a Toulouse !), Capucine (pour les co-voiturages et quelques co-vélotages), et Bruno (en plein apprentissage du français !).

Je souhaite également une bonne continuation aux autres membres du laboratoire TELECOM, ainsi qu'un bon début de thèse à tous ceux qui arrivent (ne suivez pas forcément les conseils de Thomas V., Maxandre, ... à la lettre).

Je remercie également mes prédécesseurs. Tout d'abord mon co-bureau pendant une partie de ma thèse : Seif. Il est également le premier supporter de nos bleus champions du monde et un parieur averti. Puis, Johan pour ses différents conseils en début de thèse dont je n'ai pas suivi la moitié. J'aurais quand même réussi à te convertir à la course à pied ! Je remercie aussi chaleureusement Enzo pour avoir pris le temps de répondre à mes questions en début de thèse malgré sa charge de travail.

Je tiens également à remercier Daniel, Marcel, Philippe et Gabriel pour avoir été mes compagnons de courses au bord du canal durant de nombreuses pauses-midi ! Ces dernières ayant contribué à ce que cette thèse se déroule avec le moins de stress possible !

L'occasion m'est également donné de remercier mes amis qui ont toujours été là pendant ces trois ans, qu'ils soient à Paris, Manchester, Zurich, et qui ont assisté à ma soutenance à distance. Je remercie donc en particulier la NewTeam pour tous les apéros, vacances en Ardèche et le soutien pendant cette thèse sur "les Ondes 3D". Cependant non je ne peux toujours pas savoir pourquoi votre Wifi ne marche pas ou pourquoi vous ne captez pas la 6 ! *"J'suis [thésard] moi, j'suis pas là pour agiter des [antennes] et jouer [avec des box] ..."* – Arthur, Kaamelott. Je remercie également Pierre, qui me supporte depuis la maternelle et à qui je souhaite bon courage pour sa thèse !

J'adresse également mille remerciements à ma famille (qui s'est agrandie pendant cette thèse) qui m'a soutenu durant toute ma scolarité et en particulier pendant ces trois années de doctorat ! Merci également pour avoir subi les (nombreuses) répétitions de soutenance. Je tiens également à envoyer un grand merci à mon grand-père qui m'a toujours soutenu !

Enfin, je remercie celle sur qui je peux compter et qui a dû me supporter pendant la rédaction de thèse en plein confinement. Merci à toi pour ton écoute de mes répétitions (et tes corrections de mon accent très français) et pour ton soutien (même quand je fais des choses peu intelligentes comme courir la SaintéLyon avec une entorse). Maintenant c'est à moi d'être avec toi pour ta fin de thèse !

Un dernier merci à tous ceux que j'ai pu croiser pendant ces trois ans...

"L'avenir [m'] appartient et [j'] en ferais ce que [j']ai décidé que [j'] en ferais aujourd'hui, dès demain."

– Commissaire Gibert, Taxi

Contents

Introduction

Context

In the domain of radio frequencies, the long-range electromagnetic wave propagation is a major issue. Fast and accurate methods for propagation over several hundreds of kilometers are needed. Methods exist for this purpose but they are never both accurate and fast.

These methods could be used in numerous applications that concern the terrestrial, naval, and space systems used for communication, surveillance, navigation, and observation in civil or defense domains.

In the civil aviation context [1], an efficient modeling tool for long-range propagation is particularly important for the definition, design, and implementation of ground facilities for communication (voice and data links), navigation (VOR, DME, and GBAS) and surveillance (primary and secondary radar). Many applications are concerned: optimization of the location of an antenna based on the terrain topology or meteorological condition impacts on existing systems (such as ducts,...) [2, 3]. Also, in the context of green transition, the study of the impact of man-made structures such as wind turbines or solar panels on the system performances [4–6].

Besides, for the definition of new systems, the prediction of the propagation to estimate the coverage and performances in different conditions (atmospheric/meteorological) plays a major role. For civil aviation systems, the primary importance of safety renders this problem particularly difficult since the accuracy of the model must be managed for any system and condition.

The effect of the atmosphere has a huge impact on propagation. The lowest part of the atmosphere, the troposphere, contains the oxygen and gases that plays a major role in radio communications. Indeed, refraction, turbulence, absorption, and diffusion impact the propagation at radio-frequency. The diffusion of the field is due to small particles in the medium, such as ice or water drops. The molecules in the atmosphere can also absorb electromagnetic waves at these frequencies. Diffusion and absorption induce attenuation of the field. Turbulence refers to small-variations of the characteristics of the atmosphere, leading to fluctuations of the signal. Lastly, refraction induces bending or ducting effects and corresponds to the large-scale variations of the atmosphere parameters. All these effects are characterized by the refraction index [7, 8], and must be taken into account in tropospheric propagation applications such as the radar-coverage prediction.

In the context of satellite to Earth communications, for example with GNSS, another atmospheric layer has an impact on the propagation: the ionosphere. This latter cor-

responds to the layer from 75 to 1000 km above earth here ions are present due to the Sun and cosmic rays energy. This layer induces a scintillation phenomenon, *i.e.*, the rapid modification of the field due to small scale structures. This effect should also be considered [9].

Finally, ground composition also affects long-range propagation. Dielectric ground and ground waves should be accurately accounted [10, 11] for long-range propagation, for example to model Millington effect [12, 13]. The relief also plays a key role since this latter induces diffractions and reflections [11, 14, 15].

Also, for application to inverse problem, where the field propagation is used to obtain atmosphere conditions, propagation methods can be used as a direct model [16]. For example, from radar sea clutter observations, the low altitude refractivity index can be estimated [17,18]. The refractive index in the troposphere can also be derived from radio-occultation (RO) configurations [16,19,20]. This corresponds to a situation where a GNSS satellite and a low-altitude (LEO) satellite are between light-of-sight and non-light-of-sight situations because of the Earth shadow.

State of the art

First, the atmospheric description from an electromagnetic point of view is presented. Then a state-of-the-art on the propagation method is introduced going from general computation method to specific propagation method in an atmospheric environment.

In electromagnetics, the refraction is described with the refractive index. In the troposphere, this index mainly depends on temperature, vapor, and atmospheric pressure. The refraction index is used as the input for the long-range propagation model. International telecommunication union (ITU) has proposed a generic worldwide averaged model for this index [21]. This index can also be computed using weather forecast models such as the AROME model of Météo France [22] or the weather research and forecasting (WRF) model [23, 24].

For the propagation, several computational methods can be used. They can be divided into two groups: rigorous and asymptotic methods. In rigorous methods, Maxwell's equations are numerically solved, whereas, with asymptotic methods, the equations are approximated by assuming a parameter tends to 0 or infinity.

Rigorous methods include the finite element method [25], the method of moments [26] and the finite difference [27]. These methods require a thin grid size of order $\lambda/10$, where λ is the wavelength. This corresponds to a mesh size of 10 cm for a 300 MHz wave propagation. Therefore, for propagation at large-distances, over tens or hundreds of km, these methods are not suitable.

Asymptotic methods are more suitable for long-range propagation scenario. Among these methods,the major tools for long distance propagations are ray tracing [28], physical optics (PO) [29], parabolic wave equation (PWE) [30], and Gaussian/wavelet-based methods [31–33].

The ray-tracing method is based on using ray optics as an asymptotic solution of Maxwell's equations in the high-frequency regime. This concept is used in geometrical optics (GO)

[28, 34] and uniform theory of diffraction (UTD) [35, 36]. In GO, the refraction and the ground reflections are taken into account by adding and bending rays. In UTD, rays are also added to account for edge or corner diffraction. Thus for the tropospheric long-range scenario, the computation time is usually unsatisfactory since the number of required rays is very high.

Physical optics [29] can be used to simulate the propagation over smooth ground in a homogeneous or very simple (linear) atmosphere. Therefore, complex inhomogeneous atmospheres can not be treated. Moreover, its accuracy decreases with relief variations and at grazing incidence. Therefore, the method can not be used in our case.

With the paraxial approximation and assuming no backward propagation, the PWE models are good candidates to model the tropospheric long-range propagation. To solve analytically the PWE, Gaussian beams methods have been introduced [31–33, 37–40]. These methods are based on the decomposition of the fields in a set of elementary beams, which are individually propagated. To describe the radiation from an aperture, lots of decomposition have been proposed, such as multimodal [41, 42], Gabor frames [43–46] or physics-based decomposition [32, 47, 48]. This method can accurately model atmospheric effects [19] but relief with large variations can not be considered.

Using split-step Fourier (SSF) methods [49] to iteratively solve the PWE allows a wide mesh size in the propagation direction to obtain an efficient method in terms of computation time. Besides, they can take into account refraction, terrain profile, and ground wave accurately [11, 14, 15, 50, 51]. A self-consistent method in 2D has then been proposed [52] to avoid spurious error due to the *a-posteriori* discretization of the propagator.

In the 2000's, SSF has been extended to 3D [50, 53]. The method can accurately take into account refraction. A self-consistent method in cylindrical coordinates has also been proposed [54], avoiding spurious error. Nevertheless, the 3D methods are not efficient in terms of memory occupation and computation time, preventing their use for long-range propagation. Thus, there is a need for a fast and memory-efficient method in 3D.

Methods based on the wavelet decomposition have been proposed to solve the PWE. Indeed, the wavelet transform has a lower complexity than the Fourier transform allowing to speed-up the propagation. Besides, the wavelet representation allows a good sparse representation. A first wavelet-based method has been proposed to solve the PWE [55,56], which is as accurate as SSF. Nevertheless, only the sparse representation capacity was used and the propagation part uses Fourier transform leading to the same complexity as SSF. Later, a split-step wavelet (SSW) method [57] was proposed where both advantages of the wavelets were used. The propagation is only performed in the wavelet domain. Thus, a lower complexity than SSF is achieved leading to a better efficiency in terms of computation time. This Ph.D. thesis is based on these latest works.

Objectives

The objective of this Ph.D. thesis is to develop a fast and accurate method for modeling the tropospheric propagation in 3D by means of wavelets. More precisely, first, the method shall be optimized in terms of memory occupation in 2D. Then, since the method needs to be accurate, a theoretical bound of the compression error of this method shall be derived.

Finally, the method must be generalized to 3D to obtain a fast method with low memory occupation.

Therefore, three main axes toward the objective have been studied:

- Based on the split-step wavelet algorithm and on the wavelet property a local split-step wavelet method is derived to minimize the memory occupation.
- A bound for the compression error of split-step wavelet is calculated. This allows to obtain a given accuracy in a given scenario, thus the accuracy will be assessed.
- The local method is extended to 3D to obtain a fast and accurate method for the long-range propagation in 3D while keeping a minimized memory occupation. The objective is to be better in terms of computation time and memory occupation than DSSF to allow modeling large domains in 3D.
- Simulations on canonical and realistic scenarios are performed to test and compare the different methods.

Outline

This thesis is composed of five chapters:

In **Chapter 1**, the objective is to extend the discrete version of SSF to 3D Cartesian coordinates. In the first section, a state-of-the-art on the parabolic wave equation (PWE) and the dedicated numerical methods is presented. Then, the discrete version of SSF in 2D developed by Zhou *et al.* [52] is presented. By only working in the discrete domain a new propagator is obtained and a self-consistent method in the sense of Chew is obtained. Finally, the method is extended to 3D Cartesian coordinates. Numerical experiments are presented.

In **Chapter 2**, the objective is to assess the accuracy of SSW. First, a state-of-the-art on the 1D discrete wavelet transform is presented. Then, the matrix version of SSW proposed by Zhou *et al.* [57] is introduced. Using the fast wavelet transform and the sparse representation allowed by the wavelets make this method efficient in terms of computation time. Nevertheless, the wavelet-to-wavelet propagator is a serious burden in terms of memory occupation. Besides, compressions are introduced in the method to obtain a sparse representation. This induce an accumulation of error throughout the propagation. Therefore, in the last section, we develop a theoretical formula to assess this compression error. This allows choosing the compression thresholds for a given scenario and accuracy with confidence.

In **Chapter 3**, the objective is to develop a new version of SSW which is memory efficient. First, a new propagator storing only the essential information is presented. Then, the local propagation method is presented. The complexity of the method and the memory size of the propagator are studied. Numerical tests are performed to validate the method and assess whether this method is efficient in terms of computation time and memory occupation. Finally, an application to a 2D radio-occultation scenario is proposed.

In **Chapter 4**, the objective is to extend SSW to 3D. Firstly, a state-of-the-art on the 2D discrete wavelet transform is presented before its use for the 3D propagation method.

Secondly, the matrix version of SSW is generalized to 3D. We show that the method is not relevant in this case since the memory occupation prevents any operational use, even for a reasonable size domain. Finally, the local strategy is then extended to 3D and we show that method is efficient in terms of complexity and memory occupation by comparison with DSSF.

In **Chapter 5**, the objective is to test the local version of SSW in 3D and to use it in some realistic scenario. First, the method is tested in free-space for validation and to show that the compression error formula is still relevant in 3D. Second, experiments with refraction where 3D effects are present are performed. Third, tests of propagation over y-invariant planar grounds with different compositions are performed to validated SSW with the local image theorem. An application to model the Millington effect is proposed. Finally, the problem of y-variant ground composition is addressed and preliminary theory and results are presented.

Chapter 1

Discrete Split-Step Fourier

Parabolic wave equation solved by means of SSF is among the most used methods for modeling the long-range propagation while considering the effects of the ground, the atmosphere and the relief. This method is widely used in various domains such as communication, surveillance, navigation and observation.

The aim of this chapter is to recall the 2D version of DSSF and then to extend DSSF to 3D in Cartesian coordinates.

In Section 1.1, a brief state-of-the-art on the parabolic equation is presented. In Section 1.2, the 2D Cartesian DSSF method is recalled. First, the configuration is exposed. Then, the continuous problem is formulated. This latter is discretized *a priori* and the 2D DSSF method is introduced. Some numerical experiments are performed to test the method and to show an example of a long-range propagation with relief and refraction. In Section 1.3, we derive the 3D Cartesian DSSF. As in 2D, we first define the configuration and the continuous problem. Secondly, the discrete problem is formulated. Then, we derive the discrete propagator and introduce the DSSF numerical scheme. Finally, some numerical experiments are performed to test the method. In Section 1.4, we conclude and recall some limitations.

1.1 State of the art

Parabolic wave equation (PWE) models have been introduced in 1946 [58] for electromagnetic propagation in inhomogeneous environments. Valid along a paraxial direction and neglecting the backward propagation, they allow to model the propagation in complex environments.

First numerical schemes in the '70s were based on finite difference (FD) [11, 59, 60] because they are straightforward to implement. The PWE approach became very attractive for problems with complex atmospheric conditions and relief with the Fourier split-step scheme introduced by Tappert in the '80s in acoustics [61,62]. The method was proved to be accurate to predict propagation losses in medium with vertically and horizontally inhomogeneous profiles. The same decade, the electromagnetic version of split-step Fourier (SSF) was introduced [3,30,63]. Split-step methods are more efficient than FD since they

allow a larger horizontal mesh. They are thus widely used in long-range propagation in acoustic and radio-wave propagation [49, 64–67].

Initially developped for 2D configurations, solving the PWE with SSF allows to take into account relief with various methods [14, 15, 68], ground waves [69, 70], refraction with phase screens [11] and impedance ground condition with the MFT [30]. The SSF-MFT method was shown to introduce spurious solutions [71]. To overcome this problem, the SSF-DMFT method was introduced in 1996 [71, 72].

With the improvement of computational abilities, the PWE and SSF were extended to 3D cases in the '2000s. As for 2D, first for the propagation of acoustic waves [73–76] and later for radio wave propagation in 3D Cartesian coordinates [50, 51, 53, 70].

However, in [54], SSF-DMFT was shown to be inconsistent with the concept of self-consistency in electromagnetics as derived by Chew [77]. To avoid the rise of spurious solutions, an *a-priori* discretization of the Helmholtz equation must be used. In SSF the propagation is discretized *a posteriori*, possibly inducing spurious solutions as shown in [52]. A discrete version of SSF (DSSF) in 2D and 3D cylindrical coordinates has been developed in [52, 54]. By discretizing *a priori* the propagation equation and by deriving a complete method from these discrete equations a self-consistent method has been obtained. This method is introduced in the next section.

1.2 2D Cartesian DSSF

In this section, the discrete SSF (DSSF) method for the propagation derived in [52] is recalled. The wave propagation is discretized *a priori* to obtain self-consistency following Chew's [77] discrete theory of electromagnetism.

First, the continuous problem is modeled, then the obtained wave equation and the boundary conditions are discretized. With this model, the discrete problem is iteratively solved using DSSF.

1.2.1 Configuration

In this section, the problem is treated in Cartesian coordinates. We assume that the field is invariant along y. Thus, the aim is to compute the forward propagation for a 2D problem.

The main direction of propagation corresponds to x, and z corresponds to the altitude axis.

The frequency is given by $f = \omega/2\pi$, with ω the angular frequency. The field is assumed to have an $\exp(j\omega t)$ time dependence which is omitted. We assume the field is known at $x = 0$ and the source is placed at $x \leq 0$. We aim at computing the forward propagation over the ground, *i.e.*, for $x \geq 0$ and $z \geq 0$.

We consider an inhomogenous atmosphere characterized by its refractive index n. Since n is close to 1, we use the refractivity defined by

$$N = 10^6 \left(n - 1\right), \tag{1.1}$$

for conveniency. Besides the refractivity can be expressed using [78]

$$N = \frac{77.6}{T}\left(P + 4810\frac{e}{T}\right), \tag{1.2}$$

with P the atmospheric pressure (hPa), T the absolute temperature (K) and e the water pressure (hPa). This expression can be used with an error of less than 0.5% for frequencies up to 100 GHz [11].

Since the long-range propagation is studied, the curvature of the Earth should be taken into account. An Earth flattening transform [11] can be performed by replacing n by the modified refractivity m. For the same reason that we use N instead of n, M replaces m and is defined by

$$M = N + 10^6\frac{z}{R_e}, \tag{1.3}$$

with R_e the Earth radius. M can be used as an input for long-range propagation models with varying atmosphere. Typical values for various places on Earth can be found in [21]. Alternatively, numerical weather model can also be used to obtain realistic refractivity profiles such as the outputs of the Weather Research and Forecasting (WRF) [23] or of the AROME model of Météo France [22].

The domain is of finite size for numerical reasons. The propagation is computed up to $x = x_{\max}$ in a domain of size $z \in [0, z_{\max}]$. The domain is also discretized along x and z. The following grid is used

$$\begin{aligned} x_{p_x} &= p_x\Delta x \qquad \text{for } p_x \in \{0, \cdots, N_x - 1\}, \\ z_{p_z} &= p_z\Delta z \qquad \text{for } p_z \in \{0, \cdots, N_z - 1\}, \end{aligned} \tag{1.4}$$

with N_x and N_z the number of points for the x-axis and z-axis, respectively, and Δx and Δz the steps along x and z, respectively. This configuration is plotted in Figure 1.1. The discretized version of $u(x, z)$ along z is denoted as $u_x[p_z]$.

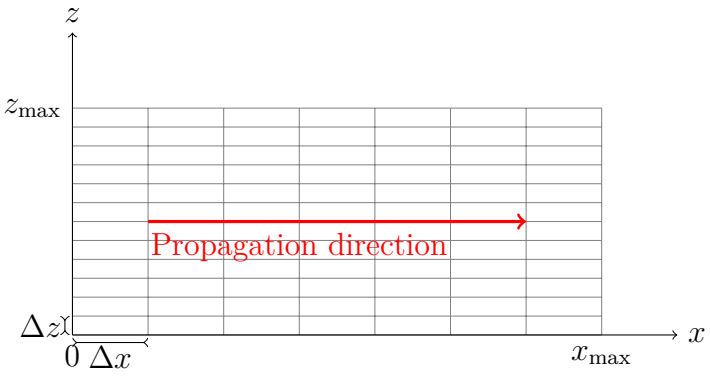


Figure 1.1: Discrete configuration.

Now that the configuration has been presented, the discrete split-step Fourier method is introduced for computing the long-range propagation.

1.2.2 The 2D DSSF method

1.2.2.a Continuous formulation

With the configuration detailed in Section 1.2.1, the electric field is a solution of the 2D Helmholtz equation

$$\nabla^2 \boldsymbol{E}(x,z) + k_0^2 n^2(x,z)\boldsymbol{E}(x,z) = 0, \tag{1.5}$$

with $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2}$ and $k_0 = 2\pi/\lambda_0$ the vacuum wavenumber.

With respect to the z-axis, the field can be decomposed in transverse electric (TE) and transverse magnetic (TM) components. In a TE case, the electric field $\boldsymbol{E}$ only has a E_y non-zero component. In a TM case, then the magnetic field $\boldsymbol{H}$ only has a H_y non-zero component.

For conciseness, we define an appropriate variable ψ working for both cases which is given by

$$\psi(x,z) = \begin{cases} E_y(x,z) \text{ in TE case,} \\ H_y(x,z) \text{ in TM case.} \end{cases} \tag{1.6}$$

From now on, a TE case is assumed, but the same calculations can be performed for the TM case.

Then, the reduced field u with reduced phase variation is introduced

$$u(x,z) = \exp(jk_0 x)\,\psi(x,z). \tag{1.7}$$

This latter varies slowly in range compared to ψ, thus a larger mesh size along x is possible. We now work with u.

Introducing u in the Helmholtz equation, we obtain

$$\frac{\partial^2 u(x,z)}{\partial x^2} - 2jk_0\frac{\partial u(x,z)}{\partial x} + \frac{\partial^2 u(x,z)}{\partial z^2} + k_0^2(n^2(x,z)-1)u(x,z) = 0. \tag{1.8}$$

The continuous formulation for the propagation of the reduced field is thus obtained.

We now introduce the boundary conditions for the continuous formulation. The ground can either be a planar electrical conductor (PEC), a planar magnetic conductor (PMC) or a dielectric ground. Thus, three different conditions are derived :

- If we assume a PEC at $z = 0$, then the reduced field fulfills at $z = 0$ a Dirichlet boundary condition
$$u|_{z=0} = 0. \tag{1.9}$$

- If we assume a PMC at $z = 0$, then the first derivative of u along z is null and the reduce field fulfills a Neumann boundary condition
$$\frac{\partial u}{\partial z}|_{z=0} = 0. \tag{1.10}$$

- If we assume a dielectric ground at $z = 0$, then this condition can be approximated by the impedance ground condition given in [79]. Thus u fulfills the Leontovitch condition
$$\frac{\partial u}{\partial z}|_{z=0} + \alpha u|_{z=0} = 0, \tag{1.11}$$

corresponding to the local version of the impedance boundary condition, where

$$\alpha = -jk_0 \cos(\theta_i) \frac{1-\Gamma}{1+\Gamma}, \tag{1.12}$$

with θ_i the angle of incidence of the plane wave and Γ the reflection coefficient. The surface roughness can be taken into account by modifying the smooth surface reflection coefficient as detailled in [67, 80–84].

Now that the continuous problem has been fully described, the same derivation is made after the discretization of (1.8) to obtain the DSSF method.

1.2.2.b Discrete formulation

The Helmholtz equation fulfilled by u is discretized along z with a finite-difference approximation of the differential operator along z. The derivative with respect to x is kept and the partial derivative equation is solved afterwards. The discrete version of the Helmholtz equation for the reduced field is given by

$$\frac{\partial^2 u_x[p_z]}{\partial x^2} - 2jk_0 \frac{\partial u_x[p_z]}{\partial x} + \mathbf{d}_z^2 u_x[p_z] + k_0^2 (n_x^2[p_z] - 1) u_x[p_z] = 0, \tag{1.13}$$

with $n_x[p_z]$ the discretized refractive index, $p_z \in [0, N_z - 1]$ and where

$$\mathbf{d}_z^2 u_x[p_z] = \frac{u_x[p_z+1] - 2u_x[p_z] + u_x[p_z-1]}{\Delta z^2}. \tag{1.14}$$

Using the PWE formalism, equation (1.13) is then factorized in forward and backward propagations, which yields

$$\left\{ \frac{\partial}{\partial x} - jk_0 \left(1 - \sqrt{1+\mathcal{Q}}\right) \right\} \left\{ \frac{\partial}{\partial x} - jk_0 \left(1 + \sqrt{1+\mathcal{Q}}\right) \right\} u_x[p_z] = 0, \tag{1.15}$$

where $\sqrt{1+\mathcal{Q}}$ corresponds to the pseudo differential operator defined by

$$\sqrt{1+\mathcal{Q}} \left(\sqrt{1+\mathcal{Q}} \left(u_x[p_z] \right) \right) = (1+\mathcal{Q}) \left(u_x[p_z] \right), \tag{1.16}$$

with $\mathcal{Q}$ the operator

$$\mathcal{Q} = \frac{1}{k_0^2} \mathbf{d}_z^2 + n_x[p_z]^2 - 1. \tag{1.17}$$

Since we only want to account for the forward propagation, we finally obtain a first order derivative equation along x for the reduced field

$$\frac{\partial u_x[p_z]}{\partial x} = jk_0 \left(1 - \sqrt{1+\mathcal{Q}}\right) \left(u_x[p_z] \right). \tag{1.18}$$

We now discretize the ground conditions to obtain the complete problem formulation. The discrete counterpart of the three considered boundary conditions are given by

$$u_x|_{p_z=0} = 0 \qquad \text{for a PEC,} \tag{1.19}$$

$$\mathbf{d}_z u_x|_{p_z=0} = 0 \qquad \text{for a PMC,} \tag{1.20}$$

$$\mathbf{d}_z u_x|_{p_z=0} + \alpha u_x|_{p_z=0} = 0 \qquad \text{for an impedance condition.} \tag{1.21}$$

The complete discrete model has thus been defined.

1.2.2.c Apodisation

For obvious numerical reasons, the domain is limited in altitude to $z_{\max}$. At the top, the energy propagates to infinity and the boundary condition must be transparent. Many methods exist, as apodization layer [11] to avoid reflections over $z_{\max}$.

Here, an apodization layer of the size N_z is used in the spatial domain. The Hanning window is applied in the apodization layer as a multiplication by a diagonal matrix denoted by $\mathbf{H}$, which diagonal elements are defined by

$$\mathbf{H}[p_z, p_z] = \begin{cases} 1 \text{ if } p_z \in [0, N_z], \\ \frac{1}{2}\left(1 + \cos\left(\pi\left(\frac{p_z \Delta z}{z_{\max}} - 1\right)\right)\right) & \text{if } p_z \in [N_z, 2N_z]. \end{cases} \tag{1.22}$$

Note that, both derivatives at N_z and $2N_z$ are zeros which ensures a smooth transition. This is illustrated in Figure 1.2.

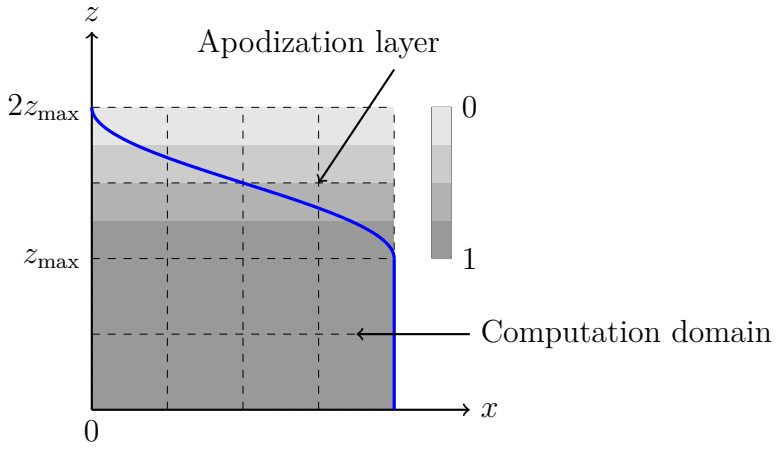


Figure 1.2: Apodisation for the domain truncation.

The Hanning window has a good accuracy for paraxial propagation and is widely used. One can use other absorbing layers as mentioned in [11].

Applying the apodization ensures that no field reaches the top boundary $2z_{\max}$ and so any convenient condition can be applied at $p_z = 2N_z$.

A convenient method to solve the PWE is the split-step method that splits the propagation into two terms.

1.2.2.d Split-step method

To compute the forward propagation, we iteratively solve (1.18) using a split-step method [11, 30, 49]. The idea is to split the propagation into two steps as follows:

- First the free-space propagation on Δx with $n = 1$ that corresponds to the propagation itself. This step is computed in the Fourier domain. Besides the ground boundary conditions are taken into account in this step. This step is comprehensively described in Section 1.2.2.e.

- Second, the environment effects are taken into account. The refraction is added with a phase screen, corresponding to a multiplication in the spatial domain. The relief is taken into account with various models given in Section 1.2.2.f.

In the following, we introduce the paraxial approximation resulting from the approximation of the pseudo-differential operator and the split-step formalism. The idea is to treat the propagation and the effects of the environment separately. We introduce next the paraxial approximation, allowing to split the $\mathcal{Q}$ into two independent operators.

With the paraxial approximation, we assume $\|\mathbf{d}_z^2\| \ll 1$ and we also have that n satisfies $n-1 \ll 1$. We aim at splitting the propagation into two terms:

- a first step corresponding to the propagation in free space, where n is assumed to be constant;
- a second step where the refraction is taken into account.

Various approximations of the pseudo-differential operator allow splitting efficiently the propagation.

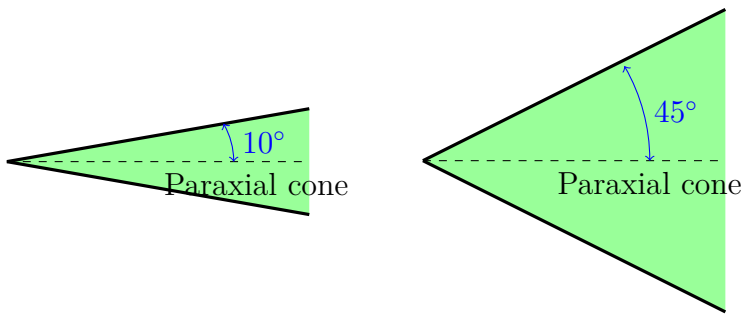
The most simple approximation is based on the first-order Tayler series expansion of the square root operator

$$\sqrt{1+\mathcal{Q}} \simeq 1 + \frac{\mathcal{A}}{2} + \frac{\mathcal{B}}{2}, \tag{1.23}$$

where

$$\begin{aligned} \mathcal{Q} &= \mathcal{A} + \mathcal{B}, \\ \mathcal{A} &= \frac{1}{k_0^2}\mathbf{d}_z^2, \\ \mathcal{B} &= n_x[p_z]^2 - 1. \end{aligned} \tag{1.24}$$

With this approximation, the propagation and the effect of the refraction are treated separately and the standard PWE (SPWE) is obtained. This later is valid for small angles around x ($\leq 10°$) [11, 85], as shown in Figure 1.3 (a),



(a) Small-angle approximation.

(b) Wide-angle approximation

Figure 1.3: Angles of validity for the paraxial approximations.

A better approximation has been introduced by Feit and Fleck [86]. The square-root operator is split efficiently to be incorporated in a split-step algorithm as

$$\sqrt{1+\mathcal{Q}} \simeq \sqrt{1+\mathcal{A}} + \sqrt{1+\mathcal{B}} - 1. \tag{1.25}$$

This approximation is exact in free space. It was shown to have a good accuracy for angles up to 45° around the propagation direction [11, 85] as illustrated in Figure 1.3 (b). From now on, only this one is used and the discrete forward propagation equation to solve is

$$\frac{\partial u_x[p_z]}{\partial x} = -j\left(\sqrt{k_0 + \mathbf{d}_z^2} - k_0\right) u_x[p_z] - jk_0\left(n_x[p_z] - 1\right) u_x[p_z]. \tag{1.26}$$

This equation is iteratively solved at increasing distances as follows

$$u_{x+\Delta x}[p_z] = \mathbf{R}\mathbf{D}u_x[p_z], \tag{1.27}$$

where

$$\mathbf{D} = \exp\left(-jk_0\left(\sqrt{1 + \mathbf{d}_z^2/k_0^2} - 1\right)\right), \tag{1.28}$$

and $\mathbf{R}$ is the diagonal operator, which diagonal elements are defined by

$$\mathbf{R}[p_z, p_z] = \exp\left(-jk_0\left(n_x[p_z] - 1\right)\right). \tag{1.29}$$

The method is illustrated in Figure 1.4, where the field is propagated from one step to another in a free-space layer (in blue) and then a phase screen is applied to take into account the refraction.

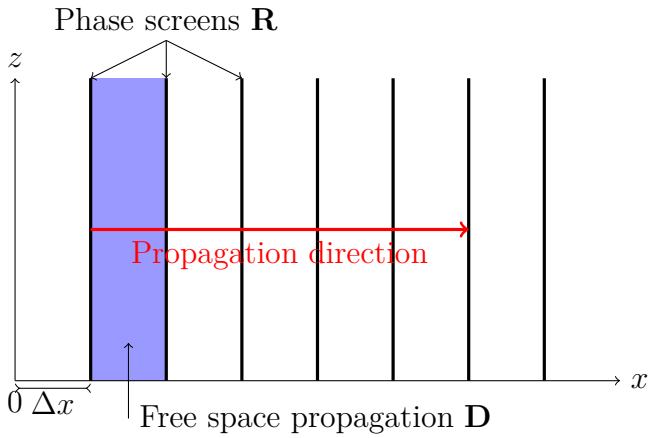


Figure 1.4: Split-step method.

Now that the split-step method has been described, a spectral transform along z is introduced in order to diagonalize the discrete second derivative $\mathbf{d}_z^2$ with consistent discrete boundary conditions.

1.2.2.e Spectral transforms

Since the propagation term $\mathbf{D}$ is difficult to compute in the spatial domain, a spectral transform along the z axis is used to diagonalize this latter. In this section, the refractive index is assumed to be $n = 1$.

The goal here is to diagonalized $\mathbf{d}_z^2$, *i.e.*, we search k_z and u such that

$$\mathbf{d}_z^2 u = -k_z^2 u. \tag{1.30}$$

To introduce the method, we assume periodic boundary conditions. In this case, the operators in z are diagonalized via a discrete Fourier Transform (DFT) and the eigenvectors are the Fourier atoms

$$u = \exp\left(-j\frac{2\pi p_z q_z}{N_z}\right). \tag{1.31}$$

The discrete spectral transform is the discrete Fourier transform (DFT) denoted by $U_x[\cdot] = \mathbf{T}_s u_x[\cdot]$ and defined by

$$U_x[q_z] = \frac{1}{\sqrt{N_z}} \sum_{p_z=0}^{N_z-1} u_x[p_z] \exp\left(-j\frac{2\pi p_z q_z}{N_z}\right), \tag{1.32}$$

with $q_z \in [0, \cdots, N_z - 1]$. The inverse spectral transform denoted $u_x[\cdot] = \mathbf{T}_s^{-1} U_x[\cdot]$ is computed as

$$u_x[q_z] = \frac{1}{\sqrt{N_z}} \sum_{p_z=0}^{N_z-1} U_x[p_z] \exp\left(j\frac{2\pi p_z q_z}{N_z}\right). \tag{1.33}$$

We now derive the formulation of the spectral propagator to obtain k_z

$$\begin{aligned}
\Delta z^2 \mathbf{d}_z^2 \left(\exp\left(-2j\frac{\pi p_z q_z}{N_z}\right)\right) &= \exp\left(-2j\frac{\pi(p_z+1)q_z}{N_z}\right) - 2\exp\left(-2j\frac{\pi p_z q_z}{N_z}\right) \\
&\quad + \exp\left(-2j\frac{\pi(p_z-1)q_z}{N_z}\right) \\
&= \left(\exp\left(-2j\frac{\pi q_z}{N_z}\right) - 2 + \exp\left(2j\frac{\pi q_z}{N_z}\right)\right)\exp\left(-2j\frac{\pi p_z q_z}{N_z}\right) \\
&= -4\sin^2\left(\frac{\pi q_z}{N_z}\right)\exp\left(-2j\frac{\pi p_z q_z}{N_z}\right).
\end{aligned} \tag{1.34}$$

As in [52], we obtain with the DFT and the periodic boundary conditions

$$k_z = \frac{2}{\Delta z}\sin\left(\frac{\pi q_z}{N_z}\right). \tag{1.35}$$

Applying the spectral transform to the forward propagation (1.26), the following differential equation along x is obtained

$$\left(\frac{\partial}{\partial x} + j\left(\sqrt[\circ]{k_0^2 - k_z^2} - k_0\right)\right) U_x[q_z] = 0, \tag{1.36}$$

where $\sqrt[\circ]{\cdot}$ corresponds to the operator giving the square root with negative imaginary part, *i.e.*,

$$\sqrt[\circ]{k_0^2 - k_z^2} = \begin{cases} \sqrt{k_0^2 - k_z^2} & \text{if} \quad |k_z| \leq k_0, \\ -j\sqrt{k_z^2 - k_0^2} & \text{if} \quad |k_z| > k_0. \end{cases} \tag{1.37}$$

In homogeneous atmosphere the propagated spectrum on Δx is given by

$$U_{x+\Delta x}[q_z] = \mathbf{P}_s[q_z, q_z] U_x[q_z], \tag{1.38}$$

with $\mathbf{P}_s$ a diagonal matrix, which diagonal elements are given by

$$\mathbf{P}_s[q_z, q_z] = \exp\left(-j\Delta x\left(\sqrt[\circ]{k_0^2 - k_z^2} - k_0\right)\right). \tag{1.39}$$

The corresponding diagonal operator is denoted $\mathbf{P}_s$.

To take the PEC ground into account, the discrete sinus transform (DST) must be used instead of the DFT, to diagonalize the operator with this boundary conditions along z. With the DST, only k_z is replaced with

$$k_z = \frac{2}{\Delta z} \sin\left(\frac{\pi q_z}{2N_z}\right). \tag{1.40}$$

For a PMC, a cosine transform (DCT) can be used to take into account the Neumann boundary condition.

For a dielectric ground, the discrete mixed Fourier transform (DMFT) [71] is used. The reduced field u is first replaced by

$$w_x[p_z] = \frac{u_x[p_z+1] - u_x[p_z-1]}{2\Delta z} + \alpha u_x[p_z]. \tag{1.41}$$

Note that other approximations of the first derivative along z can be used [72]. The impedance boundary condition for w becomes

$$\begin{aligned} w_x[0] &= 0, \\ w_x[N_z] &= 0. \end{aligned} \tag{1.42}$$

Thus, w follows a Dirichlet condition at the ground boundary and the previous method can be used. However, this transform introduces additional constants corresponding to the ground and roof waves. These ones are propagated with separate propagators. The details of the DMFT transform are given in Appendix A.

1.2.2.f Irregular relief modeling

With the PWE model, irregular relief with slight slopes can be considered. The easiest way to compute the effect of the relief is the staircase model described in [11] and comprehensively introduced in [52], Appendix B. Other methods [14, 15, 68] can be considered. For example in [15] a curvilinear transformation is performed to take into account the terrain with better accuracy. The operator denoted by $\mathbf{L}$ models the relief in the space domain.

Now that all the operations for the propagation with DSSF have been fully described, the complete algorithm is given in the next section.

1.2.2.g Algorithm of DSSF

Aggregating all the steps described in the previous sections, the propagation from p_x to $p_x + 1$ is computed as

$$u_{p_x+1} = \mathbf{HRLT}_s^{-1}\mathbf{P}_s\mathbf{T}_s u_x, \tag{1.43}$$

where the different operators have been defined in the previous sections.

In this algorithm, the discretization is applied *a priori* contrary to what is done in SSF [30] where the discretization is performed *a posteriori*. Thus, a self-consistent algorithm in

the sense of Chew [77] is obtained, avoiding the rise of spurious solutions. Numerical experiments are performed in [52] to show the advantages of the self-consistent method over SSF.

Now the complexity of the method is discussed. First a FFT of complexity $O\left(N_z' \log\left(N_z'\right)\right)$, with $N_z' = N_z + N_{\text{apo}}$ is applied to the field. Then the propagation which amounts to the multiplication of a diagonal matrix by a N_z'-elements vector is computed with a complexity $O\left(N_z'\right)$. Then, the field is expressed in the spatial domain with an inverse FFT of complexity $O\left(N_z' \log\left(N_z'\right)\right)$. Finally, the phase screen and the apodisation are of complexity $O\left(N_z'\right)$. Overall the complexity for N_x iterations is driven by

$$O\left(N_x N_z' \log\left(N_z'\right)\right). \tag{1.44}$$

Some numerical tests are performed in the following section to validate DSSF in a simple case and to show an example of long-range propagation.

1.2.3 Numerical tests

In this section two numerical experiments are performed.

For both numerical tests the source is a complex source point (CSP) [31] located at a complex position $(x_s, y_s, z_s) \in \mathbb{C}\times\mathbb{R}\times\mathbb{R}$ with $x_s = x_{w0} - jk_0W_0^2/2$, where x_{w0} corresponds to the position of the source and W_0 to its width. An analytical expression of its radiation in 2D is given by

$$u(r) = A\frac{j}{4}H_0^{(2)}\left(k_0 r\right), \tag{1.45}$$

with A a normalization constant, $H_0^{(2)}$ the Hankel function and

$$r = \sqrt{(x-x_s)^2 + (z-z_s)^2}, \tag{1.46}$$

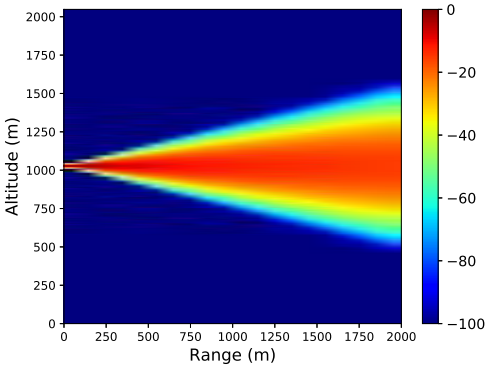
where $\text{Re}\left(\sqrt{(x-x_s)^2 + (z-z_s)^2}\right) > 0$.

1.2.3.a Free space simulation to compare SSF and DSSF

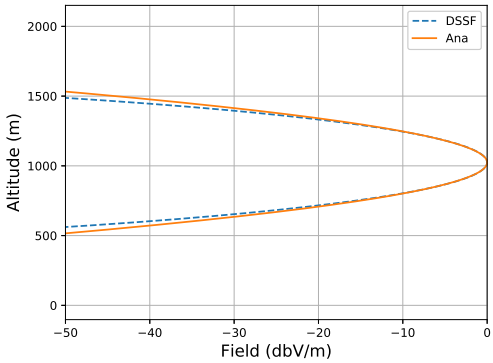
For this test, the complex source point at $f_0 = 300$ MHz is placed at $x_{w0} = -50$ m with $W_0 = 3$ m and $z_s = 1024$ m. The propagation is computed on 2000 m with $\Delta x = 100$ m. The domain is of vertical size 2048 m. The high is such that the propagation never reaches the ground.

First, a test is performed with $\Delta z = 0.5\lambda = 0.5$ m. The field u, normalized with respect to its maximal value, is given in Figure 1.5 (a). The fields normalized with respect to their maximal values computed with DSSF and the analytical expression for the CSP are plotted at the last iteration in Figure 1.5 (b). We can see that below -20 dB from the maximum both results slighly differ due to the finite-difference approximation applied to the propagation equation.

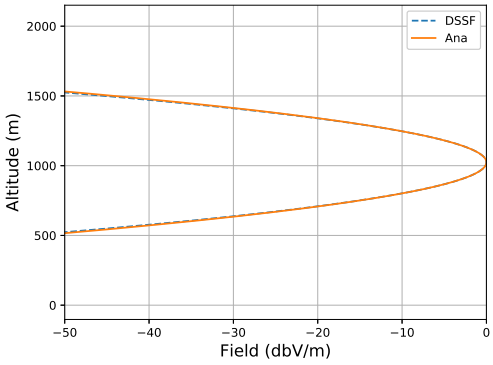
Second, a test is performed where the vertical mesh is reduced to $\Delta z = 0.2\lambda = 0.2$ m, the normalised fields at the last iteration obtained with DSSF and the analytic expression for



(a) Normalised field (dB) computed with DSSF with $\Delta z = 0.5$ m.



(b) Normalized final fields (dB) obtained with DSSF and analytical formula with $\Delta z = 0.5$ m.



(c) Normalized final fields (dB) obtained with DSSF and analytical formula with $\Delta z = 0.2$ m.

Figure 1.5: Free space propagation of a 2D CSP at altitude 1024 m.

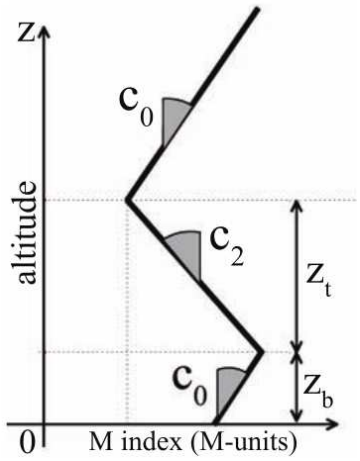


Figure 1.6: Trilinear model of the atmosphere.

the CSP are plotted in Figure 1.5. Note that the difference is reduced significantly and that both fields now match.

More tests with DSSF and its difference with SSF are performed in [52]. The main conclusion is that the steps should be lower with DSSF due to the additional finite-difference. Nevertheless, the self-consistency avoids spurious solutions in some cases as shown with a very dry ground in [52].

Now that DSSF has been validated in a simple case. A more realistic one with atmosphere and relief is now performed.

1.2.3.b Realistic test case

In this section, a test of DSSF in a case of atmospheric long-range propagation with relief is performed.

The propagation of a complex source point (CSP) in a domain with a trilinear atmosphere and two triangular reliefs is computed. The CSP parameters are: a frequency $f = 300$ MHz, with coordinates $x_{w0} = -50$ m and $z_s = 50$ m, with a waist size of $W_0 = 5$ m. The value of A in this case is 1 dBV/m.

We consider a refractive duct modelled by a trilinear atmosphere [87], see Figure 1.6. The parameters for the refraction index are $M_0 = 330$ M-units, $z_b = 241$ m, $z_t = 391$ m, with gradients $c_0 = 0.118$ M-units/m and $c_2 = -0.5$ M-units/m.

The relief is chosen as 2 small triangles of heights 100 m and 200 m. The impedance ground is of parameters $\epsilon_r = 20.0$ and $\sigma = 0.02$ S/m.

The domain is of size $x_{\max} = 100$ km in horizontal and $z_{\max} = 2048$ m in vertical. An apodization window is added. The grid size is 200 m in horizontal and 0.5 m in vertical.

The electric field computed with DSSF is represented in Figure 1.7. The effects of both the atmosphere and the relief (shadowing, reflection, and diffraction) can be observed. This result has been compared with SSF in [52] for validation.

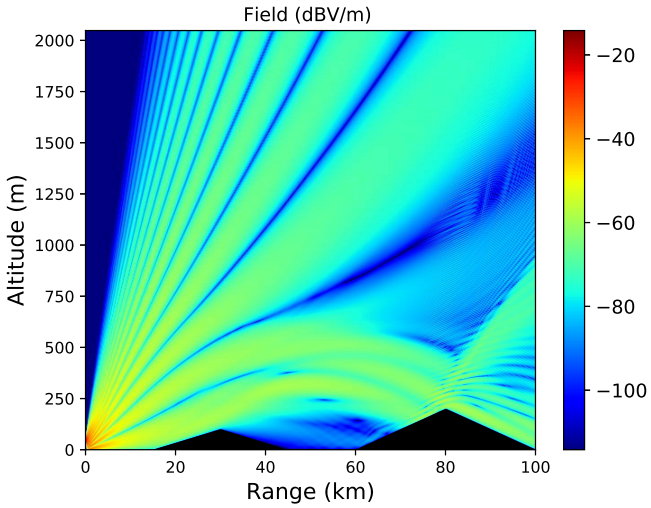


Figure 1.7: Electric field (dBV/m) computed with DSSF along the distance.

Now, that the 2D version of DSSF has been recalled, we extend it to 3D in Cartesian coordinates.

1.3 3D Cartesian DSSF

In [52,54], a self-consistent version of DSSF in cylindrical coordinates has been developed. Here, we develop the version in Cartesian coordinates. We assume no relief and only a scalar 3D problem is treated here. In the literature we can find some methods to take the relief into account [50, 51, 53], however, this task deserves a deep study which is beyond the scope of this Ph.D. Besides, we first want to obtain a fast method for the scalar 3D problem, before adding more physical phenomenon, such as the relief.

1.3.1 Configuration

The aim is to compute the propagation in the (x, y, z) Cartesian coordinate system. As for the 2D case, x is the main direction of propagation and the field is assumed to have an $\exp(j\omega t)$ time dependance. The field is assumed to be known at $x = 0$ with the source placed at $x \leq 0$. The forward propagation is computed in the region $x \geq 0$ and over the ground with $z \geq 0$, see Figure 1.8.

In a preliminary step, we assume that n is constant, *i.e.*, the atmosphere is homogeneous. The refraction condition is taken into account in Section 1.3.3.

The considered domain is of finite size with $x \in [0, x_{\max}]$, $y \in [-y_{\max}/2, y_{\max}/2]$ and $z \in [0, z_{\max}]$. The domain is discretized along the three directions as

$$\begin{aligned} x_{p_x} &= p_x \Delta x, \text{ for } p_x \in 0, \cdots, N_x - 1, \\ y_{p_y} &= p_y \Delta y, \text{ for } p_y \in -N_y/2, \cdots, N_y/2, \\ z_{p_z} &= p_z \Delta z, \text{ for } p_z \in 0, \cdots, N_z - 1, \end{aligned} \tag{1.47}$$

with $N_x = x_{\max}/\Delta x$, $N_y = y_{\max}/\Delta y$ and $N_z = z_{\max}/\Delta z$ the number of points for each direction. The discretized version of $u(x, y, z)$ is denoted as $u_x[p_y, p_z]$. The discrete domain is fully described in Figure 1.8.

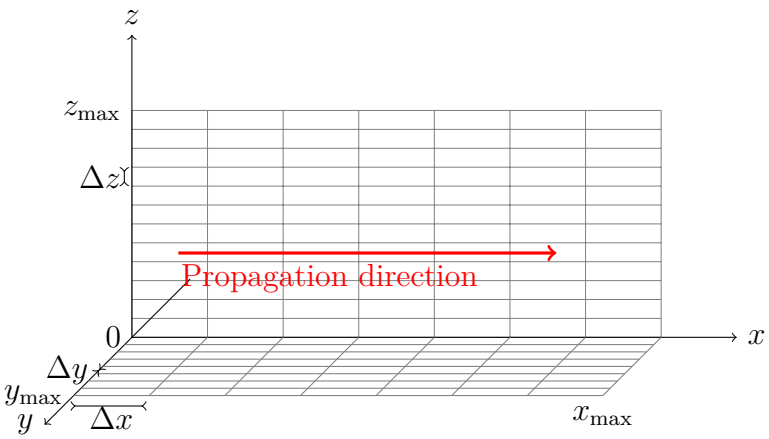


Figure 1.8: 3D discretized domain.

As in 2D, an apodization window is used to enforce the field to decrease slowly to the end of the domain. A Hanning window, of size $N_{y_{\text{apo}}}$ is used at each side in the y direction and at the top boundary condition on z, with a size $N_{z_{\text{apo}}}$, as shown in Figure 1.9. This operation is denoted by **H**. This condition allows to use any convenient boundary condition at the top and lateral boundaries of the domain. To introduce the method periodic boundary condition are introduced at each sides of the domain.

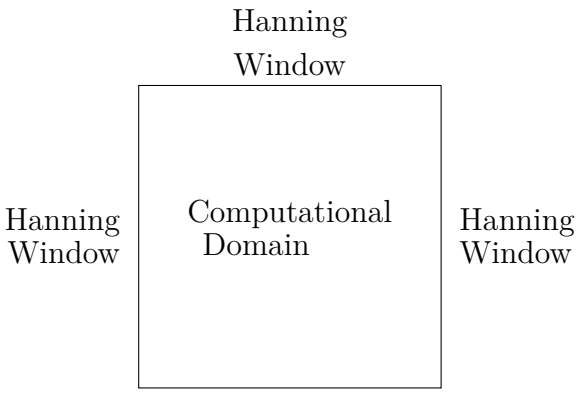


Figure 1.9: Configuration in the yOz-plane.

Now that the configuration has been described the DSSF method in 3D is introduced. As in 2D, first, the continuous problem is described. Then the free-space propagator is obtained by solving the propagation equation with $n = 1$. Finally, the refraction is taken into account with a phase screen.

1.3.2 3D DSSF method in Cartesian coordinates

1.3.2.a Continuous formulation

The field can be decomposed in a transverse electric (TE) and transverse magnetic (TM) components with respect to z with the Hertz magnetic and electric potentials [88]. Since n is assumed to be constant, the electric field is then calculated as

$$\mathbf{E} = k_0^2 n^2 \mathbf{\Pi}_e + \nabla\nabla\cdot\mathbf{\Pi}_e - k_0\zeta_0 n \nabla\times\mathbf{\Pi}_m, \tag{1.48}$$

with k_0 the wavenumber, ζ_0 the vacuum impedance, and $\mathbf{\Pi}_e$ and $\mathbf{\Pi}_m$ the electric and magnetic vectorial potentials, respectively.

Studying the TE polarisation (the TM one can be formulated likewise), the electric vectorial potential can be expressed with the potential ψ as

$$\mathbf{\Pi}_e = \psi\hat{z}. \tag{1.49}$$

In 3D Cartesian coordinates, the potential is solution of the scalar wave equation [11]

$$\frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial y^2} + \frac{\partial^2\psi}{\partial z^2} + k_0^2 n^2 \psi = 0. \tag{1.50}$$

The previous equation could be solved directly by going in the spectral domain as in cylindrical coordinates in [54]. Nevertheless, since a wide angle approximation is used to treat the refraction with phase screens, we introduce u, as in 2D, with reduced phase variation

$$u(x,y,z) = \exp\left(jk_0 x\right)\psi(x,y,z). \tag{1.51}$$

Introducing u in (1.50), the following equation is obtained

$$\frac{\partial^2 u}{\partial x^2} - 2jk_0\frac{\partial u}{\partial x} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} + k_0^2(n^2-1)u = 0. \tag{1.52}$$

For different ground compositions, u fulfills the followings conditions:

- If the ground is a PEC, then we have

$$u|_{z=0} = 0. \tag{1.53}$$

- If the ground is a PMC, then u follows a Neumann condition

$$\frac{\partial u}{\partial z}|_{z=0} = 0. \tag{1.54}$$

- Finally, for a dielectric ground condition at $z = 0$, the condition is approximated to a Leontovitch condition

$$\frac{\partial u}{\partial z}|_{z=0} + \alpha u|_{z=0} = 0. \tag{1.55}$$

  The surface roughness is easily taken into account by modifying this latter coefficient as given in [80–83, 89].

The continuous problem has now been fully introduced. In the next section, the formulation is discretized to obtain an *a-priori* discretization for a self-consistent algorithm.

1.3.2.b Discrete formulation

In order to obtain a self-consistent Cartesian SSF (DSSF) formulation, the wave equation (1.52) is discretized along y and z. This yields

$$\frac{\partial^2 u_x[p_y,p_z]}{\partial x^2} - 2jk_0\frac{\partial u_x[p_y,p_z]}{\partial x} + \mathbf{d}_y^2 u_x[p_y,p_z] + \mathbf{d}_z^2 u_x[p_y,p_z] + k_0^2(n^2-1)u_x[p_y,p_z] = 0, \quad (1.56)$$

with

$$\mathbf{d}_y^2 u_x[p_y,p_z] = \frac{1}{\Delta y^2}\left(u_x[p_y+1,p_z] - 2u_x[p_y,p_z] + u_x[p_y-1,p_z]\right),$$
$$\mathbf{d}_z^2 u_x[p_y,p_z] = \frac{1}{\Delta z^2}\left(u_x[p_y,p_z+1] - 2u_x[p_y,p_z] + u_x[p_y,p_z-1]\right).$$

The propagation equation is now discretized.

To fully obtain a discrete formulation of the problem, we now focus on the ground composition. The different bottom conditions become:

- For a PEC the condition in the discrete domain is now

$$u_x[p_y,0] = 0 \qquad \forall p_y. \quad (1.57)$$

- For a PMC bottom condition, the discretized potential fullfils

$$\mathbf{d}_z u_x[p_y,0] = 0 \qquad \forall p_y. \quad (1.58)$$

- For a dielectric ground composition, the discrete version of the Leontovitch condition is

$$\mathbf{d}_z u_x[p_y,0] + \alpha u_x[p_y,0] = 0 \qquad \forall p_y. \quad (1.59)$$

The discrete problem has been fully described. As in 2D, we first assume periodic conditions at each side to introduce the propagation method. The extension to a PEC, a PMC and an impedance condition is presented afterwards.

As in 2D a split-step scheme is used, and the forward propagation equation follows

$$\frac{\partial u_x[p_y,p_z]}{\partial x} = jk_0\left(1-\sqrt{1+\mathcal{Q}}\right)u_x[p_y,p_z], \quad (1.60)$$

with

$$\mathcal{Q} = \frac{1}{k_0^2}\left(\mathbf{d}_y^2 + \mathbf{d}_z^2\right) + n_x^2[p_y,p_z] - 1. \quad (1.61)$$

The wide angle approximation is also used with

$$\sqrt{1+\mathcal{Q}} \simeq \sqrt{1+\frac{1}{k_0^2}(\mathbf{d}_y^2+\mathbf{d}_z^2)} + (n_x[p_y,p_z]-1). \quad (1.62)$$

Thus, equation (1.52) becomes

$$\frac{\partial u_x[p_y,p_z]}{\partial x} = -j(\sqrt{k_0^2+\mathbf{d}_y^2+\mathbf{d}_z^2}-k_0)u_x[p_y,p_z] - jk_0(n_x[p_y,p_z]-1)u_x[p_y,p_z] \quad (1.63)$$

As in 2D, we aim at diagonalizing $\mathbf{d}_y^2$ and $\mathbf{d}_z^2$, *i.e.* obtaining k_y, k_z and u such that

$$\begin{aligned}\mathbf{d}_y^2 u &= -k_y^2 u,\\ \mathbf{d}_z^2 u &= -k_z^2 u.\end{aligned} \tag{1.64}$$

The spectral transform is applied directly on the discrete wave equation (1.63). Since periodic boundary conditions at each sides are assumed, a discrete Fourier transform is applied along y and z to diagonalize $\mathbf{d}_y^2$ and $\mathbf{d}_z^2$ with the Fourier atoms as eigen vectors. We denote this operation by $\mathbf{T}_s$. $\tilde{u_x}$ the spectral transform of u is thus given by

$$\tilde{u}_x[q_y, q_z] = \frac{1}{\sqrt{N_y}}\frac{1}{\sqrt{N_z}}\sum_{p_y=0}^{N_y-1}\sum_{p_z=0}^{N_z-1} u_x[p_y, p_z]\exp\left(-2j\frac{\pi p_y q_y}{N_y}\right)\exp\left(-2j\frac{\pi p_z q_z}{N_z}\right), \tag{1.65}$$

with $q_y \in [0, N_y[$ and $q_z \in [0, N_z[$. Applying the spectral transform (1.65) to (1.63) yields

$$\frac{\partial \tilde{u}_x\left[q_y, q_z\right]}{\partial x} + \left(\sqrt[\circ]{k_0^2 - k_y^2 - k_z^2} - k_0\right)\tilde{u}_x[q_y, q_z] = 0, \tag{1.66}$$

where

$$k_y = \frac{2}{\Delta y}\sin\left(\frac{\pi q_y}{N_y}\right), \tag{1.67}$$

$$k_z = \frac{2}{\Delta z}\sin\left(\frac{\pi q_z}{N_z}\right). \tag{1.68}$$

By going in the spectral domain $\mathbf{d}_y^2$ and $\mathbf{d}_z^2$ are diagonalised. k_y is obtained by calculating

$$\begin{aligned}\Delta y^2 \mathbf{d}_y^2\left(\exp\left(-2j\frac{\pi p_y q_y}{N_y}\right)\right) &= \exp\left(-2j\frac{\pi(p_y+1)q_y}{N_y}\right) - 2\exp\left(-2j\frac{\pi p_y q_y}{N_y}\right)\\ &\quad + \exp\left(-2j\frac{\pi(p_y-1)q_y}{N_y}\right)\\ &= \left(\exp\left(-2j\frac{\pi q_y}{N_y}\right) - 2 + \exp\left(2j\frac{\pi q_y}{N_y}\right)\right)\exp\left(-2j\frac{\pi p_y q_y}{N_y}\right)\\ &= -4\sin^2\left(\frac{\pi q_y}{N_y}\right)\exp\left(-2j\frac{\pi p_y q_y}{N_y}\right).\end{aligned} \tag{1.69}$$

Dividing by Δy^2 both terms, k_y is obtained. The same calculations can be done to obtain k_z.

Equation (1.66) is then solved iteratively along the propagation axis x in the spectral domain

$$\tilde{u}_{x+\Delta x} = \mathbf{P}\tilde{u}_x, \tag{1.70}$$

with $\mathbf{P}[q_y, q_z] = \exp\left(j\left(\sqrt[\circ]{k_0^2 - k_y^2 - k_z^2} - k_0\right)\Delta x\right)$ the spectral propagator, which is of size (N_y, N_z). The operator $\sqrt[\circ]{\cdot}$ corresponds to the square root with a negative imaginary part. The propagated potential in free space is denoted by $u_{x+\Delta x}$ and obtained with an inverse spectral transform $\mathbf{T}_s^{-1}$.

For a PEC ground composition a sine transform in place of the Fourier transform along z must be used. Moreover, k_z is changed by

$$k_z = \frac{2}{\Delta z}\sin\left(\frac{\pi q_z}{2N_z}\right). \tag{1.71}$$

For a PMC ground condition a DCT is used. Lastly, for a dielectric ground composition, the spectral transform is applied to w, instead of u, defined by the DMFT change of variable [71] as follows

$$w_x[p_y, q_z] = \frac{u_x[p_y, p_z + 1] - u_x[p_y, p_z - 1]}{2\Delta z} + \alpha u_x[p_y, p_z]. \tag{1.72}$$

w follows a Dirichlet boundary condition at the bottom. The method described previously can be performed directly on w. In Appendix B, the complete scheme to propagate w and retrieve the reduced field u from w is given. The ground wave propagator for ground parameters invariant along y, introduced in the DMFT, is also defined.

1.3.3 Introducing a slowly varying refractive index

Secondly, the effects of the refraction are taken into account. We consider, as in 2D, a slowly varying index $n(x, y, z)$. To take into account the curvature of the Earth m can be considered in place of n [11] as in the previous section. Also since m is close to 1, one can use M instead. Therefore, as with 3D-SSF [11], a phase screen is applied denoted by the operator $\mathbf{R}$. The elements of the diagonal operator $\mathbf{R}$ are defined by

$$\mathbf{R}[p_y, p_z] = \exp\left(-jk_0\left(n_x[p_y, p_z] - 1\right)\Delta x\right). \tag{1.73}$$

Hence, after coming in the space domain the propagated potential is obtained with

$$u_{x+\Delta x} = \mathbf{R}u^{fs}_{x+\Delta x}. \tag{1.74}$$

To sum up, the potential is computed marching on in distances as follows

$$u_{x+\Delta x} = \mathbf{R}\mathbf{T}_s^{-1}\mathbf{P}\mathbf{T}_s u_x. \tag{1.75}$$

Now, the numerical scheme and its complexity can be described.

1.3.4 Numerical scheme

As with other split-step methods, the field is computed iteratively at increasing distance from the source. An apodization windows at each sides of y and at the top of the domain are applied. Lastly, between two screens, the potential is propagated by going back and forth in the spectral and space domains.

Thus, the field is obtained iterating at increasing distances from the source with one step propagation defined as

$$u_{x+\Delta x} = \mathbf{H}\mathbf{R}\mathbf{T}_s^{-1}\mathbf{P}\mathbf{T}_s u_x. \tag{1.76}$$

Next, the complexity of the algorithm is studied. The spectral transform is performed using a 2D fast Fourier transform (FFT) of complexity $O\left(N'_y N'_z\left(\log(N'_y) + \log(N'_z)\right)\right)$, with $N'_y = N_y + 2N_{\text{apo}}$. The propagation corresponds to a complexity of $O(N'_y N'_z)$. Next, the inverse spectral transform of the same complexity as the FFT is performed. Finally,

the phase screen is also of complexity $O(N_y N_z)$. All these steps are done N_x times. Thus, the final complexity is multiplied by N_x and driven by

$$O\left(N_x N'_y N'_z \left(\log(N'_y) + \log(N'_z)\right)\right). \tag{1.77}$$

For the memory size aspect, a propagator of size (N'_y, N'_z) must be saved and its size increases with the size of the domain. Also, a phase screen of the same size needs to be stored.

1.3.5 Numerical tests

In this section, numerical tests to validate the Cartesian version of DSSF are performed. The aim is to show that the 3D extension of DSSF in Cartesian coordinates works with good accuracy. The propagation of the field radiated by a CSP [31] is studied.

In 3D the CSP is located at a complex position $(x_s, y_s, z_s) \in \mathbb{C} \times \mathbb{R} \times \mathbb{R}$ with $x_s = x_{w0} - jk_0 W_0^2/2$, where x_{w0} corresponds to the position of the source and W_0 to its width. An analytical expression of its radiation in 3D is defined as

$$u_{3D}(r) = -A\frac{\exp(-jk_0 r)}{4\pi r}, \tag{1.78}$$

with

$$r = \sqrt{(x-x_s)^2 + (y-y_s)^2 + (z-z_s)^2}, \tag{1.79}$$

where $\mathrm{Re}\left(\sqrt{(x-x_s)^2 + (y-y_s)^2 + (z-z_s)^2}\right) > 0$.

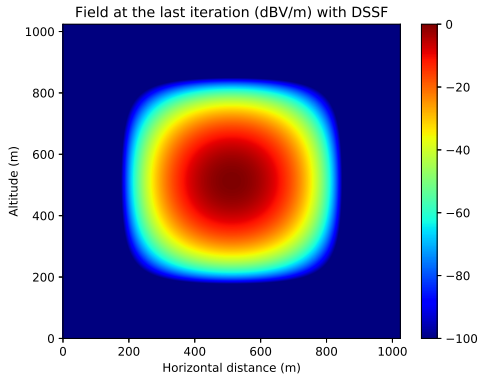
1.3.5.a Free space propagation

This test aims at validating the DSSF method in a free-space propagation configuration.

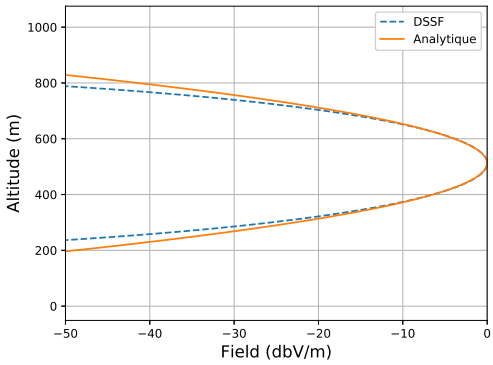
The parameters for the complex source are $f_0 = 300$ MHz, $x_{w0} = -50$ m, $W_0 = 5$ m and $(y_s, z_s) = (512, 512)$ m. The propagation is computed on $x_{\max} = 2$ km. The domain is of size 1024×1024 m in y and z directions. The size is chosen to avoid that the field reaches the ground. The different steps are chosen as $\Delta x = 100$ m, $\Delta y = \Delta z = \lambda = 1$ m. The atmosphere is assumed to be constant with $n = 1$. We compare the field obtained at the last iteration with the analytical formulation for the CSP (1.78). Figure 1.10 (a) shows the normalised final field obtained in the yOz plan with DSSF. In Figure 1.10 (b) a cut in the $x0z$ plan of the normalised final fields obtained with DSSF and equation (1.78) are plotted. Finally the normalised difference between both methods at the last iteration in the yOz-plane is shown in Figure 1.10 (c).

Figure 1.10 shows that the field obtained with DSSF is in adequation with the results previously obtained in 2D. Besides, the maximum of the normalised error is -15 dB. As expected, this error is on the side of the field as can be observed in Figures 1.10 (b) and (c). We can also note that the field has no cylindric symmetry at the edge. This error is due to the finite difference approximation in the propagation equation.

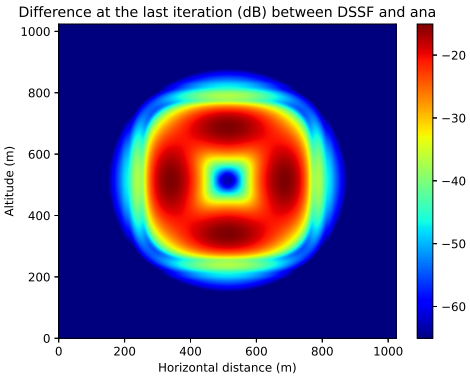
We now reduce the steps in y and z direction to 0.5λ to reduce the error. Figure 1.11 presents the results obtained with the reduced steps. The same figures in (a), (b), and



(a) Normalised field by DSSF in plan yOz (dB).



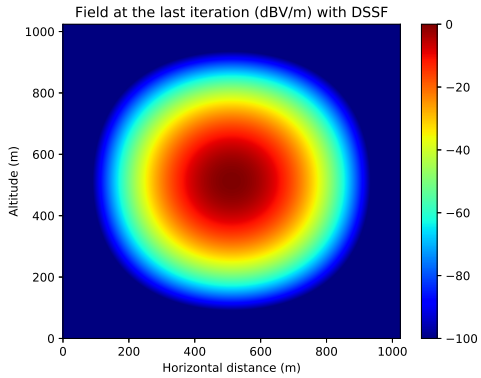
(b) Analytical method and DSSF in plan xOz (dB).



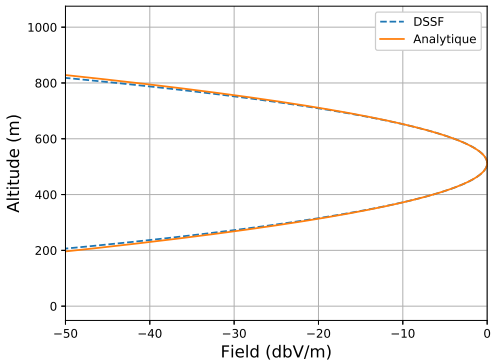
(c) Normalised difference at the last iteration (dB).

Figure 1.10: Normalised fields obtained at the last iteration with DSSF and compared to the CSP analytical expression in free space.

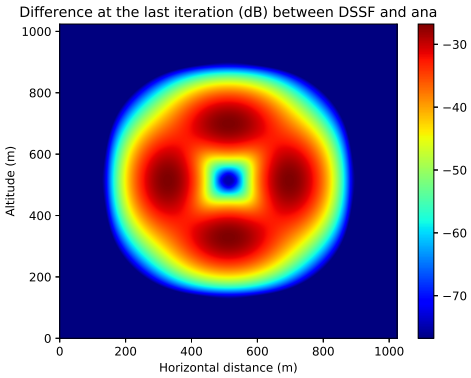
(c) as in the previous test are plotted. The normalized error has decreased to −27 dB. With the grid size reduction, we can see that the side errors are reduced as expected, and consistent with the results obtained in 2D since the error due to the finite difference is reduced. Note that the now the field has a cylindric symmetry, as expected.



(a) Normalised field by DSSF in plan yOz (dB).



(b) Analytical method and DSSF in plan xOz (dB).



(c) Normalised difference at the last iteration (dB).

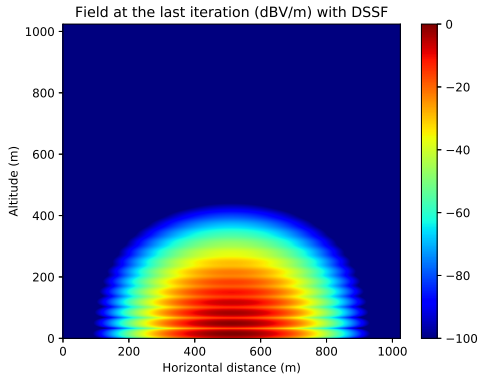
Figure 1.11: Normalised fields obtained at the last iteration with DSSF and compared to the CSP analytical expression in free space.

For the computation time aspect, the propagation has been obtained in 4 min with 140 s to pre-compute the free-space propagator. The major problem here is the memory requirement. For the propagator, the memory size was 150 MB for $\Delta y = \Delta z = 1$ m and is now 600 MB for $\Delta y = \Delta z = 0.5$ m, as expected since the size of the domain has been quadrupled. This expensive cost would further increase at higher frequencies or for a larger size of the computation domain. For example, for a 10 GHz scenario, to obtain $\Delta y = \Delta z = \lambda/2$, a grid size of 0.015 m in each directions is needed, inducing a propagator of order 650 GB, which is not achievable on a standard computer.

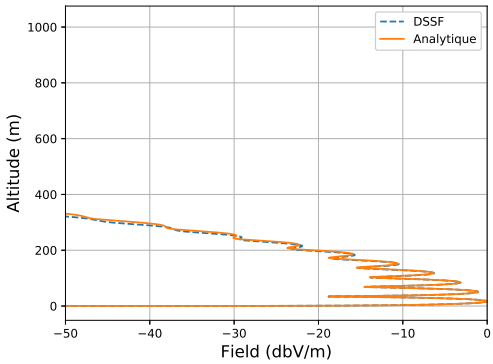
1.3.5.b Propagation over a PEC ground

In this test, we simulate the propagation over a planar PEC ground. We compare the results to the one obtained with the analytic formula of the CSP, combined with an image CSP to account for the ground reflection.

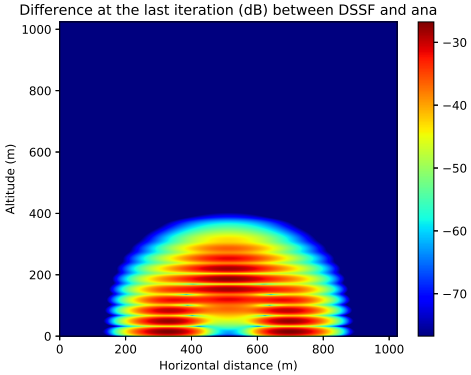
The same parameters as in the previous test with the smaller grid size is used, except for the source altitude which is decreased to $z_s = 30$ m. Figure 1.12 shows the results at the last iteration. The same figures in (a), (b), and (c) as in the previous test are plotted. First, (b) shows that we accurately obtain the ground interferences in terms of extrema levels and positions. Besides, the normalized error at the last iteration is of order -27 dB. The differences are due to the finite difference scheme applied to the propagation equation.



(a) Normalized field by DSSF in plan yOz (dB).



(b) Vertical cuts at $N_y/2$ of the normalized final fields.



(c) Normalised difference at the last iteration (dB).

Figure 1.12: Normalised fields obtained at the last iteration with DSSF and compared to the CSP analytical expression with a PEC ground condition.

The propagation is computed with DSSF in 6 min in total with an acceptable error in this case. Now, the propagation over an impedance ground with the DMFT is tested.

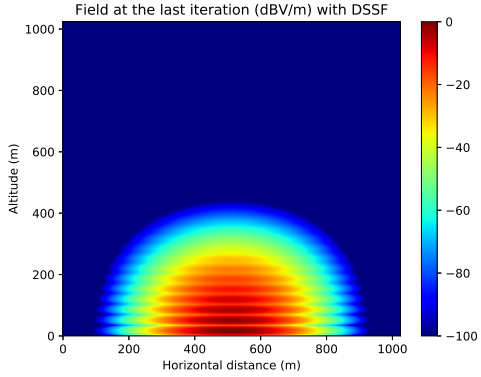
1.3.5.c Propagation over a dry ground

We aim at showing that the propagation over a dry ground is accurately modeled.

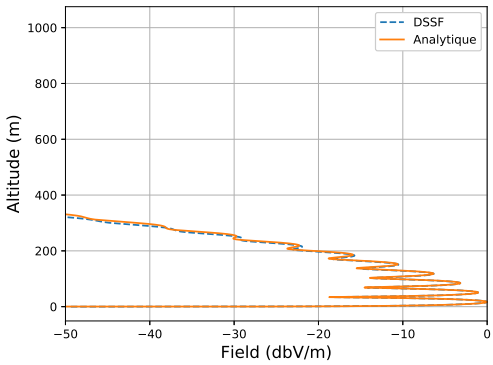
The propagation is computed with DSSF-DMFT. This result is compared to the analytic formula of the CSP propagation, for which the reflection is accounted via the Fresnel coefficient. This approach is valid in the far-field zone of the CSP. We use the same parameters for the domain, for the source and for the discretization as in the test with

the PEC. We consider an impedance condition for the ground with parameters $\epsilon_r = 20$ and $\sigma = 0.02$ S/m. The horizontal polarization is considered here. The results obtained at the last iteration are plotted in Figure 1.13.

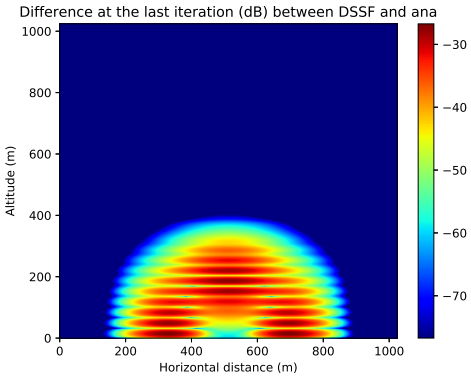
The same figures in (a), (b), and (c) as in the previous test are plotted. The maximum and minimum due to the ground interferences are the same with both methods. The normalized error at the last iteration is of -27 dB. Since, we are in the far field zone, as in the previous tests, differences are most likely due to the finite difference scheme applied to the propagation equation. Thus, the DSSF-DMFT method allows to compute the propagation over an impedance ground with good accuracy.



(a) Normalised field by DSSF in plan yOz (dB).



(b) Vertical cuts at $N_y/2$ of the normalized final fields.



(c) Normalised difference at the last iteration (dB).

Figure 1.13: Normalised fields obtained at the last iteration with DSSF and compared to the CSP analytical expression with a dielectric ground condition.

1.4 Conclusion

In this chapter, the DSSF has been recalled in 2D and extended in 3D.

Firstly, a state-of-the-art on the PWE and the SSF method has been presented.

Secondly, the 2D version of DSSF has been recalled. The continuous problem has been defined. An *a-priori* discretization has been applied to obtain a self-consistent method.

The discretized problem has thus been defined. Using a split-step method with a paraxial approximation, refraction, and propagation terms are treated separately. With this formalism, the problem is solved iteratively by going back and forth in the spectral and spatial domains. The discrete spectral propagator has been calculated and the complete numerical scheme based on the same steps as SSF has been described. Numerical tests have been performed to validate the approach. First, a propagation without ground has been computed. Results are accurate, but we have pointed out that the finite difference introduces numerical errors when the vertical step is too large. Finally, a long-range propagation simulation with refraction, relief, and an impedance ground has been successfully performed.

Thirdly, we have derived DSSF in 3D Cartesian coordinates. As in 2D, the continuous problem has been exposed. The latter has been discretized *a priori*. With a paraxial assumption, the propagation is computed with a split-step method. In free space, taking into account only the forward propagation, a first-order derivative equation has been obtained. This latter is efficiently solved by going in the spectral domain, obtaining a diagonal operator for the propagation. The refraction is then taken into account in the space domain with phase screens. Then, the algorithm and its complexity have been introduced. Numerical tests have been performed to show that DSSF performs well in terms of accuracy but is limited in terms of memory and computation time requirements.

Since the computation time and the memory requirement needed for DSSF become a significant burden for a very large domain such as radio-occultation scenario in 2D or any 3D case, this motivates the development of a new method with better performances on both aspects.

Chapter 2

Matrix Split-Step Wavelet in 2D

2.1 Introduction

A wavelet is a zero-mean oscillating function that is localized both in space and frequency. Wavelet atoms are obtained by dilating a mother wavelet. A wavelet basis is constructed by translating these atoms. The wavelet transform consists in decomposing the signal over this basis. The localization in both space and frequency allows wavelets to take into account both local and global properties of the signal. This locality of the atoms is the major difference with the Fourier transform.

The wavelet transform was defined in the '80s by Jean Morlet and Alex Grossman [90]. Since its formalization by searchers as Yves Meyers [91], Ingrid Daubechies [92] and Stéphane Mallat [93, 94], it has become quite popular in a lot of domains. The fast wavelet transform (FWT) and the sparse representation of regular signals have participated in this takeoff.

In particular, the FWT is of complexity $O(N)$, with N the number of elements, lower than the complexity of the fast Fourier transform (FFT), which is $O(N \log(N))$. The second advantage is that with wavelet decomposition a lot of coefficients are close to 0. Thus, compression allows for obtaining an efficient sparse representation.

The wavelet transform is used in various domains. In signal theory and image processing, they are used for denoising [95, 96] and data compression [97, 98]. For example, the compression is used for the image format JPEG2000 [99]. In electromagnetics, the wavelets are mostly used as basis-tests functions in finite elements [100] and integral equations [101–103], and in FDTD [104] because of their compression capacity. Recently, the low complexity of the FWT has been used in wave propagation methods [105]. They were introduced for the electromagnetic wave propagation in [55, 56] and in split-step wavelet (SSW) [57].

This latter is an iterative method to solve the PWE, shown to be faster than SSF [57]. As for SSF, each iteration follows two steps. The field is first decomposed in the wavelet domain and compressed. The wavelet coefficients are then propagated in free space using a pre-computed propagator. This propagator stores the compressed wavelet-to-wavelet propagations. Relief and atmosphere are accounted for in a second step.

Both compressions on the signal and the propagator introduce errors that accumulate

throughout the propagation. A heuristic formula has been introduced in [57] to predict the accuracy of SSW with the number of iterations. This formula was empirically deduced from numerical tests.

This chapter introduces the matrix version of SSW [52, 57]. Besides, we derive here a theoretical formula to assess the error at the end of the domain depending on the compression threshold values. This can be used to pre-compute the thresholds to achieve an expected error.

In Section 2.2, the 1D multilevel discrete wavelet transform is introduced. First, the construction of a wavelet basis is recalled. Then, the discrete wavelet transform is presented. The fast wavelet transform to rapidly compute the wavelet coefficients is introduced. We then recall some properties and characteristics of the wavelets. In Section 2.3, SSW [52,57] is introduced. First, an overview of the method is proposed. Then, the two main steps of SSW are presented. To conclude this section, numerical tests are performed to show the advantages and limitations of the method. In Section 2.4, we obtain a closed-form formula of the compression error accumulating with the number of iterations N_x. We use this expression to tune the thresholds. In Section 2.5, we conclude and recall limitations of matrix SSW.

2.2 1D discrete wavelet transform

In this section, the 1D multilevel discrete wavelet transform is presented. First we introduce how to construct a wavelet basis. Then, the discrete wavelet transform and the associated fast transform are presented. The sparse representation with wavelets is then introduced. Finally, characteristics of the wavelet and the choice of the parameters are presented.

We denote by $\mathbb{N}$ the set of natural numbers, $\mathbb{Z}$ the set of integers and $\mathbb{R}$ the real numbers.

2.2.1 Wavelet family

In this section, the construction of a wavelet family and a wavelet basis are explained.

A wavelet is a function $\psi \in \mathcal{L}^2(\mathbb{R})$ such that ψ is of zero mean

$$\int_{-\infty}^{\infty} \psi(z)\, dz = 0, \tag{2.1}$$

with $\|\psi\|_2 = 1$.

From this first function, called the mother wavelet, the wavelet family can be defined. Denoting this family by $\mathcal{F}$, we have

$$\mathcal{F} = \left\{ \psi_{l,p}(z) = 2^{-l/2} \psi\left(2^{-l} z - p\right) \right\}_{(l,p)\in\mathbb{N}\times\mathbb{Z}}, \tag{2.2}$$

with $2^{-l/2}$ a normalization factor to keep $\|\psi_{l,p}\|_2 = 1$. In this definition l corresponds to the dilation level. This latter allows to dilate the wavelet to cover lower parts of the spectrum, *i.e.*, capture the slower variations of the signal. Furthermore, p corresponds

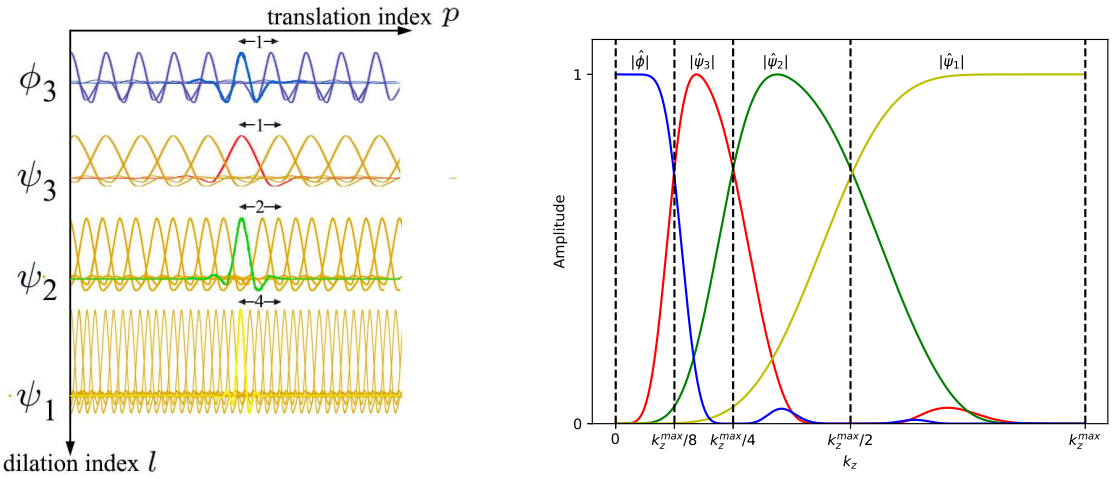
to the translations needed to cover the spatial domain. If $l \in \mathbb{N}$ and $p \in \mathbb{Z}$, then $\mathcal{F}$ is an orthonormal basis of $\mathcal{L}^2(\mathbb{R})$ [94]. In practice, the decomposition is only computed for $l \in [1, L]$, with L the maximul level of decomposition, and the family is no more a basis because the lowest part of the spectrum is not covered.

An additional function, the scaling function $\phi_{L,p} \in \mathcal{L}^2(\mathbb{R})$, of non-zero mean, is thus added to the family

$$\mathcal{B} = \{\phi_{L,p}(z), \psi_{l,p}(z)\}_{(l,p)\in[1,L]\times\mathbb{Z}}, \tag{2.3}$$

in order to obtain an orthonormal basis of $\mathcal{L}^2(\mathbb{R})$.

In Figure 2.1 (a) a wavelet basis in the spatial domain with a maximum level $L = 3$ is plotted. Figure 2.1 (b) shows the spectral coverage associated to the wavelets plotted in (a).



(a) Spatial domain [52].

(b) Spectral domain.

Figure 2.1: Example of a wavelet basis in the spatial domain and of its spectral coverage.

With figure 2.1 (a), the first remark is that the wavelets are well localized in the spatial domain. With l increasing the support of the wavelets doubles. Slower variations of the signal are taken into account. The translated versions of the wavelets and scaling function allow to cover the entire spatial domain. Figure 2.1 (b) shows that wavelets are also localized in the spectral domain. Indeed, the scaling function covers the lowest part until the continuous of the spectral domain. For the wavelets increasing l allows covering lower parts of the spectrum. In the plane wave decomposition, it corresponds to the more paraxial waves. The spectral domain is thus covered. To conclude, we obtain an orthonormal basis of $L^2(\mathbb{R})$ with functions localized in both space and frequency.

Now, that the wavelet basis has been introduced, the discrete transform associated with these atoms is introduced.

2.2.2 Discrete transform

We want to decompose a discrete function u of the variable p_z with $p_z \in \mathbb{Z}$. This function is denoted by $u[p_z]$. The wavelets and the scaling function are also discretized and function of p_z.

First, the previously constructed basis in the continuous domain is still a basis for the discrete domain.

The discrete wavelet transform is defined as an application that associates a discrete function u with a sequence of coefficients $a_{L,p}$ and $d_{l,p}$, such that

$$u[p_z] = \sum_{p\in\mathbb{Z}} a_{L,p}\phi_{L,p}\left[p_z\right] + \sum_{l\in[1,L]}\sum_{p\in\mathbb{Z}} d_{l,p}\psi_{l,p}\left[p_z\right], \tag{2.4}$$

with L the maximum decomposition level. This decomposition is denoted by $\mathbf{W}$.

The coefficients $a_{L,p}$ are called the approximation coefficients. They can be computed with the inner product of the signal u with the scaling function ϕ_L. They describe the signal with a lower resolution and take into account the slowest variations. Coefficients $d_{l,p}$ are called the detail coefficients and can be obtained with the inner product between u and the wavelets at different levels l. They give the faster variations of the signal with l decreasing and thus the details of the signal are obtained. Thus a multi-resolution decomposition of the signal is obtained.

In practice, the computation is performed in a finite domain of size N_z. Therefore, the translation coefficients p is limited to $[0, N_z/2^l[$ for $l \in [1, L]$, respectively. For the scaling function this is limited to $p \in [0, N_z/2^L[$. In practice, for domain of finite size boundary conditions need to be applied to the discrete wavelet transform. To keep an orthonormal basis, periodic boundary conditions are used [94].

For conciseness, the wavelet coefficients are from now on denoted by

$$U_l[p] = \begin{cases} a_{L,p} & , \text{ for } l = 0, p \in [0, N_z/2^L - 1] \\ d_{l,p} & , \text{ for } l \in [1, L]\,, p \in [0, N_z/2^l - 1]. \end{cases} \tag{2.5}$$

The decomposition (2.4) becomes

$$u\left[p_z\right] = \sum_{p\in[0,N_z/2^L[} U_0[p]\chi_0[p_z - p] + \sum_{l\in[1,L]}\sum_{p\in[0,N_z/2^l[} U_l[p]\chi_l\left[p_z - p\right], \tag{2.6}$$

with

$$\chi_l[p_z - p] = \begin{cases} \phi_{L,p}[p_z], & \text{for } l = 0 \\ \psi_{l,p}[p_z], & \text{for } l \in [1, L]\,. \end{cases} \tag{2.7}$$

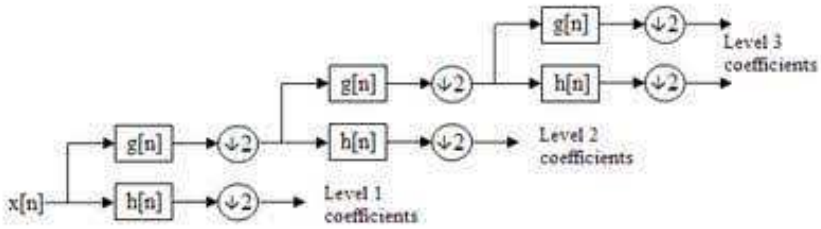
In practice, the inner products are not computed. These coefficients are rather obtained with the FWT introduced in the next section, for which the computational cost is largely cheaper.

2.2.3 The fast wavelet transform (FWT)

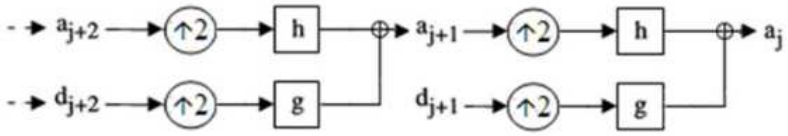
Mallat [94] has proven that any pair of wavelet ψ and scaling function ϕ can be specified using mirror filters, denoted by h and g. Their impulse responses are given by

$$\phi\left[p_z\right] = \sqrt{2}\sum_{p\in\mathbb{Z}} h_p\phi\left[2p_z - p\right], \tag{2.8}$$

$$\psi\left[p_z\right] = \sqrt{2}\sum_{p\in\mathbb{Z}} g_p\psi\left[2p_z - p\right], \tag{2.9}$$



(a) FWT



(b) inverse FWT

Figure 2.2: Decomposition and recomposition using mirror filters (FWT) [94]

with h_p and g_p the elements of h and g, respectively.

An implementation using these filters allows a fast wavelet decomposition of a given signal u because of the decimation at each step and because these filters have a support of small size. In this case the coefficients are obtained by iterating

$$a_{l+1,p} = \sum_{q \in \mathbb{N}} h_{p-2q} a_{l,q}, \tag{2.10}$$

$$d_{l+1,p} = \sum_{q \in \mathbb{N}} g_{p-2q} a_{l,q}. \tag{2.11}$$

The initialization a_0 corresponds to the very signal $u[p_z]$. Coefficients $a_{l,p}$ are $d_{l,p}$ are then succesively obtained by convolutions with mirror filters and downsamplings by 2 as shown in Figure 2.2 (a).

To recompose the signal, mirror filters are used after interpolation to obtain an upsampling by 2 of the signal, as shown in Figure 2.2 (b). Thus a_0, corresponding to $u[p_z]$, is obtained from the coefficients at the end of the inverse FWT.

This algorithm is effective because the coefficients are computed for each level l and not for each position p_z with convolutions. For a domain of finite size N_z, the complexity has been shown to be $O(N_z)$, which is lower than the complexity of the FFT $O(N_z \log(N_z))$. Besides, the sparse representation allowed by the wavelets is a huge advantage, as shown in the next section.

2.2.4 Sparse representation

One of the main advantages that explains the popularity of the wavelet transform is its sparse representation capacity [94]. In signal processing, for example, the format JPEG2000 is based on this sparse representation to store images with low memory usage.

By choosing well the wavelet family and the maximum level of decomposition lots of coefficients are closed to 0 due to the fast decreasing of the coefficients. Thus, the coefficients

below a certain threshold V_s are set to 0

$$U_l^c[p] = \begin{cases} 0 & \text{if } |U_l[p]| \leq V_s, \\ U_l[p] & \text{otherwise.} \end{cases} \tag{2.12}$$

Note that a higher threshold set more coefficient to 0. This compression, denoted by operator $\mathbf{C}$, is called "hard threshold" [94]. After the compression step, the vector $U_l^c[p]$ is generally sparse.

The compression rate is defined by

$$\text{CR} = \frac{\text{Number of zeros}}{\text{Total number of coefficients}}. \tag{2.13}$$

The sparse representation, defined by a high CR, is one of the main advantages of the wavelets. Nevertheless, obtaining a good compression ratio depends mainly on two parameters, that are defined in the next section.

2.2.5 Wavelet characteristics and paramaters

In this section, we briefly recall some important properties about wavelets and their effects on the coefficients obtained after decomposition. The two main properties that are recalled are the number of vanishing moments and the size of the support of the wavelet. Then, suitable parameters with respect to our signals are chosen.

2.2.5.a Vanishing moments and support size

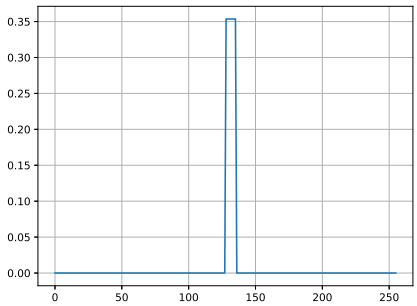
The first property is the number of vanishing moments of the wavelet [93, 94]. This number, denoted by n_v, is defined as follows

$$\forall k \in \mathbb{N}, 0 \leq k \leq n_v - 1, \int_{[0,1]} z^k \psi(\alpha) = 0. \tag{2.14}$$

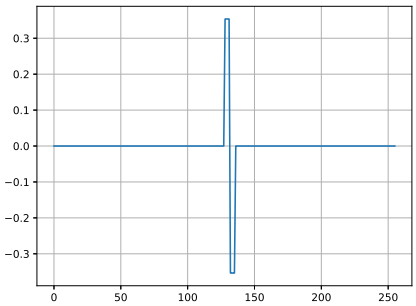
This property describes how well a wavelet can approximate a polynomial on a finite support and thus a smooth signal, with few coefficients. Indeed, with a smooth signal if n_v increases, then the wavelet coefficients of this signal decrease fastly to 0. Thus, few of them are needed to accurately describe the signal. Therefore, a high compression ratio is achieved. For each wavelet family, a mother wavelet can be specified for any value of n_v.

In Figure 2.3, an example of the mother wavelet and scaling function associated with the Daubechies family with $n_v \in \{1, 3, 6\}$ is given.

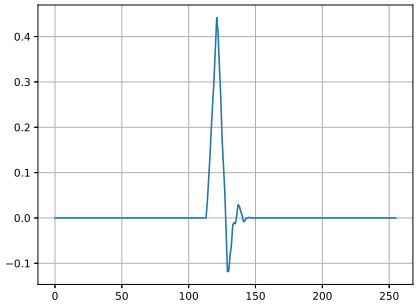
On the other hand, if a wavelet has n_v vanishing moments, then its support is at least of size $2n_v - 1$. The Daubechies wavelets are the ones with a support of size exactly $2n_v - 1$. Yet, to take into account steep variations of the function with few coefficients, this is necessary to have a support as small as possible. Thus, both criteria cannot be optimized at the same time.



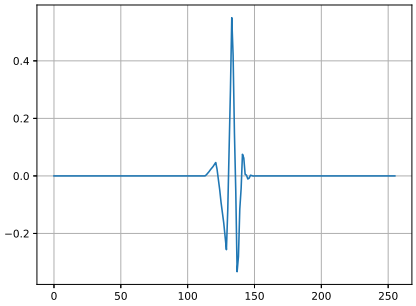
(a) Scaling function $n_v = 1$.



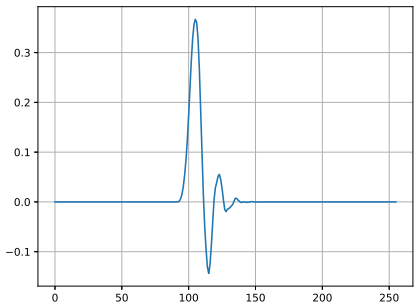
(b) Mother wavelet $n_v = 1$.



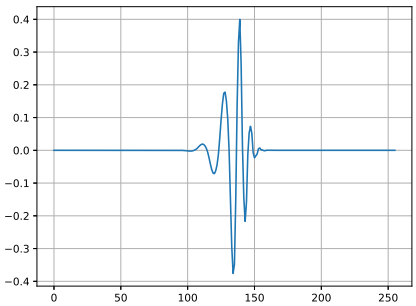
(c) Scaling function $n_v = 3$.



(d) Mother wavelet $n_v = 3$.



(e) Scaling function $n_v = 6$.



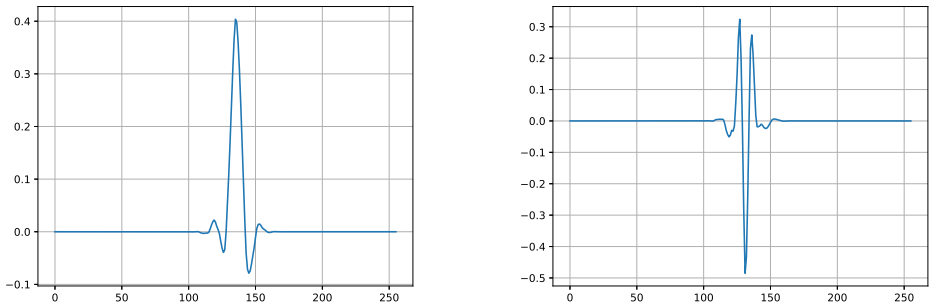
(f) Mother wavelet $n_v = 6$.

Figure 2.3: Scaling function and mother wavelet for the Daubechies family with $n_v \in \{1, 3, 6\}$.

2.2.5.b Choice of the wavelet parameters

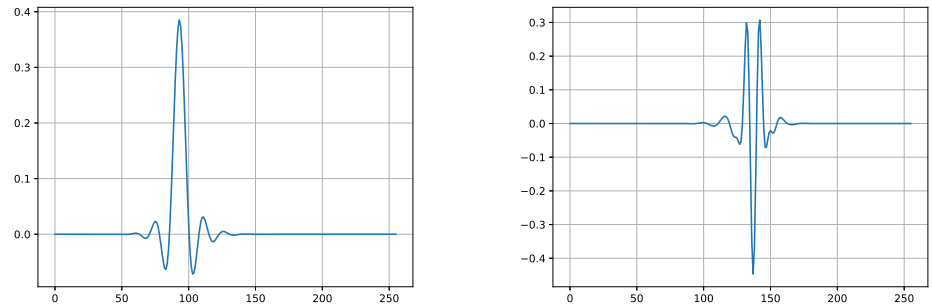
In our case, for electromagnetic field representation, other properties seem important. First, we need a discrete wavelet family because we are in a discrete domain. Secondly, since the free-space propagator is symmetric over z, symmetric or quasi-symmetric wavelets seem a better choice.

This limits the choice to three families. The first one is the Daubechies wavelet [92] with minimal support but they are asymmetric. A Daubechies wavelet basis is represented in Figure 2.3. Figure (a) shows the scaling function of this family and (b) the mother wavelet. To add symmetry, the symlet [94], which are quasi-symmetric, have been introduced and are represented in Figure 2.4. Finally, coiflets proposed in [106] and illustrated in Figure 2.5 are widely used for their symmetry property but the support is no more of minimal size. This last point is problematic if the signal has lots of singularities or details of size less than the wavelet support. In Table 2.1, these properties are summarized.



(a) Scaling function. (b) Mother wavelet.

Figure 2.4: Scaling function and mother wavelet for the symlet family.



(a) Scaling function. (b) Mother wavelet.

Figure 2.5: Scaling function and mother wavelet for the coiflet family.

Family	Daubechies	symlets	coiflets
Vanishing moments	n_v	n_v	$2n_v$
Support	$2n_v - 1$	$2n_v - 1$	$6n_v - 1$
Symmetry	-	+	++

Table 2.1: Summary of the properties of the different wavelet families.

Since the physical signals we are dealing with are mostly smooth functions, the number of vanishing moments must be high enough to describe the signal with few coefficients. In [101], $n_v = 8$ is said to allow a representation with single-precision accuracy and a good sparse representation. Nevertheless, a high n_v increases support and computation time. In [52, 57], numerical tests over the number of vanishing moments have been performed. In these tests, we can see that $n_v = 6$ gives a good compromise between sparsity and accuracy.

Finally, as in [52, 57], the symlet family is used here with $n_v = 6$. The maximum level has an effect of the computation time and tests over L in [52, 57] have shown that $L = 2$ or $L = 3$ give good results. Thus $L = 3$ is chosen to have a better sparse representation.

2.2.6 Conclusion

In this section, the wavelet representation of a field with a sparse vector has been introduced. Some important properties of the wavelets have also been recalled.

First, the constructions of a wavelet family and of a wavelet basis have been explained. Using this basis, the discrete wavelet transform to represent a signal over the atoms of this basis has been introduced. The fast wavelet transform has then been presented, it allows us to compute the coefficients of the wavelet transform in a short computation time. Finally, some important properties of the wavelet have been recalled and the choice of the wavelet parameters has been discussed.

In the next section, the wavelet transform and its compression capacity are used to compute the long-range wave propagation with split-step wavelet.

2.3 Split-step wavelet

In this section, SSW is presented. The configuration and discretization are the same as with DSSF in Chapter 1.

2.3.1 Introduction to the split-step wavelet method

SSW is an iterative method to compute the field at increasing distances from the source, by going back and forth from the wavelet domain to the spatial domain. The ground is supposed to be a PEC. The scheme in Figure 2.6 sums up the SSW algorithm. A step of SSW on Δx to obtain $u_{x+\Delta x}$ from u_x is computed as follows:

1. The sparse vector of the wavelet coefficients of u_x is obtained using a FWT (operator $\mathbf{W}$) and a compression of threshold V_s (operator $\mathbf{C}$)

$$U_x = \mathbf{CW}u_x. \tag{2.15}$$

   A high compression allows a faster propagation whereas a low compression is better for accuracy.

2. The propagated coefficients $U_{x+\Delta x}$ are obtained using the compressed propagator $\mathbf{P}$ (threshold V_p)
$$U_{x+\Delta x} = \mathbf{P}U_x. \tag{2.16}$$
The propagator accounts for all the wavelet-to-wavelet propagations, corresponding to a scattering matrix. Two methods of computation of this propagator will be studied in this thesis: the one developed by H. Zhou [57] , which is presented in Section 2.3.2, and a more efficient one [107], presented in Chapter 3.

3. The free space propagated field $u^{\text{fs}}_{x+\Delta x}$ is obtained by applying an inverse FWT
$$u^{\text{fs}}_{x+\Delta x} = \mathbf{W}^{-1}U_{x+\Delta x}. \tag{2.17}$$
$u^{\text{fs}}_{x+\Delta x}$ is expressed in the space domain.

4. The ground is taken into account using the local image method [57]. Refraction (phase screen) and relief are taken into account in the space domain as with DSSF, see chapter 1. These operations are represented by operators $\mathbf{L}$, $\mathbf{R}$, $\mathbf{H}$ for the relief, the refraction and the apodisation, respectively. This finally yields
$$u_{x+\Delta x} = \mathbf{HRL}u^{\text{fs}}_{x+\Delta x}. \tag{2.18}$$

Note that two more parameters need to be fixed: the thresholds V_s and V_p. This point is adressed in Section 2.4.

2.3.2 Computing the matrix propagator

In this section, we recall the computation of the scattering matrix of size (N_z, N_z) $\mathbf{P}$ [57]. The element $\mathbf{P}_{(l,p)(l',p^{l'}_t)}$ corresponds to the level l and position p of the wavelet transform apply to the propagated wavelet of level l' at position p_t. Properties of the wavelets are taken into account to efficiently fill the matrix. For conciseness, the level 0 is treated as the level L.

First the translation property of the wavelets is recalled. The position grids of the wavelets are dilated by 2 at each level. Considering two levels l and l', for the level after and before propagation, we have a translation property defined as follows:

- if $l = l'$, then a translation of 1 on l corresponds to the same translation on l'.
- if $l' < l$, then there are $2^{l-l'}$ times as many wavelets on l' than on l. Thus, a translation 1 on l corresponds to a translation of $2^{l-l'}$ on level l'.
- if $l' > l$, there are $2^{l'-l}$ times less wavelets on l' than on l. Thus, a translation of $2^{l'-l}$on l corresponds to a translation of 1 on l'.

Also, this property shows that with l decreasing a translation of less than 2^{L-l} induces a different wavelet decomposition because the grid at different levels do not match, see Figure 2.7. Therefore, all 2^{L-l} wavelets need to be propagated for a level l, leading to translations of $p^l_t \in [0, 2^{L-l}[$ for all $l \in [1, L]$.

The computation of the matrix is done in three steps [57]. Firstly, one centered wavelet $\chi_{l',0}[p_z]$ for all $p_z \in [0, N_z - 1]$ of each level $l' \in [1, L]$ is propagated on Δx using DSSF.

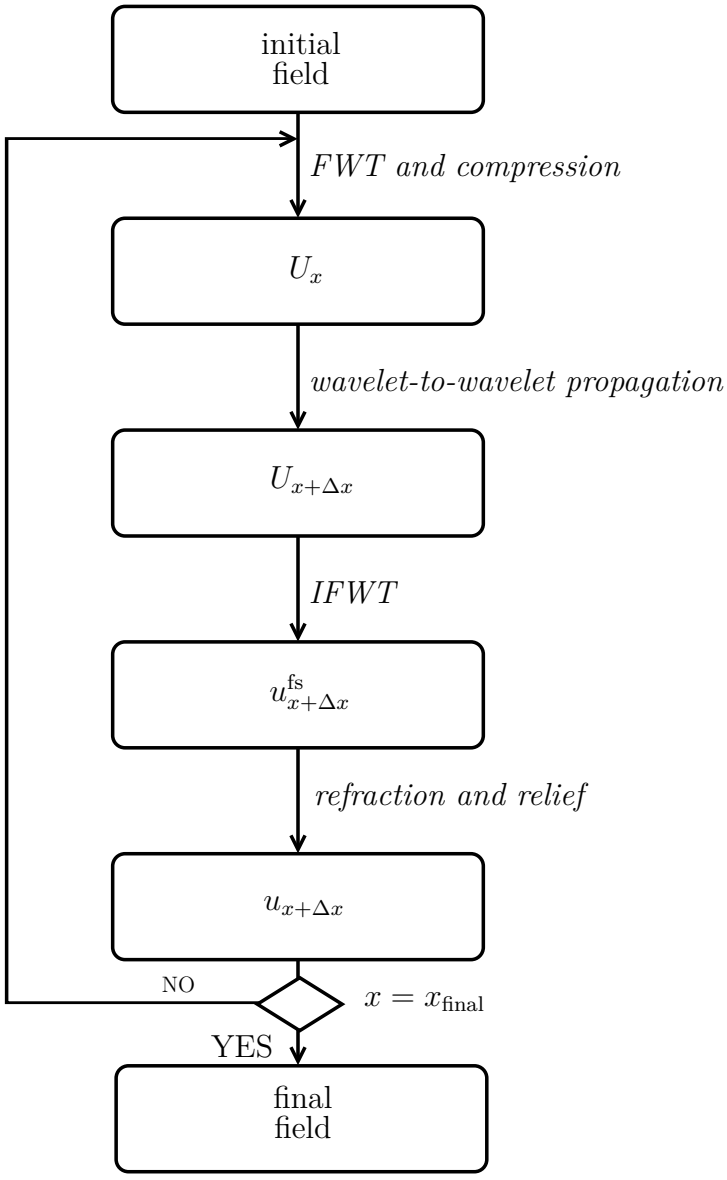


Figure 2.6: Overview of SSW.

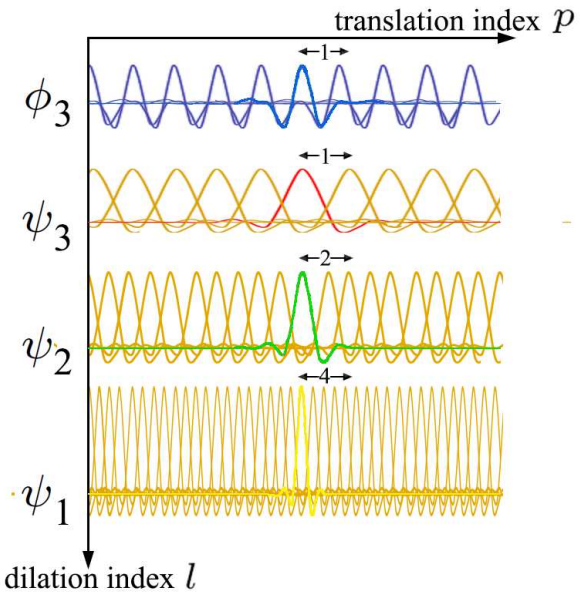


Figure 2.7: Space grid of the wavelets with $L = 3$ [57].

The field associated with the wavelet is obtained by computing the inverse FWT of a wavelet decomposition where only the center coefficient of level l' is put to 1. We obtain $\chi_{l',\Delta x}[p_z]$. Other methods can be used in place of DSSF [55, 56].

Secondly, the wavelets are translated. For one level l', we need to obtain all $2^{L-l'}$ the elementary propagations. To reduce the number of propagations, the translation property is used, as shown in Figure 2.8. Thus, each propagated wavelet of level l' is translated of $p_t \in [0, 2^{L-l}[$ along z to obtain $\chi_{l',\Delta x}[p_z + p_t]$ for all $p_z + p_t \in [0, N_z - 1]$.

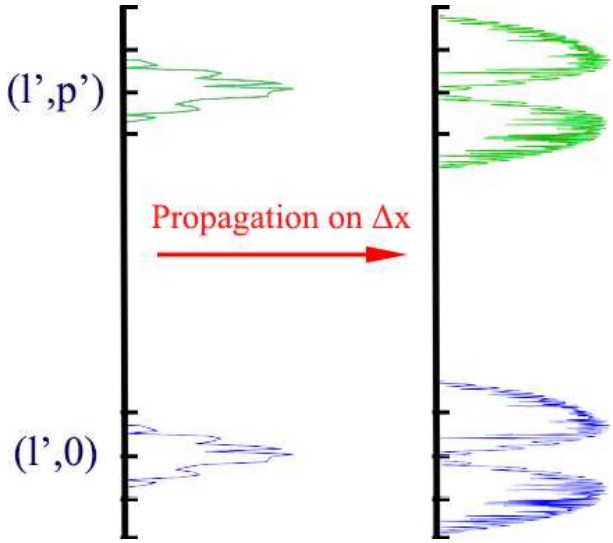


Figure 2.8: Illustration of wavelet propagations on Δx at level l' [57].

The propagated wavelets $\chi_{l',\Delta x}[p_z + p_t^{l'}]$ are then decomposed and compressed with threshold V_p to obtain $\mathbf{P}_{(l,p)(l',p_t^{l'})}$. Finally, the other elements are then filled with the replicas of the elementary propagations, and the matrix contains the propagations for all levels and positions.

The matrix is pre-computed just once for a given scenario. In total, the matrix is thus of

size (N_z, N_z) and its computation only requires $L+1$ DSSF and 2^L translations. Therefore, the main drawback of the propagator is its memory size, which rapidly increases with N_z.

The propagation is then computed with a matrix-vector product between the precomputed propagator $\mathbf{P}$, and the wavelet coefficients of the field U_{p_x}.

In the next section, an efficient way to account for the ground is presented.

2.3.3 Local image theorem

In this section, we present a method to consider the ground with a few additional points for SSW. We first assume a PEC ground and then generalize to an impedance ground.

PEC ground condition Contrary to DSSF, we can not change the wavelet transform with the appropriate transform – the sine transform – to efficiently consider the ground, because the wavelet transform is not related to a particular boundary condition. The first method that could be used is the image theorem, but the computation domain should be doubled, which is not efficient in terms of computational cost. A better method has been introduced in [52, 57]. While the image theorem doubles the number of points, the local image algorithm adds a thinner image layer. The main point of this method is that, at each step, the field in this thin image layer is updated by symmetry.

For a PEC the following steps are performed:

1. At x, the field u_x is of size N_z. Adding a local image layer under u_x on N_{im}, the new vector u_x^t of size $N_z + N_{im}$ is obtained as follows

$$u_x^t[p_z] = \begin{cases} u_x[p_z] & \text{for } p_z \in [1, N_z - 1], \\ 0 & \text{for } p_z = 0, \\ -u_x[-p_z] & \text{for } p_z \in [-N_{im}, -1]. \end{cases} \tag{2.19}$$

2. The field u_x^t is propagated using the SSW method to obtain $u_{x+\Delta x}^t$.

3. The thin image layer is removed. The propagated field is expressed as

$$u_{x+\Delta x}[p_z] = u_{x+\Delta x}^t[p_z] \qquad \text{for } p_z \in [0, N_z - 1]. \tag{2.20}$$

This method is repeated at each step. Note that, a spurious reflection appears over the bottom of the thin image layer. Nevertheless, N_{im} is chosen to ensure that this error does not reach the computation domain. Besides, the field is updated at each step using (2.19), clearing this error.

In practice N_{im} is chosen to be greater than the width of a wavelet after propagation. This width is theoretically known [107, 108] and N_{im} is thus pre-computed. Since the propagation support of a wavelet is much smaller than the computational domain, we have $N_{im} \ll N_z$, in practice.

Impedance ground condition First, an impedance ground condition is treated using the DMFT change of variable. With this change of variable, the space wave and the ground wave are treated separately, as with DSSF. Then, by linearity, both contributions are summed to obtain the total field.

For the space wave component, the same method is applied to w the field obtained from u by the DMFT change of variable and defined as (1.41). Therefore w follows a Dirichlet boundary condition and is propagated with the SSW with the local image theorem. Then, u is retrieved from w with the classical DMFT method.

In vertical polarisation, for the ground wave, the propagation is treated separately, with the same propagator as for DSSF, as detailed in Appendix A. Its contribution is added to the space wave so as to obtain the total propagated field.

Now that the SSW method has been recalled, its complexity is analyzed.

2.3.4 Complexity

In this section, the complexity of each step of SSW is given and compared to the one of DSSF. Since in the spatial domain, both methods are the same, we only compare their propagation steps.

First, a FWT and a compression are applied to the field. This step is of complexity $O(N_z + N_{im}) \simeq O(N_z)$, since $N_{im} \ll N_z$. Then the propagation step is shown [52, 57] to be of complexity $O(N_e)$ with $N_e = N_s N_m$, where N_m and N_s the number of non-zero coefficients of the propagator $\mathbf{P}$ and of the decomposition of the field, respectively. First, we have $N_e \ll N_z^2$ because of the compression. Second, simulations in [57] show that N_e can be less or of order N_z with a good choice of the threshold. Next, to come back in the spatial domain an inverse FWT of complexity $O(N_z)$ is applied. Finally, the total complexity is driven by

$$O(N_x \max(N_e, N_z)) \simeq O(N_x N_z). \tag{2.21}$$

As a reminder for DSSF the overall complexity is driven by

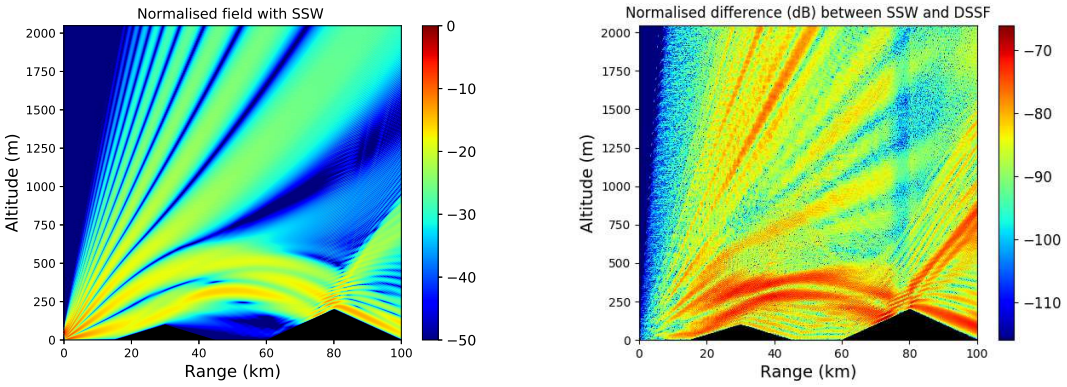
$$O(N_x N_z \log(N_z)). \tag{2.22}$$

Thus, if the compression rate is high enough, the SSW method should be faster than DSSF.

Note that, for SSW a precomputed propagator $\mathbf{P}$ is needed. This step has a complexity of

$$O(L N_z \log(N_z) + 2^L N_z). \tag{2.23}$$

This step is a serious limitation in terms of complexity for domains of large sizes, *e.g.* at high frequencies or for radio-occultation simulations. Besides, in terms of memory size, this propagator also becomes a burden for very large domains in 2D, since its occupation is rapidly increasing with N_z. This problem is specifically addressed in Chapter 3. The next section is dedicated to the validation of SSW.



(a) Normalized field obtained with SSW (dB).

(b) Normalised difference betwenn SSW and DSSF (dB).

Figure 2.9: Electrical field obtained with SSW and normalised difference with DSSF for the realistic scenario.

2.3.5 Numerical tests

Numerous numerical tests to validate SSW have been performed in [52, 57], showing the advantage of SSW over DSSF in terms of computation time when the compression rate is high. Besides the tests show that the error added by SSW is negligible and can be managed with the thresholds. In this section, the same realistic test as with DSSF in Section 1.2.3.b is performed. The same parameters are used.

In a few words, the propagation of a CSP over a dry ground with two triangular relief in the presence of an atmospheric duct is studied.

For this test, we introduce the normalised thresholds v_s and v_p such that

$$\begin{aligned} V_s &= v_s \max\left(|U_0|\right), \\ V_p &= v_p \max\left(|\mathbf{P}|\right). \end{aligned} \tag{2.24}$$

The thresholds v_s and v_p are then chosen to

$$\begin{aligned} v_s &= 3.16 \times 10^{-5}, \\ v_p &= 3.16 \times 10^{-5}. \end{aligned} \tag{2.25}$$

The electric field computed with SSW is presented in Figure 2.9 (a). The effects of both the atmosphere and the relief (shadowing, reflection, and diffraction) can be observed. The normalised difference between SSW and DSSF on the domain is plotted in Figure 2.9 (b). We can see that the error is negligible.

In terms of computation time, SSW needed 26 s to compute the propagator and 7.28 s to perform the propagation. For DSSF the propagation was computed in 6 s. Note that the SSW propagation part is a laboratory code in Python. The propagation in DSSF is computed with only a Python diagonal matrix-vector product, which is optimized in Python. For the SSW propagation part, a sparse matrix-sparse vector product should be computed, but in Python this is not possible. Therefore, a sparse matrix-dense vector or dense matrix-sparse vector is computed instead, which is not optimal and add a sparse

to dense vector or matrix computation. Thus, the computation time of the same order between both methods is explained. In terms of memory size requirement, with SSW a propagator of size 50 Mo was needed.

Tests performed in [52, 57] have shown that SSW has a better computation efficiency than DSSF with a negligible error in other cases. Nevertheless, the memory size of the propagator remains the main burden of this method.

In the next section, we derive a theoretical formula for the compression error. Also, we show that the heuristic formula proposed in [57] was too optimistic for some numerical tests.

2.4 Theoretical formula for the compression error

We have seen in Section 2.3, that with SSW two numerical parameters, V_s and V_p, are needed for the propagation. Both have a huge impact on the field propagation since the accuracy and the speed of the method highly depend on them. Indeed, V_s and V_p introduce two terms of error that accumulate while marching on in distances. Our goal is to be able to assess these two parameters to obtain a desired final error.

In [52, 57], a heuristic formula for the compression error has been given. In this section, we obtain a theoretical closed-form formula for the accumulated compression error. Thus, the signal and propagator thresholds V_s and V_p can be deduced from an expected final accuracy.

For the demonstration, we use the normalised thresholds v_s and v_p [94, 109] defined by equation (2.24). We also recall that the operator norm of an operator $\mathbf{P}$ corresponds to [110]

$$\|\mathbf{P}\|_{op} = \sup_{U\neq 0} \frac{\|\mathbf{P}U\|_2}{\|U\|_2}. \tag{2.26}$$

From power conservation, the operator norm of the free-space propagator $\mathbf{P}$ is equal to 1 ($\|\mathbf{P}\|_{op} = 1$). This is because:

- If there is no evanscent wave and the propagation does not reach any boundaries, then we have $\|\mathbf{P}u\|_2 = \|u\|_2$.
- In any other cases (environment losses, apodization, evanescent waves, ...), we have $\|\mathbf{P}u\|_2 < \|u\|_2$.

Assuming that both compression errors are independent, they are calculated separately and added so as to obtain the total accumulated error. First, a closed-form formula for an upper bound of the signal compression error is derived. However, this expression depends on the vertical size and is too large in practice. So a sharper approximate bound is proposed based on the properties of the wavelet decomposition. Secondly, a formula for the upper bound of the propagator compression is computed and the total error expression is deduced. Finally, numerical tests are performed to show the relevancy of the expressions.

2.4.1 Signal compression error

The objective of this section is to study how the signal compression error accumulates with N_x. We assume that $V_s \neq 0$ and $V_p = 0$. The propagator has no compression. The error due to the threshold V_s on signal (operator C_{V_s}) after N_x iterations is defined by

$$\delta^s_{N_x} = \frac{\|\tilde{U}_{N_x} - U_{N_x}\|_2}{\|U_0\|_2}, \tag{2.27}$$

with

$$\tilde{U}_{N_x} = (\mathbf{P}\mathbf{C}_{V_s})^{N_x} U_0 \qquad \text{and} \qquad U_{N_x} = \mathbf{P}^{N_x} U_0, \tag{2.28}$$

the compressed and uncompressed propagated coefficients, respectively. U_0 is the wavelet decomposition of the initial field.

For one iteration, the error is given by

$$\delta^s_1 = \frac{\|\mathbf{P}\mathbf{C}_{V_s} U_0 - \mathbf{P} U_0\|_2}{\|U_0\|_2}. \tag{2.29}$$

We introduce ϵ_0 the compression term due to C_{V_s} defined by

$$\mathbf{C}_{V_s} U_0 = U_0 + \epsilon_0. \tag{2.30}$$

Using (2.26) and putting (2.30) in (2.29) gives an upper-bound for the compression error

$$\delta^s_1 \leq \frac{\|\epsilon_0\|_2}{\|U_0\|_2} \leq v_s \frac{\sqrt{N_z - M}\max\left(|u_0|\right)}{\|U_0\|_2} \leq v_s \frac{\sqrt{N_z}\max\left(|u_0|\right)}{\|U_0\|_2}, \tag{2.31}$$

with $M \ll N_z$ the number of non-zero coefficients after compression. This bound is large and increases with N_z, which is not satisfying from a physical point of view.

Thus, we now derive a sharper bound from the properties of wavelets and the smoothness of the physical signals we deal with. In this case, the coefficients rapidly decrease to 0. We rewrite the norm of the error as follows

$$\|\epsilon_0\|_2^2 = v_s^2 \max^2\left(|U_0|\right) \sum_{m=0}^{N_z-1} |\hat{\epsilon}_m|^2, \tag{2.32}$$

with the coefficients $|\hat{\epsilon}_m| \leq 1$ corresponding to the normalised amplitudes of the wavelet coefficients of the error indexed in decreasing order. Following [109,111], error components are bounded by

$$|\hat{\epsilon}_m| \leq C_\epsilon (m+1)^{-n_v}, \tag{2.33}$$

with $m \in [0, N_z - 1]$, n_v the number of vanishing moments and C_ϵ a constant depending only on the smoothness of the field and of the wavelets. Putting (2.33) in (2.32), an upper bound for $\|\epsilon_0\|_2$ is obtained

$$\|\epsilon_0\|_2 \leq v_s \max\left(|U_0|\right) C_\epsilon \sqrt{\sum_{m=0}^{N_z-1} (m+1)^{-2n_v}}. \tag{2.34}$$

Firstly, for $n_v \geq 2$ the sum converges rapidly close to 1 (*e.g.* for $n_v = 2$ the sum is about 1.082). Secondly, the constant C_ϵ decreases with n_v and becomes of order 1, *i.e.* between

1/5 and 5. This result is illustrated with numerical tests in Section 2.4.3.a. Finally, since C_ϵ is of order 1 and $\max(|U_0|) \leq \|U_0\|_2$, we use in practice that $\max(|U_0|)\, C_\epsilon \lesssim \|U_0\|_2$, where $\lesssim$ means inferior or close to.

Thus, the bound is approximated by

$$\|\epsilon_0\|_2 \lesssim v_s \|U_0\|_2 \qquad \text{thus} \qquad \delta_1^s \lesssim v_s. \tag{2.35}$$

In practice, this result shows a very good accuracy with numerous numerical tests performed in [52, 102].

For 2 iterations, we compare the propagations with and without compression

$$\delta_2^s = \frac{\|(\mathbf{PC}_{V_s})(\mathbf{PC}_{V_s})U_0 - \mathbf{PP}U_0\|_2}{\|U_0\|_2}. \tag{2.36}$$

We define the second compression error ϵ_1 such as $\mathbf{C}_{V_s}(\mathbf{P}U_0 + \mathbf{P}\epsilon_0) = \mathbf{P}U_0 + \mathbf{P}\epsilon_0 + \epsilon_1$. The expression of the error is calculated as

$$\begin{aligned} \delta_2^s &= \frac{1}{\|U_0\|_2} \|\mathbf{PP}U_0 + \mathbf{PP}\epsilon_0 + \mathbf{P}\epsilon_1 - \mathbf{PP}U_0\|_2 \\ &\leq \frac{1}{\|U_0\|_2} \left(\|\epsilon_0\|_2 + \|\epsilon_1\|_2\right). \end{aligned} \tag{2.37}$$

Supposing (2.35) is true for the 2nd iteration, we obtain

$$\frac{\|\epsilon_1\|_2}{\|U_0\|_2} \lesssim \frac{v_s(\|U_0\|_2 + \|\epsilon_0\|_2)}{\|U_0\|_2}. \tag{2.38}$$

Since, with appropriate threshold, $\|\epsilon_0\|_2$ is negligible to $\|U_0\|_2$, we have

$$\delta_2^s \lesssim 2v_s. \tag{2.39}$$

By induction, the signal compression error after N_x horizontal iterations fulfills

$$\delta_{N_x}^s \lesssim N_x v_s. \tag{2.40}$$

The appropriate threshold $V_s = v_s \max(|U_0|)$ can now be computed with (2.40).

2.4.2 Propagator compression error

We now assume that $V_s = 0$ and $V_p \neq 0$. The error $\delta_{N_x}^p$ due to the compression of the propagator after N_x iterations is studied. It is defined by

$$\delta_{N_x}^p = \frac{\|\tilde{U}_{N_x} - U_{N_x}\|_2}{\|U_0\|_2}, \tag{2.41}$$

where $\tilde{U}_{N_x}$ corresponds to the coefficients propagated N_x times with the compressed operator denoted as $\mathbf{P} + \Delta\mathbf{P}$. Furthermore U_{N_x} corresponds to the ones propagated with $\mathbf{P}$.

From [109], we have the norm operator of ΔP bounded by

$$\|\Delta \mathbf{P}\|_{op} = \sup_{U\neq 0} \frac{\|\Delta \mathbf{P} U\|_2}{\|U\|_2} \leq v_p. \tag{2.42}$$

For one iteration the expression of the error is given by

$$\delta_1^p = \frac{\|\tilde{U}_1 - U_1\|_2}{\|U_0\|_2} = \frac{\|\Delta \mathbf{P} U_0\|_2}{\|U_0\|_2}. \tag{2.43}$$

Following (2.42), we have

$$\delta_1^p \leq v_p. \tag{2.44}$$

Using the same notations and methodology as for 1 iteration and since $\|P\|_{op} = 1$, we obtain for 2 iterations using (2.42)

$$\begin{aligned} \delta_2^p &= \frac{\|\tilde{U}_2 - U_2\|_2}{\|U_0\|_2} \\ &\leq \frac{\|\Delta \mathbf{P}\mathbf{P} U_0\|_2 + \|\mathbf{P}\Delta \mathbf{P} U_0\|_2 + \|\Delta \mathbf{P}^2 U_0\|_2}{\|U_0\|_2} \\ &\leq 2v_p + v_p^2. \end{aligned} \tag{2.45}$$

Neglecting the term v_p^2 (since $v_p \ll 1$), δ_2^m is shown to be less than or of order $2v_p$. By induction, we finally obtain

$$\delta_{N_x}^p \lesssim v_p N_x. \tag{2.46}$$

Formula (2.46) allows to choose the adequate threshold $V_p = v_p \sup(|\mathbf{P}|)$ for a given error and scenario.

Assuming that both errors are independent, we finally derive a closed-form expression for the accumulated compression error

$$\delta_{N_x} \lesssim (v_s + v_p) N_x. \tag{2.47}$$

For a given maximum expected error $\delta_{N_x}^{\max}$ and number of iterations N_x, the normalised thresholds are computed as

$$v_s = \frac{\delta_{N_x}^{\max}}{2N_x} \text{ and } v_p = \frac{\delta_{N_x}^{\max}}{2N_x}. \tag{2.48}$$

Thus, we derive the unnormalised thresholds

$$V_s = \frac{\delta_{N_x}^{\max}}{2N_x} \max(|U_0|) \text{ and } V_p = \frac{\delta_{N_x}^{\max}}{2N_x} \max(|\mathbf{P}|). \tag{2.49}$$

Now some numerical tests are performed to compare the closed-form expression with numerical root mean square (RMS) error due to compressions.

2.4.3 Numerical tests

In this section, several numerical tests are performed:

- To illustrate the convergence of the constant C_ϵ introduced in the theoretical analysis in Section 2.4.1.
- To show that for a source with smooth variations the formula is relevant.
- To test for a source with steep variations that the formula is still relevant to manage the error.
- To show that the formula is conservative in a context of realistic propagation with losses.

The two different sources are a CSP, introduced in Chapter 1, and a uniform aperture. This latter is defined by a position x_s and z_s, and by its width w_z. The field on the aperture is given by

$$u(0,z) = \begin{cases} E_0 & \text{if } z_s - w_z/2 \leq z \leq z_s + w_z/2, \\ 0 & \text{otherwise.} \end{cases} \tag{2.50}$$

Under the far field assumption, the field radiated from this aperture can be computing using the stationary phase theorem [112] and is given by

$$u(x,z) = E_0 w_z \cos(\theta) \sqrt{\frac{jk_0}{2\pi d}} \text{sinc}\left(\frac{k_0 w_z \sin(\theta)}{2}\right) \exp(-jk_0 d), \tag{2.51}$$

with $d = \sqrt{(x-x_s)^2 + (z-z_s)^2}$ and θ the angle with the x-axis.

2.4.3.a Convergence of C_ϵ

In this test, we aim at showing that the constant C_ϵ introduced in section 2.4.1 becomes of order 1 with n_v. First the test is performed with the CSP as the initial field, then with the uniform aperture.

We introduce C'_ϵ, which is defined by

$$C'_\epsilon = \frac{\|\epsilon_0\|_2}{v_s \max\left(|U_0|\right)}, \tag{2.52}$$

with v_s the signal threshold. We want to study the asymptotic behavior of C'_ϵ with N_z. We aim at showing that this constant converges with N_z and becomes of order 1, in order to assess whether the approximation is relevant.

Complex source point The decomposition and compression with a threshold of $v_s = 10^{-2}$ of the field of a complex source point (CSP) are performed. The frequency is 300 MHz and the CSP is placed at $x_s = -50$ m with $W_0 = 5$ m and $z_s = 256$ m. The vertical size of the domain is $z_{\max} = 512$ m. The Daubechies family with $L = 3$ is chosen. The convergence is tested for $n_v \in \{2, 6, 8\}$.

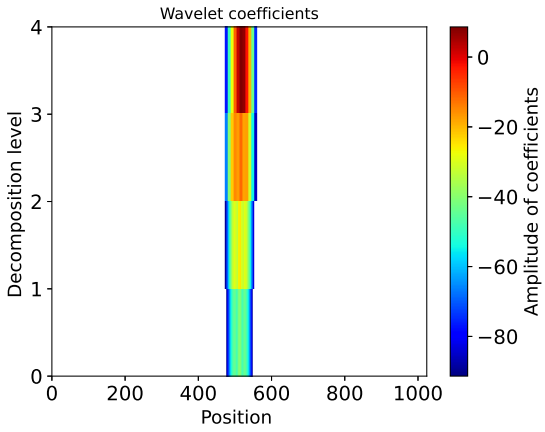
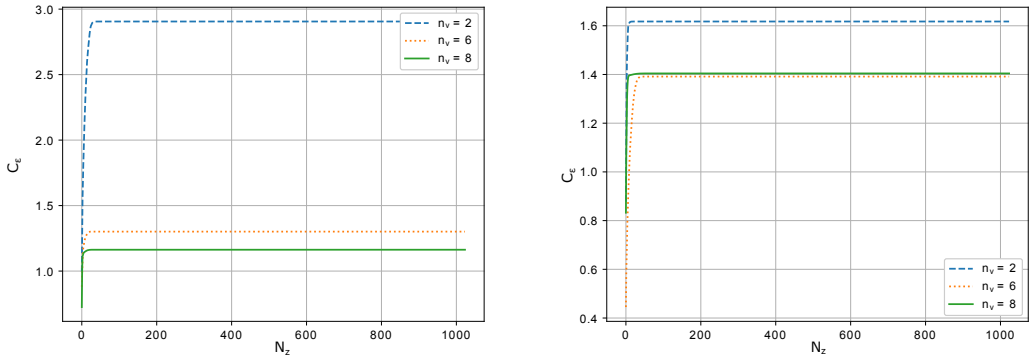


Figure 2.10: Wavelet decomposition of the field radiated from a CSP.

First, the wavelet decomposition with $n_v = 2$ is presented in Figure 2.10 with respect to the position p_z and the level of the decomposition $l \in [1, 3]$. Level 4 corresponds to the scaling function. Since the signal is smooth, few coefficients are needed and they are mostly on the higher levels.

The evolution of C'_ϵ with N_z is plotted in Figure 2.11.



(a) $v_s = 10^{-2}$ (b) $v_s = 10^{-4}$

Figure 2.11: Evolution of C'_ϵ with N_z for a CSP: (a) a threshold $v_s = 10^{-2}$ is applied, (b) a threshold $v_s = 10^{-4}$ is applied.

The first conclusion is that with n_v increasing, C'_ϵ is of order 1, as expected. We can also note that the convergence with N_z is faster with n_v high. Besides, decreasing v_s induces that the constant becomes of order 1 for the higher n_v. Thus, we can conclude that the approximation of the upper bound for the signal compression is accurate for a smooth signal.

Uniform aperture Now, we aim at showing that the same result is relevant for a signal with steep variations. The same test is performed for the field radiated from a uniform aperture of width $w_z = 5$ m. The other parameters remain unchanged.

The wavelet decomposition with $n_v = 2$ is presented in Figure 2.12 with respect to the

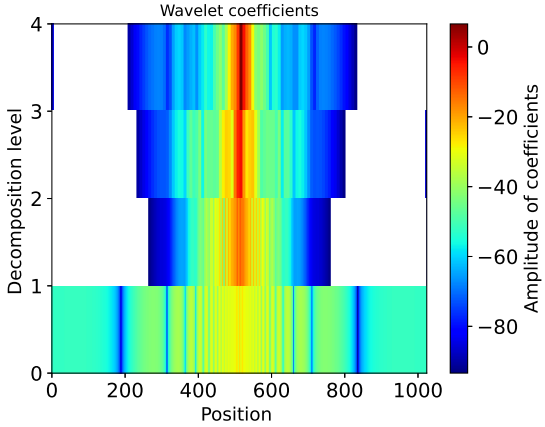
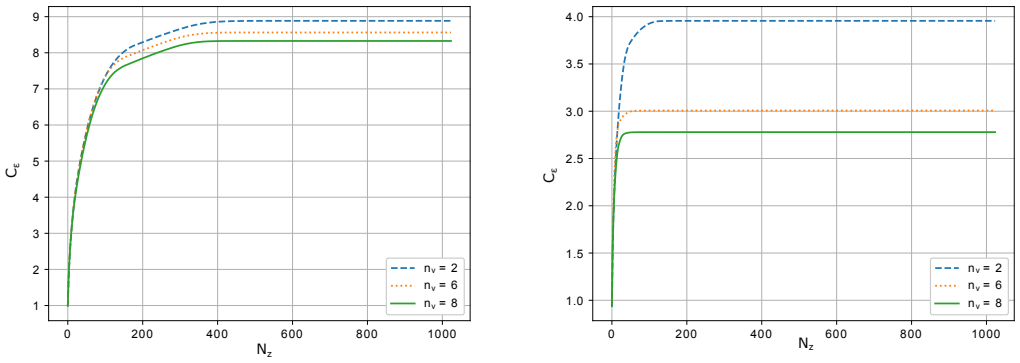


Figure 2.12: Wavelet decomposition of the field radiated by a uniform aperture.

position and the level of the decomposition. The signal has steeper variations, thus more coefficients are needed to represent it.

The evolution of C'_ϵ with N_z is presented in Figure 2.13.



(a) $v_s = 10^{-2}$ (b) $v_s = 10^{-4}$

Figure 2.13: Evolution of C'_ϵ with N_z for a uniform aperture. (a) A threshold $v_s = 10^{-2}$ is applied. (b) A threshold $v_s = 10^{-4}$ is applied.

The same conclusion can be made on the convergence with n_v, as expected. We can add that in this case decreasing the threshold induces a faster convergence with N_z of the constant for all n_v. Nevertheless, since the signal has steeper variations, we are not as close to 1. Nevertheless, the constant C'_ϵ remains low compared to the domain size (N_z) and of order 1. Therefore, the approximation is still relevant. Besides, in long-range propagation, thresholds are in general of order 10^{-4} or below as can be seen in the following tests and in [52, 57]. Furthermore, the RMS error is also below v_s in these cases [52]. Thus, the approximation remains accurate.

2.4.3.b Test of the compression error formula for a CSP

In this section, we test formulas (2.40), (2.46) and (2.48) in a first propagation scenario, with a source with smooth variations. We aim at showing that the formulas are relevant,

which means that they allow to bound the error.

The source is a CSP at $f_0 = 300$ MHz of width $W_0 = 5$ m and is placed at $x_s = -50$ m and $z_s = 1024$ m in a domain of vertical size $z_{\max} = 2048$ m. The domain is of size $x_{\max} = 2000$ m in horizontal. The steps are $\Delta x = 20$ m and $\Delta z = 0.5$ m. Thus, we have $N_x = 100$. For the wavelet parameters, the symlet with $n_v = 6$ and a maximum level of $L = 3$ are chosen.

The RMS error between compressed and uncompressed propagations is computed for different values of N_x and compared to the closed-form formulas. Thresholds are chosen so as to obtain a maximum given error of -30 dB at the final iteration.

Firstly, a threshold is only applied on the signal, *i.e.*, $v_p = 0$. Secondly, a threshold is only applied on the propagator, *i.e.*, $v_s = 0$. Thirdly, both thresholds are applied. Using the aforementioned formulas we obtain the following thresholds for the three cases

$$\begin{aligned} &v_s = 3.16 \times 10^{-4}\,, && v_p = 0, && \text{1st case,} \\ &v_s = 0\,, && v_p = 3.16 \times 10^{-4}, && \text{2nd case,} \\ &v_s = 1.6 \times 10^{-4}\,, && v_p = 1.6 \times 10^{-4}, && \text{3rd case.} \end{aligned}$$

To illustrate the configuration, the field obtained in the third case is presented in Figure 2.14, with respect to distance and altitude.

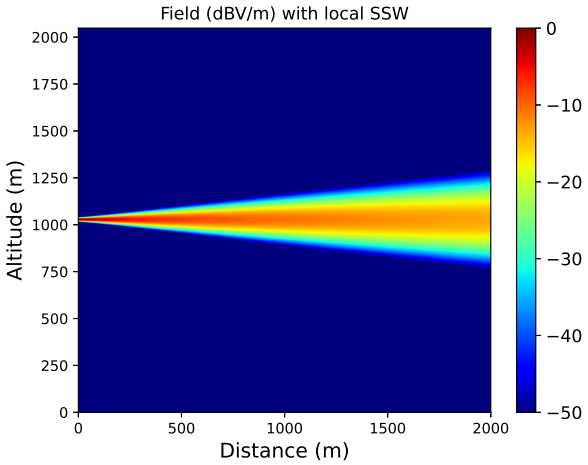


Figure 2.14: Field obtained with SSW.

First, we show that the constant C'_ϵ is close to 1 along the propagation. The constant is computed for each iteration and the result is plotted in Figure 2.15. As expected, the constant is of order 1, showing that the approximation introduced in Section 2.4.1 is relevant.

Then for all the tests, the RMS errors are computed and given in Figure 2.16 in continuous lines. The analytic bounds are plotted in dotted lines.

As expected, Figure 2.16 shows that the closed-form expression for the accumulated signal compression error is never reached. The computed thresholds allow bounding the error below the desired maximum for a paraxial field.

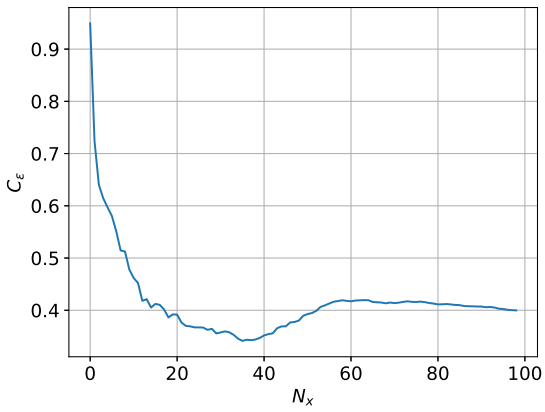
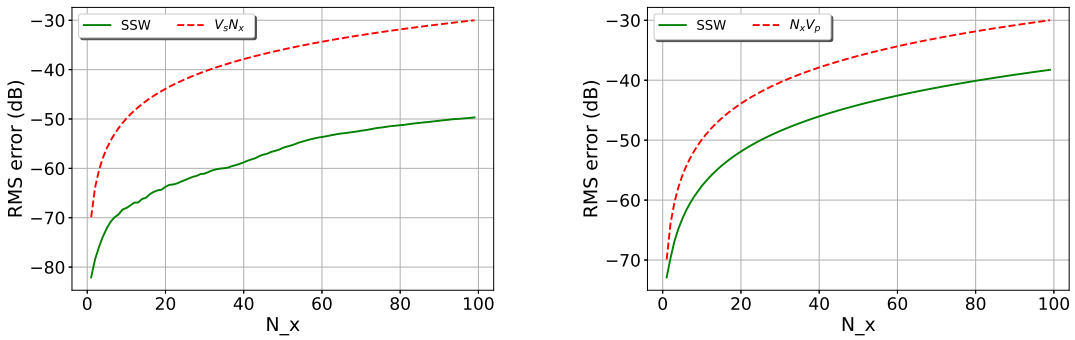
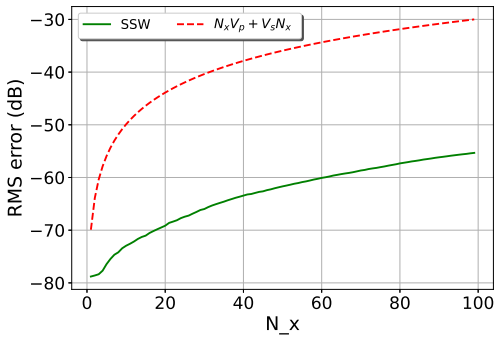


Figure 2.15: Evolution of C'_ϵ with the number of iterations N_x.



(a) First case: $v_s = 3.1 \times 10^{-4}$.

(b) Second case: $v_p = 3.1 \times 10^{-4}$.



(c) Thirst case: $v_s = v_p = 1.6 \times 10^{-4}$

Figure 2.16: Evolution of the RMS error with N_x for the field radiated from a CSP.

We also compute a linear regression to find the optimal α such that $\delta^s_{N_x} \sim v_s N_x^\alpha$ and $\delta^p_{N_x} \sim v_p N_x^\alpha$. For the signal compression, we obtain $\alpha = 0.98$ a bit lower than the value proposed here, *i.e.*, 1, but greater than the heuristic value proposed in [57], *i.e*, 0.5. For the propagator, $\alpha = 0.99$ is obtained, close to the value proposed here. This shows that the heuristic formula proposed in [57] was too optimistic.

2.4.3.c Test of the compression error formula for a uniform aperture

Second, we perform the same three tests with a source with steeper variations to show the relevancy of the formulas (2.40), (2.46) and (2.48).

The source is a uniform aperture at $f_0 = 300$ MHz of size $w_y = 10$ m and is placed at $x_s = 0$ and $z_s = 1024$ m in a domain of vertical size $z_{\max} = 2048$ m. The other parameters remain the same.

Since the domain is of the same size and the final desired error is the same, the three thresholds remain the same

$$\begin{aligned} &v_s = 3.16 \times 10^{-4}\,, && v_p = 0 && \text{1st case,} \\ &v_s = 0\,, && v_p = 3.16 \times 10^{-4} && \text{2nd case,} \\ &v_s = 1.6 \times 10^{-4}\,, && v_p = 1.6 \times 10^{-4} && \text{3rd case.} \end{aligned}$$

First, to illustrate the configuration, an example of the field obtained with SSW with $v_s = 1.6 \times 10^{-4}$ and $v_p = 1.6 \times 10^{-4}$ is given in Figure 2.17, with respect to distance and altitude.

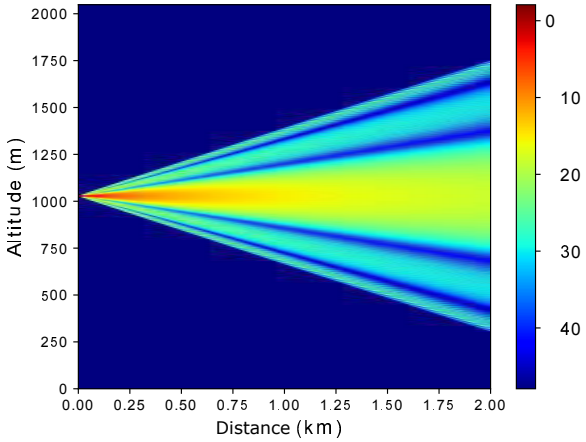


Figure 2.17: Field obtained with SSW.

As in 2.4.3.b, we compute the constant C'_ϵ at each step of propagation N_x to show that the constant is low and close to 1. This experiment is illustrated in Figure 2.18. As expected, the constant C'_ϵ is of order 1 so that the approximation proposed in Section 2.4.1 is relevant.

Then for all the tests, the RMS errors are computed and given in Figure 2.19 in continuous lines. The analytic bounds are plotted in dotted lines.

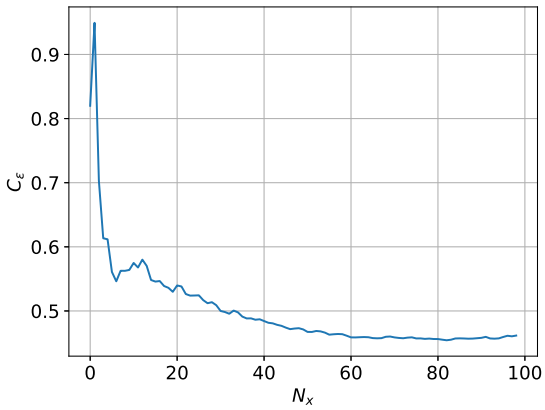
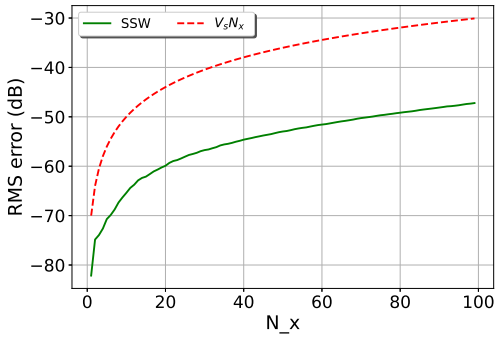
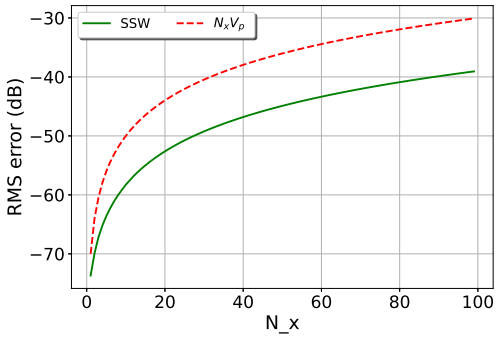


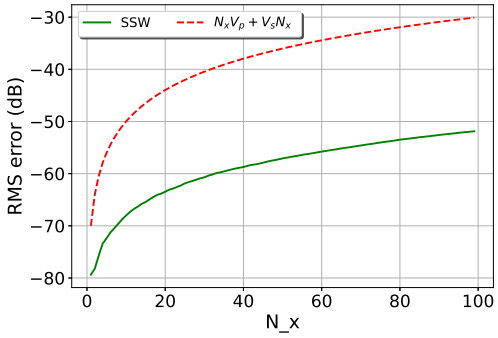
Figure 2.18: Evolution of C'_ϵ with the number of iterations N_x.



(a) First case: $v_s = 3.1 \times 10^{-4}$.



(b) Second case: $v_p = 3.1 \times 10^{-4}$.



(c) Thirst case: $v_s = v_p = 1.6 \times 10^{-4}$

Figure 2.19: Evolution of the RMS error 2D with N_x for the field radiated by a uniform aperture.

As expected and as in the previous test, Figure 2.19 shows that the closed-form expression for the accumulated signal compression error is never reached. The thresholds computed allow to bound the error below the desired maximum for a less paraxial field.

As in 2.4.3.b, we can compute a linear regression to find the optimal α such that $\delta^s_{N_x} \sim v_s N_x^\alpha$ and $\delta^p_{N_x} \sim v_p N_x^\alpha$. For the signal compression and the propagator compression, we obtain $\alpha = 0.96$ and $\alpha = 0.97$, respectively. The same conclusions hold.

Our proposed bounds are never reached, as expected, which shows the relevancy of these latter.

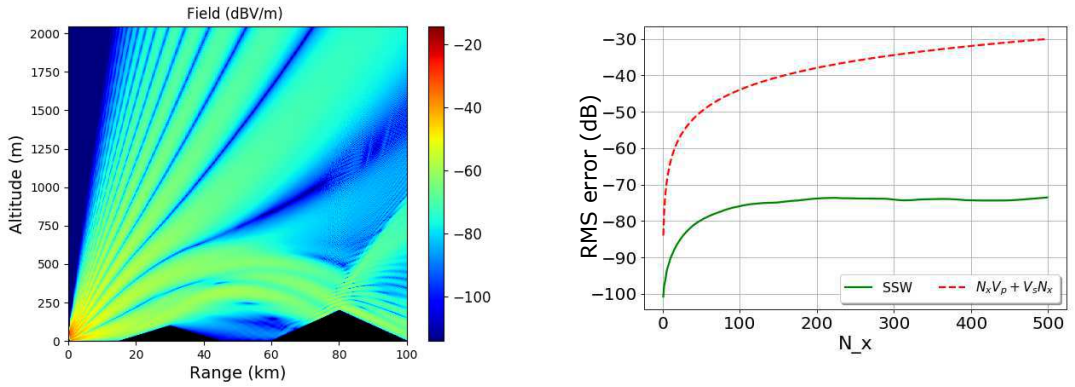
2.4.4 Realistic test case

Finally, a test in realistic conditions with refraction and relief is performed. We aim at showing that the proposed formula is still relevant.

The same test as in section 2.3, with the same parameters is performed. Now we aim at obtaining an error of -30 dB at the final iteration, giving for the thresholds

$$v_s = 3.16 \times 10^{-5}, v_p \quad = 3.16 \times 10^{-5}. \tag{2.53}$$

In Figure 2.20, the field in dBV/m is plotted in (a) and the RMS error evolution is plotted in (b). The analytic bound is also plotted in (b). We can see that the bound is not reached and that the final error is significantly smaller than the desired error. This is mostly due to the apodization layer in which energy is leaving the computational domain, reducing the total error. Therefore, our formula is conservative in a realistic domain.



(a) Field obtained with SSW (dBV/m).

(b) RMS error evolution (dB).

Figure 2.20: Field obtained with SSW and Evolution of the RMS error for the realistic test case.

To conclude this section, the expression obtained for the accumulated compression error can now be used in SSW to automatically set V_s and V_p for a given accuracy.

2.5 Conclusion

In this chapter, a state-of-the-art on the wavelet transform in 1D and the split-step wavelet method have been presented. Then, formulas to compute the compression threshold for SSW have been derived.

Firstly, a brief state-of-the-art on the discrete multilevel wavelet decomposition has been presented. The wavelet family and its associated discrete wavelet transform have been properly defined. Then, the fast wavelet transform to obtain the set of wavelet coefficients with a low complexity has been described. We also have introduced the sparse representation capacity of the wavelets. Finally, some important wavelet properties for the wavelet representation and for the theoretical analysis have been recalled.

Secondly, the SSW method has been recalled. This is an iterative method that computes the propagation of a field by going back and forth in the wavelet and spatial domains. First, the field is represented as a sparse set of coefficients after applying a FWT and a compression. Then, the coefficients are propagated with a pre-computed scattering matrix storing all the compressed wavelet-to-wavelet propagations. The strategy to compute the propagator efficiently has been recalled. Then, an inverse FWT is applied to come back to the spatial domain and to treat the effects of the environment as with DSSF. Both compressions introduce errors. With a high compression rate, this method has a better computation time than DSSF, but the memory needed by the propagator is a burden for propagation over very large domains in 2D and for a generalization to 3D.

Thirdly, we have calculated how the signal compression error and the propagator compression error accumulate while marching on in distances. The expressions of both compression errors have been summed to obtain a closed-form expression that assesses the accumulated compression error. Numerical tests in 2D have been performed. In these tests, we have computed V_s and V_p for a given maximum error and scenario. Our proposed bounds are never reached, as expected, which show their relevancy. To conclude, the expression we have obtained for the accumulated compression error can now be used in SSW to automatically set V_s and V_p for a given accuracy.

The next chapter is dedicated to the improvement of SSW, by modifying the computation and storage of the operator in order to reduce the memory size requirement.

Chapter 3

Local Split-Step Wavelet in 2D

3.1 Introduction

In Chapter 2, a matrix version of split-step wavelet [57], denoted in this chapter by mSSW, has been introduced for the 2D case with an invariance along y. Complexity theory shows that this method is better than DSSF in terms of computation time when a high compression rate is achieved. Nevertheless, this method cannot be directly generalized to 3D because of the memory cost of the propagator. Numerical tests [20, 52] show that propagators of order 1 Go are needed for large 2D scenarios. Besides this propagator is computed once and for all for a given scenario (grid size), preventing any versatility.

In this chapter, we propose an improved version of SSW to reduce the memory size requirement while keeping the advantage of computation time. The duplications needed to fill the matrix are avoided to only store the essential information with few propagators. The whole set of propagators corresponds to the minimal subset of the matrix essential to the propagation.

Also, to further limit the computation time requirement the limited support of the wavelets is accounted to significantly speed up the propagator computation. Thus, the propagators could be computed on the fly. Therefore, versatility is added to the method.

With this set, the propagation is locally performed for each non-zero coefficient of the wavelet decomposition of the field and summed afterwards. Therefore, this method is called local SSW and is denoted by lSSW.

The content of this chapter corresponds to several of my submitted or accepted publications [107, 108, 113, 114].

The chapter is organized as follows. In Section 3.2, we comprehensively introduce the local SSW method. Firstly, the configuration of the problem is recalled. Secondly, the general SSW method is reminded to highlight where the differences between mSSW and lSSW occur. Thirdly, the computation of the set of propagators is introduced. We then present the local propagation. We conclude the section with the numerical scheme of lSSW and discuss its complexity. In Section 3.3, numerical experiments to test the method are performed. First, we assess that using the limited support of the wavelets allows reducing the computation time. Then, we show that the compression error formula obtained in the previous chapter can still be used. This implies that the accuracy is preserved. Finally, a

realistic test case scenario is performed, where mSSW and lSSW are compared in terms of accuracy, memory requirement, and computation time. In Section 3.4, an application of the section to model the forward propagation with a radio-occultation (RO) scenario is performed. The advantage of lSSW on such a large scenario is presented. Section 3.5 concludes the chapter.

3.2 Local propagation in SSW

In this section lSSW, the local version of SSW, is presented. The configuration and discretization are the same as with DSSF in Chapter 1.

3.2.1 Overview of the method

The local SSW (lSSW) method follows the same steps as the matrix SSW (mSSW) presented in the previous chapter. The propagation part is the only difference. Whereas mSSW uses a scattering matrix and matrix-vector product, we introduce a local propagation based on the minimal needed number of wavelet-to-wavelet propagators.

To describe the method we first assume that the ground is a PEC.

A step of SSW on Δx with u_x known is computed as follows:

1. The sparse vector of the wavelet coefficients of u_x is obtained using a FWT (operator $\mathbf{W}$) and a compression of threshold V_s (operator $\mathbf{C}$)

$$U_x = \mathbf{C}\mathbf{W}u_x. \tag{3.1}$$

2. The propagated non-zero coefficients $U_{x+\Delta x}$ are obtained using the propagator P

$$U_{x+\Delta x} = PU_x. \tag{3.2}$$

   In the previous chapter the propagator was a pre-computed matrix, denoted by $\mathbf{P}$ [57], that stored all the wavelet-to-wavelet propagations such that the propagation was a sparse matrix-vector product. Here, we propose the use of a set of local propagators [107, 108, 114], denoted by P. This is described in Sections 3.2.2 to 3.2.3. Note that, the notation $\mathbf{P}$ corresponds to a matrix whereas P is not a matrix. This latter corresponds to an operator that sums all the local propagations of each non-zero wavelet coefficient of the field.

3. The free-space propagated field $u^{\text{fs}}_{p_x+1}$ is obtained by applying an inverse FWT

$$u^{\text{fs}}_{x+\Delta x} = \mathbf{W}^{-1}U_{x+\Delta x}. \tag{3.3}$$

   $u^{\text{fs}}_{x+\Delta x}$ is expressed in the space domain.

4. Refraction (phase screen) $\mathbf{R}$, relief $\mathbf{L}$ and apodisation $\mathbf{H}$ are taken into account in the space domain as with mSSW

$$u_{x+\Delta x} = \mathbf{H}\mathbf{R}\mathbf{L}u^{\text{fs}}_{x+\Delta x}. \tag{3.4}$$

3.2.2 Computing the propagators

In this section, we introduce the efficient computation of the set of propagators [107].

To compute efficiently the propagators two important properties of wavelets are used:

- The invariance by translation property to store only the essential information and to compute and store only the essential information.
- The small size of the support of one wavelet, denoted by N_l, to reduce the computation time. We note N_l^{p} the support needed after propagation.

The propagations of as few as possible elementary wavelets are stored and used to propagate all the wavelet coefficients of the field. These propagations are stored in a set denoted by P. P_{l,p_t^l} corresponds to a N_l^{p}-sized vector, $N_l^{\mathrm{p}} \ll N_z$, with the wavelet coefficients of a propagated l level wavelet at position p_t^l. Therefore P is not a matrix but corresponds to a set of propagators of reduced size. Then, for each coefficient of the wavelet decomposition of the field, an element of this set returns the propagated coefficients. Summing all these elementary propagations leads to obtaining the propagated field.

3.2.2.a Computing P_{l,p_t}

First, we explain the steps to compute the local propagator.

We note l the level of the wavelet to propagate and $l' \in [1, L[$ the levels of the coefficients after propagation. As for mSSW, we have a translation property that we recall as follows for a given l':

- if $l = l'$, then a translation of 1 on l corresponds to the same translation on l'.
- if $l' < l$, then there are $2^{l-l'}$ times as many wavelets on l' than on l and a translation 1 on l corresponds to a translation of $2^{l-l'}$ on level l'.
- if $l' > l$, then it is the inverse case, and on l' there are $2^{l'-l}$ times less wavelets than on l. Thus, a translation of $2^{l'-l}$on l corresponds to a translation of 1 on l'.

The space grids of the wavelets are dilated by 2 at each level. Consequently, the decomposition of a wavelet at position p of level l is 2^{L-l}-periodic at each level l. Considering this, the propagation operator in the wavelet domain is 2^L-periodic. Therefore, only 1 period for each level is stored such that the memory requirements are minimum, see Figure 3.1. We store no redundant information, whereas the scattering matrix contains all the propagations for all levels and positions.

Algorithm 1 shows how to compute the set of propagators using this property. The computation of this set can be decomposed into two steps. First, for each level l, the wavelet $\chi_{l,0}[p_z]\,\forall p_z \in [0, N_z - 1]$ at $x = 0$ is propagated on a distance Δx using DSSF. The wavelet $\chi_{l,\Delta x}[p_z]$ is obtained. Other methods can be used to compute the wavelet propagations, *e.g.* [115]. It could increase precision but the computation time would also be impacted.

Secondly, the wavelet field $\chi_{l,\Delta x}[p_z]$ is translated of $p_t^l \in [0, 2^{L-l}[$ to obtain $\chi_{l,\Delta x}[p_z + p_t^l]\,\forall p_z + p_t^l \in [0, N_z - 1]$ for all the needed translations. Then, it is decomposed and compressed (with threshold V_p) to obtain P_{l,p_t}. These propagations are then stored for

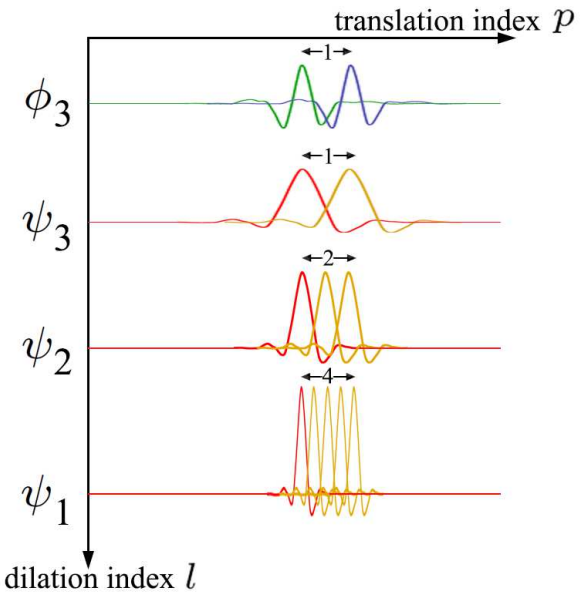


Figure 3.1: 1 period of the wavelet space grid with $L = 3$.

all levels l and translations p_t^l as a set of propagators $\left(P_{l,p_t^l}\right)_{l\in[0,L],p_t^l\in[0,2^{L-l}[}$. Using the translations we only need to compute $L+1$ propagations and to perform 2^L FWT. Thus, only 2^L propagators are stored. Zhou *et al.* [57] uses the translation property to efficiently fill the matrix by duplicating the propagations, thus redundant information was stored [52]. Here, the set of propagators corresponds to the minimal subset of this matrix required for the propagation.

Algorithm 1 Computing the propagators

Inputs: maximum level L, wavelet basis
Output: set of propagators $\left(P_{l,p_t^l}\right)_{l\in[0,L],p_t^l\in[0,2^{L-l}[}$
\\ at each wavelet level
for $l \in [1, L]$ **do**
  $\chi_{l,0}[p_z] \leftarrow$ wavelet at level l and position 0 .
  $\chi_{l,\Delta x}[p_z] \leftarrow$ propagated wavelet with DSSF.
  \\ 2^{L-l} periodicity of the decomposition
  for $p_t^l \in [0, 2^{L-l}[$ **do**
    $\chi_{l,\Delta x}[p_z + p_t^l] \leftarrow$ translate $\chi_{l,\Delta x}[p_z]$ along p_t^l points.
    $P_{l,p_t^l} \leftarrow$ apply FWT and compression to $\chi_{l,\Delta x}[p_z + p_t^l]$.
  end for
end for

3.2.2.b Reducing the computation time with the wavelet support

Now the small size of the support N_l of the wavelet is used to speed up the computation of this set. Each wavelet at level l has a support of finite size theoretically known [94]. In practice, the support is much smaller than the size of the domain ($N_l \ll N_z$). Therefore, propagation can be performed on a reduced domain, using N_l^{p}. This latter depends only on the level l and the step size Δx. To assess the value of this parameter, we use the

support before propagation N_l, and the angle of propagation corresponding to the validity domain of the wide-angle PWE ($\pm\pi/4$) [11] as shown in Figure 3.2. Using the propagation step Δx, and the discretization step Δz the number of points after propagation is given by

$$N_l^{\mathrm{p}} = \left\lceil N_l + \sqrt{2}\frac{\Delta x}{\Delta z} \right\rceil, \tag{3.5}$$

where $\lceil \cdot \rceil$ denotes the ceiling function.

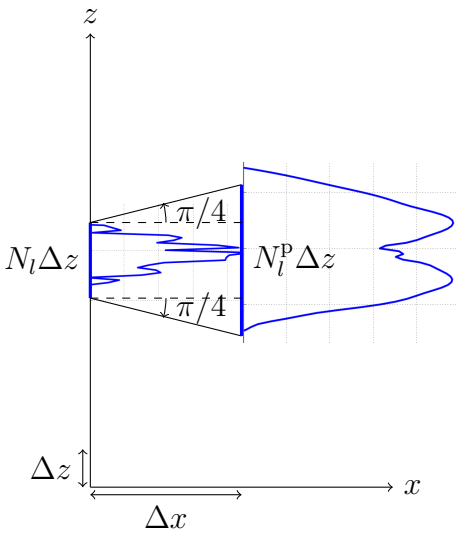


Figure 3.2: Wavelet supports before and after propagation on one step.

For example, with $\Delta x = 100$ m and $\Delta z = 1$ m using the symlet with $n_v = 6$ the support of the mother wavelet is $2n_v - 1 = 5$ m, with $L = 3$. Thus, for $l \in [1, L]$ we have $N_l = l(2n_v - 1) = 5l$ m and $N_l^{\mathrm{p}} = 5l + 100 \times \sqrt{2}$. Hence, the support after propagation is of at most 156 points. Since in practice domain are of order thousand of points this latter is much smaller than the total domain. Besides, N_l^{p} remains the same even with N_z increasing if Δx and Δz remain constant.

Thus, the DSSF propagation on N_l^{p} is much faster than on the overall domain.

Besides N_l^{p} does not depend on N_z. Then, the propagation time for computing$P$ is independent of N_z. This method is cheap in terms of memory size occupation. Moreover, for large domains ($N_l^{\mathrm{p}} \ll N_z$) the computation of the set (of complexity $O(\sum_l N_l^{\mathrm{p}} \log(N_l^{\mathrm{p}}))$) is faster than the computation of one step of propagation (of complexity $O(N_z)$). Therefore, the propagator can be computed on the fly during the propagation if one wants to change the grid size, adding versatility to the method.

3.2.3 Propagation with the local propagators

In this section, the local propagation method using the previously defined set is introduced. The presentation firstly focuses on one non-zero wavelet coefficient. The generalization consists in adding the contributions of all non-zero coefficients to finally obtain the propagated wavelet coefficients $U_{x+\Delta x}$.

The computations details can be found in Algorithm 2. First, the non-zero coefficients $\alpha_l[p_z]$ of the wavelet decomposition of the field U_x are obtained with a FWT and compression (with threshold V_s). For each coefficient, we need to compute the corresponding propagator P_{l,p_t^l} in the set, from the level l and position p_z of $\alpha_l[p_z]$ following

$$p^l_{t_{p_z}} = p_z \mod 2^{L-l}. \tag{3.6}$$

This latter must be computed because we only store the elementary propagations in the set. Then, we compute the elementary propagation for this coefficient by multiplying the local propagator by $\alpha_l[p_z]$. Finally, all these elementary propagations are translated to original position p_z of the coefficient and summed as

$$U_{x+\Delta x} = \sum_{l,p_z} \alpha_l[p_z] P_{l,p^l_{t_{p_z}}}[\cdot + p_z] \tag{3.7}$$

to obtain the propagated wavelet coefficients. An inverse FWT finally gives the propagated field in free space.

The compressions on the signal V_s and on the propagators V_p error introduce an error that can be managed using formula (2.49).

Algorithm 2 Free-space propagation with local propagators

Inputs: field u_x, elementary propagator list $\left(P_{l,p_t^l}\right)_{l\in[0,L],p_t^l\in[0,2^{L-l}[}$

Output: propagated field $u^{fs}_{x+\Delta x}$ in free-space
$U_{x+\Delta x} \leftarrow$ empty list of wavelet coefficients (initialization)
$u_x \leftarrow$ field at p_x.
$U_x = CWu_x \leftarrow$ sparse wavelet representation of the field
for each non-zero coefficient $\alpha_l[p_z]$ of U_x **do**
  $l, p_z \leftarrow$ level and position of the coefficient
  $p^l_{t_{p_z}} \leftarrow$ choose the operator number using (3.6)
  $P_{l,p^l_{t_{p_z}}} \leftarrow$ corresponding elementary propagator
  $U_{x+\Delta x} \leftarrow U_{x+\Delta x} + \alpha_l[p_z] P_{l,p_{t_{p_z}}}$
end for
$u^{fs}_{x+\Delta x} \leftarrow \mathrm{IFWT}(U_{x+\Delta x})$

3.2.4 Complexity and memory requirements

For the complexity of the propagation part, the local method is driven by the same term as mSSW

$$O(N_x N_e), \tag{3.8}$$

with N_x the number of horizontal iterations and $N_e = N_s N_p$, where N_s and N_p correspond to the number of non-zero elements in the signal and propagator, respectively. With a high compression ratio, we have N_e of the same order as N_z and thus the lSSW method, as mSSW, is better than DSSF.

The main problem of mSSW was the propagator of size (N_z, N_z) computed with a complexity of order

$$O(LN_z \log(N_z) + 2^L N_z). \tag{3.9}$$

Therefore, the computation time and the memory size of the propagator were increasing rapidly with N_z, and the method could not be used for large 2D domains nor the 3D cases. Besides for large domain, since only few elements are stored, we also become faster because the time to access to any entries of P is reduced.

With the local propagation, only a limited number of propagators are stored. The whole set contains 2^L vectors of size at most N_p, with N_p the maximum number of non-zero wavelet coefficients of the propagator. This size is thus highly reduced since in practice $2^L \ll N_z$ and $N_p \ll N_z$.

Besides, in terms of computation time, the complexity is now driven by

$$O(\sum_{l\in[1,L]} N_l^{\mathrm{p}} \log(N_l^{\mathrm{p}}) + \sum_{l\in[1,L]} 2^{L-l} N_l^{\mathrm{p}}). \tag{3.10}$$

Therefore, the complexity is also reduced a lot since $N_l^{\mathrm{p}} \ll N_z$.

Using the same parameters as for the example in Section 3.2.2.b, we obtain $N_l^{\mathrm{p}} = 5l + 100 \times \sqrt{2}$ and

$$\text{Complexity} = O(\sum_{l\in[1,3]} N_l^{\mathrm{p}} \log(N_l^{\mathrm{p}}) + \sum_{l\in[1,3]} 2^{L-l} N_l^{\mathrm{p}}). \tag{3.11}$$

In this case the sum corresponds to

$$\begin{aligned} \sum_{l\in[1,3]} N_l^{\mathrm{p}} \log(N_l^{\mathrm{p}}) + \sum_{l\in[1,3]} 2^{L-l} N_l^{\mathrm{p}} &= 729 + 757 + 787 + 584 + 302 + 156, \\ &= 3315. \end{aligned} \tag{3.12}$$

The complexity is of order one step of computation $O(N_z)$ since in practice N_z is of order thousand of points. Therefore, the grid-size can be changed during the propagation for a cost of one propagation step.

In conclusion, the local version of SSW should overall be better than mSSW in the two requirements, *i.e.,* the memory size and the computation time of the propagator.

We now perform some numerical tests to assess that lSSW performs well in both criteria in comparison with mSSW.

3.3 Numerical tests

Numerical tests are performed. First, we show the interest of the reduced support. A realistic test case with long-range propagation is shown. For all experiments, the same wavelet parameters are used. The symlet6 family is used with the maximum level $L = 3$, as discussed in Chapter 2 and in [52, 57].

3.3.1 Interest of the reduced support

This test shows the improvement in terms of computation time using the wavelet supports for their propagations.

We compute the set of propagators with DSSF for several values of N_z. The propagators are computed on N_l^{p} points and compared to propagators obtained by using the total vertical domain N_z. We test this method for wavelets of each level $l \leq 3$. The steps Δx and Δz are 50 m and 0.5 m, respectively. In Figure 3.3, we plot the time to compute the set of propagators with respect to the number of vertical points N_z.

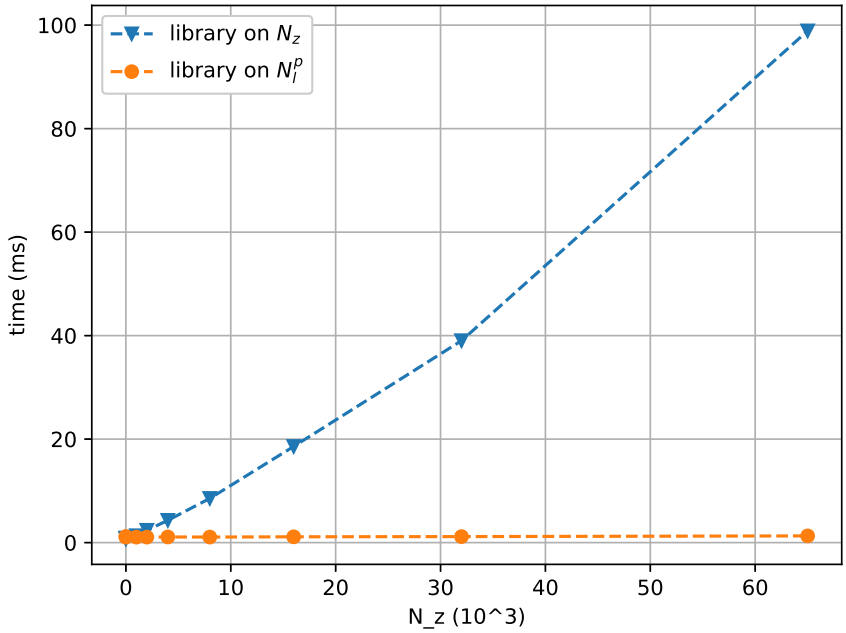


Figure 3.3: Computation time (ms) to obtain the set of propagators with DSSF on the total domain and on N_l^{p}.

Figure 3.3 shows that the time to generate the propagators on N_l^{p} is independent of N_z, as expected. Limiting the computation domain saves a lot of computation time for large N_z. Numerical tests show that, as expected, no error is added using N_l^{p} instead of N_z. This demonstrates the interest of considering the support after propagation on N_l^{p}. The following simulation illustrates that the local method with N_l^{p} does not add any error in a realistic test case.

3.3.2 Propagation in free space

With this test, we aim at validating lSSW in a free-space scenario.

The source is a uniform aperture at $f_0 = 300$ MHz of size $w_y = 10$ m and is placed at $x_s = 0$ and $z_s = 1024$ m in a domain of vertical size $z_{\max} = 2048$ m. The domain is of size $x_{\max} = 2000$ m in horizontal. The steps are $\Delta x = 20$ m and $\Delta z = 0.5$ m. Thus, we have $N_x = 100$. For the wavelet parameters, the symlet with $n_v = 6$ and a maximum level of $L = 3$ are chosen.

For a desired error at the final iteration of -30 dB the following thresholds are computed with (2.49)

$$v_s = 1.6 \times 10^{-4}\,, v_p = 1.6 \times 10^{-4}. \tag{3.13}$$

First, the field obtained with lSSW is given in Figure 3.4, with respect to distance and altitude.

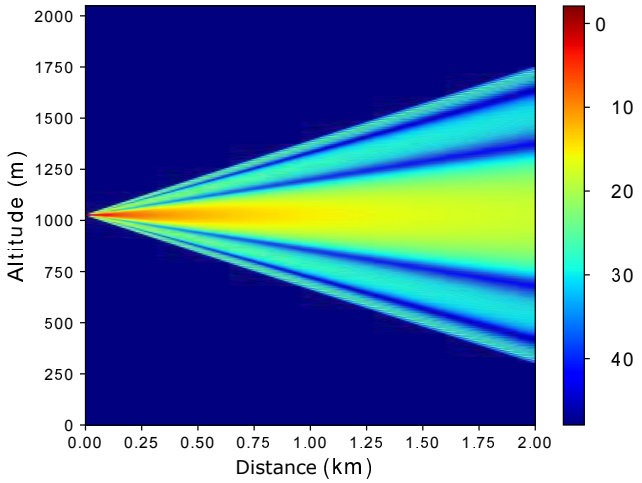


Figure 3.4: Field obtained with SSW.

Then, the RMS error is computed and given in Figure 3.5 in continuous lines. The analytic bound is plotted in dotted lines.

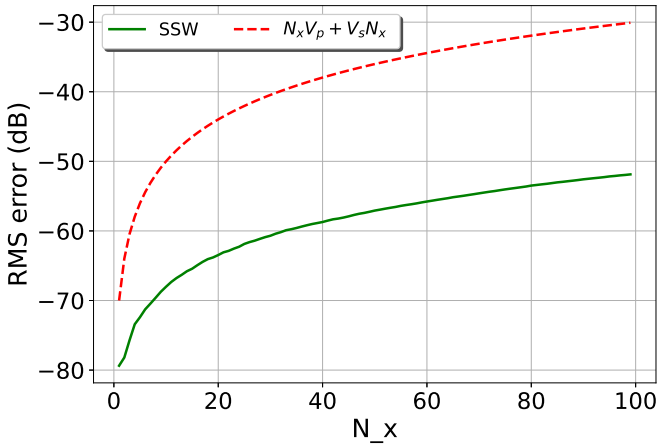


Figure 3.5: Evolution of the RMS error with N_x for the field radiated by a uniform aperture with $v_s = v_p = 1.6 \times 10^{-4}$.

As expected, Figure 3.5 shows that we obtain the same results as with mSSW. The thresholds computed allow to bound the error below the desired maximum. Thus, lSSW is validated for a simple case of free-space propagation.

3.3.3 Propagation with atmosphere and relief

This section aims at comparing lSSW and mSSW in terms of computation time, memory size, and accuracy in a realistic scenario.

The propagation of a complex source point (CSP) between Pau (43°17′51″N,0°22′07″E) and Toulouse (43°36′15″N, 1°26′36″E) is computed using both versions of SSW (lSSW and mSSW). An atmospheric duct is added. The relief between the two cities is taken into account.

The CSP parameters are: a frequency $f = 3$ GHz, with coordinates $x_s = x_{w0} + jk_0W_0^2/2$, with $x_{w0} = -50$ m and $W_0 = 1$ m and $z_s = 50$ m from the ground. The value of A in equation (1.45) is set to 1 dBV/m.

First, we consider a domain of size $x_{max} = 150$ km (distance between the two cities) and $z_{max} = 1024$ m. The grid size is $\Delta x = 200$ m in horizontal and $\Delta z = \lambda = 0.1$ m in vertical. The impedance ground is of parameters $\epsilon_r = 20.0$ and $\sigma = 0.02$ S/m.

We consider a refractive duct modelled by a trilinear refractive index [87]. The parameters are $M_0 = 330$ M-units, $z_b = 241$ m, $z_t = 391$ m, with gradients $c_0 = 0.118$ M-units/m and $c_2 = -0.5$ M-units/m.

The signal and propagator thresholds are $v_s = 2.1 \times 10^{-5}$ and $v_p = 2.1 \times 10^{-5}$, respectively, so as to obtain an error of -30 dB at the last iteration using the formula (2.49). The image layer for applying the local image method is of size $0.1 z_{max}$.

The electric field using lSSW is represented in Figure 3.6. The effects of both the atmosphere – bending of the wave – and the relief – interference and shadow zone – can be seen on that figure. For lSSW and mSSW, the RMS difference with DSSF at each iteration p_x is shown in Figure 3.7. We can see that the RMS error evolution is exactly the same with mSSW and lSSW, as expected.

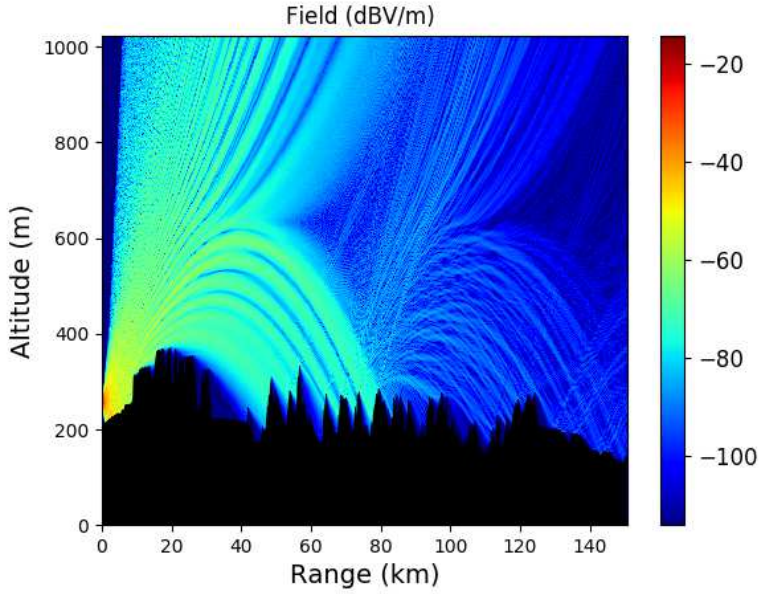


Figure 3.6: Electric field (dBV/m) in the vertical plane obtained by lSSW.

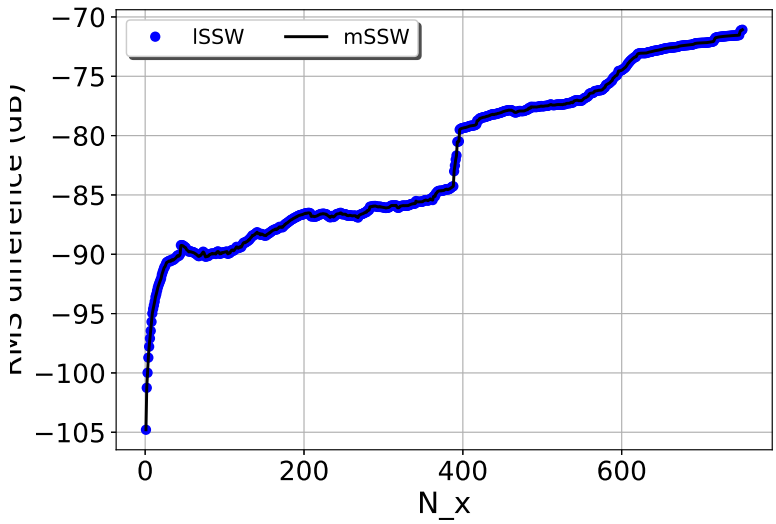


Figure 3.7: RMS difference of lSSW and mSSW with DSSF at each iteration.

Table 3.1 presents the times required to compute the propagator, one step of propagation for each method, the total time for the propagation, and the memory needed for mSSW and lSSW. The initialization time corresponds to the time to compute the set of propagators or the propagation matrix.

Table 3.1: Times and memory size needed with lSSW and mSSW.

method	mSSW	lSSW
Initialization time (s)	135	0.1
One step propagation (s)	0.01 to 0.3	0.01 to 0.4
Total time (s)	160	27.1
Propagator memory size	252 MB	117 kB

Results show that lSSW is better than mSSW in terms of both computation time and memory size. First, in terms of propagation lSSW is as fast as mSSW but the initialization time is much larger for mSSW: 1000 times larger. Thus lSSW allows being faster overall. Also, the time to compute the set of propagators (0.1 s) is of the order of one step of propagation and allows to update the propagators with acceptable effort if needed. In terms of memory, lSSW needs 2000 times less occupation in this scenario than mSSW.

Then, we double the number of vertical points ($z_{\max} = 2048$ m) to show further the interest of lSSW against mSSW. All other parameters remain the same. We obtain the results of Table 3.2 for the times and memory size.

As expected, the time to initialize and the memory size of the propagator remain the same with N_z increasing. Regarding the time and memory size, lSSW becomes very attractive in this case, since both parameters have significantly increased with mSSW. Finally, the test shows that the error with DSSF remains the same at the last iteration. Thus, the new version of SSW allows computing fields for large scenarios with few memory requirements. In the next section, lSSW and mSSW are compared with a scenario where the memory

Table 3.2: Times and memory size needed with $z_{\max} = 2048$ m with lSSW and mSSW.

method	mSSW	lSSW
Initialization time (s)	285	0.1
One step propagation (s)	0.02 to 0.4	0.02 to 0.5
Total time (s)	339	56.1
Propagator memory size	505 MB	117 kB

size and the computation time of the propagator are the main concerns.

3.4 Application to radio-occultation

Radio occultation (RO) consists in inferring atmospheric data from the radio frequency (RF) link between two satellites in the limit of line of sight. This is generally used with the RF link between GPS and LEO satellites [16, 116, 117]. In this configuration, the propagation is altered by the refractive condition (troposphere and ionosphere) of the Earth. This altered data carry information about the atmospheric conditions. Since 1995 inverse RO techniques have been used to estimate the physical properties of the Earth atmosphere using inversion techniques [118, 119]. The problem is here simplified to a static 2D RO scenario.

RO inversion methods need to be validated. To do so, a direct model that accurately computes the field for a given atmosphere condition are necessary. This model should run in a reasonable time. Indeed in a large domain, as the one considered here, the computation time required with SSF becomes prohibitive. Other methods have been proposed to eliminate this constraint. Using Gaussian beams as an asymptotic solution of the PE, L'Hour *et al.* [120] have proposed a fast method for field propagation in large domain and applied it to RO [33, 121]. mSSW has also been used as a direct method for RO scenarios [20] but both the time and memory size needed by the matrix become prohibitive to reduce the vertical step to be more accurate. In this section, we use lSSW as the direct method to obtain the field on the overall domain of a RO scenario. With its low requirements in time and memory size for the propagator, lSSW decreases the computation time and allows to compute the propagation with thinner grid size, compatible with an operational use.

The Cartesian system with (x, z) coordinates is used. In this scenario, we consider a GPS satellite at frequency $f_0 = 1.575$ GHz with a gain of 16 dBi and power of 25 W. The receiver is a LEO satellite. The total distance between both satellites is $x_{\max} = 29200$ km, corresponding to the x-axis coordinate of the LEO satellite. The goal is to obtain the field near the LEO satellite.

No ionosphere is included in the simulation. This could be added using ITU-R P531. The refractive index in the troposphere is computed using ITU recommendations [21]. More realistic models could use the Weather Research and Forecasting model (WRF) [23, 24] or the AROME model from Météo-France [22]. Note that N is used instead of M since Earth is treated as relief with a staircase model because of the size of the domain.

The configuration of the computation domain is shown in Figure 3.8. Axis $z = 0$ is placed

at the center of this window. Moreover, the first vertical is set at $x_0 = 24986$ km from the GPS satellite. Thus, the horizontal size of propagation is $x_{\max} - x_0 = 4214$ km. The vertical size is chosen to $65,536$ km. The steps are set to $\Delta x = 1$ km and $\Delta z = 1$ m $\approx 5\lambda$. The propagation is thus computed on $N_x = 4214$ points and $N_z = 65.536$ points. The Earth shape is considered as a sphere. The Earth ground is considered as a PEC and a local image layer of size $0.1N_z$ is added. Before reaching the Earth apodizations layers are added at the top and the bottom of the domain. When propagating in the Earth radius, only an apodization window at the top is considered, since the Earth is treated with the local image theorem and the staircase model. Finally, after leaving the earth radius apodization layers of size N_z are added at the top and the bottom of the domain. For the source, a windowed plane wave is used.

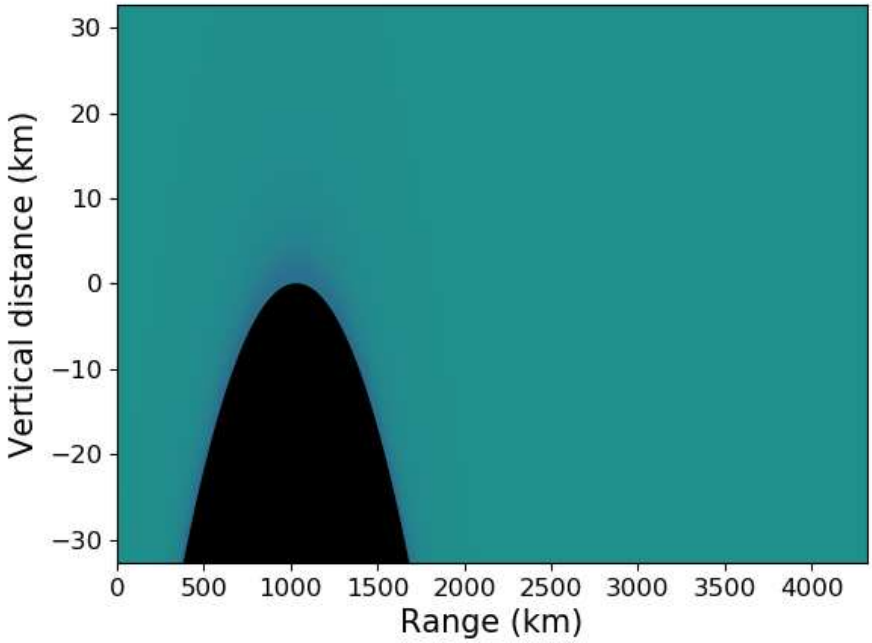


Figure 3.8: Configuration of the computation domain for the RO scenario.

The refractivity N is represented in Figure 3.9. Earth is represented in black.

The compression thresholds are set to obtain a negligible final compression error of -30 dB.

The electrical field computed with lSSW is represented in Figure 3.10. The two vertical dotted lines show the limit of the propagation over the Earth. We can see the effect of the Earth with the interference pattern of the field and the shadow zone.

The results in terms of time and memory size with lSSW and mSSW are shown in Table 3.3. First, with lSSW we have a better computation time and memory size than with mSSW. In particular, lSSW initialization time is 50000 less than the one with mSSW, but the propagation time is a bit slower. Overall lSSW is faster than mSSW. For the memory requirement concerns, the impact of the local method becomes clear since lSSW is better of a factor of 25000 against mSSW. Thus with lSSW a smaller grid size is possible, whereas reducing by two the grid size is not even possible with mSSW with a desktop computer.

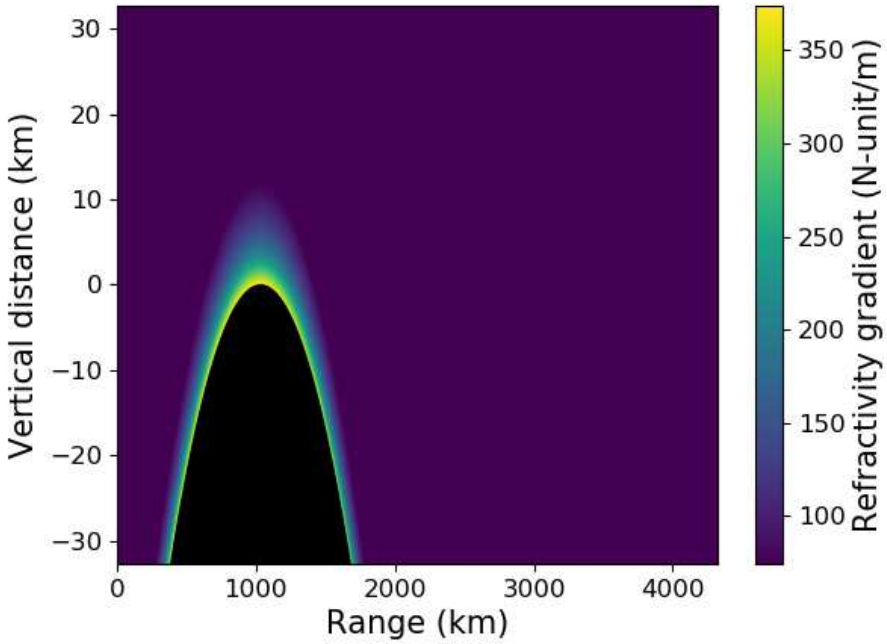


Figure 3.9: N index along the propagation scene.

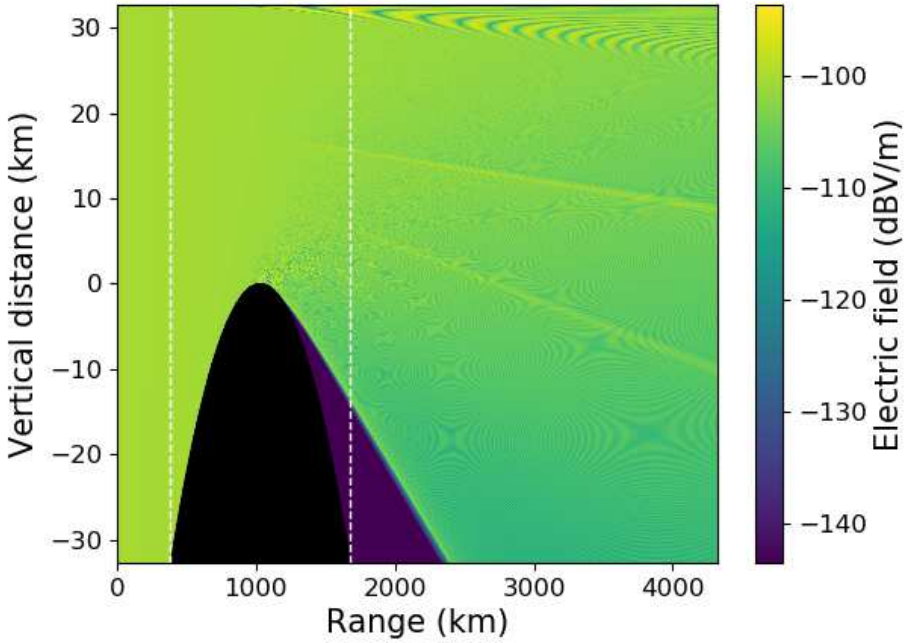


Figure 3.10: Electrical field (dBV/m) obtained with lSSW in scenario 4.

Table 3.3: Times and memory size needed with lSSW and mSSW.

method	mSSW	lSSW
Initialization time (s)	585	0.017
Propagation time (s)	374	466
Total time (s)	959	466
Propagator memory size	759 MB	42 kB

Also, since the time to generate the propagators is negligible to the propagation, horizontal steps can be changed between vacuum and ground propagations if needed. Therefore one can reduce the grid size when the propagation reaches the Earth radius to be more accurate in zone where there are faster variations of the field. Thus, the local method adds versatility.

To conclude, lSSW is a good method to model the field with a RO scenario, since the computation time and memory requirements are low in comparison to mSSW.

3.5 Conclusion

In this chapter, the local method for SSW, namely lSSW, has been introduced. It is based on an efficient strategy to compute the propagators using the support of the wavelet for the propagation, and the storage of only the essential information.

First, an overview of the SSW method has been given. We have reminded that the wavelet coefficients are propagated using a pre-computed propagator. The propagation can be either a matrix-vector product or a sum of local propagations. The other steps (relief and atmosphere are taken into account in the space domain) are independent of the choice of the propagation step. Then, the method to compute the local propagators has been comprehensively detailed. Based on the translation property of the wavelets, few propagators are computed and stored. Using the wavelet supports as vertical domains for the propagation of these wavelets allows to be faster and to use less memory. Thus, the propagators are generated independently of the vertical size of the domain for given horizontal and vertical steps. The propagation is then performed by summing all the local propagations associated with the wavelet coefficients of the field.

Numerical tests have shown that lSSW does not add any additional error compared to its matrix counterpart, mSSW. In terms of computation time, the time to propagate the wavelet remains constant regardless of the domain size as expected. For complex environment simulations and large domain scenarios, the same electric fields as with mSSW are obtained. Nevertheless, the memory size is significantly better with lSSW and allows to compute propagations in larger domains. The computation time needed for initialization is also better. Finally, the time to compute the propagators is short compared to one step of propagation adding versatility to the method.

An application to a 2D radio-occultation scenario has been performed. We have shown that lSSW allow computing the forward propagation in this case in a short time and with a limited memory size. lSSW could thus be used as the forward model for inversion in radio-occultation.

This new version of SSW allows tackling the problem of memory size and computation time for the propagator in mSSW. Since the computation time for the propagators is short compared to the propagation time, adaptative steps can be introduced with this method. Besides the improvements obtained in the memory and time requirements are an essential step towards an efficient 3D version of SSW, where local propagators gain an additional dimension.

Chapter 4

3D SSW Using 2D Separable Wavelets

4.1 Introduction

In Chapter 1, the 3D Cartesian DSSF method has been introduced. Nevertheless, we have shown that the computation time and the memory size required by the propagator are a major burden for the 3D case.

In Chapter 2, the 1D multilevel discrete wavelet transform has been presented to introduce the 2D split-step wavelet method. In Chapter 3, a new version of the split-step wavelet method that performs better in terms of memory occupation has been proposed.

This chapter introduces the 2D multilevel discrete wavelet transform based on a separable wavelet basis. The 2D decomposition is then used for the 3D SSW method. Indeed the low complexity of 2D FWT and its ability for sparse representations of fields are used to obtain a propagation method fast and of low memory requirement. These considerations are crucial in 3D electromagnetic wave modeling.

This chapter is organized as follows. In Section 4.2, we introduce the 2D multilevel discrete wavelet transform. First, the construction of the separable wavelet basis is presented. Then, the discrete wavelet transform and several notations are introduced. Finally, the sparse representation provided by the wavelets and the compression are explained. In Section 4.3 the principal characteristics of the wavelets are recalled. Then, several experiments are performed to choose the wavelet family, the number of vanishing moments, and the maximum level of decomposition. Lastly, the fast Fourier transform and the fast wavelet transform applied on 2D signals are compared in terms of computation time. In Section 4.4, we generalize SSW to 3D. First, the matrix version of SSW is described and we show why this straightforward extension is not relevant. Then, to overcome this problem the local version of SSW is generalized to 3D. Section 4.5 concludes the chapter.

4.2 The 2D wavelet transform

A straightforward approach to perform a 2D wavelet transform would be, as for the Fourier transform, to do one 1D wavelet transform in each direction. Nevertheless, the dilations and information along both directions would be mixed and invariance properties would be lost. In general this approach is avoided and a 2D wavelet transform with a separable basis is preferred [94].

This section introduces the 2D multilevel discrete wavelet transform based on the separable wavelet basis, *i.e.*, for which the dilations are performed simultaneously on the 2 directions.

As for the 1D case, we first introduce the construction of the 2D wavelet basis. Then, the discrete wavelet transform and its fast algorithm are presented. Finally, compression with a hard threshold is introduced.

4.2.1 2D wavelet family

For the 2D case, we use the separable family of wavelets [94]. In few words, a separable wavelet basis corresponds to dilations and translations of four combinations of wavelet and scaling function applied along both axes. In 1D the basis is composed of one dilated and translated wavelet and a translated scaling function. In 2D, three dilated and translated wavelets and one translated scaling function are needed.

First, the construction of this kind of wavelet family is introduced. A separable wavelet family is defined by three mother wavelets

$$\psi^h(y,z) = \phi(y)\psi(z)\,,\ \psi^v(y,z) = \psi(y)\phi(z)\ \text{ and }\ \psi^d(y,z) = \psi(y)\psi(z), \tag{4.1}$$

where ψ and ϕ correspond to the 1D mother wavelet and scaling function as defined in Chapter 2. Since these functions are constructed from 1D wavelets, they are of zero mean and normalized for the 2-norm, *i.e.*, $\|\cdot\|_2 = 1$. As seen in 1D, see Section 2.2, the scaling function corresponds to a low-pass filter, whereas the mother wavelet corresponds to a high-pass filter. Therefore, the first wavelet corresponds to a low-pass filter in y-direction and a high-pass filter in z direction, thus it describes well the horizontal elements of a signal. The same argumentation shows that the two others describe the vertical and the diagonal elements, respectively. Thus, the upper script h, v, and d refer to the preferred direction of each wavelet: horizontal, vertical, and diagonal, respectively.

By dilating of l and translating of t_y, t_z these three wavelets, a wavelet family $\mathcal{F}$ is obtained

$$\mathcal{F} = \left\{ \frac{1}{2^l}\psi^h\left(\frac{y-2^l t_y}{2^l}, \frac{z-2^l t_z}{2^l}\right), \frac{1}{2^l}\psi^v\left(\frac{y-2^l t_y}{2^l}, \frac{z-2^l t_z}{2^l}\right), \right. \\ \left. \frac{1}{2^l}\psi^d\left(\frac{y-2^l t_y}{2^l}, \frac{z-2^l t_z}{2^l}\right) \right\}_{l\in\mathbb{N},(t_y,t_z)\in\mathbb{Z}^2}. \tag{4.2}$$

The parameter l corresponds to the dilation level. Smaller values of l cover lower parts of the spectrum. The parameters t_y and t_z correspond to the translation in y and z axes, respectively. They allow to cover the spatial domain. Furthermore, $1/2^l$ corresponds to

the scaling factor introduced to ensure $\|\cdot\|_2 = 1$. Since the wavelets are separable, the variation in each directions y or z can be studied independently.

This family corresponds to a basis of $\mathcal{L}^2(\mathbb{R}^2)$ [94]. We denote each element of the family by

$$\psi^o_{l,p}(y,z) = \frac{1}{2^l}\psi^o\left(\frac{y-2^l t_y}{2^l}, \frac{z-2^l t_z}{2^l}\right), \tag{4.3}$$

with $p = (t_y, t_z)$, the translation parameter, and $o \in \{h, v, d\}$, the orientation parameter.

In practice, the level of decomposition is limited to a maximum, denoted by L. Thus, the family $\mathcal{F}$ is not a basis anymore. To retrieve a basis, we add the scaling function

$$\phi_{L,p}(y,z) = \frac{1}{2^L}\phi\left(\frac{y-2^L t_y}{2^L}\right)\phi\left(\frac{z-2^L t_z}{2^L}\right). \tag{4.4}$$

This last function corresponds to the product of the 1D scaling function by itself and is of non-zero mean. Thus, this corresponds to a low-pass filter in both directions. Therefore, the scaling function covers the lowest part of the spectrum until the continuous part.

The basis $\mathcal{B}$ of $\mathcal{L}^2(\mathbb{R}^2)$ is finally obtained

$$\mathcal{B} = \{\phi_{L,p}(y,z), \psi^o_{l,p}(y,z)\}_{l\in[1,L], p\in\mathbb{Z}^2, o\in\{h,v,d\}}. \tag{4.5}$$

We present these three different mother wavelets and their respective spectral coverage (Fourier transform of each wavelet) in Figure 4.1. Figures (a), (c), and (e) correspond to the three wavelets in the spatial domain, while, Figures (b), (d), and (f) correspond to the Fourier transform of each wavelet.

On Figure 4.1 (b), for the horizontal wavelet, as expected, the spectral coverage corresponds to the one of a low-pass filter in y direction and a high pass in z direction. Therefore, this wavelet returns the horizontal detail of a signal. The same conclusion can be made for the two other wavelets, which are covering the vertical and diagonal direction respectively. Also, dilating these three wavelets results in covering the lower part of the spectral. We can also note that all the wavelets are of zero mean, as expected.

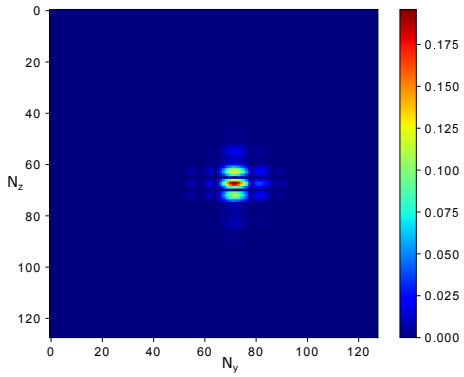
Also, we can note that the horizontal and vertical wavelets are obtained with a rotation of 90° from each other if the same grid size is applied in y and z directions.

Example of a scaling function and its Fourier transform are shown in Figure 4.2. In Figure (a) the scaling function in the spatial domain is presented, while its spectral transform is given in (b).

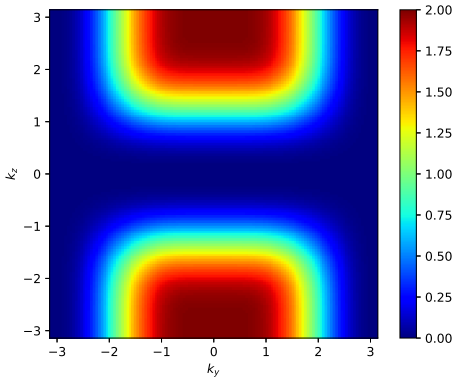
First, Figure 4.2 (b) shows that the scaling function is of non-zero mean. Second, the scaling function is a low-pass filter in both directions allowing to cover the lowest part of the spectrum, as expected from its construction. Finally, by summing the spectral coverages of the wavelets and the scaling function, one can see that the entire Fourier domain is described.

4.2.2 2D discrete wavelet transform

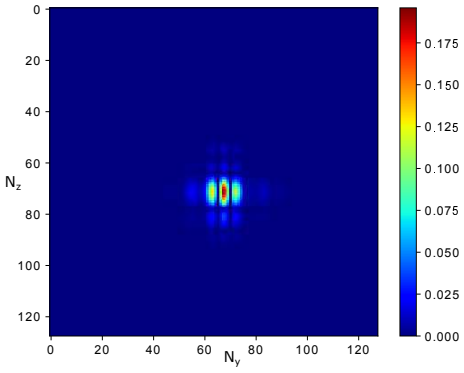
The decomposition In this section, we study a discrete signal $u[p_y, p_z]$ with $p_y, p_z \in \mathbb{Z}^2$. For any discrete domain the previously defined basis $\mathcal{B}$ is still a basis.



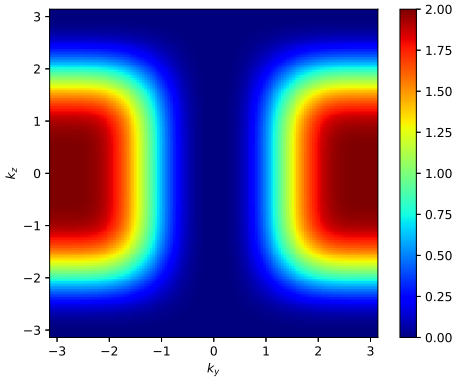
(a) Hozitontal wavelet in the spatial domain.



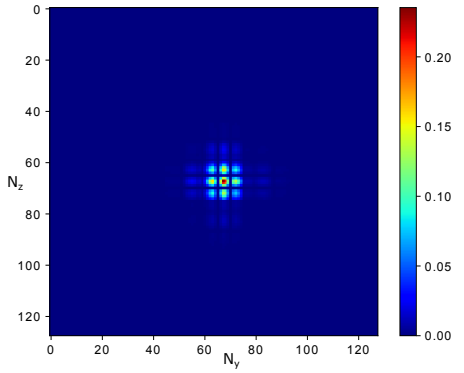
(b) Spectral representation of the horizontal wavelet.



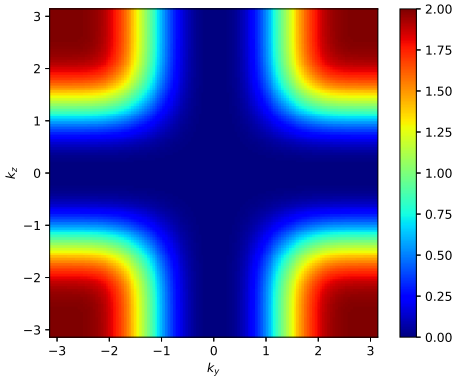
(c) Vertical wavelet in the spatial domain.



(d) Spectral representation of the vertical wavelet.

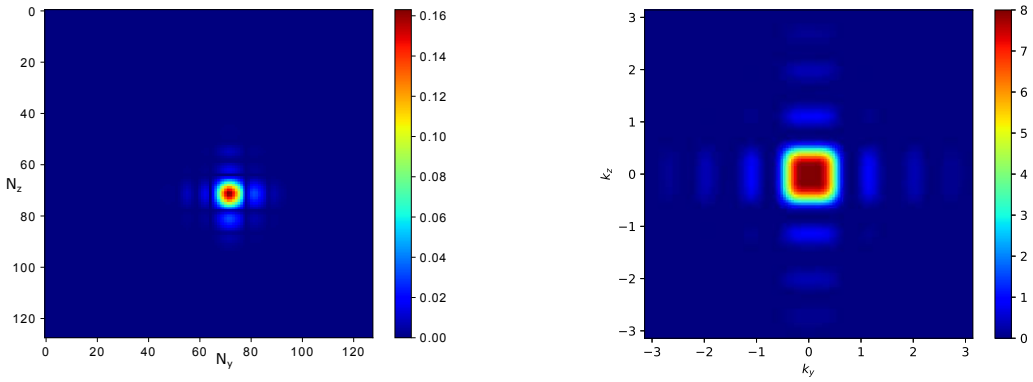


(e) Diagonal wavelet in the spatial domain.



(f) Spectral representation of the diagonal wavelet.

Figure 4.1: Spatial and spectral representation of the three mother wavelets (Symlet family of order 6)



(a) Spatial domain.

(b) Spectral domain.

Figure 4.2: Scaling function.

The discrete wavelet transform in 2D is defined as an application that, for a discrete signal u, returns a list of coefficients $a_{L,p}$ and $d^o_{l,p}$ such that

$$u[p_y, p_z] = \sum_{p\in\mathbb{Z}^2} a_{L,p}\phi_{L,p}[p_y, p_z] + \sum_{o\in\{h,v,d\}} \sum_{l=1}^{L} \sum_{p\in\mathbb{Z}^2} d^o_{l,p}\psi^o_{l,p}[p_y, p_z]. \tag{4.6}$$

In this definition, the coefficients $a_{L,p}$ are called the approximation coefficients. They can be obtained by the inner product between the signal and the scaling function. These coefficients describe the slowest variations of the signal. The coefficients $d^o_{l,p}$ are called the details coefficients and can be obtained with inner products between the signal and the wavelets. They return faster variations of the signal while l decreases. Thus details of the signals are obtained. Therefore, a multiresolution decomposition of the signal is obtained. The associated operator is denoted by $\mathbf{W}$.

In practice, the computation is performed in a finite domain of size (N_y, N_z). Therefore, the translation coefficients t_y and t_z are limited to $[0, N_y/2^l[$ and $[0, N_z/2^l[$ for $l \in [1, L]$. For the scaling function they are limited to $(t_y, t_z) \in [0, N_y/2^L[\times[0, N_z/2^L[$. Thus, boundary conditions need to be applied to the discrete wavelet transform. To keep an orthonormal basis, periodic boundary conditions are used.

From now on, for conciseness, the wavelet coefficients are denoted by

$$U^o_l[t_y, t_z] = \begin{cases} a_{L,p}, & \text{for } l = 0 \text{ and } \forall (t_y, t_z) \in [0, N_y/2^L[\times[0, N_z/2^L[\\ d^o_{l,p}, & \text{for } l \in [1, L] \text{ and } o \in \{h, v, d\}, \forall (t_y, t_z) \in [0, N_y/2^l[\times[0, N_z/2^l[. \end{cases} \tag{4.7}$$

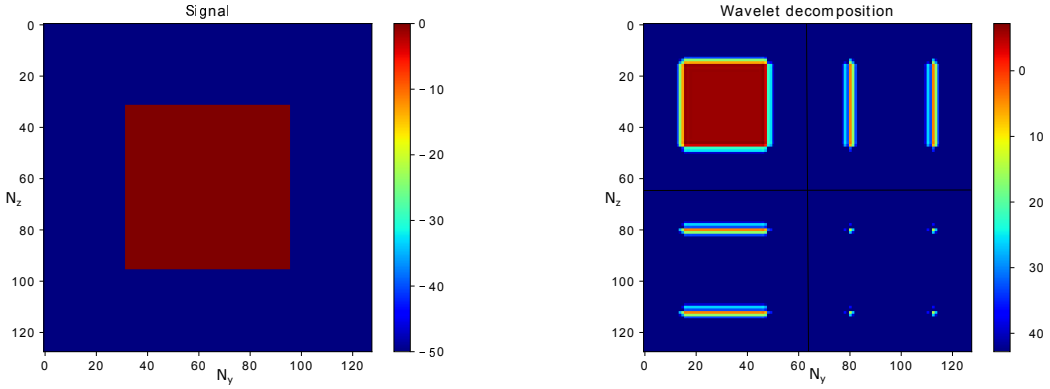
Therefore, $\forall (p_y, p_z) \in [0, N_y] \times [0, N_z]$, the discrete wavelet decomposition is synthetically expressed as

$$u[p_y, p_z] = \sum_{l=0}^{L\prime} \sum_{o\in\{h,v,d\}} \sum_{(t_y,t_z)\in\mathbb{Z}^2} U^o_l[t_y, t_z]\chi^o_l[p_y - t_y, p_z - t_z], \tag{4.8}$$

with $\sum'$ corresponding to a sum not applied to $l = 0$ and

$$\chi^o_l[p_y - t_y, p_z - t_z] = \begin{cases} \phi_{L,p}[p_y, p_z], & \text{for } l = 0 \\ \psi^o_{l,p}[p_y, p_z], & \text{for } l \in [1, L] \text{ and } o \in \{h, v, d\}. \end{cases} \tag{4.9}$$

To further show what information contains each wavelet coefficient, the decomposition of a square is studied. The signal is presented in Figure 4.3 (a). Only a one-level wavelet decomposition of the square signal is performed and the result is given in Figure 4.3 (b). At the top right corner, the scaling function coefficients are plotted. The horizontal, vertical, and diagonal coefficients are presented at the bottom left, top right and bottom right, respectively. Note that each sub-figure containing the different wavelet coefficients is half the size of the input image on both dimensions.



(a) Square signal (dB).

(b) Square signal decomposition.

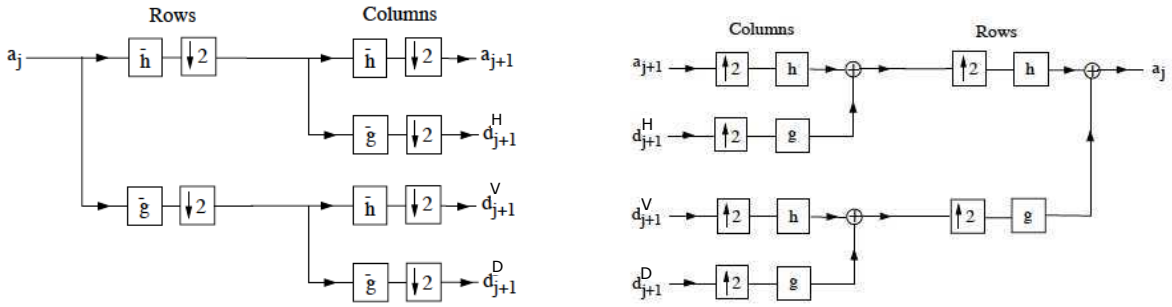
Figure 4.3: Example of a one level discrete wavelet decomposition with a square signal.

Figure 4.3 (b) shows that the approximation coefficients return the signal with a lower resolution: the steep gradient on the edge of the square is not captured by the scaling function. When steep horizontal variations occur, such as the horizontal square ridge, then the wavelet coefficients are maximum, whereas for any other variations they are close to 0. For the vertical and diagonal wavelets, the same conclusions can be made: vertical and diagonal coefficients are associated with the vertical ridge and each vertex of the square, respectively.

The fast wavelet transform As in 1D, the inner products are not computed and the FWT [94] is used to obtain the coefficients in a short time. The method is the same as in 1D, see Chapter 2 and Figures 4.4 (a) and (b) for the scheme of the direct and inverse FWT, respectively. The signal is decomposed through a series of mirror filters along y and z and decimations of a factor 2. The recomposition is computed by interpolating the signal and convolving the signal with mirror filters at each step. The complexity of the FWT is proven to be driven by $O(N_y N_z)$ [94]. Thus, a first advantage of the wavelet transform is that the FWT is faster than the 2D FFT, which is of complexity $O(N_y N_z(\log(N_y) + \log(N_z)))$. The second advantage that explains the popularity of the method is the sparse representation capacity of the wavelets.

4.2.3 Compression

As said in Section 2.4.1, for a regular signal the wavelet coefficients are rapidly decreasing to 0. Therefore, one of the main applications of the wavelet transform is data compression, such as with the JPEG2000 image format.



(a) FWT. (b) Inverse FWT.

Figure 4.4: Fast wavelet transform [94].

The sparse representation is achieved by forcing to 0 coefficients that are close to 0. In practice, a hard threshold is applied and the elements of the compressed representation are defined by

$$\tilde{U}_l^o[t_y, t_z] = \begin{cases} 0 & \text{for} \quad |U_l^o[t_y, t_z]| \leq V_s, \\ U_l^o[t_y, t_z] & \text{otherwise}, \end{cases} \tag{4.10}$$

with V_s a chosen threshold. This correspond to a compression, which is denoted by operator $\mathbf{C}$.

In general, if the threshold is well chosen, the number of non-zero coefficients N_s of $\tilde{U}$ is much smaller than the original size $N_y N_z$ of the signal U, *i.e.*, $N_s \ll N_y N_z$. Thus a sparse representation is obtained. The accuracy is directly related to V_s, as shown in Section 2.4. In Figure 4.5, the decomposition of the square signal has been compressed with $V_s = 10^{-3}$. The null coefficients are represented in white.

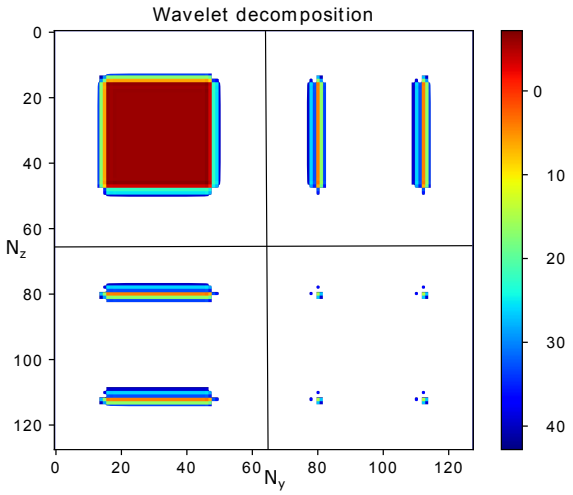


Figure 4.5: Compressed one-level wavelet decomposition of the square signal.

The RMS error between the reconstruction of the signal from the sparse set of wavelet coefficients and the initial signal is below -60 dB $= 20\log(V_s)$, as expected. Therefore, only a few coefficients are needed to store the signal while keeping good accuracy.

In the next section, the choice of the parameters to obtain a good sparse representation while keeping a good accuracy for electromagnetic signals is discussed.

4.3 Choice of the wavelet parameters

In this section, we remind the main parameters of the wavelets and their different effects on a signal decomposition. Then, some numerical tests are performed to select a set of parameters suitable for the electromagnetic signals we deal with, as performed in Section 2.2.5 for the 2D SSW.

4.3.1 Wavelet characteristics

Firstly, the compression ratio (CR) is defined as in 1D by equation (2.13). We want to choose the parameters that allow a high CR, while keeping a good accuracy, defined with the RMS error.

Secondly, the parameters we can set are:

- The wavelet family, which for discrete wavelets is either the Daubechies, the symlets, or the coiflets.
- The number of vanishing moments n_v of the mother wavelet, defined in Section 2.2.5. As mentioned in Chapter 2, for a smooth signal like an electromagnetic field, the wavelet coefficients are rapidly decreasing to 0 with n_v increasing. Therefore, a high n_v allows representing a smooth signal with few coefficients. Thus, a high compression ratio is achieved.
- The support of the wavelet N_l. Smaller supports allow to take into account steep variations of a signal, with fewer coefficients. Nevertheless, the size of the mirror filters associated to the family depends on the support. Therefore a wider support increases the computation time of the FWT.
- The symmetry or quasi-symmetry of the wavelet family is also important for us since the free-space propagation preserves the symmetry with z. Intuitively quasi-symmetric wavelets might be better to model a symmetric phenomenon.
- The maximum level of the wavelet decomposition is also important because of its impact on the computation time and CR.

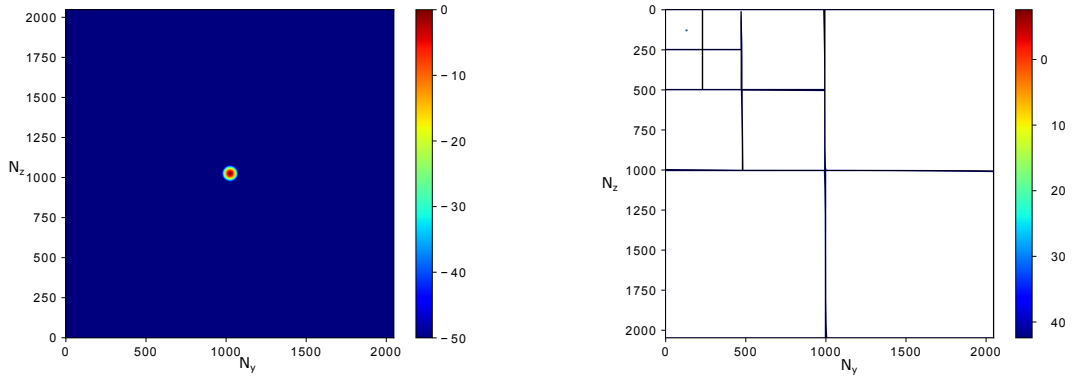
Moreover, N_l and n_v are linked together, since the support is at least of size $2n_v - 1$. Therefore, both can not be optimized at the same time and a trade-off must be made. Since the physical signals we deal with are regular, we show next that a high n_v is preferable.

4.3.2 Numerical tests

Some numerical tests for the different parameters with different electromagnetic signals are now performed to obtain a good set. In all tests, we decompose a signal, either the field of a CSP or radiated by a uniform aperture, we apply compression with a threshold V_s, and finally, compute the RMS error between the recomposed signal and the input signal. The CR is also computed.

The frequency is $f = 300$ MHz. The domain is of size $(2048, 2048)$ m. Two different sources are considered:

- a CSP is placed at $x_s = -500$ m and $y_s = z_s = 1024$ m with a waist size $W_0 = 20$ m. This first field is used because of its smooth variations. Steps of $\Delta y = \Delta z = 1$ m are used, leading to a total size of $N_y \times N_z = 4.19 \times 10^6$.
- a uniform aperture, the aperture is placed $x_s = -500$ m. The vertical and horizontal positions of the source are chosen to $y_s = z_s = 1024$ and the width is $w_y = w_z = 5$ m. This second field is used because of its steeper variations. Steps of $\Delta y = \Delta z = 0.5$ m are used, leading to a totalsize of $N_y \times N_z = 1.69 \times 10^7$.



(a) (b)

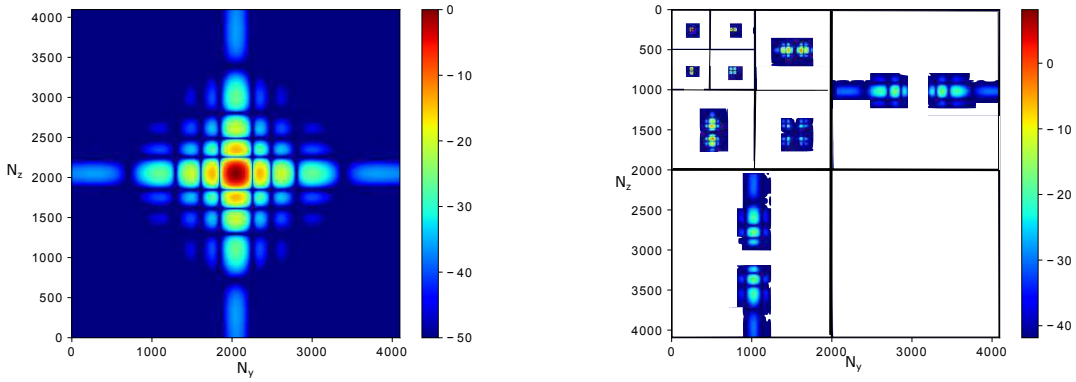
Figure 4.6: Field (dB) radiated from a CSP (a) and its wavelet decomposition (b).

The field radiated by the CSP and its wavelet decomposition (with symlet family, $n_v = 5$, $L = 3$ and $v_s = 10^{-3}$) are plotted in Figure 4.6 (a) and (b). At the top left the scaling function coefficients are presented. Then the same order for each orientation as for Figure 4.3, is used and repeated for each level. For $l \in [1, L]$, the size of each sub-figure containing the wavelet coefficients are $N_z/2^l$. As expected, the scaling functions return the signal with a lower resolution. Then the wavelet coefficients return the details in each direction. Nevertheless, since the signal is smooth, very few coefficients are needed ($CR = 99\%$).

Then, for the aperture, the field and its wavelet decomposition are presented in Figure 4.7. Here, since the signal has steep variations, more coefficients are required ($CR = 75\%$). As expected, the approximation coefficients correspond to the main lobe of the field. Also, the horizontal and vertical sidelobes are only visible in the associated wavelet coefficients showing the effect of both the horizontal and vertical wavelets. Furthermore, diagonal coefficients correspond to the diagonal variations of the field. Finally, we can see that with the level l decreasing, faster variations of the signals are obtained, corresponding to lower sidelobes.

4.3.2.a Test on the wavelet types

For the discrete wavelet transform, we need orthogonal, and discrete wavelet families of finite support. This reduces the choice to three common wavelet families that are the



(a) (b)

Figure 4.7: Field (dB) radiated from a uniform aperture (a) and its wavelet decomposition (b).

Daubechies, the symlets, and the coiflets [94]. These families have been described in Chapter 2. Thus, we just briefly remind the main features of each one. The Daubechies family has the support of minimum size for a given number of vanishing moments but is asymmetric. The symlets are a modified version of the Daubechies family, which are quasi-symmetric. Finally, the coiflets are a near symmetric wavelet family but their support is not of minimum size. We now perform tests to choose among one of these families, with $n_v = 5$, $L = 3$ and $v_s = 10^{-3}$.

First, we test with the radiation of a CSP, which has smooth variations. The compression results are summarized in Table 4.1.

family	Daubechies	symlets	coiflets
non-zero coefficients	424	429	232
CR (%)	99.9	99.9	99.9
RMSE (dB)	-77.3	-77.3	-76.6

Table 4.1: Compression of the field radiated from a CSP with $V_s = 10^{-3}$ for different families.

Since the decomposed signal is very smooth, only few coefficients are needed. We also see that for the three families, results are similar in terms of CR and RMSE. Note that, since the field radiated by a CSP is symmetric, fewer coefficient are needed with the coiflets that are quasi-symmetric. Nevertheless, since the CR is the same with the three families, we can choose the one with the best RMSE. Thus, the symlets seem a good choice.

Second, we test with the field radiated from a uniform aperture, which variations are steeper. The results are summarized in Table 4.2.

In this case, since the field has steeper variations, more coefficients are needed to characterize the signal. Since the field is symmetric, the coiflets, which is a near symmetric family are good ones.

We can also see that the results are almost the same with the three families in both cases, showing that in our case the choice of the family is not the main parameter.

family	Daubechies	symlets	coiflets
non-zero coefficients ($\times 10^6$)	4.24	4.24	3.69
CR (%)	75	75	78
RMSE (dB)	-62.8	-62.8	-63

Table 4.2: Compression of the field radiated from a uniform aperture with $V_s = 10^{-3}$ for different families.

The numerical tests show that for a field with smooth variations the symlet family seems a good choice, whereas the coiflets are better for steeper variations. Nevertheless, since the support of the coiflets is not of minimum size, the FWT computation time would be higher. Therefore, the symlets are used from now on. We now perform tests on the vanishing moments.

4.3.2.b Test on the vanishing moments

In this test, the number of vanishing moments n_v varies from 2 to 8. The family is the symlets with a maximum level of decomposition of $L = 3$. Firstly, a test with the field radiated by the CSP is performed. The compression results are summarized in Table 4.3.

n_v	2	3	4	5	6	7	8
non-zero coefficients	2637	923	509	429	383	359	307
CR (%)	99.9	99.9	99.9	99.9	99.9	99.9	99.9
RMSE (dB)	-66	-70.8	-70.8	-77.3	-79.1	-78.1	-77.1

Table 4.3: Compression of the field radiated from a CSP with $V_s = 10^{-3}$ with respect to the number of vanishing moments.

Notice that since the field is regular, as expected from formula (2.33) in Section 2.4 the CR is increasing with n_v. Nevertheless, after $n_v = 6$ the RMSE decreases, thus this latter seems to be a suitable choice for n_v.

Secondly, a test with a field radiated by a uniform aperture is performed. The results in terms of CR and RMS error are summarized in Table 4.4.

n_v	2	3	4	5	6	7	8
non-zero coefficients ($\times 10^6$)	5.32	4.78	4.47	4.24	4.09	3.97	3.88
CR (%)	68.2	71.6	73.3	74.7	75.6	76.3	76.9
RMSE (dB)	-62.4	-61.9	-62.7	-62.8	-62.9	-62.9	-62.9

Table 4.4: Compression of the field radiated from a uniform aperture with $V_s = 10^{-3}$ with respect to the number of vanishing moments.

Since the field is still regular, the CR is also increasing with n_v. Then, $n_v = 6$ seems also a good choice with a field with steeper variations. Therefore, from now on we use the symlet with $n_v = 6$. Now, tests to choose the maximum level of decomposition are performed.

4.3.2.c Test on the maximum level of decomposition

In this section, the tests are performed with the symlets with $n_v = 6$ for L varying from 2 to 5.

First, a test with the field radiated by a CSP is performed. The CR and RMSE for the different maximum levels of decomposition are summarized in Table 4.5.

L	2	3	4	5
non-zero coefficients	752	383	439	570
CR (%)	99.9	99.9	99.9	99.9
RMSE (dB)	-78.4	-79.1	-77.3	-77.6

Table 4.5: Compression of the field radiated from a CSP with $V_s = 10^{-3}$ with respect to the maximum level of decomposition.

We find that the maximum level of decomposition has few effects on both the RMSE and the CR. Second, a test with a uniform aperture as the source is performed. The results are summed up in Table 4.6.

L	2	3	4	5
non-zero coefficients ($\times 10^6$)	4.12	4.09	4.09	4.09
CR (%)	75	75.6	75.6	75.6
RMSE (dB)	-62.9	-62.9	-62.9	-62.9

Table 4.6: Compression of the field radiated from a uniform aperture with $V_s = 10^{-3}$ with respect to the maximum level of decomposition.

Once again the maximum level of decomposition has almost no effect on the RMS error and CR. Therefore we choose $L = 3$ as in 1D.

4.3.2.d Computation times of FFT and FWT

The computation times for applying the FFT and the FWT on 2D signals are compared. For this test, the grid sizes in each direction N_y and N_z are chosen such that they are equal, $N_y = N_z$ and power of 2. We expect that the FWT will be faster than the FFT and that the difference will increase with the number of points. The computation time for one decomposition with respect to the number of elements $N_y N_z$ is plotted in Figure 4.8.

The results of Figure 4.8 are not what was expected. The FFT is faster than the FWT. Besides, with the number of elements increasing the difference between both times is increasing. This may be due to a more efficient implementation of FFT over FWT, leading to reduced computation times [122]. Indeed, the FFT is widely used and has been better optimized.

4.3.3 Conclusion

In this section, some numerical tests have been performed to choose a suitable set of parameters for the signals we are dealing with. We have seen that the symlet with $n_v = 6$

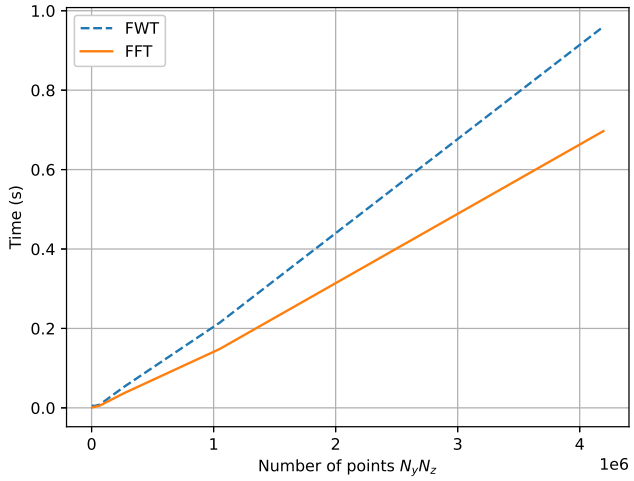


Figure 4.8: Comparison of the computation time of the FFT and the FWT with respect to N_yN_z.

and $L = 3$ seems to be a good choice for both fields with smooth and steeper variations.

We can also note that the RMS error is below $20\log(V_s) = -60$ dB regardless of the parameter values, corroborating the theoretical analysis in Chapter 2.

Then, a numerical test has been performed to compare the computation time of FFT and FWT. Unexpectedly, FFT is slightly faster than the FWT.

4.4 Extension of SSW to 3D

In this section, we extend the matrix version of SSW to 3D using the 2D wavelet transform. Also, we show why this implementation is not usable in 3D.

4.4.1 Configuration

In this section, the 3D configuration of the problem is recalled. We use the same configuration as the one with 3D Cartesian DSSF, see Chapter 1.

In few words, we study the forward propagation over the ground in the Cartesian coordinated system, *i.e.*, towards $x \geq 0$ and above $z \geq 0$. The source is placed at $x_s \leq 0$ and the field is known at $x = 0$. The domain is of finite size with $x \in [0, x_{max}]$, $y \in [-y_{max}, y_{max}]$ and $z \in [0, z_{max}]$.

The computation domain is discretized with the following grid

$$x_{p_x} = p_x \Delta x \text{ for } p_x \in \{0, \cdots, N_x - 1\}, \tag{4.11}$$
$$y_{p_y} = p_y \Delta y \text{ for } p_y \in \{N_y/2, \cdots, N_y/2\}, \tag{4.12}$$
$$z_{p_z} = p_z \Delta z \text{ for } p_z \in \{0, \cdots, N_z - 1\}, \tag{4.13}$$

with Δx, Δy and Δz the steps along x, y and z, respectively. The discretized version in y and z directions of a function $u(x, y, z)$ is denoted by $u_x[p_y, p_z]$.

4.4.2 Overview of the method

In this section, an overview of the 3D method is given. We first assume that the ground is a PEC.

The SSW method is based on the wavelet transform in place of the spectral transform in DSSF. The potential is computed marching in on distances, by going back and forth in the wavelet and spatial domains. As with SSF, 4 steps are performed to compute the field from one step to another. One step to obtain $u_{x+\Delta x}$ from u_x is computed as follows.

1. The field is represented by a sparse vector of N_s elements U_x computed by applying the fast wavelet transform (denoted by $\mathbf{W}$) and compression with hard threshold V_s (denoted by $\mathbf{C}$)
$$U_x = \mathbf{CW}u_x. \tag{4.14}$$

2. The propagator P accounts for all the wavelet-to-wavelet propagations. As in 2D propagation, two methods could be used. A straightforward method is to generalize the method of [57] that uses a scattering matrix $\mathbf{P}$ to the 3D case, which is presented in Section 4.4.4.a. This method is shown to not be relevant due to the memory occupation of the matrix. Therefore, a more efficient method based on the generalization of the local propagators is presented in Section 4.4.5. The propagated wavelet coefficients $U_{x+\Delta x}$ are
$$U_{x+\Delta x} = PU_x. \tag{4.15}$$

3. The free-space propagation is computed in the space domain with an inverse FWT (denoted by $\mathbf{W}^{-1}$)
$$u^{fs}_{x+\Delta x} = \mathbf{W}^{-1}U_{x+\Delta x}. \tag{4.16}$$

4. Apodisation windows (operator $\mathbf{H}$) are applied at each side of the domain and at the top of the domain to avoid undesired reflections. Refraction (operator $\mathbf{R}$) is also applied in the spatial domain to obtain the propagated field
$$u_{x+\Delta x} = \mathbf{RH}u^{fs}_{x+\Delta x}. \tag{4.17}$$

To take into account the ground, as in 2D [57], the local image method is used. It allows to accurately describe the effects of the ground with few additional points. As in 2D, the DMFT and the local image method are used for impedance ground conditions, see Chapter 2 and 3. For a y-invariant ground composition, the ground waves are propagated with the same propagators. In the case of a y-variant ground, the method can not be used and preliminary works are presented in Section 5.5. The relief is not accounted in 3D in this thesis work.

4.4.3 Wavelet properties to efficiently compute the propagator

The propagator stores the wavelet-to-wavelet propagations. Three wavelet properties are used to compute P efficiently. They are introduced now.

4.4.3.a Translation invariance

First, since a separable wavelet basis is used in 2D, the translation invariance property of the 1D wavelet remains valid for each directions. Thus, for one direction and two levels l and l', we recall that:

- if $l' = l$ then a translation of 1 on level l corresponds to the same one on level l'.
- if $l' < l$ then a translation of 1 on level l corresponds to a translation of $2^{l-l'}$ on level l'.
- if $l' > l$ then a translation of $2^{l'-l}$ on level l corresponds to a translation of 1 on level l'.

We note $p^l_{t_y}$ and $p^l_{t_z}$ the translations in each direction needed in the wavelet domain. Since the previous property is true in both directions y and z, for each level l we have $(p^l_{t_y}, p^l_{t_z}) \in [0, 2^{L-l}[\times[0, 2^{L-l}[$. Thus, we must store a total of $2^{2(L-l)}$ propagated wavelets for each orientation. This corresponds to all the translations in each direction y and z and to the coupled translations. This is shown in Figure 4.9, where all these translations are presented, with the upper-script corresponding to the level $l \in [1, L]$. In this case we have

$$\begin{aligned} &\text{for } l = 1 (p^1_{t_y}, p^1_{t_z}) \in [0, 4[\times[0, 4[, \quad \text{in black,} \\ &\text{for } l = 2 (p^2_{t_y}, p^2_{t_z}) \in [0, 2[\times[0, 2[, \quad \text{in blue,} \\ &\text{for } l = 3 (p^3_{t_y}, p^3_{t_z}) \in [0, 1[\times[0, 1[, \quad \text{in red.} \end{aligned} \tag{4.18}$$

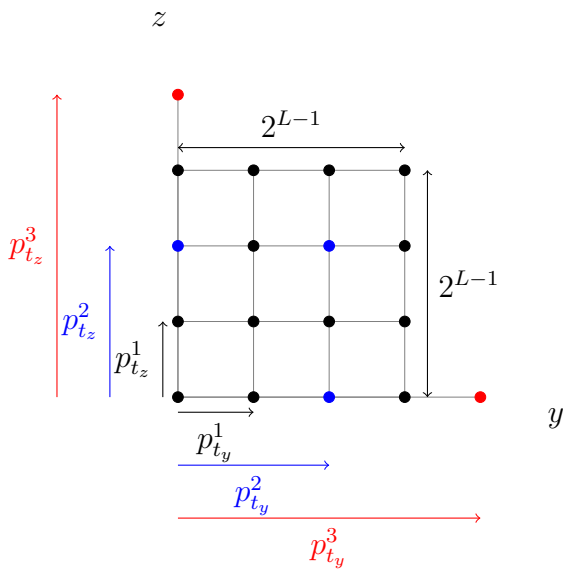


Figure 4.9: Number of translations needed at each level with $L = 3$.

4.4.3.b Rotation invariance

Secondly, when $\Delta y = \Delta z$ the rotation invariance property can be used to reduce the number of DSSF needed. In this case for a level l, the vertical wavelet corresponds to the rotation of $\pi/2$ of the horizontal wavelet.

Therefore, if $\Delta y = \Delta z$, then the propagation of the vertical wavelet is computed as the rotation of the propagated horizontal wavelet. The wavelet $\chi_{l,v,0}(\Delta x)$ is thus obtained as a rotation of $\pi/2$ of the propagated horizontal wavelet $\chi_{l,h,0}(\Delta x)$. Thus, one less propagation is needed at each level.

The computation time of the propagators can be further reduced using the last property of the wavelets.

4.4.3.c Wavelet support

Lastly, the support of the wavelet is used to further reduce the computation time of the set of propagators. Since this support is small in comparison to the domain size, the propagation on the wavelet support is faster.

Firstly, the support of the wavelet after propagation has to be computed. We use the same methodology as in 2D but in both directions y and z. We denote $(N_{l_y}^{\mathrm{p}} \times N_{l_z}^{\mathrm{p}})$ the support needed after propagation in each direction at a level l. To calculate these parameters, for each level l, the theoretical support of the wavelet $N_{l_y} \times N_{l_z}$ and the validity domain of the parabolic wave equation, $\pi/4$, are used. Using the propagation steps Δx, Δy and Δz the number of points in each direction is given by

$$N_{l_y}^{\mathrm{p}} = N_{l_y} + \sqrt{2}\frac{\Delta x}{\Delta y}, \tag{4.19}$$

$$N_{l_z}^{\mathrm{p}} = N_{l_z} + \sqrt{2}\frac{\Delta x}{\Delta z}. \tag{4.20}$$

The propagation using these supports is thus much faster than on the overall original domain. Besides the computation time to compute the propagators becomes independent of N_y and N_z.

4.4.4 3D matrix SSW

4.4.4.a Computing the scattering matrix

In this part, the generalization to the 3D of the scattering matrix is introduced, denoted by $\mathbf{P}$. Each element of the propagator $\mathbf{P}_{[l,(t_y,t_z)][l',(p_{t_y}^{l'},p_{t_z}^{l'})]}$ corresponds to the level l and position (t_y, t_z) of decomposition of the propagated wavelet of level l' at position $(p_{t_y}^{l'}, p_{t_z}^{l'})$.

For conciseness, the scaling function is treated as a diagonal wavelet of level $l = L$. The wavelet-to-wavelet propagator is thus computed in three steps.

Firstly, one centered wavelet $\chi_{l',0}^{o}[p_y, p_z]\,\forall (p_y, p_z) \in [0, N_y - 1] \times [0, N_z - 1]$ of each level $l' \in [1, L]$ and each orientation $o \in \{h, v, d\}$ is propagated on Δx on its reduced support $N_{l_y}^{\mathrm{p}} \times N_{l_z}^{\mathrm{p}}$ using DSSF. The field associated to the wavelet is obtained as in 2D, by computing the inverse FWT of a wavelet decomposition where only the center coefficient of level l' is set to 1. Thus, $\chi_{l',\Delta x}^{o}[p_y, p_z]$ is obtained. Therefore $3L + 1$ DSSF are needed.

Moreover, if the steps are the same in y and z directions, then the rotation invariance property can be used and the propagated vertical wavelet is computed by rotation of the horizontal one, leading to a reduction of the computational cost to $2L + 1$ DSSF.

Secondly, the wavelets are translated in order to obtain all the needed propagations, as in 2D. Thus each centered propagated wavelet of level l' is translated of $p_{t_y}^{l'} \in [0, 2^{L-l}[$ and $p_{t_z}^{l'} \in [0, 2^{L-l}[$ along y and z so as to obtain $\chi^o_{l',\Delta x}[p_y + p_{t_y}^{l'}, p_z + p_{t_z}^{l'}]$. This corresponds to the propagations of all the points presented in Figure 4.9.

Finally, the propagated wavelets are decomposed and compressed (threshold V_p) in order to obtain $\mathbf{P}_{[l,(t_y,t_z)][l',(p_{t_y}^{l'},p_{t_z}^{l'})]}$, with t_y and t_z the wavelet position at level l. The other elements are filled by duplicating the elementary propagations previously obtained to obtain the scattering matrix on the whole domain. Thus, redundant information is stored.

The propagation is then computed with a sparse matrix-sparse vector product between the sparse scattering matrix $\mathbf{P}$ and the sparse matrix of coefficients U_x.

Now that the method to compute the scattering and the propagation has been introduced, we focus on the size of the matrix and show that it becomes a major burden for the 3D generalization.

4.4.4.b Scattering matrix size

In 2D, we already saw that this matrix could become a serious burden in terms of memory occupation. In 3D the scattering matrix, which corresponds to all the wavelet-to-wavelet propagations, should store N_yN_z matrices of size N_yN_z, leading to a number of $(N_yN_z)^2$ elements.

For a given number of points, we compute the total number of elements of the matrix without any compression and with only 10% elements stored, *i.e.*, a sparsity of 90%. For this last, the memory occupation in octet for a complex (16 octets) is also computed. This is summed up in Table 4.7.

Domain	64×64	128×128	256×256	512×512	1024×1024
N_yN_z	4096	16384	65536	262 144	1 048 576
Whole matrix	1.67×10^7	2.68×10^8	4.29×10^9	6.87×10^{10}	1.1×10^{12}
Sparse matrix	1.67×10^6	2.68×10^7	4.29×10^8	6.87×10^9	1.1×10^{11}
Mem. occupation	27 MB	430 MB	6.9 GB	110 GB	1.8×10^3 GB

Table 4.7: Number of coefficients needed for the scattering matrix $\mathbf{P}$.

Table 4.7 shows that even if a sparse scattering matrix is pre-computed the number of elements needed becomes a serious burden even for a scenario of reasonable size. The memory size and computation time of the propagator will thus be a major problem, preventing from using it. Besides, the propagation time will also be long since the search for an element will be problematic. Thus, this method is not used. This has motivated the generalization of the local method to the 3D case instead of the matrix one, which is presented in the following section.

4.4.5 Local SSW

In this part, we detail the computation of the local propagators in 3D [123]. The idea is the same as in 2D, we want to compute and store only the essential information. Thus, only

the minimal number of wavelets is propagated and kept in a set denoted by P. The element $P_{l,o,(p^l_{t_y},p^l_{t_z})}$ of the set P, of size $N^{\mathrm{p}}_{l_y} \times N^{\mathrm{p}}_{l_z}$, corresponds to the wavelet coefficients of the propagated wavelet at position $(p^l_{t_y}, p^l_{t_z})$ of level l and orientation o. The set P corresponds to the minimal subset of the scattering matrix $\mathbf{P}$ needed for the propagation. Then, for each coefficient of the wavelet decomposition of the field, an element of this set returns the associated propagated coefficients. Summing all these elementary propagations yields the propagated field.

4.4.5.a Computation of $P_{l,o,(p_{t_y},p_{t_z})}$

Here, to obtain a usable propagator only $N_t = 1 + 3\sum_{l=1}^{L} 2^{2(L-l)}$ local propagators are stored, such that the memory occupation is minimum. This number N_t corresponds to all the needed propagation: 1 for the scaling function and then all the original and translated propagations for all three orientations. These propagators correspond to each point plotted in Figure 4.9. No redundant information is computed and stored here.

The computation of the set of propagators is presented. This computation can be divided in 3 steps:

1. For each level l and orientation o the wavelet $\chi_{l,o,0}[p_y, p_z]$ at position $x = 0$ is propagated of Δx using a 3D DSSF on the reduced support $N^{\mathrm{p}}_{l_y} \times N^{\mathrm{p}}_{l_z}$. Thus, $\chi_{l,o,\Delta x}[p_y, p_z]\, \forall (p_y, p_z) \in [0, N_y - 1] \times [0, N_z - 1]$ is obtained.
2. Using the translation invariance property, all wavelets of level l and orientation o are translated in each direction of $p^l_{t_y}, p^l_{t_z} \in [0, 2^{L-l}[^2$, leading to $2^{2(L-l)}$ translations in total for each level l and orientation o. We obtain the wavelets $\chi_{l,o,\Delta x}[p_y + p^l_{t_y}, p_z + p^l_{t_z}]\, \forall (p_y + p^l_{t_y}, p_z + p^l_{t_z}) \in [0, N_y - 1] \times [0, N_z - 1]$.
3. The translated wavelet $\chi_{l,o,\Delta x}[p_y + p^l_{t_y}, p_z + p^l_{t_z}]$ is then decomposed with a 2D FWT and compressed (hard threshold V_p) to obtain the propagated coefficients that are stored in $P_{l,o,(p^l_{t_y},p^l_{t_z})}$. This set only stores $1 + 3\sum_{l=1}^{L} 2^{2(L-l)}$ sparse propagators.

We need $3L + 1$ DSSF and $1 + 3\sum_{l=1}^{L} 2^{2(L-l)}$ FWT to compute the propagators. The computation time can be reduced using the invariance by rotation between vertical and horizontal wavelets. If $\Delta y = \Delta z$, the propagation of the vertical wavelet is computed as the rotation of $\pi/2$ of the propagated horizontal wavelet. This yields a reduction of the number of the needed propagations to $2L + 1$. The number of FWT is also reduced to $1 + 2\sum_{l=1}^{L} 2^{2(L-l)}$.

4.4.6 Propagation by means of local propagators

The propagation works as in 2D. To briefly introduce the idea of the propagation, a non-zero wavelet coefficient of u_x is propagated using its associated propagator. We obtain one elementary propagation. This one is then added to the coefficients containing all the other elementary propagations. Thus, the vector $U_{x+\Delta x}$ of the propagated coefficients is iteratively computed.

Now, the details of the propagation are given. First the non-zero coefficients of U_x, denoted by $\alpha^o_l[p_y, p_z]$, are obtained by applying a FWT and compression (threshold V_s)

to u_x. For each coefficient, we compute the associated propagator $P_{l,o,p^l_{t_{py}},p^l_{t_{pz}}}$ with

$$p^l_{t_{py}} = p_y \mod 2^{L-l}, \tag{4.21}$$
$$p^l_{t_{pz}} = p_z \mod 2^{L-l}. \tag{4.22}$$

The elementary propagation then corresponds to $\alpha^o_l[p_y,p_z]P_{l,o,p^l_{t_{py}},p^l_{t_{pz}}}$. All these elementary propagations are translated to the original coefficient positions p_y, p_z and summed so as to obtain $U_{x+\Delta x}$ as follows

$$U_{x+\Delta x} = \sum_{l,o,p_y,p_z} \left(\alpha^o_l[p_y,p_z]P_{l,o,p^l_{t_{py}},p^l_{t_{pz}}}\right)[\cdot+p_y,\cdot+p_z]. \tag{4.23}$$

An inverse FWT gives the free-space propagated field.

Finally, formula (2.49) allows to compute the thresholds in order to manage the compression error.

4.4.7 Complexity and memory

Now that the propagation method has been described, the complexity of the method is presented and compared to DSSF.

First, a 2D FWT is applied. This operation is of complexity $O(N_y(N_z+N_{im})) \sim O(N_yN_z)$ because $N_{im} \ll N_z$. For the propagation part the number of operations N_o needed is

$$N_o = \sum_{l,o,p_y,p_z \in \mathcal{I}_{nz}(U)} \mathcal{N}_{nz}(U^o_{l,x})\mathcal{N}_{nz}(P_{l,o,p^l_{t_{py}},p^l_{t_{pz}}}), \tag{4.24}$$

with $\mathcal{I}_{nz}$ and $\mathcal{N}_{nz}$ the indices and the number of non-zero elements of the vector, respectively. Finally we have

$$N_o = N_pN_s, \tag{4.25}$$

with

$$N_p = \mathcal{N}_{nz}(P_{l,o,p_{t_{py}},p_{t_{pz}}})$$
$$N_s = \mathcal{N}_{nz}(U_x).$$

With compression thresholds V_s and V_p, the numbers N_s and N_p are both much smaller than N_yN_z, leading to $N_o \ll (N_yN_z)^2$. If a high CR is obtained then $N_o \sim N_yN_z$ can be achieved. But N_o largely depends on the compression rates. Then a inverse FWT of complexity $\mathcal{O}(N_yN_z)$ is applied. Multiplying each complexity steps by N_x, the total complexity is driven by

$$O(N_xN_o) \text{ or } O(N_xN_yN_z). \tag{4.26}$$

Comparing the complexity at each step with DSSF we obtain the Table 4.8. If good compression rates on the signal and propagators are obtained then lSSW becomes more effective, but this mostly depends on the wavelet parameters.

For the computations of the the propagators, we need $3L+1$ DSSF and $N_t = 1 + 3\sum_{l=1}^{L} 2^{2(L-l)}$ FWT but on a support of size $N^p_{l,y}N^p_{l,z} \ll N_yN_z$. Thus, the complexity is driven by

$$O\left(N^p_{l,y}N^p_{l,z}\left(\log(N^p_{l,y}) + \log(N^p_{l,z})\right) + N_tN^p_{l,y}N^p_{l,z}\right). \tag{4.27}$$

method	DSSF	lSSW
transforms	$O\left(N_y N_z \left(\log(N_y) + \log(N_z)\right)\right)$	$O(N_y N_z)$
one propagation step	$O(N_y N_z)$	$O(N_o)$
inverse transforms	$O\left(N_y N_z \left(\log(N_y) + \log(N_z)\right)\right)$	$O(N_y N_z)$

Table 4.8: Computational complexity of DSSF and lSSW

Therefore, the complexity is drastically reduced using the wavelet support.

Besides, in practice, since $N^p_{l,y} N^p_{l,z} \ll N_y N_z$ the time to compute the set of propagators is of the same order than of the time required for one step of propagation. This allows to change the grid size with acceptable effort if needed, adding versatily to the method.

Finally, for the memory occupation, with lSSW a sparse set of $1 + 3\sum_{l=1}^{L} 2^{2(L-l)}$ sparse elements is stored and thus only $N_p \ll N_y N_z$ elements are stored, whereas $\mathrm{CR}(N_y N_z)^2$ elements were needed for the scattering matrix, with CR the compression rate. Therefore, the memory size is highly reduced with this method. Nevertheless, the memory requirement relies also mostly on the compression rate obtained with threshold V_p. Also, we only have to save $N_s \ll N_y N_z$ elements for the field.

To conclude, a good compression rate improves both main requirements of 3D: memory size and computation time.

4.5 Conclusion

This chapter has introduced the 2D multilevel discrete wavelet transform using a separable wavelet basis. With the aim of the extension of SSW to 3D. Two different methods for the propagation have been introduced and the local one has been shown to be relevant for 3D long-range propagation.

Firstly, a brief sreminder on the 2D discrete wavelet transform with separable families has been presented. The 2D separable wavelet basis and its associated discrete wavelet transform have been defined. The FWT has been recalled and notations for conciseness have been introduced. The sparse representation capacity of the wavelets has also been introduced. We have recalled important wavelet characteristics. Some numerical tests have been performed to choose a good set of parameters for the physical signals we deal with. Besides, the FFT and FWT computation times in Python have been compared. Surprisingly FFT is faster than FWT. This should be noted for future comparisons of propagation methods.

Secondly, the extension of SSW to 3D has been introduced. We have recalled the general method and introduced the computation of the scattering matrix. Nevertheless, we have shown that the memory size necessary for the propagator is a serious problem since for domains of reasonable size the number of elements needed for the sparse matrix is prohibitive. Therefore, the method is not used in 3D.

This latter has motivated the generalization of the local SSW method to 3D. The extension has been presented. The local method has been shown to be relevant since the memory occupation is low and can now be used for 3D propagation scenarios.

Chapter 5

Validation and Operational Use of the 3D Local SSW

5.1 Introduction

In the previous chapter, the local SSW method has been generalized to 3D.

The objectives of this chapter are to validate the 3D version of lSSW with several numerical experiments and to show some practical uses of the method. We compare lSSW to DSSF and show that the method is better in terms of memory occupation and is of the same order in terms of computation time.

The chapter is organized as follows. In Section 5.2, some experiments in free space are performed in order to validate the method. In Section 5.3, tests with canonical refractions are performed. First, we compute the propagation of a CSP when a 45°-slanted atmosphere is assumed. Second, a bidirectional-duct scenario is studied to show 3D behaviors. In Section 5.4, lSSW with the local image method is validated with the propagation of a CSP over two different planar grounds: a PEC and a dielectric. Then, a realistic test of lSSW with the propagation over an island, where the Millington recovery effect should appear, is proposed. In Section 5.4.3, the problem of y-variant ground is introduced. We show that the DMFT is not valid anymore in this case. Section 5.6 concludes the chapter.

5.2 Free-space propagation

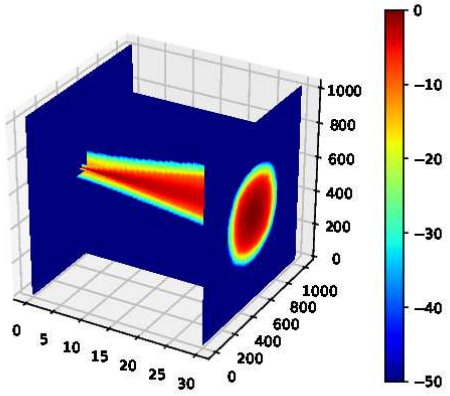
In order to validate the lSSW method, tests of free-space propagations are performed and compared to DSSF. Two different sources are studied: a CSP and a uniform aperture.

For the wavelet parameters, the symlet with $n_v = 6$ and a maximum level of decomposition of $L = 3$ are used. The thresholds are computed using formulas (2.49) in order to obtain a final accuracy of -30 dB.

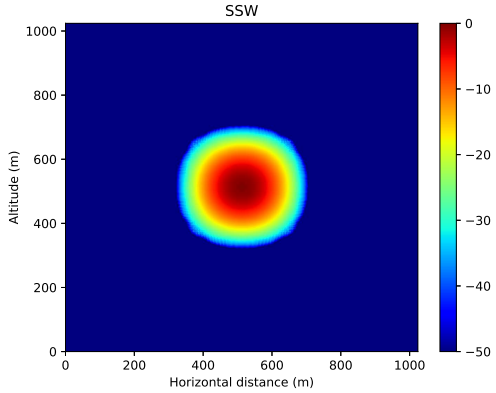
5.2.1 CSP scenario

The propagation of a CSP is studied. The frequency is 300 MHz. The domain is of size $x_{\max} = 1500$ m and $y_{\max} = z_{\max} = 1024$ m. The steps are 50 m in x and 1 m in y and z directions. This leads to $N_x = 30$ and $N_y = N_z = 1024$.

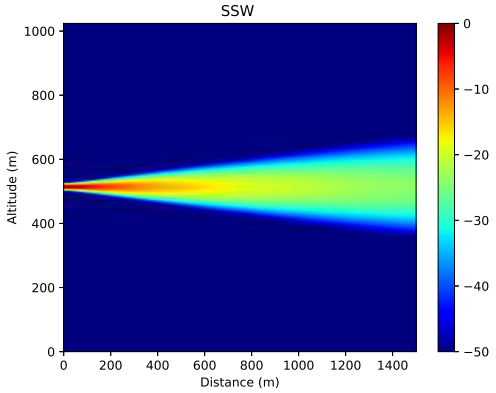
The source is placed at $x_s = -50$ m and $y_s = z_s = 512$ m. The waist size is $W_0 = 5$ m. In Figure 5.1 (a), the propagation of the field along the propagation direction is plotted. In Figures 5.1 (b) and (c), the final normalized fields in the $y0z$-plane and $x0z$-plane are plotted. Finally, in (d) the RMS difference between lSSW and DSSF with the number of iterations is plotted.



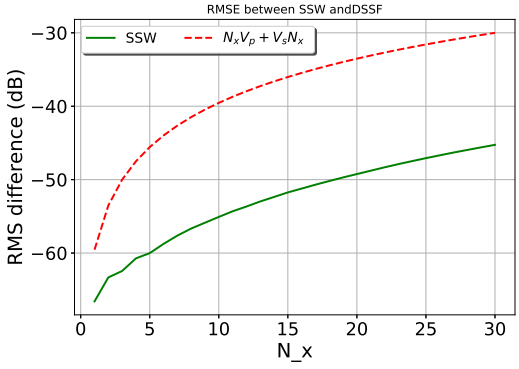
(a) 3D plot of the normalized field (dB) obtained with lSSW.



(b) Final normalized field (dB) obtained with lSSW in yOz-plane.



(c) Final normalized field (dB) obtained with lSSW in xOz-plane.



(d) RMS difference evolution (dB).

Figure 5.1: Propagation of the field radiated by a CSP in free space.

The final normalised difference is -36 dB. First, the difference with DSSF is negligible, thus the method is validated in this case. Second, we can see that the bound is never reached. This corroborates the theoretical formulas developped in Section 2.4. To conclude, the method is validated in this case.

Now, the memory size and computation times of lSSW and DSSF are summed up in

Table 5.1 and compared. Propagator time and memory correspond to the time to compute the matrix in DSSF (1.70) and the set of propagators in lSSW, see Section 4.4.5.a, and their respective memory occupation. Propagation time and memory correspond to the time to compute the propagation on the whole domain with both methods and to the memory size of the field for DSSF and the wavelet coefficients for lSSW.

method	DSSF	lSSW
propagator time (s)	36	1.7
propagation time (s)	60	100
propagator memory (MB)	151	1.5
propagation memory (MB)	151	0.05 to 2.1

Table 5.1: Computation time and memory occupation needed by lSSW and DSSF.

The first conclusion is that in terms of memory occupation lSSW is better than DSSF by a factor of 100. Also, the computation time for the propagator with lSSW is better than with DSSF. Nevertheless, the propagation part is faster with DSSF than lSSW. This can be due to many parameters. First, the compression ratio could be higher by allowing a reduction of accuracy. Second, as we have seen in Section 4.3.2.d, the FFT is faster than the FWT even if the complexity of the latter is better. Finally, the lSSW code is a laboratory Python code including loops, whereas the matrix-vector product for the DSSF propagation is optimized in Python.

Now, we reduce the grid size in y and z direction by a factor of 2 to have $\Delta y = \Delta z = 0.5$ m. The other parameters remain the same. In this case, the results of Figure 5.2 are obtained. The normalized fields in the yOz and xOz planes obtained with lSSW at the last iteration are plotted in Figures 5.2 (a) and (b). The RMS difference evolution is displayed in Figure 5.2 (c).

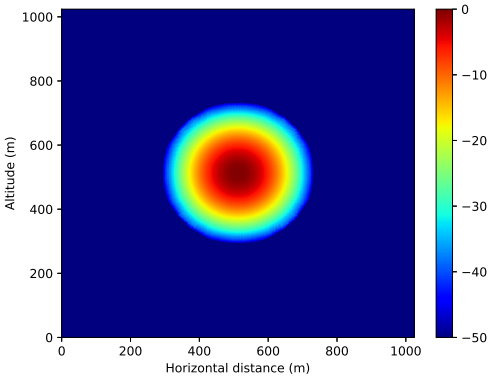
First, as expected, the RMS difference is still below the theoretical bound. The final normalized error is of order -45 dB and is thus negligible. In terms of memory occupation and computation time the results are summed up in Table 5.2.

method	DSSF	lSSW
propagator time (s)	138	8.3
propagation time (s)	343	330
propagator memory (MB)	604	10.2
propagation memory (MB)	604	0.05 to 2.1

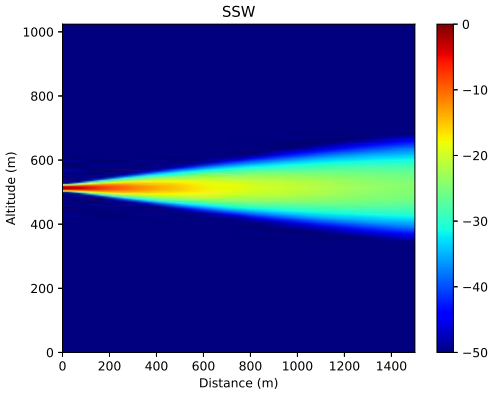
Table 5.2: Computation time and memory occupation needed by lSSW and DSSF when $\Delta y = \Delta z = 0.5$ m.

This time, lSSW method is better than DSSF in memory size of the propagator and in computation time. Therefore, in this case, the lSSW method is validated.

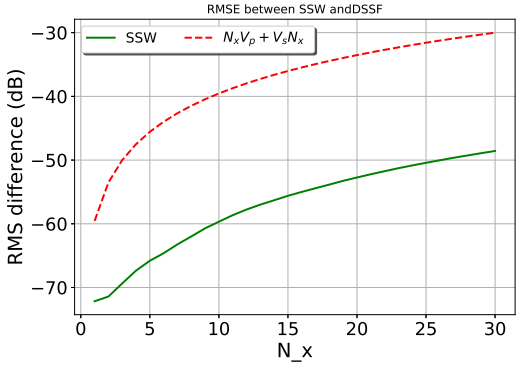
Now, a test with a field with steeper variations is performed.



(a) Final normalized field (dB) obtained with lSSW in yOz-plane



(b) Final normalized field (dB) obtained with lSSW in xOz-plane

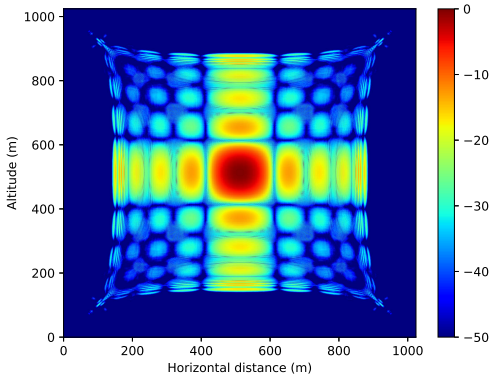


(c) RMS difference evolution (dB).

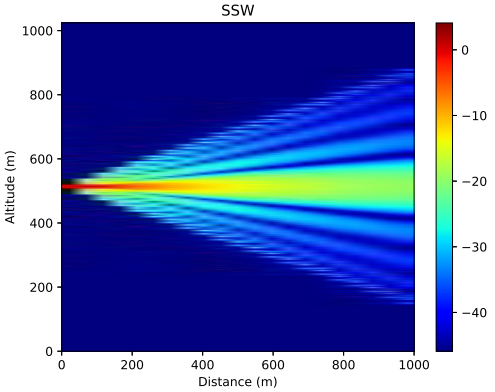
Figure 5.2: Propagation of the field radiated by a CSP in free space with $\Delta y = \Delta z = 0.5$ m.

5.2.2 Aperture scenario

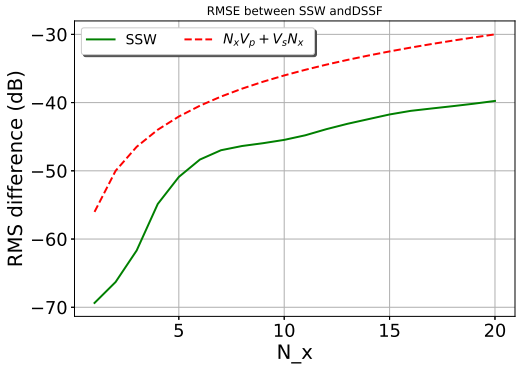
In this section, the propagation of the field radiated by a uniform aperture over 1000 m is studied. The source is placed at $x_s = 0$ m and $y_s = z_s = 512$ m. The width are $w_y = 5$ m and $w_z = 5$ m in y and z directions, respectively. The other parameters remain the same as the previous simulation. In Figure 5.3 (a) and (b), we plot the final normalized fields obtained with lSSW in the $y0z$ and $x0z$-planes. In (c), the RMS error evolution between lSSW and DSSF with N_x is plotted.



(a) Final normalized field (dB) obtained with lSSW in yOz-plane



(b) Final normalized field (dB) obtained with lSSW in xOz-plane



(c) RMS error evolution (dB).

Figure 5.3: Propagation of the field radiated by a uniform aperture in free space with $\Delta y = \Delta z = 0.5$ m.

The final normalized error is of -34 dB, still below the theoretical formula developped in Section 2.4. Also, in Figure 5.3, we can see that the first lobes are properly taken into account. Note that, due to the compression and grid size the last sidelobes are note taken into account. The results in terms of memory and computation times are summed up in Table 5.3.

Since more coefficients are needed with a uniform aperture, the computation time is better with DSSF. Also the CR is lower and storing the wavelet coefficients has a higher memory cost than for the CSP. Nevertheless, as expected, the memory size needed with lSSW is 60 times better than with DSSF for the propagator and 7 times better for the field.

method	DSSF	lSSW
propagator time (s)	136	8.3
propagation time (s)	278	3600
propagator memory (MB)	604	10.2
propagation memory (MB)	604	2 to 80

Table 5.3: Computation time and memory occupation needed by lSSW and DSSF for the propagation of a uniform aperture source.

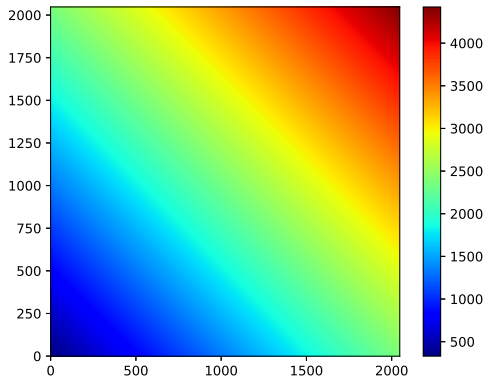


Figure 5.4: M-index (M-units/m) on the yOz-plane.

To conclude this part, lSSW is validated in these cases and the compression thresholds (2.49) are still relevant. Also, the memory size is better with lSSW. Nevertheless, the computation time depends on the source and the number of coefficients needed to describe the field. In the next sections, more complex scenarios are treated.

5.3 Propagation with refraction

5.3.1 45°-slanted linear atmosphere propagation

The method is now applied to a long-range 3D scenario with a 45°-slanted refractivity index.

The configuration parameters are as follows: the frequency is 300 MHz. The propagation is computed over 10 km with steps of 200 m in x-direction. The domain is of size 2048×2048 m in y and z directions with $\Delta y = \Delta z = 1$ m. The source is a complex source point placed at $x_s = -50$ m with a waist size of $W_0 = 10$ m and at $y_s = z_s = 1024$ m.

The 45°-slanted linear atmosphere is modeled by a linear refractive index in both directions with $c_0 = 1$ M-units/m. This value is chosen in order to obtain significant refraction effects. The M-index on the yOz-plane is presented in Figure 5.4, corresponding to the phase screen applied.

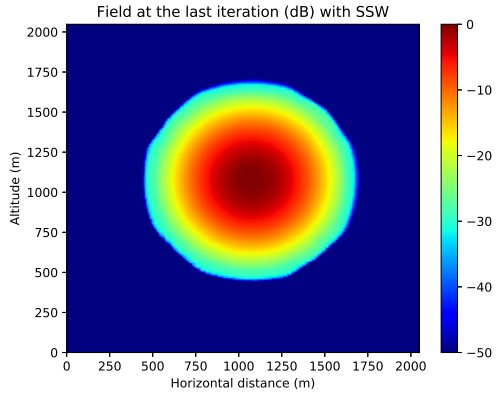
For the wavelet parameters, the same as before are used. Only, the final expected error is decreased to -20 dB to accelerate lSSW.

The propagation is computed with lSSW and DSSF for validation. Also, the shift of the beam center is compared to ray theory [124], where the shift is given theoretically by

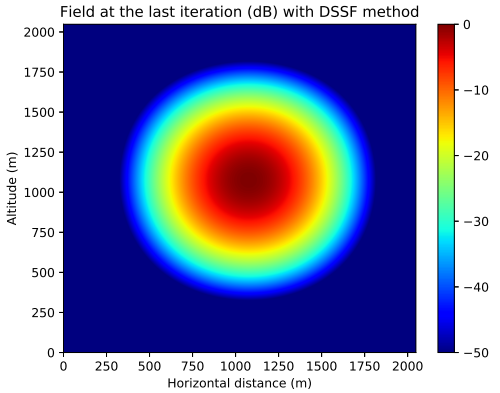
$$l_{\text{shift}} = \frac{x_{\max}^2 c_0 \times 10^{-6}}{2}. \tag{5.1}$$

Thus a shift of around 50 m is expected at the end.

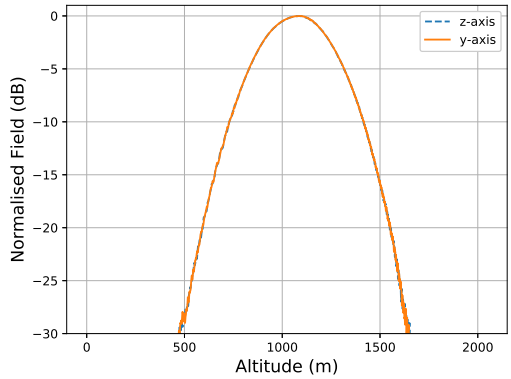
In Figures 5.5 (a) and (b), the normalized fields obtained with lSSW and DSSF are plotted. In (c), vertical and horizontal cuts at $N_y/2$ and $N_z/2$ of the normalized final fields obtained with lSSW are displayed. Finally, the normalized difference between lSSW and DSSF is plotted in (d).



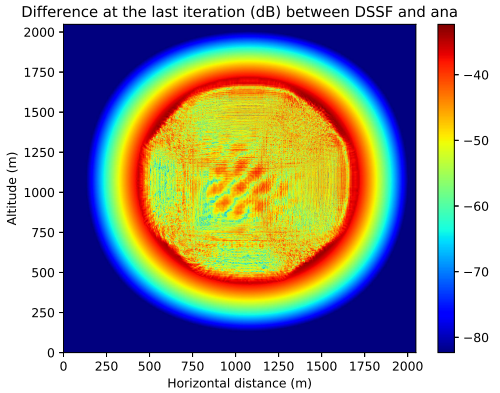
(a) Final normalized field (dB) obtained with lSSW in yOz-plane



(b) Final normalized field (dB) obtained with DSSF in yOz-plane



(c) Vertical/horizontal cuts of the final normalized field obtained (dB) with lSSW.



(d) Final normalized difference (dB) between lSSW and DSSF in yOz-plane

Figure 5.5: Propagation of the field radiated by a CSP in a 45°-slanted linear atmosphere.

First, the final difference is at most -32 dB, below the theoretical -20 dB, thus the error is consistent with the theory. We can also note that the field below -40 dB is not present with lSSW due to the compression. In addition, the fields begin to differ below -25 dB in both direction, due to the compression. Besides, note that due to the compression the gradual degradation with radius near the edge, see Figure 5.5 (b), is not obtained with lSSW. Nevertheless, the effects in y and z directions are almost the same since shifts

of 52 m and 47 m are obtained in y and z. Thus, we obtain the same results with an acceptable error and close to the theoretical shift. Hence, the effects of refraction are properly taken into account with lSSW. Therefore, good results are obtained wth lSSW in this case.

Note that, the displacement along y could not be taken into account with a $N\times$2D SSF [52, 54].

Second, the computation time and memory with DSSF and lSSW are summed up in Table 5.4.

method	DSSF	lSSW
propagator time (s)	133	6.5
propagation time (min)	9.2	10
propagator memory (MB)	604	5.1
propagation memory (MB)	604	0.03 to 3.4

Table 5.4: Computational cost of lSSW and DSSF in terms of computation time and memory occupation.

As before, the memory is better with lSSW by a factor of 100. Besides, the computation time with DSSF and lSSW are of the same order.

5.3.2 Bidirecional-duct scenario

Now, a more complex long-range propagation scenario with 3D effects along y and z is studied. A test with a two-duct refractive index along both directions y and z is performed. Thus, we expect to obtain the same behavior in both directions. With a $N\times$2D-DSSF [54], the effects in only one direction could be considered.

The two-ducts are modeled with a tri-linear model of atmosphere in both directions y and z. The parameters for the refractivity index are the same in both directions. These latter are as follows: $M_0 = 330$ M-units, $y_b = z_b = 950$ m and $y_t = z_t = 100$ m, $c_0 = 0.118$ M-units/m and $c_2 = -1$ M-units/m. The modified refractive index M is plotted in Figure 5.6. It corresponds to the phase screen applied at each step. The other parameters remain the same as in the previous simulation.

In Figures 5.7 (a) and (b), the normalized fields obtained with lSSW and DSSF are plotted. In (c), vertical and horizontal cuts at $N_y/2$ and $N_z/2$ of the normalized final fields obtained with lSSW are displayed to see the effect in both directions. Finally, the normalized difference between lSSW and DSSF is plotted in (d).

We can see the 3D effects due to the 2 ducts along y and z directions. In y and z-directions the maximum values of the final fields are located at 33 m and 32 m, respectively. Thus, the two values are the same within an acceptable error. In Figure 5.7 (c), the cut over y and z axes show that in both directions the fields are matching. Also, the error with DSSF is below -30 dB, as expected. Therefore, lSSW is successfully tested and 3D effects are successfully taken into account.

In terms of computation time and memory, the same results as with the linear atmosphere are obtained. Therefore, lSSW works well in a complex environment and allows to be

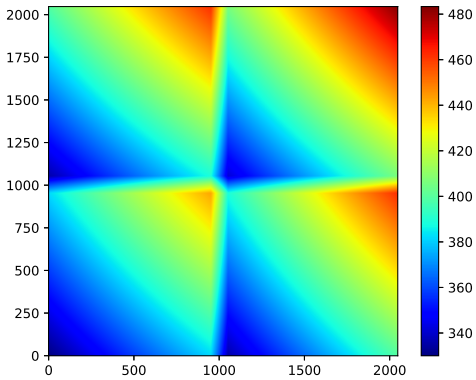
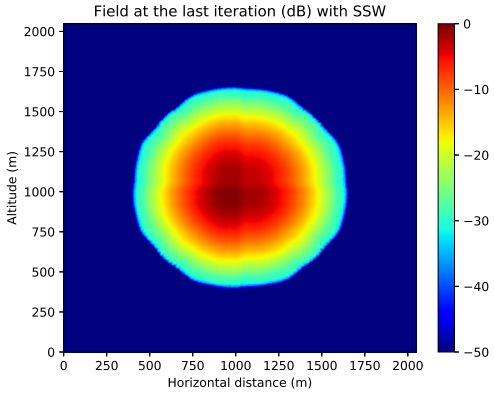
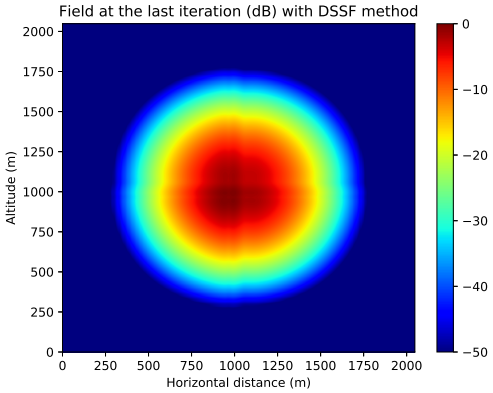


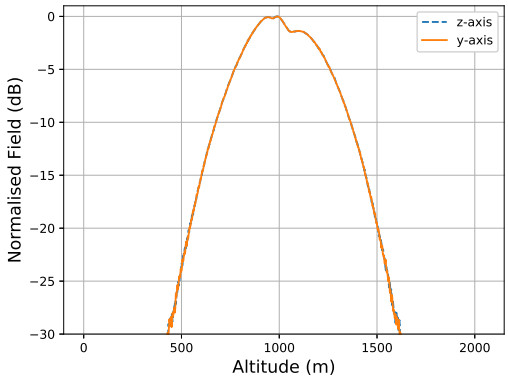
Figure 5.6: M-index (M-units/m) on the yOz-plane.



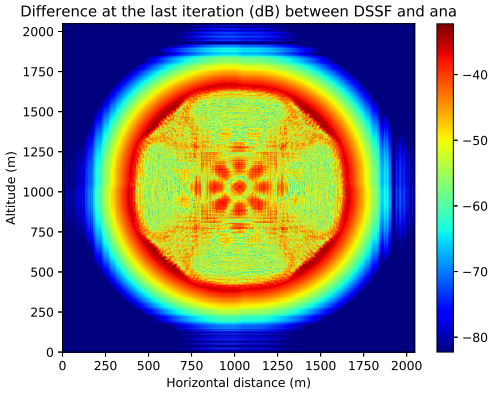
(a) Final normalized field (dB) obtained with lSSW in yOz-plane.



(b) Final normalized field (dB) obtained with DSSF in yOz-plane.



(c) Vertical/horizontal cuts of the final normalized field (dB) obtained with lSSW.



(d) Final normalized difference (dB) between lSSW and DSSF in yOz-plane

Figure 5.7: Propagation of the field radiated by a CSP in a bidirectional-duct scenario.

better in terms of memory size and with the same order of computation time.

5.4 Propagation over y-invariant planar ground

Now, simulations over y-invariant planar grounds with different compositions are performed. In this case the DMFT, see Appendix B, is valid. First, the method is validated with a PEC ground. Then, a propagation over a dielectric ground is computed. Finally, at a lower frequency a propagation over a ground-sea-ground configuration is studied. We aim at validating the 3D lSSW-DMFT with local image method.

5.4.1 PEC ground

First, a simulation over a PEC ground is performed. We aim at validating the local image method in 3D, where a thin image layer is added to take into account the ground, see Section 2.3.3.

The configuration is as follows: the frequency is 300 MHz. The propagation is computed over 5 km with steps of 200 m in x-direction. The domain is of size 2048×2048 m in y and z directions with $\Delta y = \Delta z = 1$ m. The source is a complex source point placed at $x_s = -50$ m with a waist size of $W_0 = 7$ m and at $y_s = 1024$ m and $z_s = 50$ m.

We use the same wavelet parameters. Thresholds for a final accuracy of -20 dB are set.

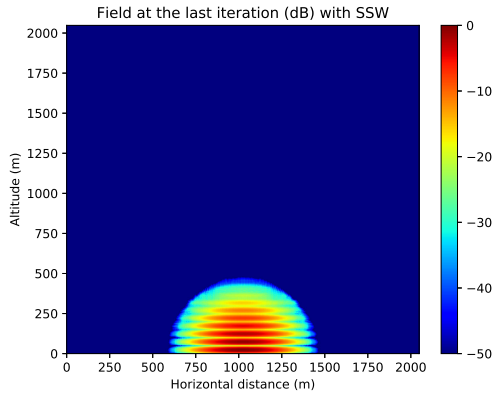
In Figures 5.8 (a) and (b), the normalized fields obtained with lSSW and DSSF are plotted. In (c), vertical cuts at $N_y/2$ of the final fields obtained with lSSW and DSSF are displayed to see if the effects of the ground are well taken into account. Finally, the normalized difference between lSSW and DSSF is plotted in (d).

The final accuracy is of -32 dB. As expected the error is negligible between lSSW and DSSF. Also, the extrema of the interference pattern due to the ground are accurately obtained. Therefore, lSSW with the local image method works well in 3D. We then compare the methods in terms of memory and computation time. This is summed up in Table 5.5.

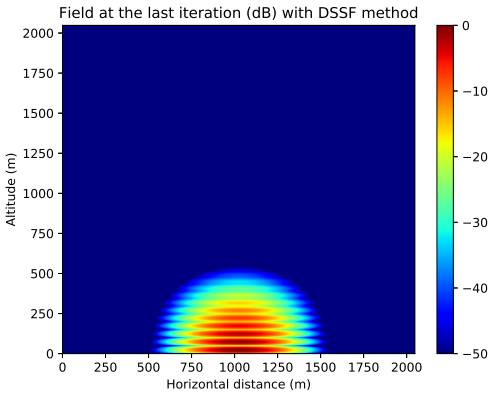
method	DSSF	lSSW
propagator time (s)	138	6.4
propagation time (s)	274	270
propagator memory (MB)	604	5.6
propagation memory (MB)	604	0.05 to 2

Table 5.5: Computational cost of DSSF and lSSW in terms of time and memory occupation.

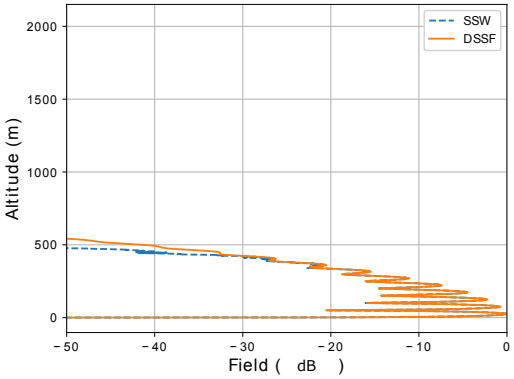
In this case, the memory size for the propagator is better, as before. Also, the total computation time is of the same order with lSSW, even if FFT is faster than the FWT, as seen in Section 4.3.2.d. We can conclude that the generalization of lSSW to 3D works well in the presence of a PEC ground. Now, a test with a dielectric ground is performed.



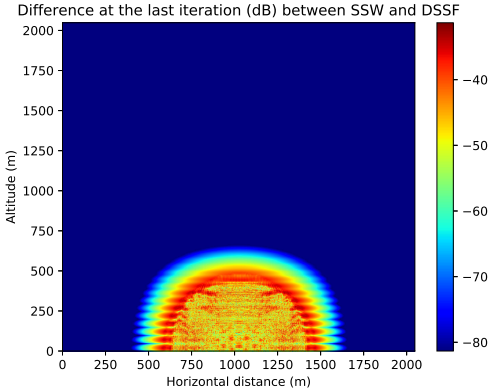
(a) Final normalized field (dB) obtained with lSSW in yOz-plane.



(b) Final normalized field (dB) obtained with DSSF in yOz-plane.



(c) Vertical cuts at $N_y/2$ of the normalized final fields (dB).



(d) Final normalized difference (dB) between lSSW and DSSF in yOz-plane.

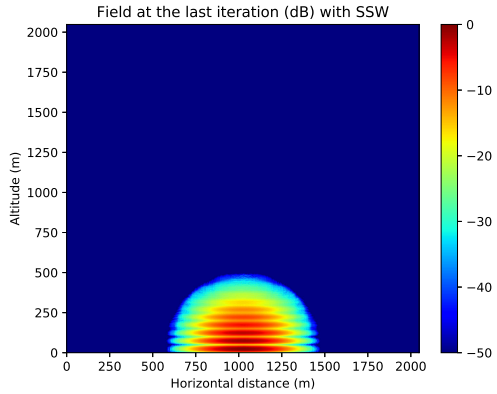
Figure 5.8: Propagation of the field radiated by a CSP over a planar PEC ground.

5.4.2 Impedance ground condition

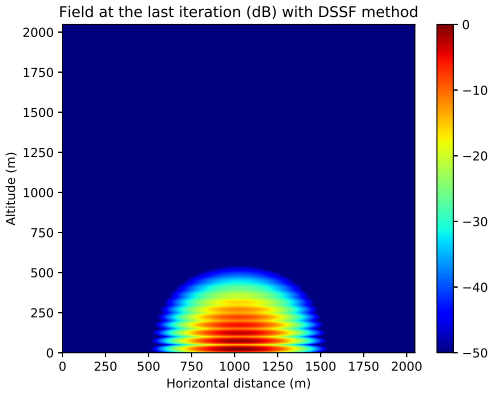
Now, we compute the propagation over a planar dielectric ground in vertical polarization. We aim at validating the propagation with lSSW and the local image method applied to w, the field obtained with the DMFT, see Section 2.3.3.

The configuration is the same as in the previous section in which the PEC ground has been replaced by a dielectric ground of parameters $\epsilon_r = 20$ and $\sigma = 0.02$ S/m.

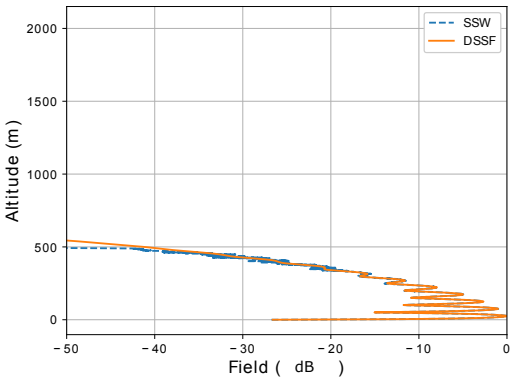
In Figure 5.9, we display the same results as in the previous section.



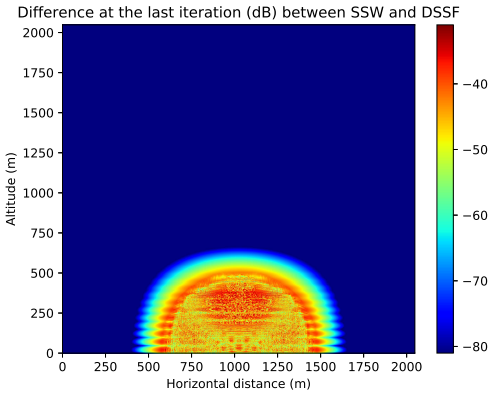
(a) Final normalized field (dB) obtained with lSSW in yOz-plane.



(b) Final normalized field (dB) obtained with DSSF in yOz-plane.



(c) Vertical cuts at $N_y/2$ of the normalized final fields (dB).



(d) Final normalized difference (dB) between lSSW and DSSF in yOz-plane.

Figure 5.9: Propagation of the field radiated by a CSP over a planar dielectric ground.

The final normalized error is of -31 dB. Also, as before the interference pattern is simulated with good accuracy. Therefore, the method works well with an impedance ground condition. In terms of memory size, the result is the same as in the previous section since the same parameters are used. Thus, lSSW allows being better for memory occupation. Nevertheless, for the computation time both lSSW and DSSF took 32 min. Therefore, SSW allows to compute the propagation in the same order of time as DSSF but with a drastically reduced memory occupation allowing to go further in distances.

5.4.3 Millington effect with infinite island

In this section, numerical tests with two different planar grounds at a frequency 75 MHz are performed using lSSW, where ground wave propagation has a high impact. We expect to observe the Millington recovery effect [12]. This would validate the DMFT with ground-wave propagation method used in 3D lSSW.

The propagation in a ground-sea-ground configuration is computed both with DSSF and lSSW.

For this test, the frequency is 75 MHz. The propagation is computed over $x_{\max} = 5$ km with $\Delta x = 100$ m. The domain is of size 1024×1024 m in y and z directions. The steps are chosen to $\Delta y = \Delta z = 4$ m. For validation an invariant source along y is considered. The source is composed of CSP a placed at $x_s = -50$ m and $z_s = 50$ m for each points in y direction.

For the wavelet parameters, the symlet with $n_v = 6$ and a maximum level of decomposition $L = 3$ are used. The thresholds are computed so as to obtain an error of -30 dB at the final iteration.

In this scenario, a sea of the size of the domain in y direction and of size 2 km in x is placed at $x = 1.5$ km. For the rest of the domain we assume a dry ground composition. This configuration is presented in Figure 5.10. The parameters are then:

- for the sea part: $\epsilon_r = 80$ and $\sigma = 5$ S/m.
- for the ground part: $\epsilon_r = 20$ and $\sigma = 0.02$ S/m.

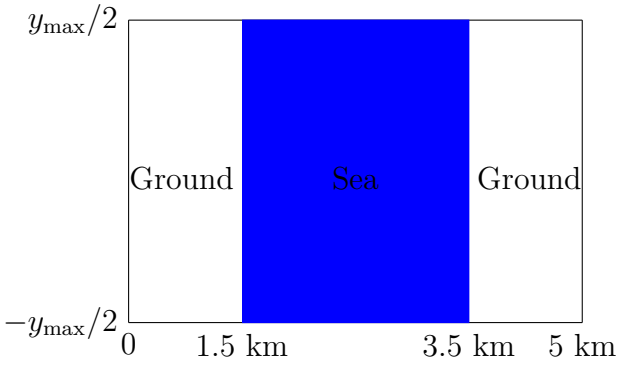
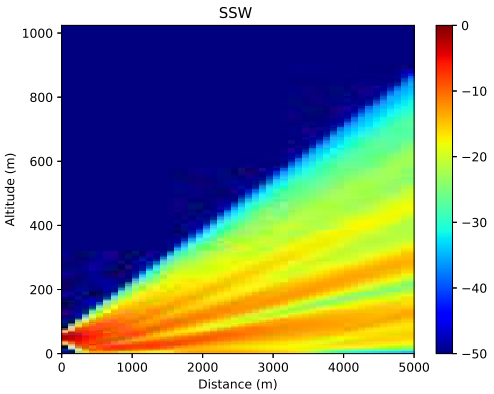


Figure 5.10: Scenario of propagation over the sea.

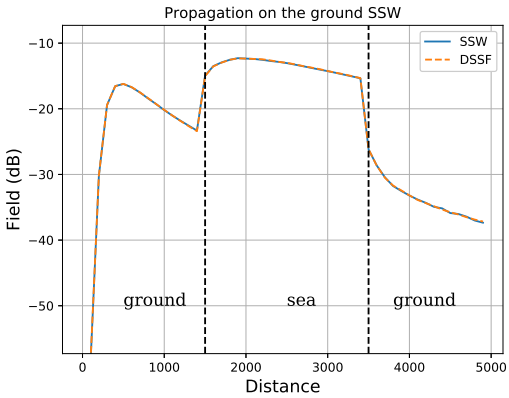
The propagation is computed with 3D lSSW, 3D DSSF and 2D lSSW. The same results in the 2D and 3D cases are expected. Over the first part of ground, the ground wave should decrease. Then, over the sea since ϵ_r and σ are high the ground wave should increase (Millington recovery effect). Finally, the field at the ground level should rapidly decrease after the sea.

The results are displayed in Figure 5.11. In Figure (a), the normalized field in the $x0z$-plane obtained with 3D lSSW is plotted. Figures (b) and (c) shows the propagation at $N_y/2$ and $z = 0$ obtained with 3D lSSW or DSSF and 2D lSSW, respectively.

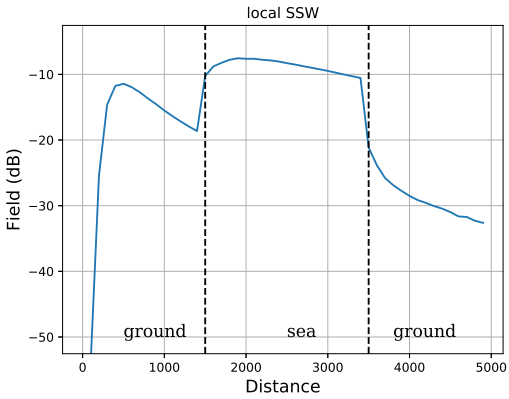
First, in 2D (c) and 3D (b) the same behavior are observed for the field at the $z = 0$. This variations correspond to the one obtained in [12]. As expected, the field at the bottom



(a) Normalised field (dB) in $x0z$-plane obtained with 3D lSSW.



(b) Propagation of the field at $z = 0$ obtained with 3D lSSW and DSSF.



(c) Propagation of the field at $z = 0$ obtained with 2D lSSW.

Figure 5.11: Propagation over a ground-sea-ground configuration.

boundary appears and then decreases until reaching the sea. At this point the Millington recovery effect is modeled, *i.e.*, the ground wave is increasing when over the sea since ϵ_r and σ are high. The ground wave behavior is the same as the one obtained in [12]. Finally, the ground wave is rapidly decreasing after the sea, as expected.

Also, the final normalised difference between 3D DSSF and 3D lSSW is -32 dB. We can see that below this value the DSSF and lSSW ground waves are slightly different.

To conclude, the 3D lSSW method allows to model ground propagation and can thus be used in complex environment. In this case, lSSW is much slower than DSSF and the memory occupation with DSSF is low, since the mesh size is wide in y and z direction. This is due to the loop for the propagation in lSSW, whereas DSSF uses a matrix-vector product for this part. Nevertheless, the main objective was to show that lSSW can be used in this case.

5.5 Propagation over a y-variant ground

In this section, we address the problem of y-variant grounds, for example, propagation over the sea with islands. In this case, we show that the DMFT, in particular the ground wave propagation, is not valid. To highlight the effect of a transversely variant impedance ground condition, we choose to work in the continuous domain for conciseness.

A first solution would be to take into account a dielectric ground as with MFT-SSF. The following change of variable would be used

$$w_x(y,z) = \frac{\partial u_x(y,z)}{\partial z} + \alpha_x(y) u_x(y,z), \tag{5.2}$$

with α_x depending on y in this case. Therefore, w is not solution of a simple wave equation, cross-terms with α_x appear in the derivative along y. Therefore, we can only work with the reduced field u.

Second, since the boundary condition in $z = 0$ depends on y, through $\alpha_x(y)$, an independent diagonalization along y and z is not possible to obtain the spectral representation. Firstly, a diagonalization along y could be considered. Since periodic boundaries conditions are assumed at the limit, a Fourier series is used. The problem is that the boundary condition at $z = 0$ in this case becomes

$$\frac{\partial \tilde{u}_x(k_y,0)}{\partial z} + \left(\tilde{\alpha_x} \otimes \tilde{u_x}\right)(k_y,0), \tag{5.3}$$

with $\tilde{u}_x$ and $\tilde{\alpha}_x$ corresponding to the Fourier transforms of u_x and α_x along y and $\otimes$ to a convolution. This condition is not local.

Secondly, a diagonalization along z could be considered. The transform would correspond to the one used in the MFT [30], but the term α_x depends on y. Therefore, convolutions would also appear.

This shows that with a ground for which the parameters are functions of the transverse components in Cartesian coordinates, no direct method can be used. A new theory should be derived.

5.6 Conclusion

In this chapter, numerical experiments in canonical scenarios have been performed to show the advantages of lSSW. In addition, propagation over an infinite island has been tested. Finally, we have shown that the DMFT can not be used when the ground parameters depend on y and that a new theory needs to be developed.

First, tests in free space have been performed to validate the method. These experiments have shown that the theoretical formula for the compression error is still relevant in 3D. Also, we have seen that in terms of memory occupation lSSW is better than DSSF. In terms of computation time, when the CR is high, lSSW can be of the same order than DSSF.

Second, tests with canonical inhomogeneous atmospheres have been proposed. Firstly, the lSSW has been applied in a 3D scenario with a 45°-slanted refractive index. The effects in both directions have been properly simulated. Secondly, a complex 3D scenario with effects along both directions has been introduced. We have considered the presence of a bi-directional duct. The effects in both directions have been well taken into account with lSSW. Besides, in terms of both computation time and memory occupation lSSW has been shown to be better than DSSF.

Third, tests over y-invariant ground have been performed. A test over a PEC has been introduced to validate the 3D lSSW with local image method. Then to test the lSSW-DMFT with local image method, a scenario with an impedance ground has been proposed. In both cases the effects of the ground have been properly simulated. The computation time and the memory occupation is better with lSSW than with DSSF. A test with an infinite sea in the middle of the domain has finally been proposed to show that the method could be used in a complex environment, implying a surface wave.

Finally, the case of ground for which the parameters are varying along the y-direction has been introduced. We have shown that the DMFT is not valid, since the parameters depend on y. A new theory must be developed to tackle this problem and to apply lSSW to a realistic island scenario.

Conclusion

This Ph.D. thesis aimed at developing a fast and accurate method for modeling the tropospheric long-range propagation in 3D. For these applications, the method should also be low in memory occupation. Several milestones toward this objective have been achieved, summarized by these three contributions:

- Based on the wavelet transform a local method of split-step wavelet has been proposed in 2D to obtain a fast method that requires low memory occupation. This method is an improvement of the matrix split-step wavelet (SSW) method previously developed. The local SSW method requires much less memory occupation.
- A theoretical bound for the compression error with SSW has been calculated. This latter allows computing the thresholds for a desired accuracy. This milestone answers to the accuracy of the method.
- Based on the local method developed in 2D, a 3D method has been proposed. This method has been shown to be good in terms of both memory occupation and computation time in comparison to split-step Fourier.

Summary

In the **first chapter**, the discrete split-step Fourier (SSF) method for long-range propagation in 2D and 3D Cartesian coordinates has been presented.

First, a state-of-the-art on the parabolic wave equation (PWE) has been presented. By neglecting the backward propagation and using the paraxial approximation, the model based on the PWE allows to model the long-range propagation in complex environments. Initial methods to solve the PWE were based on the finite difference scheme. But to benefit from a wider grid size, iterative split-step methods were introduced.

The 2D discrete version of SSF has been reminded. Assuming a PEC, DSSF has been derived from the discrete Helmholtz equation. Thus, the obtained method is self-consistent. This latter has been extended to impedance ground with the discrete mixed Fourier transform (DMFT) to obtain the DSSF-DMFT method. Numerical tests that validate the method have been presented.

Then, DSSF has been extended to the 3D Cartesian coordinates problem. To derive the method a PEC has been assumed. The continuous problem has first been described. Then, an entire discrete formulation has been introduced to obtain the discrete propagator and a self-consistent method. The method has been extended to impedance grounds with

the DMFT. Numerical tests have been performed to validate the methods. Nevertheless, the computation time and the memory occupation of the propagator are problematic for long-range scenarios.

In the **second chapter**, a theoretical bound for the compression error in the split-step wavelet method has been derived.

The 1D discrete wavelet transform has been introduced. This latter uses wavelets, short length oscillating function, as atoms. They are obtained by dilating and translating a mother wavelet. The main advantages of the wavelet decomposition are its fast transform, faster than the fast Fourier transform, and the sparse representation allowed by the wavelets.

Then, the matrix split-step wavelet (SSW) method has been presented. This method, as SSF, is an iterative method to compute the field marching on in distances. Using the fast wavelet transform and compression, this method is fast.

The compressions that are applied on the field and on the propagator induce an accumulation of error throughout the propagation. A theoretical bound has been derived for this compression error. The formula obtained allows to compute the thresholds for a given scenario and a chosen accuracy. Therefore, the accuracy of SSW is ensured.

Nevertheless, the memory occupation and the computation time of the propagator are serious burdens. Therefore, the matrix is pre-computed once and for all.

In the **third chapter**, a new local SSW method has been introduced. By storing only the necessary information for the propagation, this method has a low memory occupation.

The method follows the same steps as the matrix SSW, only the propagation part is different. Here, a set of propagators and the local propagation have been introduced to limit the memory occupation. Indeed, the translation invariance property of the wavelet basis is used to reduce the computation time and also to only store a set of propagators of minimal size. This set, contrary to the matrix, stores no redundant information. Also, the wavelet support of finite size, which is theoretically known, is used to further limit the memory occupation and computation time of this set. With the local propagators, the propagation is performed by summing the contributions of all the non-zero coefficients of the wavelet decomposition of the field. We also have shown that the time to compute the set is of the same order as the time of one step of propagation making the method versatile since adaptive steps are now possible.

Then, numerical tests have been performed to validate the method and show its advantages. Simulations have shown that the use of the wavelet support is relevant, significantly reducing the time to compute the propagators. Long-range propagation experiments have also been performed. These tests show that the local version is as accurate as of the matrix version of SSW but is better in terms of memory occupation while the propagation time is of the same order. Finally, an application to a 2D static scenario of radio-occultation has been presented, where the advantages of local SSW were highlighted.

In the **fourth chapter**, the local version of SSW has been extended to 3D.

A state-of-the-art on the 2D wavelet transform has been presented. Using a separable wavelet basis composed of the dilations and translations of three mother wavelets, any signal can be decomposed into wavelet coefficients. This 2D decomposition maintains

its advantages: its fast transform, faster than the fast Fourier transform, and the sparse representation capacity of the wavelets.

Numerical experiments have been performed to choose a good set of wavelet parameters for the signals we are dealing with. The different wavelet characteristics have been presented and their effects on the wavelet decomposition have been described. The wavelet parameters have then been chosen with a parametric study on these parameters with two different sources.

The sparse representation and computational advantages of the wavelet transform have been used to derive a 3D version of SSW. First, the matrix version of SSW has been extended to 3D, but we have shown that even for a reasonable domain size the memory occupation would be a serious problem and the method would not be usable. Hence, the 3D local version of SSW has been derived. Using the same methodology as in 2D, only the essential propagation information is stored in the set of propagators, making the memory occupation minimal.

In the **fifth chapter**, we have shown that the 3D local SSW method works well in some canonical and practical scenarios.

Numerical experiments in free space have been proposed to validate the method and show that the theoretical formula for the compression error is still relevant. Also, the advantage of lSSW in terms of memory occupation has been highlighted. The computation time has been shown to seriously depends on the compression ratio.

Then, different refractive models have been considered, where 3D effects must be taken into account. The effects have been properly simulated for inhomogeneous atmospheres in both directions. This shows the advantage of a 3D method against a $N\times$2D method. Besides, the lSSW method has been shown to be better than DSSF in both computation time and memory requirements.

To validate the 3D lSSW-DMFT with the local image method, tests with various ground compositions have been performed. The effects of the ground have been well simulated. In these cases, lSSW was better than DSSF in both memory occupation and computation time. A test with an infinite island has also been proposed to show that the method can accurately model the Millington effect in this case.

Nevertheless, the problem of a ground with parameters varying along the transverse direction has been introduced. We have shown that the DMFT is not valid in this case and that this problem is difficult. Thus, a new theory must be derived.

Perspectives

In this thesis, we have derived a local split-step wavelet method for 2D and 3D. In terms of memory occupation and computation time, the tests have shown that the method is satisfying both in both 2D and 3D. Nevertheless, several future works could be considered. They concern the implementation, the models, the tests, and the applications.

As numerical tests have shown, the propagation part in 2D and 3D is of the same order or slower with SSW than with DSSF. This is mostly because the propagation of SSW is a laboratory Python code whereas DSSF uses the optimized matrix-vector product of

Python. Therefore a first task could be to transpose the code into a compiled language such as Fortran. In this case, the time comparison between both methods would be fair.

Moreover, in SSW since the wavelets have supports of small sizes, the apodization method could be optimized. Indeed, in this Ph.D. thesis a Hanning window of the same size as the vertical domain is used to removed spurious reflections at the limit of the domain. However, a more efficient method could be developed based on the localization of the wavelets, as for the ground with the local image method.

Besides, more numerical experiments could be performed. First, SSW could be compared to the Gaussian beam-based method, since this method has also been shown to be more efficient than SSF in 2D. In particular, they could be compared in a radio-occultation scenario. Further tests with this configuration should be considered for validation. The method could also be compared to other propagation methods such as ray tracing.

Also, for RO scenario, the computation time could be reduced by reducing the computational domain. Using a conformal mapping along a preferred trajectory that goes between the two satellites, as for the relief in 2D, the propagation could follow this direction with a reduced window.

Second, a comparison of results of SSW with measurements could be considered. As in [54], the propagation through a duct retrieved from clutter data inversion could be modeled with 3D SSW.

The refraction model could also be improved to model the atmospheric scintillations effects, *i.e.*, the rapid modification of radio waves caused by small scale structures in the atmosphere. To this end, the phase-screen formulations should be modified [9].

In addition, for realistic tests and applications, the relief should be introduced in the 3D split-step wavelet method. For this latter, many challenges should be tackled. Indeed, depolarization of the field occurs and must be considered. Thus, TM and TE should be propagated together, modeling the reflection over lateral reliefs is also a challenging task.

Finally, the problem of grounds with parameters varying along the transverse direction should be tackled. As we have shown, with a spectral representation along y a boundary condition with a convolution appears. Maybe, by assuming that the parameters are piecewise constant a method based on the wavelets could be derived, since they are widely used to take steep variations into account.

Applications

They are various possible applications of the method. Firstly, the method could be used to predict the radar coverage, for either ground waves or space waves based systems. Also, the method could be used for the definition, design, and implementation of the ground facilities and systems. Indeed, ENAC has a code to model the effects of wind turbines on VOR systems [4–6] where the propagation from the VOR to the turbines is model with PWE and SSF. This part could be replaced with SSW.

Second, as we have previously mentioned the method could be used as a direct model in RO configurations. Nevertheless, it can also be used as the direct model for other

inversion problems, such as refractivity from clutter. Matrix SSW is already used for this application in the in-progress Ph.D. thesis of Uygar Karabaş at ENAC [125].

The method could be applied to model the ionosphere propagation [9]. The ionosphere corresponds to the ionized part of the atmospheric layer, from 75 to 1000 km above Earth. This part contains positively charged atoms due to the high of the Sun and the cosmic rays. This has effects on the field, for example, scintillations. Therefore, this layer has an influence on the propagation between satellites and Earth. SSW could be used as an efficient tool for this kind of problem.

Finally, in another domain, the method could be used in acoustic problems. Since the PWE model with SSF is widely used in underwater acoustic [126–128], the SSW method could be used in place of SSF.

Appendix A

2D Discrete Mixed Fourier Transform and Propagator

In this chapter, the 2D discrete mixed Fourier transform for DSSF is introuduced. This is based on [52, 71]

A.1 Spectral transform

First, the field is substituted by the variable w as follows

$$w_x[p_z] = \frac{u_x[p_z+1] - u_x[p_z-1]}{2\Delta z} + \alpha u_x[p_z], \tag{A.1}$$

with $p_z \in [1, N_z - 2]$. For w the impedance boundary condition on u becomes a Dirichlet condition.

Second, for $q_z \in [0, N_z - 1]$ the DMFT spectral transform is given by [71]

$$\begin{aligned} W_x[0] &= Q \sum_{p_z=1}^{N_z-2\,\prime} r_0^{p_z} u_x[p_z], \\ W_x[q_z] &= \sum_{p_z=1}^{N_z-2} \sin\left(\frac{\pi q_z p_z}{N_z}\right) w_x[p_z], \quad \text{for } q_z \in [1, N_z-2], \\ W_x[N_z-1] &= Q \sum_{p_z=1}^{N_z-2\,\prime} (-r_0)^{p_z} u_x[p_z], \end{aligned} \tag{A.2}$$

with the first and last terms weighted by $1/2$, which is indicated by the prime.

For a vertical polarization α has a positive real part. Therefore r_0 is defined by

$$r_0 = \sqrt{1 + (\alpha\Delta z)^2} - \alpha\Delta z, \tag{A.3}$$

In case of a horizontal polarization, since α has a negative real part, we have

$$r_0 = -\sqrt{1 + (\alpha\Delta z)^2} - \alpha\Delta z, \tag{A.4}$$

Since the terms r_0 and $-1/r_0$ correspond to the roots of

$$r_0^2 + 2\alpha\Delta z r_0 - 1 = 0, \tag{A.5}$$

we have $|r_0| \leq 1$. Therefore, singularities are avoided in the expression

$$Q = \frac{2(1-r_0^2)}{(1+r_0^2)(1-r_0^{2(N_z-1)})}. \tag{A.6}$$

Note that $W_x[0]$ and $W_x[N_z-1]$ correspond to the surface waves propagating at $z=0$ and $z=z_{\max}$. On the other hand, the space wave corresponds to the vector $W_x[q_z]$ for $q_z \in [1, N_z-2]$. In the next section, the spectral propagators needed to propagate both are derived.

A.2 Spectral propagator

The spectral propagators are obtained by solving

$$\frac{\partial^2 u_x}{\partial x^2} - 2jk_0\frac{\partial u_x}{\partial x} + \mathbf{d}_z^2 u_x = 0, \tag{A.7}$$

with $\mathbf{d}_z^2$ the finite difference approximation of the differential operator along z defined in Section 1.2.2.b.

The diagonal propagator $\mathbf{P}_s$ is thus defined by

$$W_{x+\Delta x}[q_z] = \mathbf{P}_s[q_z, q_z] W_x[q_z]. \tag{A.8}$$

For the space wave, the propagator $\mathbf{P}_s$ for $q_z \in [1, N_z-2]$ is applied. On the other hand, the surface waves $W_x[0]$ and $W_x[N_z-1]$ are propagated with their own propagators. These propagators are defined as follows:

- Since for the space waves $w_x[0]=0$, the propagator is the same as the one for a PEC condition DSSF introduced in Section 1.2.2.e and its diagonal elements are given by

$$\mathbf{P}_s[q_z, q_z] = \exp\left(-j\Delta x\left(\sqrt[\circ]{k_0^2 - k_z^2} - k_0\right)\right), \tag{A.9}$$

  with $\sqrt[\circ]{\cdot}$ defined in (1.37) and

$$k_z = \frac{2}{\Delta z}\sin\left(\frac{\pi q_z}{2N_z}\right). \tag{A.10}$$

- For the surface waves, we need to propagate the terms $W_x[0]r_0^{p_z}$ and $W_x[N_z](-1/r_0)^{p_z}$ that satisfy (A.7). By inserting these coefficients in (A.7) and solving the forward propagation equation, the propagators are defined by

$$\begin{aligned}
\mathbf{P}_s[0,0] &= \exp\left(-j\Delta x\left(\sqrt{k_0^2 + \frac{r_0 + r_0^{-1} - 2}{\Delta z^2}} - k_0\right)\right), \\
\mathbf{P}_s[N_z-1, N_z-1] &= \exp\left(-j\Delta x\left(\sqrt{k_0^2 + \frac{(-r_0) + (-r_0)^{-1} - 2}{\Delta z^2}} - k_0\right)\right).
\end{aligned} \tag{A.11}$$

A.3 Inverse spectral transform

By applying an inverse sine transform to $W_{x+\Delta x}$, the free-space propagated field $w^{fs}_{x+\Delta x}$ is obtained. An inverse DMFT allows to retrieve $u^{fs}_{x+\Delta x}$ from $w^{fs}_{x+\Delta x}$, $W_{x+\Delta x}[0]$ and $W_{x+\Delta x}[N_z]$.

The free-space propagated field is given by

$$u^{fs}_{x+\Delta x}[p_z] = \hat{u}_{x+\Delta x}[p_z] + B_1 r_0^{p_z} + B_2(-1/r_0)^{p_z}, \tag{A.12}$$

with $p_z \in [0, N_z - 1]$, and $\hat{u}_{x+\Delta x}[p_z]$ a particular solution of (A.1). The two other terms correspond to the general solutions of (A.1), with the constant B_1 and B_2 given by

$$\begin{aligned} B_1 &= W_{x+\Delta x}[0] - Q \sum_{p_z=0}^{N_z-1\ \prime} \hat{u}_{x+\Delta x}[p_z] r_0^{p_z}, \\ B_2 &= W_{x+\Delta x}[N_z - 1] - Q \sum_{p_z=0}^{N_z-1\ \prime} \hat{u}_{x+\Delta x}[p_z](-r_0)^{p_z}. \end{aligned} \tag{A.13}$$

The particular solution $\hat{u}$ can be obtained by setting $\hat{u}_{x+\Delta x}[0] = 0$ and $\hat{u}_{x+\Delta x}[N_z - 1] = 0$. Then a double-pass method is applied to solve the resulting system of equations. For $p_z \in [1, N_z - 2]$, with $\eta[0] = 0$ the recursion

$$\eta[p_z] - r_0 \eta[p_z - 1] = 2 w_{x+\Delta x}[p_z] \Delta z, \tag{A.14}$$

gives the first pass . The second pass is computed with the backward recursion

$$\hat{u}_{x+\Delta x}[p_z + 1] + 1/r_0 \hat{u}_{x+\Delta x}[p_z] = \eta[p_z], \tag{A.15}$$

for $p_z \in [N_z - 2, 0]$, with $\hat{u}_{x+\Delta x}[N_z - 1] = 0$.

Following these steps, $u^{fs}_{x+\Delta x}$ is retrieved.

Appendix B

3D DMFT for a Transversely Invariant Ground Condition

In this chapter, the DMFT method to propagate the field above a dielectric ground with DSSF in Cartesian coordinates is introduced. For the method to be valid, a ground which parameters are invariant along y is assumed.

B.1 Numerical scheme

In this section, the numerical scheme to propagate the field u_x from x to $x+\Delta x$ is defined.

The method follows these steps:

1. A DFT along y is applied on the field u_x. Therefore, $\tilde{U}_x[q_y, p_z]$ with $q_y \in [0, N_y - 1]$ and $p_z \in [0, N_z - 1]$ is obtained.
2. The DMFT change of variable is performed along z to obtain the space and surface waves:
   (a) w_x is computed according to
   $$w_x[q_y, p_z] = \frac{\tilde{U}_x[q_y, p_z + 1] - \tilde{U}_x[q_y, p_z - 1]}{2\Delta z} + \alpha \tilde{U}_x[q_y, p_z], \tag{B.1}$$
   for $p_z \in [1, N_z - 2]$.
   (b) W_x taking into account the space and surface waves is obtained from
   $$\begin{aligned} W_x[q_y, 0] &= Q \sum_{p_z=0}^{N_z-1\ \prime} r_0^{p_z} \tilde{U}_x[q_y, p_z], \\ W_x[q_y, 0] &= \sum_{p_z=1}^{N_z-2} \sin\left(\frac{\pi q_z p_z}{N_z}\right) w_x[q_y, p_z], \\ W_x[q_y, N_z - 1] &= Q \sum_{p_z=0}^{N_z-1\ \prime} (-r_0)^{p_z} \tilde{U}_x[q_y, p_z], \end{aligned} \tag{B.2}$$
   with r_0 defined by equation (A.3) in vertical polarisation, (A.4) in horizontal polarisation and Q defined in Appendix A.

3. For $q_z \in [1, N_z - 2]$, using the diagonal propagator introduced in Section 1.3.2.b W_x is propagated. Furthermore $W_x[q_y, 0]$ and $W_x[q_y, N_z - 1]$ are propagated with their own surface-wave propagators defined in Section B.2.
4. $\tilde{U}_{x+\Delta x}$ is computed with the inverse DMFT algorithm as introduced in Appendix A.
5. $u^{fs}_{x+\Delta x}$ is finally obtained by applying an inverse Fourier transform along y.

Note that, since the change of variable is linear and α does not depend on y, the DFT along y can be performed after the change of variable.

B.2 Surface wave propagators

In this section, assuming r_0 is invariant along y, the propagators for the surface waves $W_x[q_y, 0]r_0^{p_z}$ and $W_x[q_y, N_z - 1](-1/r_0)^{p_z}$ are obtained.

With the above hypothesis α is constant over the ground for a position x.

The discretized version along y and z of the wave equation assuming $n = 1$ is given by

$$\frac{\partial^2 u_x}{\partial x^2} - 2jk_0\frac{\partial u_x}{\partial x} + \mathbf{d}_y^2 u_x + \mathbf{d}_z^2 u_x + k_0^2 u_x = 0, \tag{B.3}$$

with $\mathbf{d}_y^2$ and $\mathbf{d}_z^2$ defined in Section 1.3.2.b.

First, we use a discrete Fourier transform along y on (B.3), leading to

$$\frac{\partial^2 \tilde{U}_x}{\partial x^2} - 2jk_0\frac{\partial \tilde{U}_x}{\partial x} - k_y^2\tilde{U}_x + \mathbf{d}_z^2\tilde{U}_x + k_0^2\tilde{U}_x = 0, \tag{B.4}$$

where k_y is given in Section 1.3.2.b.

Second, the surfaces waves are solution of (B.4). This leads to

$$\begin{aligned}
&\frac{\partial^2 W_x[q_y, 0]}{\partial x^2} - k_y^2 W_x[q_y, 0] + \frac{r_0 + r_0^{-1} - 2}{\Delta z^2} W_x[q_y, 0] + k_0^2 W_x[q_y, 0] = 0,\\
&\frac{\partial^2 W_x[q_y, N_z - 1]}{\partial x^2} - k_y^2 W_x[q_y, N_z - 1]\\
&+ \frac{-r_0 + (-r_0)^{-1} - 2}{\Delta z^2} W_x[q_y, N_z - 1] + k_0^2 W_x[q_y, N_z - 1] = 0.
\end{aligned} \tag{B.5}$$

Assuming only a forward propagation and factorizing with a PWE formalism, the propagators are given by

$$\begin{aligned}
\mathbf{P}_s[q_y, 0] &= \exp\left(-j\Delta x\left(\sqrt{k_0^2 - k_y^2 + \frac{r_0 + r_0^{-1} - 2}{\Delta z^2}} - k_0\right)\right),\\
\mathbf{P}_s[q_y, N_z - 1] &= \exp\left(-j\Delta x\left(\sqrt{k_0^2 - k_y^2 + \frac{(-r_0) + (-r_0)^{-1} - 2}{\Delta z^2}} - k_0\right)\right).
\end{aligned} \tag{B.6}$$

Therefore, the 3D DMFT has been defined and can be used to propagate over dielectric grounds.

Appendix C

Summary in French (Résumé en français)

C.1 Introduction

En radio-fréquence, la propagation des ondes électromagnétiques est un problème majeur. Ainsi, des méthodes de modélisation rapides et précises sont nécessaires. Celles-ci sont utiles pour de nombreuses applications concernant les systèmes de communications, surveillance, navigation et observation.

Dans ce contexte, plusieurs effets sont à prendre en compte. Tout d'abord dans la troposphère – couche la plus basse de l'atmosphère– la propagation est affectée par la réfraction, les turbulences, la diffusion et l'absorption. Ces deux dernières induisent une atténuation du champ quand la réfraction implique des phénomènes de courbure de rayons ou de conduits [2, 3]. Ces effets sont caractérisés avec l'indice de réfraction [7, 8].

Pour des applications de communication satellite-sol, comme en GNSS, l'ionosphère joue également un rôle important. En effet, cette couche de l'atmosphère induit des phénomènes de scintillation.

Enfin, les effets du terrain sont aussi à prendre en compte pour avoir un modèle précis de la propagation sur de longues distances. L'onde de sol, la réflexion au niveau du sol, les interférences doivent être pris en compte de façon efficace [11, 14, 15].

Dans ce contexte, l'objectif de cette thèse est de développer une méthode rapide et précise pour modéliser la propagation troposphérique en 3D. Il est nécessaire pour que cette méthode soit utilisable qu'elle soit efficace à la fois en temps de calcul mais aussi en taille mémoire. Pour arriver à cet objectif, trois axes ont été étudiés:

- En se basant sur une méthode de propagation existante basée sur les ondelettes (split-step wavelet) [57], et en utilisant les propriétés des ondelettes, une méthode plus efficace en terme de taille mémoire a été développée en 2D. Le but est d'obtenir une méthode efficace à la fois en temps de calcul et en taille mémoire pour le passage à la 3D.
- Une formule théorique pour l'erreur de compression avec split-step wavelet a été obtenue. Celle-ci permet de connaître et gérer la précision de la méthode dans un

environnement donné.

- Enfin, la méthode de propagation en ondelettes a été généralisée à la 3D pour obtenir une méthode précise et efficace en temps de calcul et taille mémoire sur des scénarios complexes.

C.2 Formulation auto-cohérente de split-step Fourier

Les modèles basés sur l'équation parabolique (EP) sont très utilisés dans ce contexte [11]. En effet, valides autour d'une direction paraxiale et en négligeant la propagation arrière, ils permettent de modéliser la propagation sur de longues distances dans des environnements complexes.

La méthode Split-step Fourier (SSF) [3, 30, 63] permet de résoudre itérativement l'EP en faisant des allers-retours entre le domaine spatial et le domaine spectral. Cela permet de faire de grands pas dans la direction de propagation. Elle est donc particulièrement efficace pour la propagation longue distance en radio-fréquence. Les effets de la réfraction sont pris en compte avec un écran de phase quand l'effet de la composition du sol passe par le choix de la transformée spectrale en sinus, cosinus, ou transformée mixte (Discrete Mixed Fourier Transform, DMFT) [71, 72]. Le relief peut également être pris en compte avec différentes méthodes.

Cependant, il a récemment été montré que SSF-DMFT n'est pas une méthode auto-cohérente au sens de Chew. En discrétisant la propagation *a posteriori* des erreurs résiduelles peuvent apparaitre. Une version discrète DSSF-DMFT [52] a donc été introduite où les équations sont discrétisées *a priori* et résolues ensuite.

Dans cette section, la méthode DSSF-DMFT en 2D est rappelée. Ensuite, la méthode est étendue à un domaine 3D cartésien.

C.2.1 DSSF 2D

3.2.1.a Configuration

Nous nous plaçons en coordonnées cartésiennes et supposons que la propagation est invariante selon y. La direction de propagation est x. Le but est de calculer la propagation itérativement selon x.

L'atmosphère est considérée inhomogène et l'indice de réfraction est noté n. Pour tenir compte de la courbure de la Terre, l'indice modifié m est utilisé. Celui-ci étant proche de 1, la réfractivité modifiée M est préférée.

Pour des raisons numériques, le domaine est de taille fini $x \in [0, x_{\max}]$ et $z \in [0, z_{\max}]$ et discrétisé comme suit

$$\begin{aligned} x_{p_x} &= p_x \Delta x \qquad \text{pour } p_x \in \{0, \cdots, N_x - 1\}, \\ z_{p_z} &= p_z \Delta z \qquad \text{pour } p_z \in \{0, \cdots, N_z - 1\}. \end{aligned} \tag{C.1}$$

La configuration est décrite en Figure C.1. La version discrète d'un champ $u(x, z)$ selon z est notée $u_x[p_z]$.

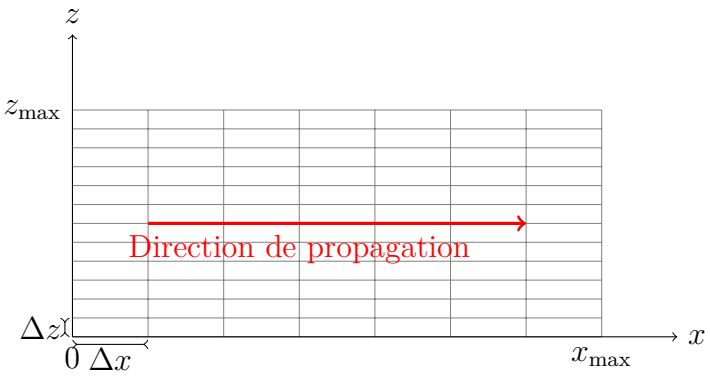


Figure C.1: Configuration discrète.

Enfin, une fenêtre d'apodisation de Hanning est appliquée en haut du domaine pour éviter les réflexions parasites. L'opérateur associé est noté **H**.

3.2.1.b La méthode 2D DSSF

Formulation continue Dans la configuration définie en polarisation transverse magnétique (TM), seule la composante y du champ magnétique est non nulle, quand en transverse électrique (TE) seule la composante y du champ électrique est non nulle. Nous définissons

$$\psi(x,z) = \begin{cases} E_y(x,z) \text{ en TE}, \\ H_y(x,z) \text{ en TM}, \end{cases} \tag{C.2}$$

pour travailler sur une seule variable. Celle-ci est solution de l'équation de Helmholtz

$$\nabla^2\psi(x,z) + k^2\psi(x,z) = 0, \tag{C.3}$$

avec $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2}$ et k le nombre d'onde.

Nous définissons le champ réduit

$$u = \exp(jk_0x)\psi(x,z), \tag{C.4}$$

à variation de phase lente selon x. Cette caractéristique permettra d'augmenter le pas de calcul horizontal Δx.

Ce dernier est solution de l'équation continue

$$\frac{\partial^2 u(x,z)}{\partial x^2} - 2jk_0\frac{\partial u(x,z)}{\partial x} + \frac{\partial^2 u(x,z)}{\partial z^2} + k_0^2(n(x,z)^2 - 1)u(x,z) = 0. \tag{C.5}$$

Les conditions au niveau du sol sont données par

$$\begin{aligned} u|_{z=0} &= 0 \text{ pour un PEC}, \\ \frac{\partial u}{\partial z}|_{z=0} &= 0 \text{ pour un PMC}, \\ \frac{\partial u}{\partial z}|_{z=0} + \alpha u|_{z=0} &= 0 \text{ pour un sol diélectrique}, \end{aligned} \tag{C.6}$$

avec

$$\alpha = jk_0\cos(\theta_i)\frac{1-R}{1+R}, \tag{C.7}$$

où θ_i est l'angle d'incidence et R le coefficient de réflexion.

Formulation discrète Le problème est discrétisé selon l'axe z pour obtenir

$$\frac{\partial^2 u_x[p_z]}{\partial x^2} - 2jk_0\frac{\partial u_x[p_z]}{\partial x} + \mathbf{d}_z^2 u_x[p_z] + k_0^2(n_x[p_z])^2 - 1)u_x[p_z] = 0, \tag{C.8}$$

avec $\mathbf{d}_z^2$ l'approximation aux différences finies de l'opérateur dérivée seconde. Les conditions à la limite au sol deviennent

$$\begin{aligned} u_x[0] &= 0 \text{ pour un PEC,} \\ \mathbf{d}_z u_x[0] &= 0 \text{ pour un PMC,} \\ \mathbf{d}_z u_x[0] + \alpha u_x[0] &= 0 \text{ pour un sol diélectrique.} \end{aligned} \tag{C.9}$$

En factorisant en propagations avant et arrière, en négligeant la propagation arrière, et en faisant une approximation grand-angle ce problème peut s'écrire dans le formalisme PE [11] comme suit

$$\frac{\partial u_x[p_z]}{\partial x} = -j\left(\sqrt{k_0 + \mathbf{d}_z^2} - k_0\right) u_x[p_z] - jk_0\left(n_x[p_z] - 1\right) u_x[p_z]. \tag{C.10}$$

Dans ce cas la propagation peut être résolue itérativement par méthode split-step comme suit :

$$u_{x+\Delta x}[p_z] = \mathbf{R}\mathbf{D}u_x[p_z], \tag{C.11}$$

avec

$$\mathbf{D} = \exp\left(-jk_0\left(\sqrt{1 + \mathbf{d}_z^2/k_0^2} - 1\right)\right), \tag{C.12}$$

et $\mathbf{R}$ un opérateur diagonal dont les éléments diagonaux sont donnés par

$$\mathbf{R}[p_z, p_z] = \exp\left(-jk_0\left(n_x[p_z] - 1\right)\right). \tag{C.13}$$

Ainsi la propagation est séparée en un terme de propagation en espace libre $\mathbf{D}$ et un terme de prise en compte de la réfraction par écran de phase $\mathbf{R}$.

Transformée spectrale Cependant le terme $\mathbf{D}$ est difficile à calculer. Ce terme de propagation est donc calculé dans le domaine spectral. En supposant $n = 1$ et avec un PEC au niveau du sol, la transformée en sinus, notée $\mathbf{T}_s$ permet de diagonaliser $\mathbf{d}_z^2$, tel que

$$\mathbf{d}_z^2 u = -k_z^2 u, \tag{C.14}$$

avec

$$k_z = \frac{2}{\Delta z}\sin\left(\frac{\pi q_z}{2N_z}\right). \tag{C.15}$$

Le terme de propagation en espace libre de l'équation (C.10) devient dans le domaine spectral

$$\frac{\partial U_x[p_z]}{\partial x} = -j\left(\sqrt[\circ]{k_0 - k_z^2} - k_0\right) U_x[p_z], \tag{C.16}$$

avec U_x la transformée en sinus de u_x et

$$\sqrt[\circ]{k_0^2 - k_z^2} = \begin{cases} \sqrt{k_0^2 - k_z^2} & \text{si} \quad |k_z| \leq k_0, \\ -j\sqrt{k_z^2 - k_0^2} & \text{si} \quad |k_z| > k_0. \end{cases} \tag{C.17}$$

La propagation en espace libre dans le domaine spectral est alors donnée par

$$U_{x+\Delta x} = \mathbf{P}_s U_x, \tag{C.18}$$

avec $\mathbf{P}_s$ un opérateur diagonal dont les éléments diagonaux sont donnés par

$$\mathbf{P}_s[q_z, q_z] = \exp\left(-j\Delta x\left(\sqrt[\circ]{k_0^2 - k_z^2} - k_0\right)\right). \tag{C.19}$$

Sol impédant Dans le cas d'un sol impédant, la discrete mixed Fourier transform (DMFT) est utilisée en lieu et place de la transformée en sinus [71, 72].

Relief Pour modéliser le relief, différentes méthodes sont possibles [11,14,15]. Pour des raisons de simplicité nous choisissons la modélisation sous forme de marches d'escalier [11]. L'opérateur associé est noté $\mathbf{L}$.

Schéma de simulation avec DSSF 2D Ainsi la propagation entre x et $x + \Delta x$ au dessus d'un sol PEC est obtenu comme suit

$$u_{x+\Delta x} = \mathbf{HRLT}_s^{-1}\mathbf{P}_s\mathbf{T}_s u_x. \tag{C.20}$$

Ce schéma est répété jusqu'à obtenir $u_{x_{\max}}$.

3.2.1.c Test numérique

Dans ce test, nous simulons la propagation d'un point source complexe (CSP) au dessus d'un sol impédant en présence d'un conduit atmosphérique et de deux reliefs triangulaires. La fréquence est 300 MHz. La source est placée en $x_s = -50$ m et $z_s = 50$ m avec une largeur $W_0 = 5$ m. La propagation est calculée sur 100 km en x et 2048 m en z avec des pas de 200 m et 0.5 m. Le conduit est modélisé par un indice de réfraction variant tri-linéairement selon z.

Le résultat pour le champ électrique est représenté en Figure C.2.

Maintenant que la méthode 2D a été présentée, nous pouvons l'étendre à la 3D en coordonnées cartésiennes.

C.2.2 DSSF 3D

Dans cette section, nous présentons DSSF 3D en coordonnées cartésiennes. Nous supposons qu'il n'y a pas de relief. En effet, prendre en compte le relief en 3D est un problème à part entière qui n'est pas traité dans cette thèse.

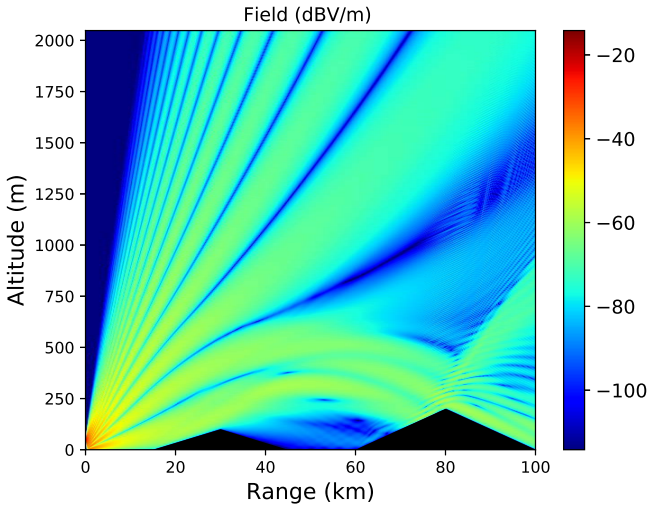


Figure C.2: Champ électrique (dBV/m) calculé avec DSSF.

3.2.2.a Configuration

En 3D nous nous plaçons en coordonnés cartésiennes avec x la direction de propagation. Nous supposons dans un premier temps que n est constant. Le domaine est de taille finie avec $x \in [0, x_{\max}]$, $y \in [-y_{\max}/2, y_{\max}/2]$ et $z \in [0, z_{\max}]$. Enfin une discrétisation selon les 3 directions est appliquée comme suit

$$\begin{aligned} x_{p_x} &= p_x \Delta x, \text{ pour } p_x \in 0, \cdots, N_x - 1, \\ y_{p_y} &= p_y \Delta y, \text{ pour } p_y \in -N_y/2, \cdots, N_y/2, \\ z_{p_z} &= p_z \Delta z, \text{ pour } p_z \in 0, \cdots, N_z - 1, \end{aligned} \tag{C.21}$$

avec $N_x = x_{\max}/\Delta x$, $N_y = y_{\max}/\Delta y$ et $N_z = z_{\max}/\Delta z$ le nombre de points dans chaque direction.

Pour éliminer les réflexions parasites en haut et sur les limites latérales du domaine des apodisations avec des fenêtres de Hanning sont appliquées. L'opérateur est noté **H**.

3.2.2.b DSSF en 3D

Formulation continue Pour la formulation continue, nous travaillons ici avec les potentiels de Hertz car le champ peut être décomposé en une combinaison de composante TE et TM. Nous travaillons ici en TE et dans ce cas avec la variable ψ. Pour la propagation le champ réduit u est défini comme en 2D

$$u(x, y, z) = \exp(jk_0 x)\psi(x, y, z), \tag{C.22}$$

pour avoir des variations plus lentes selon x. Ce dernier est solution de

$$\frac{\partial^2 u(x,y,z)}{\partial x^2} - 2jk_0 \frac{\partial u(x,y,z)}{\partial x} + \frac{\partial^2 u(x,y,z)}{\partial y^2} + \frac{\partial^2 u(x,y,z)}{\partial z^2} + k_0^2(n^2-1)u(x,y,z) = 0. \tag{C.23}$$

Enfin pour obtenir la formulation complète du problème, les conditions au niveau du sol sont données par

$$\begin{aligned} u|_{z=0} &= 0 \text{ pour un PEC,} \\ \frac{\partial u}{\partial z}|_{z=0} &= 0 \text{ pour un PMC,} \\ \frac{\partial u}{\partial z}|_{z=0} + \alpha u|_{z=0} &= 0 \text{ pour un sol diélectrique.} \end{aligned} \tag{C.24}$$

Formulation discrète Nous discrétisons maintenant selon y et z pour obtenir la formulation discrète suivante

$$\frac{\partial^2 u_x[p_y,p_z]}{\partial x^2} - 2jk_0\frac{\partial u_x[p_y,p_z]}{\partial x} + \mathbf{d}_y^2 u_x[p_y,p_z] + \mathbf{d}_z^2 u_x[p_y,p_z] + k_0^2(n^2-1)u_x[p_y,p_z] = 0, \tag{C.25}$$

avec les conditions aux limites suivantes

$$\begin{aligned} u_x[p_y,0] &= 0 \text{ pour un PEC,} \\ \mathbf{d}_z u_x[p_y,0] &= 0 \text{ pour un PMC,} \\ \mathbf{d}_z u_x[p_y,0] + \alpha u_x[p_y,0] &= 0 \text{ pour un sol diélectrique.} \end{aligned} \tag{C.26}$$

Dans ce cas avec le formalisme PE en ne considérant que la propagation vers l'avant, l'équation à résoudre est

$$\frac{\partial u_x[p_z]}{\partial x} = -j\left(\sqrt{k_0 + \mathbf{d}_y^2 + \mathbf{d}_z^2} - k_0\right) u_x[p_z] - jk_0\left(n_x[p_z] - 1\right) u_x[p_z]. \tag{C.27}$$

Transformée spectrale Comme en 2D, la propagation est séparée en propagation espace libre $n = 1$ et prise en compte de la réfraction avec un écran de phase. Pour la partie en espace libre, en supposant une limite PEC en bas du domaine et périodique sur les bords du domaine, on diagonalise les opérateurs $\mathbf{d}_y^2$ et $\mathbf{d}_z^2$ avec une transformée de Fourier et en sinus respectivement. La transformation spectrale est notée $\mathbf{T}_s$. On obtient alors

$$\begin{aligned} \mathbf{d}_y^2 u &= -k_y^2 u, \\ \mathbf{d}_z^2 u &= -k_z^2 u, \end{aligned} \tag{C.28}$$

avec

$$\begin{aligned} k_y &= \frac{2}{N_y}\sin\left(\frac{\pi q_y}{N_y}\right), \\ k_z &= \frac{2}{N_z}\sin\left(\frac{\pi q_z}{2N_z}\right). \end{aligned} \tag{C.29}$$

Dans le domaine spectral, l'équation de propagation devient alors

$$\frac{\partial \tilde{u}_x[p_z]}{\partial x} = -j\left(\sqrt[\circ]{k_0 - k_y^2 - kz^2} - k_0\right)\tilde{u}_x[p_z], \tag{C.30}$$

avec $\tilde{u}_x$ la transformée spectrale de u_x et

$$\sqrt[\circ]{k_0^2 - k_y^2 - k_z^2} = \begin{cases} \sqrt{k_0^2 - k_y^2 - k_z^2} & \text{si} \quad |k_z| \leq k_0, \\ -j\sqrt{k_y^2 + k_z^2 - k_0^2} & \text{si} \quad |k_z| > k_0. \end{cases} \tag{C.31}$$

Ainsi, dans le domaine spectral, la propagation entre x et $x + \Delta x$ est donnée par

$$\tilde{u}_{x+\Delta x} = \mathbf{P}_s \tilde{u}_x, \tag{C.32}$$

avec $\mathbf{P}_s$ un opérateur dont les éléments sont donnés par

$$\mathbf{P}_s[q_y, q_z] = \exp\left(-j\Delta x \left(\sqrt[\circ]{k_0^2 - k_y^2 - k_z^2} - k_0\right)\right). \tag{C.33}$$

Introduction de la réfraction Pour prendre en compte les effets de l'atmosphère, la méthode de l'écran de phase est utilisée avec l'opérateur diagonal $\mathbf{R}$. Les éléments de ce dernier sont donnés par

$$\mathbf{R}[p_y, p_z] = \exp\left(-jk_0 \left(n_x[p_y, p_z] - 1\right) \Delta x\right). \tag{C.34}$$

Sol impédant Dans le cas d'un sol impédant invariant dans la direction transverse y, la DMFT est utilisée [71, 72].

Schéma de propagation En conclusion, la propagation sur un pas entre x et $x + \Delta x$ est donnée par

$$u_{x+\Delta x} = \mathbf{R}\mathbf{T}_s^{-1}\mathbf{P}\mathbf{T}_s u_x. \tag{C.35}$$

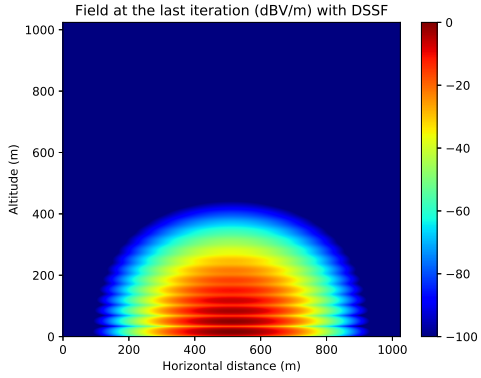
3.2.2.c Tests numériques

L'objectif du test numérique est de valider la méthode 3D DSSF pour la propagation au-dessus d'un sol diélectrique. La source utilisée est un CSP. Dans ce cas, en champ lointain, on peut utiliser la formule analytique du champ d'un point source complexe et les réflexions sont prises en compte avec le coefficient de Fresnel.

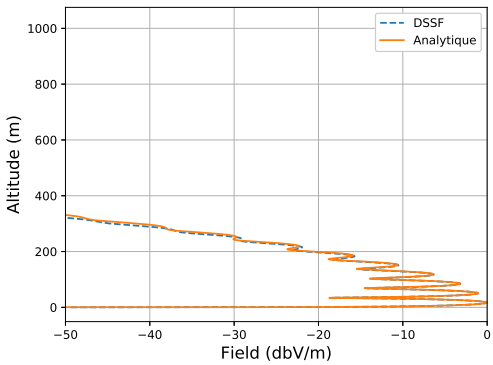
Pour cette simulation la fréquence est de 300 MHz. Le domaine est de taille 2 km selon x et 1024×1024 en y et z. Les pas sont choisis à $\Delta x = 100$ m et $\Delta y = \Delta z = 0.5$ m. La source est placée en $x_s = -50$ m, $y_s = 512$ m et $z_s = 0$ m. Sa largeur est $W_0 = 5$ m. Les paramètres du sol sont $\epsilon_r = 20.0$ et $\sigma = 0.02$ S/m.

Les résultats sont données en Figure C.3. En (a) une coupe sur le plan $y0z$ du dernier champ normalisé obtenu avec DSSF est donnée. Les coupes en $N_y/2$ des champs normalisés obtenus avec DSSF et la méthode analytique à la dernière itération sont données en figure (b). Enfin, en (c) la différence normalisée à la dernière itération entre les champs obtenus avec les deux méthodes est tracée dans le plan $y0z$.

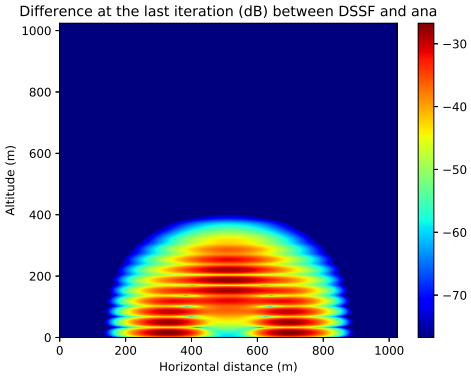
La différence normalisée à la dernière itération est au maximum de -27 dB. De plus les extremums dus aux réflexions sur le sol sont bien représentés. La méthode est donc validée. Cependant dans ce cas la méthode nécessite 600 Mo pour la sauvegarde de l'opérateur $\mathbf{P}_s$. De plus elle nécessite sur ce cas environ 10 min de temps de calcul.



(a) Champ normalisé obtenu avec DSSF dans le plan yOz (dB).



(b) Coupe vertical à la dernière itération en $N_y/2$ des champs normalisés obtenus avec les deux méthodes (dB).



(c) Différence normalisée à la dernière itération (dB).

Figure C.3: Champs finaux normalisés obtenus avec DSSF et la méthode analytique.

C.2.3 Conclusion

En conclusion, dans cette section la méthode auto-cohérente DSSF en 2D a été rappelée. Ensuite, nous avons étendu cette méthode à la 3D en coordonnées cartésiennes.

Des tests numériques ont permis de montrer que la méthode donne des résultats avec une bonne précision. Cependant les temps de calcul et la taille mémoire du propagateur en 3D limitent l'utilisation de cette méthode dans le cas de la propagation sur de longues distances dans un environnement complexe.

Cette dernière conclusion motive l'introduction d'une méthode plus efficace en temps de calcul et taille mémoire pour la 3D.

C.3 Split-step wavelet en 2D

Dans ce chapitre une méthode alternative à DSSF en 2D, dénommée split-step wavelet (SSW), est introduite. Elle est basée sur la transformée discrète en ondelettes 1D [94]. En effet les ondelettes ont deux avantages. Tout d'abord la transformée en ondelettes rapide (fast wavelet transform, FWT) est de complexité plus faible que la transformée de Fourier rapide (fast Fourier transform, FFT). Ceci permettrait donc de gagner du temps. De plus, la représentation en ondelettes est en général parcimonieuse, permettant de gagner en taille mémoire et en temps de calcul. Ainsi, une méthode de propagation plus efficace en temps de calcul et taille mémoire est introduite en 2D pour ensuite être généralisée à la 3D.

La configuration est la discrétisation est la même qu'en Section 3.2.1.a.

C.3.1 Principe générale de la méthode

La méthode SSW est basée sur le même principe que DSSF. Le champ est calculé itérativement de plus en plus loin de la source. Ainsi le schéma de propagation entre x et $x + \Delta x$ est le suivant

1. Le vecteur creux de coefficients d'ondelettes de u_x est obtenu avec une transformée en ondelettes ($\mathbf{W}$) et une compression de seuil V_s ($\mathbf{C}$)

$$U_x = \mathbf{CW}u_x. \tag{C.36}$$

2. Les coefficients d'ondelettes propagés $U_{x+\Delta x}$ sont obtenus à l'aide de du propagateur d'ondelettes-à-ondelettes P

$$U_{x+\Delta x} = PU_x. \tag{C.37}$$

Cette opérateur correspond à la propagation en espace libre des ondelettes. Deux méthodes sont possibles pour cette propagation. Une première en utilisant une matrice de propagation $\mathbf{P}$ dans laquelle toutes les propagations d'ondelettes-à-ondelettes sont stockées. Dans ce cas la propagation est un produit matrice-vecteur et la méthode est notée mSSW [57]. Une seconde méthode avec une bibliothèque d'opérateurs P, où seules les propagations essentielles sont stockées et utilisées pour

propager chaque coefficient non nul de U_x. Dans ce cas la propagation se fait pour chaque coefficient et la méthode est notée lSSW [107].

3. La propagation en espace libre du champ est obtenue avec une transformée en ondelettes inverse
$$u^{\text{fs}}_{x+\Delta x} = \mathbf{W}^{-1} U_{x+\Delta x}. \tag{C.38}$$

4. La réfraction et le relief sont pris en compte comme avec DSSF et une fenêtre d'apodisation est appliquée en haut du domaine pour obtenir le champ propagé
$$u_{x+\Delta x} = \mathbf{HRL} u^{\text{fs}}_{x+\Delta x}. \tag{C.39}$$

Pour la prise en compte du sol la méthode des images locale a été mise au point par Zhou et al. [57]. Celle-ci permet de prendre en compte le sol sur un petit nombre de points en utilisant le caractère local des ondelettes.

C.3.2 Propagation en espace libre

Pour construire la matrice ou la bibliothèque, deux propriétés des ondelettes sont utilisées: l'invariance par translation et la taille réduite du support des ondelettes (notée N_l^{p}). Le niveau 0 correspondant à la fonction d'échelle est traitée comme le niveau L.

Tout d'abord une ondelette centrée $\chi_{l',0}[p_z]$ de chaque niveau $l' \in [1, L]$ est propagée avec DSSF pour obtenir $\chi_{l',\Delta x}[p_z]$. La propagation par DSSF se fait sur le support des ondelettes N_l^{p}, car $N_l^{\text{p}} \ll N_z$. Ensuite, en utilisant la propriété d'invariance par translation des ondelettes, les ondelettes sont translatées de $p_t \in [0, 2^{L-l}[$ selon z afin d'obtenir les propagations nécessaires $\chi_{l',\Delta x}[p_z+p_t]$. Ces dernières sont décomposées en ondelettes ($\mathbf{W}$) et compressées avec un seuil V_p ($\mathbf{C}$), résultant sur la propagation d'ondelettes-à-ondelettes P_{l,p_t}.

Pour la matrice ces vecteurs d'ondelettes élémentaires sont stockés dans $\mathbf{P}_{(l,p)(l',p_t)}$ pour chaque l. Les autres éléments de la matrice sont obtenus en répliquant ces éléments. Ainsi une matrice de taille $N_z \times N_z$ est obtenue. La propagation s'effectue par un produit matrice-vecteur. Le calcul et la sauvegarde de cette matrice deviennent donc problématiques quand N_z augmente et cette méthode n'est donc pas adéquate pour la 3D.

Pour la bibliothèque seulement les éléments P_{l,p_t} sont stockés. Celle-ci correspond donc au sous-ensemble minimale de la matrice nécessaire à la propagation. La taille mémoire est ainsi grandement diminuée, rendant la méthode généralisable à la 3D. La propagation s'effectue en sommant les propagations élémentaires associées à chaque coefficient d'ondelette non nul du champ.

Cette méthode permet en utilisant les avantages des ondelettes d'être théoriquement plus rapide que DSSF et de limiter les besoins en taille mémoire.

C.3.3 Borne de l'erreur de compression

Comme nous l'avons vu deux paramètres de plus sont nécessaires avec SSW comparé à DSSF : les seuils de compression V_s et V_p. Ces derniers permettent d'avoir une bonne

représentation parcimonieuse. Cependant ces compressions introduisent une erreur qui s'accumule avec les itérations et dont la valeur dépend des paramètres de compression.

Une formule théorique a donc été calculée pour prévoir évolution de l'erreur de compression avec ces seuils et avec le nombre d'itérations. Ceci afin de maitriser l'erreur finale maximale en choisissant de bons seuils de compression.

Nous définissons l'erreur comme

$$\delta_{N_x} = \frac{\|\tilde{U}_{N_x} - U_{N_x}\|_2}{\|U_0\|_2}, \tag{C.40}$$

où U_{N_x} correspond aux coefficients d'ondelettes propagés sans compression à l'itération N_x, $\tilde{U}_{N_x}$ à ceux propagés avec compression, et U_0 aux coefficients du champ initial.

Dans ce cas nous avons montré que l'erreur évolue selon

$$\delta_{N_x} = (v_s + v_p)N_x, \tag{C.41}$$

avec v_s et v_p les deux seuils normalisés tels que

$$V_s = v_s \max(|U_0|) \quad \text{et} \quad V_p = v_p \max(|\mathbf{P}|). \tag{C.42}$$

De plus comme les erreurs sur le champ et sur le propagateur sont indépendantes, pour une erreur donnée $\delta_{N_x}^{\max}$, les seuils sont calculés comme suit

$$V_s = \frac{\delta_{N_x}^{\max}}{2N_x} \max(|U_0|) \qquad \text{et} \quad V_p = \frac{\delta_{N_x}^{\max}}{2N_x} \max(|\mathbf{P}|). \tag{C.43}$$

Ainsi les seuils peuvent être choisis automatiquement pour une précision voulus sur un scénario donnés.

C.3.4 Tests numériques

Dans cette partie, des tests numériques pour valider la méthode et montrer son efficacité sont effectués.

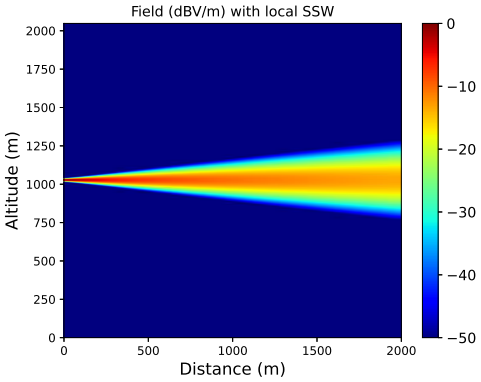
3.3.4.a Propagation en espace libre

Tout d'abord la méthode et la borne de l'erreur de compression sont validées sur un scénario de propagation du champ d'un CSP en espace libre. La fréquence est 300 MHz pour ce test. Le domaine est de taille $x_{\max} = 2$ km et $z_{\max} = 2048$ m avec des pas de 20 m et 0.5 m. La source est placée en $x_s = -50$ m et $z_s = 1024$ m avec une largeur de $W_0 = 5$ m.

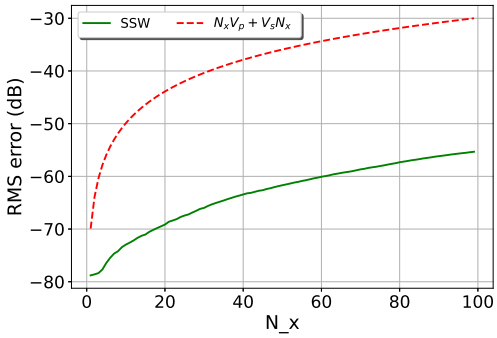
Pour les paramètres d'ondelettes, les symlet 6 avec $L = 3$ sont utilisées. Les seuils sont calculés pour une erreur de -30 dB à la dernière itération.

Dans ce cas le champ obtenu avec SSW est représenté en Figure C.4 (a). L'évolution de l'erreur RMS entre SSW et DSSF avec le nombre d'itérations est présentée en Figure (b).

Cette dernière montre que l'erreur est bien bornée par la formule théorique, comme attendu. De plus, l'erreur de compression est négligeable. La méthode SSW est donc validée dans ce cas.



(a) Champ obtenu avec SSW.



(b) Évolution de l'erreur RMS avec le nombre d'itération.

Figure C.4: Propagation d'un point source complexe en espace libre avec SSW.

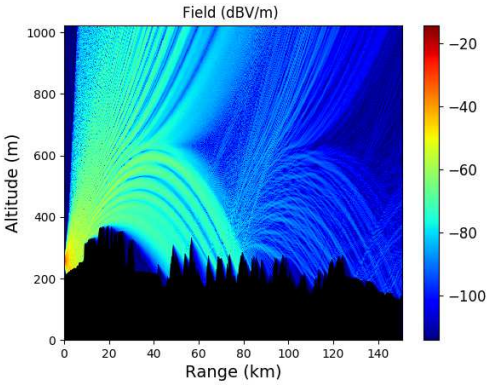
3.3.4.b Propagation en environnement complexe

Dans ce test nous comparons mSSW et lSSW pour montrer les avantages de cette dernière dans un environnement complexe.

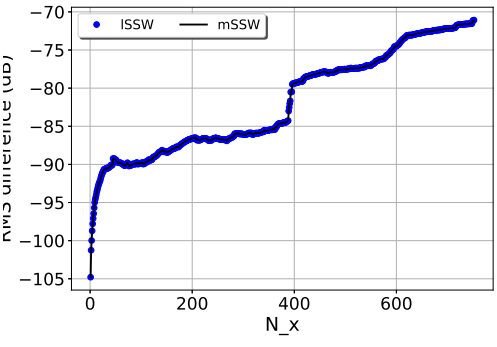
La fréquence pour ce test est 3 GHz. Nous souhaitons modéliser la propagation entre Pau et Toulouse. Le relief entre les deux villes est pris en compte. De plus, un conduit atmosphérique est présent. La source est un CSP placé en $x_s = -50$ m et $z_s = 50$ m au dessus du sol, de largeur $W_0 = 1$ m. Le domaine est de taille $x_{\max} = 150$ km et $z_{\max} = 1024$ m avec des pas de 20 m et 0.1 m.

Pour les ondelettes les mêmes paramètres sont utilisés et les seuils sont calculés pour une erreur attendue de -30 dB.

Les résultats de la Figure C.5 sont obtenus. En (a) le champ électrique obtenu avec lSSW est tracé. En (b), l'évolution de l'erreur RMS entre mSSW et DSSF, et lSSW et DSSF est présentée.



(a) Champ électrique (dBV/m) obtenu avec SSW.



(b) Évolution de l'erreur RMS avec lSSW et mSSW.

Figure C.5: Propagation d'un point source complexe entre Pau et Toulouse.

Tout d'abord, l'erreur à la dernière itération est bien en dessous de l'erreur attendue. De

plus, elle est négligeable. Enfin, comme attendu l'évolution de l'erreur avec mSSW et lSSW est la même.

En terme de temps de calcul et taille mémoire, les résultats sont présentés sur le tableau C.1.

Table C.1: Temps de calcul et taille mémoire nécessaire à mSSW et lSSW.

Méthode	mSSW	lSSW
Initialisation (s)	585	0.017
Propagation (s)	374	466
Total (s)	959	466
Taille mémoire du propagateur	759 Mo	42 ko

On peut conclure que lSSW permet de gagner fortement en taille mémoire et en temps de calcul pour le propagateur. De plus, même si la propagation est un peu plus longue, le temps total est bien inférieur avec lSSW.

C.3.5 Conclusion

Dans cette section, nous avons rappelé la méthode mSSW. Ensuite, nous avons introduit la méthode lSSW dans le but d'améliorer mSSW en temps de calcul et taille mémoire nécessaire.

Une formule pour calculer les seuils pour une erreur choisie a aussi été introduite, permettant de gérer la précision de la méthode.

Des tests numériques ont permis de valider SSW avec DSSF. De plus ces tests montrent que lSSW est beaucoup plus efficace que mSSW en terme de taille mémoire. Ainsi lSSW peut être généralisé à la 3D.

C.4 Split-step wavelet en 3D

Dans cette section, la méthode lSSW est généralisée à la 3D. Comme pour DSSF 3D, le relief et donc la dépolarisation ne sont pas pris en compte ici. Le but est dans un premier temps d'avoir une méthode efficace avant d'ajouter des phénomènes physiques.

La méthode est basée sur la transformée discrète en ondelettes séparables 2D [94]. Celle-ci a une complexité plus faible que la FFT et permet donc de gagner en temps de calcul. De plus, comme en 2D, les ondelettes permettent une représentation parcimonieuse permettant un gain important en taille mémoire.

C.4.1 Principe de la méthode

Le principe de la méthode est le même que pour lSSW 2D. Le schéma de propagation entre x et $x + \Delta x$ est rappelé

$$u_{x+\Delta x} = \mathbf{RLW}^{-1} P \mathbf{CW} u_x. \tag{C.44}$$

Comme en 2D, le sol est pris en compte avec la méthode des images locale. Celle-ci permet de prendre en compte les réflexions à l'aide d'une couche image très mince. Cette méthode est donc efficace en terme de temps de calcul.

Enfin, l'erreur de compression évolue de la même façon qu'en 2D et les seuils peuvent donc être choisis avec la formule (C.43).

C.4.2 Construction de la bibliothèque

Pour la construction de la bibliothèque, trois propriétés des bases d'ondelettes séparables 2D sont utilisées: l'invariance par translation, la taille réduite du support des ondelettes et l'invariance par rotation entre deux des ondelettes de la base. Cela permet de limiter le temps de calcul et la taille mémoire de la bibliothèque de propagateur.

La bibliothèque se construit de la même façon qu'en 2D, en 3 pas :

1. Pour chaque niveau l et orientation o, l'ondelette 2D $\chi_{l,o,0}[p_y, p_z]$ à la position $x = 0$ est propagée de Δx avec 3D DSSF sur son support réduit $N_{l_y}^{\mathrm{p}} \times N_{l_z}^{\mathrm{p}}$. Ainsi, $\chi_{l,o,\Delta x}[p_y, p_z]\, \forall (p_y, p_z) \in [0, N_y - 1] \times [0, N_z - 1]$ est obtenue.
2. En utilisant l'invariance par translation, toutes les ondelettes propagées de niveau l et orientation o sont translatées dans chaque direction de $p_{t_y}, p_{t_z} \in [0, 2^{L-l}[^2$, induisant $2^{2(L-l)}$ translations au total pour chaque niveau l et orientation o. Nous obtenons les ondelettes $\chi_{l,o,\Delta x}[p_y + p_{t_y}, p_z + p_{t_z}]\, \forall (p_y + p_{t_y}, p_z + p_{t_z}) \in [0, N_y - 1] \times [0, N_z - 1]$
3. Les ondelettes translatées $\chi_{l,o,\Delta x}[p_y + p_{t_y}, p_z + p_{t_z}]$ sont ensuite décomposées avec une FWT 2D et compressées (hard threshold V_p) afin d'obtenir les coefficients propagés qui sont stockés dans la bibliothèque $P_{l,o,(p_{t_y},p_{t_z})}$. Celle-ci contient donc $1 + 3 \times \sum_{l=1}^{L} 2^{2(L-l)}$ propagateurs parcimonieux.

Dans le cas où $\Delta y = \Delta z$, la propriété d'invariance par rotation entre les ondelettes horizontales et verticales peut être utilisée. Ainsi la propagation de l'une est déduite par la rotation de 90° de la propagation de l'autre, réduisant le nombre de DSSF nécessaires et donc le temps de calcul de la bibliothèque.

Pour la propagation, comme en 2D, chaque coefficient d'ondelettes non nul du champ est propagé avec son opérateur associé issu de la bibliothèque. Ces propagations élémentaires sont translatées et sommées pour obtenir la propagation totale.

Cette méthode permet grâce aux propriétés des ondelettes un gain fort en taille mémoire et en temps de calcul. Des tests de validation et de comparaison avec DSSF sont effectués dans la section suivante.

C.4.3 Tests numériques

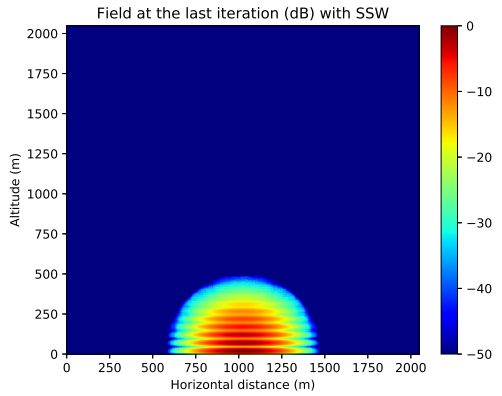
3.4.3.a Propagation au dessus d'un sol impédant

Tout d'abord la méthode lSSW-DMFT avec méthode des images locale est validée avec un test de propagation au-dessus d'un sol impédant.

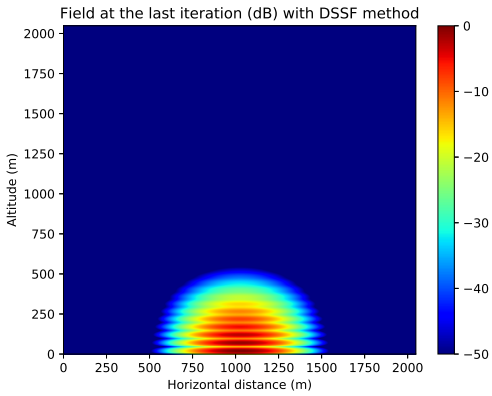
La configuration est la suivante : la fréquence est 300 MHz. La propagation est modélisée sur 5 km avec un pas de 200 m dans la direction x. Le domaine est de taille 2048×2048 m en y et z avec $\Delta y = \Delta z = 1$ m. La source est un CSP placé en $x_s = -50$ m, $y_s = 1024$ m et $z_s = 50$ m, dont la largeur est $W_0 = 7$ m. Les paramètres pour le sol sont $\epsilon_r = 20$ et $\sigma = 0.02$ S/m.

Pour les paramètres d'ondelettes, les symlets d'ordre 6 et un niveau maximum de décomposition $L = 3$ sont utilisés. De plus, les seuils sont choisis pour une erreur de -20 dB.

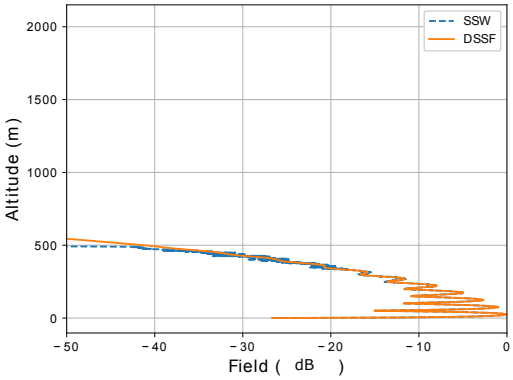
En Figures C.6 (a) et (b), les champs normalisés obtenus avec lSSW et DSSF sont présentés. Sur la figure (c), des coupes verticales en $N_y/2$ des champs finaux obtenus avec lSSW et DSSF sont montrées. Enfin, la différence normalisée entre lSSW et DSSF est présentée en Figure (d).



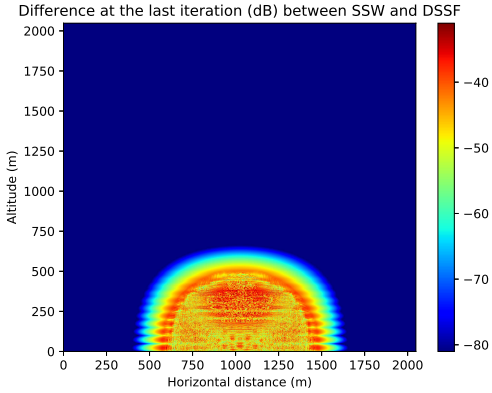
(a) Champ normalisé obtenu avec lSSW dans le plan yOz à la dernière itération.



(b) Champ normalisé obtenu avec DSSF dans le plan yOz à la dernière itération.



(c) Coupes verticales en $N_y/2$ des champs normalisés.



(d) Différence normalisée entre lSSW et DSSF dans le plan yOz à la dernière itération.

Figure C.6: Propagation du champ d'un CSP au dessus d'un sol impédant.

La différence à la dernière itération est de -32 dB. Comme prévu, celle-ci est négligeable. De plus, les extremums dus aux réflexions sur le sol sont bien modélisés. Ainsi, la méthode lSSW-DMFT avec les images locales est validée 3D.

En terme de taille mémoire, le propagateur de DSSF nécessite 604 Mo quand celui de lSSW nécessite 6.4 Mo. La méthode lSSW est donc plus efficace que DSSF en taille mémoire. Quant au temps de calcul, les deux méthodes sont du même ordre dans ce cas.

3.4.3.b Propagation dans un conduit atmosphérique bi-directionnel

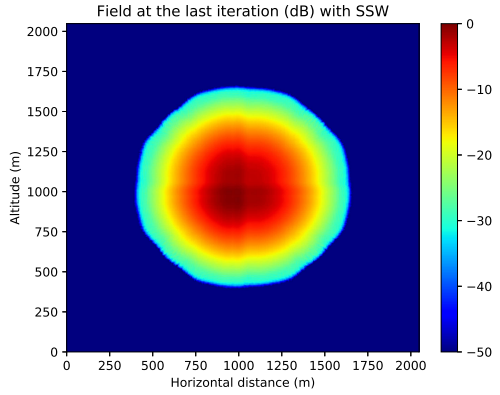
Ensuite, un test où des effets 3D sont à prendre en compte est proposé. Dans celui-ci la propagation dans une atmosphère inhomogène avec un conduit bidirectionnel est modélisée. Dans ce cas, on s'attend à obtenir les mêmes variations dans les deux directions avec lSSW. Ces effets ne peuvent être pris en compte avec une méthode $N\times$2D.

La configuration est la suivante: la fréquence est 300 MHz. La propagation est modélisée sur 10 km avec un pas de 200 m dans la direction x. Le domaine est de taille 2048×2048 m en y et z avec $\Delta y = \Delta z = 1$ m. La source est un CSP placé en $x_s = -50$ m, et $y_s = z_s = 1024$ m dont la largeur est $W_0 = 10$ m.

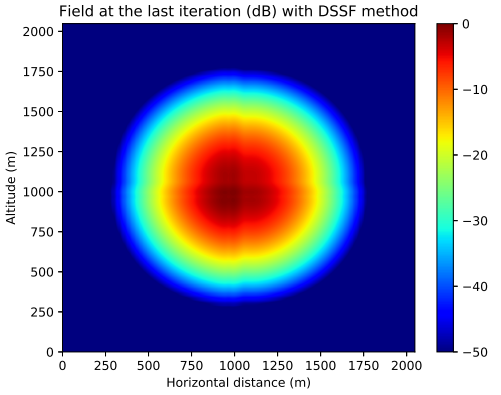
Le conduit bidirectionnel est modélisée par un indice de réfraction variant tri-linéairement dans chaque direction. Les paramètres dans les deux directions sont :$M_0 = 330$ M-units, $y_b = z_b = 950$ m et $y_t = z_t = 100$ m, $c_0 = 0.118$ M-units/m et $c_2 = -1$ M-units/m.

Les mêmes paramètres sont utilisés pour la transformée en ondelettes.

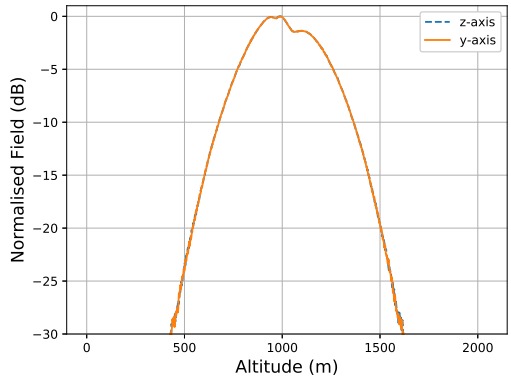
Les résultats obtenus sont présentés sur la Figure C.7.



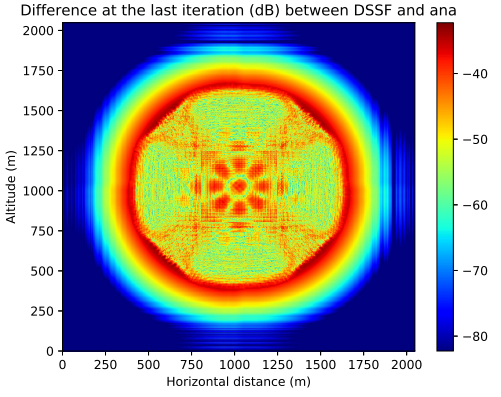
(a) Champ normalisé obtenu avec lSSW dans le plan yOz à la dernière itération.



(b) Champ normalisé obtenu avec DSSF dans le plan yOz à la dernière itération.



(c) Coupes horizontale et verticale du champ normalisé final obtenu avec lSSW.



(d) Différence normalisée entre lSSW et DSSF dans le plan yOz à la dernière itération.

Figure C.7: Propagation du champ rayonné par un CSP en présence d'un conduit bidirectionnel.

Nous pouvons observer que les effets 3D de l'atmosphère inhomogène sont bien pris en compte. De plus, l'erreur avec DSSF en dessous de -30 dB est négligeable. Ainsi lSSW permet de bien modéliser des effets 3D complexes et a été testé avec succès dans ce cas.

Dans un second temps, les temps de calcul et l'occupation mémoire nécessaire aux deux méthodes sont présentés sur le Tableau C.2.

Méthode	DSSF	lSSW
Propagateur (s)	133	6.5
Propagation (min)	9.2	10
Taille mémoire propagateur (Mo)	604	5.1
Taille mémoire en propagation (Mo)	604	0.03 to 3.4

Table C.2: Occupation mémoire et temps de calcul nécessaires à lSSW et DSSF.

Ce tableau montre que lSSW est bien plus efficace que DSSF en terme de taille mémoire. La propagation nécessite cependant un peu plus de temps avec lSSW. Cela est en parti dû à l'implémentation de lSSW en Python où de nombreuses boucles sont nécessaires quand DSSF nécessite uniquement des produits matrice-vecteur effectués en langage compilé (fonction Python).

3.4.3.c Effet Millington

Enfin, la propagation dans un scénario terre-mer-terre est modélisée. Dans celui-ci, on s'attend à observer l'effet Millington [12].

Pour ce test, la fréquence est 75 MHz. La propagation est calculée jusqu'à $x_{\max} = 5$ km avec $\Delta x = 100$ m. Le domaine est de taille 1024×1024 m dans les directions y et z. Les pas sont $\Delta y = \Delta z = 4$ m. La source est invariante selon y et composée d'un CSP placé en $x_s = -50$ m et $z_s = 50$ m pour chaque point selon la direction y.

Les mêmes paramètres d'ondelettes sont utilisés, excepté l'erreur qui est choisie à -30 dB dans ce cas.

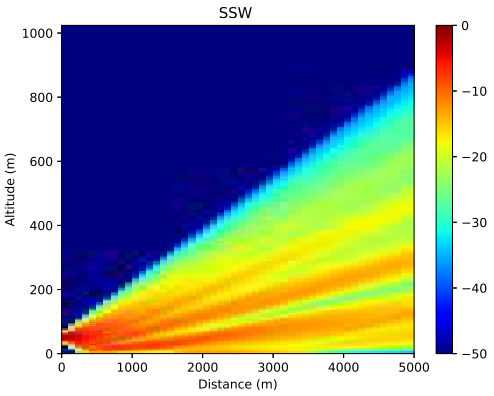
Dans ce scénario, une mer de la taille du domaine en y et de taille 2 km en x est placée à $x = 1.5$ km. Dans le reste du domaine, nous supposons un sol sec. Les paramètres des différents sols sont les suivant :

- Pour la mer: $\epsilon_r = 80$ et $\sigma = 5$ S/m.
- Pour le sol sec : $\epsilon_r = 20$ et $\sigma = 0.02$ S/m.

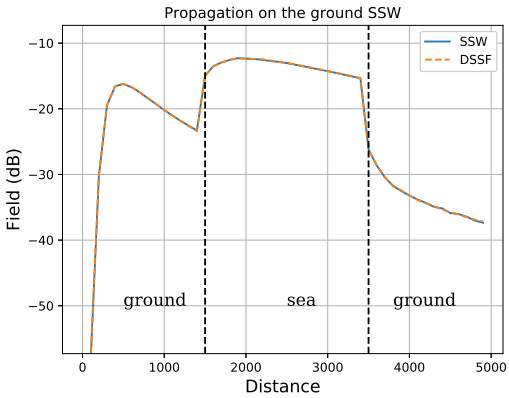
La propagation est modélisée avec 3D lSSW, 3D DSSF et 2D lSSW. On s'attend à obtenir les mêmes résultats avec ces différentes méthodes.

Les résultats sont présentés en Figure C.8. En (a), le champ normalisée dans le plan $x0z$ obtenu avec 3D lSSW est montré. En Figures (b) et (c), nous présentons la propagation de l'onde de sol obtenue avec les méthodes 3D ou la méthode 2D, respectivement.

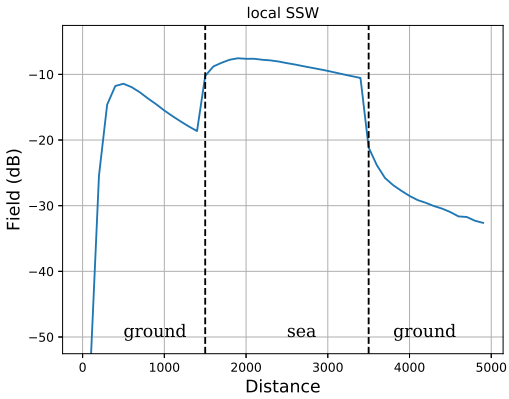
Tout d'abord, nous pouvons voir que l'onde de sol suit le même comportement avec les méthodes 3D ou avec la méthode 2D, comme attendu. Il est le même que celui présenté dans [12].



(a) dans le plan $x0z$ avec 3D lSSW.



(b) au sol avec 3D lSSW et DSSF.



(c) au sol avec 2D lSSW.

Figure C.8: Champ normalisé dans la configuration sol-mer-sol.

De plus, la différence normalisée entre 3D DSSF et 3D lSSW est de -32 dB à la dernière itération. Nous pouvons remarquer qu'en dessous de cette valeur les ondes de sol obtenues avec les deux méthodes sont légèrement différentes.

C.4.4 Conclusion

Dans cette section, nous avons étendu la méthode SSW à la 3D. Des tests numériques ont également été effectués pour valider la méthode et montrer ses avantages.

Dans un premier temps, la méthode 3D de SSW a été introduite. Tout d'abord, l'algorithme général a été rappelé. Ensuite, la méthode pour calculer efficacement le propagateur a été présentée.

Dans un second temps, des tests pour valider la méthode et montrer son avantage en terme de taille mémoire par rapport à DSSF ont été effectués. Ces tests montrent que SSW modélise bien des effets complexes en 3D et les effets du sol. De plus l'occupation mémoire nécessaire à SSW est bien inférieure à celle de DSSF. Le temps de calcul reste cependant à améliorer car il est pour l'instant du même ordre que DSSF.

C.5 Conclusion

L'objectif principal de cette thèse était de développer une méthode rapide et précise pour la modélisation de la propagation longue-distance en 3D. Dans ce cas, la méthode doit être efficace en occupation mémoire. Plusieurs jalons pour atteindre l'objectif ont été atteints. Ceux-ci peuvent se résumer à ces trois contributions:

- En se basant sur la transformée en ondelettes, une méthode locale de split-step wavelet en 2D a été proposée. Celle-ci permet une nette amélioration en terme de taille mémoire par rapport à la version matricielle de SSW.
- Une borne théorique pour l'erreur de compression faite avec SSW a été démontrée. Celle-ci permet de calculer les seuils pour une précision demandée et un scénario donné. Cette contribution répond à la question de la précision de la méthode.
- Enfin, en se basant sur la méthode locale en 2D, une extension à la 3D a été introduite. Cette méthode présente un grand avantage en terme de taille mémoire par rapport à split-step Fourier.

Pour améliorer cette méthode plusieurs pistes sont à envisager. Tout d'abord le code peut être transposé dans un langage compilé pour rendre la comparaison avec split-step Fourier plus représentative. De plus, pour la 3D les effets du relief et la dépolarisation restent à ajouter. Ces phénomènes sont en effet difficiles à prendre en compte. Il faudrait aussi résoudre le problème de la propagation au-dessus d'un sol impédant dont les paramètres varient dans la direction transverse.

De nombreuses applications sont possibles dans divers domaines. En aviation civile, la méthode peut être utilisée pour calculer des couvertures radar. Elle peut aussi être utilisée comme modèle direct pour divers problèmes inverses. Enfin en acoustique elle peut être utilisée à la place de split-step Fourier pour la propagation sous-marine.

Bibliography

[1] International Civil Aviation Organization. *Annexes to the Convention on International Civil Aviation*. ICAO, 2009.

[2] M. P. M. Hall. Effects of the troposphere on radio communication. *IEWS*, 8, 1980.

[3] H. W. Ko, J. W. Sari, and J. P. Skura. Anomalous microwave propagation through atmospheric ducts. *Johns Hopkins APL Technical Digest*, 4:12–26, 1983.

[4] L. Claudepierre. *Simulateur Électromagnétique d'Erreur VOR par Méthodes Déterministes: Application aux Parcs Éoliens*. PhdThesis, Université Paul Sabatier - Toulouse III, 2015.

[5] L. Claudepierre, R. Douvenot, A. Chabory, and C. Morlaas. A deterministic VOR error modeling method - Application to wind turbines. *IEEE transactions on Aerospace and Electronic Systems*, 53(1):247–257, 2017.

[6] A. Chabory, S. Ben-Hassine, C. Morlaas, and R. Douvenot. A stastistical model for assessing the impact of wind turbines on conventionnal VOR. In *Specialist Meeting on Electromagnetic Waves and Wind Turbines (EMWT2019)*, 2019.

[7] B. R. Bean and E. J. Dutton. *Radio Meteorology*, volume 92. Superintendent of Documents, US GovernmentPrint. Office, 1966.

[8] L. Barclay. *Propagation of Radiowaves*, volume 2. IET, 2003.

[9] H. Galiègue. *Modélisation des Effets des Scintillations Ionosphériques sur la Propagation des Ondes Électromagnétiques en Bande L aux Latitudes Polaires.* PhdThesis, Université Paul Sabatier - Toulouse III, 2015.

[10] T. B. A. Senior and J. L. Volakis. *Approximate Boundary Conditions in Electromagnetics*. Number 41. IET, 1995.

[11] M. Levy. *Parabolic Equation Methods for Electromagnetic Wave Propagation*. Number 45. IET, 2000.

[12] G. Millington. Ground-wave propagation across a land/sea boundary. *Nature*, 163(4134):128–128, 1949.

[13] C. Bourlier and G. Kubicke. Ground wave propagation along an inhomogeneous rough surface in the HF band: Millington effect for a flat Earth. *IEEE Transactions on geoscience and remote sensing*, 49(4):1374–1382, 2010.

[14] A. E. Barrios. A terrain parabolic equation model for propagation in the troposphere. *IEEE Transactions on Antennas and Propagation*, 42(1):90–98, 1994.

[15] R. Janaswamy. A curvilinear coordinate-based split-step parabolic equation method for propagation predictions over terrain. *IEEE Transactions on Antennas and Propagation*, 46(7):1089–1097, 1998.

[16] E. R. Kursinski, G. A. Hajj, J. T. Schofield, R. P. Linfield, and K. R. Hardy. Observing Earth's atmosphere with radio occultation measurements using the Global Positioning System. *Journal of Geophysical Research: Atmospheres*, 102(D19):23429–23465, 1997.

[17] P. Gerstoft, L. T. Rogers, J. L. Krolik, and W. S. Hodgkiss. Inversion for refractivity parameters from radar sea clutter. *Radio Science*, 38(3), 2003.

[18] R. Douvenot, V. Fabbro, P. Gerstoft, C. Bourlier, and J. Saillard. A duct mapping method using least squares support vector machines. *Radio Science*, 43(06):1–12, 2008.

[19] C-A. L'Hour. *Modélisation de la Propagation Électromagnétique en Milieux Inhomogènes Basée sur les Faisceaux Gaussiens-Application à la Propagation en Atmosphère Réaliste et à la Radio-occultation entre Satellites*. PhD thesis, Université de Toulouse, Université Toulouse III-Paul Sabatier, 2017.

[20] R. Douvenot, A. Chabory, and S. Rougerie. Modelling the radiowave propagation with a split-step wavelet method for radio occultation. In *2018 22nd International Microwave and Radar Conference (MIKON)*, pages 562–564. IEEE, 2018.

[21] Recommendation ITU P.834-9 (12/2017): Effects of tropospheric refraction on radiowave propagation.

[22] Y. Seity, P. Brousseau, S. l. Malardel, G. Hello, P. Bénard, F. Bouttier, C. Lac, and V. Masson. The AROME-France convective-scale operational model. *Monthly Weather Review*, 139(3):976–991, 2011.

[23] X-Y. Huang, Q. Xiao, D. M. Barker, X. Zhang, J. Michalakes, W. Huang, T. Henderson, J. Bray, Y. Chen, Z. Ma, et al. Four-dimensional variational data assimilation for WRF: Formulation and preliminary results. *Monthly Weather Review*, 137(1):299–314, 2009.

[24] J. G. Powers, J. B. Klemp, W. C. Skamarock, C. A. Davis, J. Dudhia, D. O. Gill, J. L. Coen, D. J. Gochis, R. Ahmadov, S. E Peckham, et al. The weather research and forecasting model: Overview, system efforts, and future directions. *Bulletin of the American Meteorological Society*, 98(8):1717–1737, 2017.

[25] J-M. Jin. *The Finite Element Method in Electromagnetics*. John Wiley & Sons, 2015.

[26] R. F. Harrington. *Field Computation by Moment Methods*. Wiley-IEEE Press, 1993.

[27] A. Taflove and S. C. Hagness. *Computational Electrodynamics: the Finite-Difference Time-Domain Method*. Artech house, 2005.

[28] D. Bouche and F. Molinet. *Méthodes Asymptotiques en Électromagnétisme*, volume 16. Springer Science & Business Media, 1994.

[29] J. S. Asvestas. The physical optics method in electromagnetic scattering. *Journal of Mathematical Physics*, 21(2):290–299, 1980.

[30] J. R. Kuttler and G. D. Dockery. Theoretical description of the parabolic approximation/Fourier split-step method of representing electromagnetic propagation in the troposphere. *Radio Science*, 26(2):381–393, 1991.

[31] G. A. Deschamps. Gaussian beam as a bundle of complex rays. *Electronics Letters*, 7(23):684–685, 1971.

[32] A. Chabory, J. Sokoloff, S. Bolioli, and P. F. Combes. Computation of electromagnetic scattering by multilayer dielectric objects using Gaussian beam based techniques. *Comptes Rendus Physique*, 6(6):654–662, 2005.

[33] C-A. L'Hour, V. Fabbro, A. Chabory, and J. Sokoloff. 2D modeling of the atmospheric refraction based on Gaussian beams. In *2016 10th European Conference on Antennas and Propagation (EuCAP)*, pages 1–3. IEEE, 2016.

[34] Y. A. Kravtsov, L. A. Ostrovsky, and N. S. Stepanov. Geometrical optics of inhomogeneous and nonstationary dispersive media. *Proceedings of the IEEE*, 62(11):1492–1510, 1974.

[35] J. B. Keller. Geometrical theory of diffraction. *JOSA*, 52(2):116–130, 1962.

[36] R. G. Kouyoumjian and P. H. Pathak. A uniform geometrical theory of diffraction for an edge in a perfectly conducting surface. *Proceedings of the IEEE*, 62(11):1448–1461, 1974.

[37] H. Kogelnik and T. Li. Laser beams and resonators. *Applied Optics*, 5(10):1550–1567, 1966.

[38] D. Marcuse. *Light Transmission Optics*. Van Nostrand Reinhold New York, 1972.

[39] P. O. Leye, A. Khenchaf, P. Pouliguen, et al. The Gaussian beam summation and the Gaussian launching methods in scattering problem. *Journal of Electromagnetic Analysis and Applications*, 8(10):219, 2016.

[40] A. Khenchaf, P. O. Leye, and P. Pouliguen. RCS complex target, Gaussian beam summation method. In *EUCAP 2016*, 2016.

[41] A. Bogush and R. Elkins. Gaussian field expansions for large aperture antennas. *IEEE Transactions on Antennas and Propagation*, 34(2):228–243, 1986.

[42] R. K. Arora and Z. Lu. Graphical depiction of the electromagnetic fields of Hermite—Gaussian modes. In *IEE Proceedings H (Microwaves, Antennas and Propagation)*, volume 139, pages 369–375. IET, 1992.

[43] D. Gabor. Theory of communication. part 1: The analysis of information. *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering*, 93(26):429–441, 1946.

[44] P. D. Einziger, S. Raz, and M. Shapira. Gabor representation and aperture theory. *Journal of Optical Society of America*, 3(4):508–522, 1986.

[45] J. J. Maciel and L. B. Felsen. Systematic study of fields due to extended apertures by Gaussian beam discretization. *IEEE transactions on antennas and propagation*, 37(7):884–892, 1989.

[46] J. J. Maciel and L. B. Felsen. Discretized Gabor-based beam algorithm for time-harmonic radiation from two-dimensional truncated planar aperture distributions. I. Formulation and solution. *IEEE Transactions on Antennas and Propagation*, 50(12):1751–1759, 2002.

[47] H-T. Chou, P. H. Pathak, and R. J. Burkholder. Novel Gaussian beam method for the rapid analysis of large reflector antennas. *IEEE Transactions on Antennas and Propagation*, 49(6):880–893, 2001.

[48] A. Chabory, J. Sokoloff, and S. Bolioli. Novel Gabor-based Gaussian beam expansion for curved aperture radiation in dimension two. *Progress In Electromagnetics Research*, 58:171–185, 2006.

[49] R. H. Hardin. Applications of the split-step Fourier method to the numerical solution of nonlinear and variable coefficient wave equations. *SIAM Review (Chronicles)*, 15, 1973.

[50] R. Janaswamy. Path loss predictions in the presence of buildings on flat terrain: A 3-D vector parabolic equation approach. *IEEE Transactions on Antennas and Propagation*, 51(8):1716–1728, 2003.

[51] Z. El Ahdab and F. Akleman. Radiowave propagation analysis with a bidirectional 3-D vector parabolic equation method. *IEEE Transactions on Antennas and Propagation*, 65(4):1958–1966, 2017.

[52] H. Zhou. *Modeling the Atmospheric Propagation of Electromagnetic waves in 2-D and 3-D using Fourier and Wavelet Transforms*. PhdThesis, Université Paul Sabatier - Toulouse III, 2018.

[53] A. Ginestet. *Modélisation de la Propagation d'une Onde Électromagnétique sur des Scènes de Grande Taille par Résolution de l'Equation Parabolique 3D Vectorielle*. PhD thesis, Université Paul Sabatier-Toulouse III, 2007.

[54] H. Zhou, A. Chabory, and R. Douvenot. A 3-D split-step Fourier algorithm based on a discrete spectral representation of the propagation equation. *IEEE Transactions on Antennas and Propagation*, 65(4):1988–1995, 2017.

[55] A. Iqbal and V. Jeoti. A split-step wavelet method for radiowave propagation modelling in tropospheric ducts. In *RF and Microwave Conference (RFM), 2011 IEEE International*, pages 67–70. IEEE, 2011.

[56] A. Iqbal and V. Jeoti. An improved split-step wavelet transform method for anomalous radio wave propagation modeling. *Radioengineering*, 23(4):987, 2014.

[57] H. Zhou, R. Douvenot, and A. Chabory. Modeling the long-range wave propagation by a split-step wavelet method. *Journal of Computational Physics*, 402:109042, 2020.

[58] M. A. Leontovich and V. A. Fock. Solution of the problem of propagation of electromagnetic waves along the earth's surface by the method of parabolic equation. *J. Phys. USSR*, 10(1):13–23, 1946.

[59] A. V. Popov. Solution of a parabolic equation of diffraction theory by a finite difference method. *USSR Computational Mathematics and Mathematical Physics*, 8(5):282–288, 1968.

[60] D. Lee, G. Botseas, and J. S. Papadakis. Finite-difference solution to the parabolic wave equation. *The Journal of the Acoustical Society of America*, 70(3):795–800, 1981.

[61] F. D. Tappert. The parabolic approximation method. In *Wave Propagation and Underwater Acoustics*, pages 224–287. Springer, 1977.

[62] F. R. Dinapoli and R. L. Deavenport. Numerical Methods of Underwater Acoustic Propagation Ocean Acoustics, 1977.

[63] G. D. Dockery. Modeling electromagnetic wave propagation in the troposphere using the parabolic equation. *IEEE Transactions on Antennas and Propagation*, 36(10):1464–1470, 1988.

[64] S. T. McDaniel. Parabolic approximations for underwater sound propagation. *The Journal of the Acoustical Society of America*, 58(6):1178–1185, 1975.

[65] R. J. Mac Arthur and D. H. O. Bebbington. Diffraction over simple terrain obstacles by the method of parabolic equation. In *ICAP91, IEE Conf. Pub*, pages 2824–2827, 1991.

[66] V. A Permyakov, M. S. Mikhailov, and E. S. Malevich. Analysis of propagation of electromagnetic waves in difficult conditions by the parabolic equation method. *IEEE Transactions on Antennas and Propagation*, 67(4):2167–2175, 2019.

[67] Y. Gao, Q. Shao, B. Yan, Q. Li, and S. Guo. Parabolic equation modeling of electromagnetic wave propagation over rough sea surfaces. *Sensors*, 19(5):1252, 2019.

[68] D. J. Donohue and J. R. Kuttler. Propagation modeling over terrain using the parabolic wave equation. *IEEE Transactions on Antennas and Propagation*, 48(2):260–277, 2000.

[69] G. Apaydin and L. Sevgi. A novel split-step parabolic-equation package for surface-wave propagation prediction along multiple mixed irregular-terrain paths. *IEEE Antennas and Propagation Magazine*, 52(4):90–97, 2010.

[70] Z. El Ahdab and F. Akleman. Groundwave propagation in a nonhomogeneous atmosphere: Prediction using 3D parabolic equation. In *2017 International Applied Computational Electromagnetics Society Symposium-Italy (ACES)*, pages 1–2. IEEE, 2017.

[71] D. Dockery and J. R. Kuttler. An improved impedance-boundary algorithm for Fourier split-step solutions of the parabolic wave equation. *IEEE Transactions on Antennas and Propagation*, 44(12):1592–1599, 1996.

[72] J. R. Kuttler and R. Janaswamy. Improved Fourier transform methods for solving the parabolic wave equation. *Radio Science*, 37(2):1–11, 2002.

[73] W. L. Siegmann, G. A. Kriegsmann, and D. Lee. A wide-angle three-dimensional parabolic wave equation. *The Journal of the Acoustical Society of America*, 78(2):659–664, 1985.

[74] M. D. Collins and S. A. Chin-Bing. A three-dimensional parabolic equation model that includes the effects of rough boundaries. *The Journal of the Acoustical Society of America*, 87(3):1104–1109, 1990.

[75] F. Sturm and J. A. Fawcett. On the use of higher-order azimuthal schemes in 3-D PE modeling. *The Journal of the Acoustical Society of America*, 113(6):3134–3145, 2003.

[76] Y. Lin, J. M. Collis, and T. F. Duda. A three-dimensional parabolic equation model of sound propagation using higher-order operator splitting and Padé approximants. *The Journal of the Acoustical Society of America*, 132(5):EL364–EL370, 2012.

[77] W. C. Chew. Electromagnetic theory on a lattice. *Journal of Applied Physics*, 75(10):4843–4850, 1994.

[78] E. K. Smith and S. Weintraub. The constants in the equation for atmospheric refractive index at radio frequencies. *Proceedings of the IRE*, 41(8):1035–1037, 1953.

[79] L. Brekhovskikh. *Waves in Layered Media*, volume 16. Elsevier, 2012.

[80] D. E. Barrick. Theory of HF and VHF propagation across the rough sea, 1, The effective surface impedance for a slightly rough highly conducting medium at grazing incidence. *Radio Science*, 6(5):517–526, 1971.

[81] D. E. Barrick. Theory of HF and VHF propagation across the rough sea, 2, Application to HF and VHF propagation above the sea. *Radio Science*, 6(5):527–533, 1971.

[82] A. R. Miller, R. M. Brown, and E. Vegh. New derivation for the rough-surface reflection coefficient and for the distribution of sea-wave elevations. In *IEE Proceedings H-Microwaves, Optics and Antennas*, volume 131, pages 114–116. IET, 1984.

[83] A. Ishimaru, J. D. Rockway, and S. Lee. Sommerfeld and Zenneck wave propagation for a finitely conducting one-dimensional rough surface. *IEEE Transactions on Antennas and Propagation*, 48(9):1475–1484, 2000.

[84] O. Benhmammouch, A. Khenchaf, and N. Caouren. Modelling roughness effects on propagation of electromagnetic waves in a maritime environment: a hybrid approach. *IET radar, sonar & navigation*, 5(9):1018–1025, 2011.

[85] J. R. Kuttler. Differences between the narrow-angle and wide-angle propagators in the split-step Fourier solution of the parabolic wave equation. *IEEE Transactions on Antennas and Propagation*, 47(7):1131–1140, 1999.

[86] M. D. Feit and J. A. Fleck. Light propagation in graded-index optical fibers. *Applied Optics*, 17(24):3990–3998, 1978.

[87] E.E. Gossard and R.G. Strauch. Radar observation of clear air and clouds. *Developments in Atmospheric Science*, 14, 1983.

[88] A. Nisbet. Hertzian electromagnetic potentials and associated gauge transformations. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 231(1185):250–263, 1955.

[89] O. Benhmammouch. *Propagation des Ondes EM et Conduits Atmosphériques au-dessus d'une Surface Maritime Rugueuse*. PhD thesis, Brest, 2011.

[90] A. Grossmann and J. Morlet. Decomposition of Hardy functions into square integrable wavelets of constant shape. *SIAM journal on mathematical analysis*, 15(4):723–736, 1984.

[91] Y. Meyer. *Ondelettes*. Hermann, 1990.

[92] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41(7):909–996, 1988.

[93] S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.

[94] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic press, 1999.

[95] M. L. Hilton. Wavelet and wavelet packet compression of electrocardiograms. *IEEE Transactions on Biomedical Engineering*, 44(5):394–402, 1997.

[96] H. A. R. Akkar, W. A. H. Hadi, and I. H. Al-Dosari. A Squared-Chebyshev wavelet thresholding based 1D signal compression. *Defence Technology*, 15(3):426–431, 2019.

[97] B. Walczak and D. L. Massart. Noise suppression and signal compression using the wavelet packet transform. *Chemometrics and Intelligent Laboratory Systems*, 36(2):81–94, 1997.

[98] F. M. Bayer, A. J. Kozakevicius, and R. J. Cintra. An iterative wavelet threshold for signal denoising. *Signal Processing*, 162:10–20, 2019.

[99] M. Rabbani. JPEG2000: Image compression fundamentals, standards and practice. *Journal of Electronic Imaging*, 11(2):286, 2002.

[100] P. C. L. Da Silva, J. P. Da Silva, and A. R. G. Garcia. Daubechies wavelets as basis functions for the vectorial beam propagation method. *Journal of Electromagnetic Waves and Applications*, 33(8):1027–1041, 2019.

[101] Z. Xiang and Y. Lu. An effective wavelet matrix transform approach for efficient solutions of electromagnetic integral equations. *IEEE Transactions on Antennas and Propagation*, 45(8):1205–1213, 1997.

[102] T. K. Sarkar, M. Salazar-Palma, and M. C. Wicks. *Wavelet Applications in Engineering Electromagnetics*. Artech House, 2002.

[103] Y. Sun, W. Song, R. Sun, and R. Sun. A novel method for fast analysis of electromagnetic scattering characteristics of PEC objects. In *2019 IEEE Asia-Pacific Microwave Conference (APMC)*, pages 1074–1076. IEEE, 2019.

[104] W. Bilgic, A. Rennings, P. Waldow, and I. Wolff. Appropriate wavelets with compact support for the compression of FDTD calculated electromagnetic fields. In *German Microwave Conference 2005*, 2005.

[105] T-K. Hong and B. L. N. Kennett. On a wavelet-based method for the numerical simulation of wave propagation. *Journal of Computational Physics*, 183(2):577–622, 2002.

[106] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, 1992.

[107] T. Bonnafont, R. Douvenot, and A. Chabory. A local split-step wavelet method for the long range propagation simulation in 2D. *Radio Science*, in Minor review.

[108] T. Bonnafont, R. Douvenot, and A. Chabory. A speed up of split-step wavelet for the computation of long range propagation. In *2020 14th European Conference on Antennas and Propagation (EUCAP)*. IEEE, 2020.

[109] A. Cohen. *Numerical Analysis of Wavelet Methods*. Elsevier, 2003.

[110] D. Charalambos and B. Aliprantis. *Infinite Dimensional Analysis: A Hitchhiker's Guide*. Springer-Verlag Berlin and Heidelberg GmbH & Company KG, 2013.

[111] R. A. DeVore, B. Jawerth, and V. Popov. Compression of wavelet decompositions. *American Journal of Mathematics*, 114(4):737–785, 1992.

[112] L. B. Felsen and N. Marcuvitz. *Radiation and Scattering of Waves*, volume 31. John Wiley & Sons, 1994.

[113] T. Bonnafont, R. Douvenot, and A. Chabory. Modélisation par méthode split-step wavelet avec propagateurs locaux. In *XXIèmes Journées Nationales Microondes*, Caen, France, May 2019.

[114] T. Bonnafont, R. Douvenot, and A. Chabory. Split-step wavelet propagation modelling using local operators. In *2019 URSI International Symposium on Electromagnetic Theory (EMTS)*, pages 1–4. IEEE, 2019.

[115] A. Iqbal and V. Jeoti. A novel wavelet-Galerkin method for modeling radio wave propagation in tropospheric ducts. *Progress in Electromagnetics Research*, 36:35–52, 2012.

[116] C. Rocken, R. Anthes, M. Exner, D. Hunt, S. Sokolovskiy, R. Ware, M. Gorbunov, W. Schreiner, D. Feng, B. Herman, et al. Analysis and validation of GPS/MET data in the neutral atmosphere. *Journal of Geophysical Research: Atmospheres*, 102(D25):29849–29866, 1997.

[117] E. R. Kursinski, G. A. Hajj, S. S. Leroy, and B. Herman. The GPS radio occulation technique. *Terr. Atmos. Ocean. Sci.*, 11(1):53, 2001.

[118] S. B. Healy and J. R. Eyre. Retrieving temperature, water vapour and surface pressure information from refractive-index profiles derived by radio occultation: A simulation study. *Quarterly Journal of the Royal Meteorological Society*, 126(566):1661–1683, 2000.

[119] S. B. Healy and J-N. Thépaut. Assimilation experiments with CHAMP GPS radio occultation measurements. *Quarterly Journal of the Royal Meteorological Society: A Journal of the Atmospheric Sciences, Applied Meteorology and Physical Oceanography*, 132(615):605–623, 2006.

[120] C-A. L'Hour, V. Fabbro, A. Chabory, and J. Sokoloff. 2-D propagation modeling in inhomogeneous refractive atmosphere based on Gaussian beams Part I: Propagation modeling. *IEEE Transactions on Antennas and Propagation*, 67(8):5477–5486, 2019.

[121] C-A. L'Hour, V. Fabbro, A. Chabory, and J. Sokoloff. 2-D propagation modeling in inhomogeneous refractive atmosphere based on Gaussian beams Part II: Application

to radio occultation. *IEEE Transactions on Antennas and Propagation*, 67(8):5487–5496, 2019.

[122] G. Strang. Wavelet transforms versus Fourier transforms. *Bulletin of the American Mathematical Society*, 28(2):288–305, 1993.

[123] T. Bonnafont, R. Douvenot, and A. Chabory. Split-step wavelet with local operators for the 3D long-range propagation. In *2021 15th European Conference on Antennas and Propagation (EUCAP)*. IEEE, 2021, submitted.

[124] Y. A. Kravtsov and Y. I. Orlov. *Geometrical optics of inhomogeneous media*. Springer-Verlag, 1990.

[125] U. Karabaş, Y. Diouane, and R. Douvenot. On the use of adjoint methods for refractivity estimation in the troposphere. In *2020 14th European Conference on Antennas and Propagation (EuCAP)*, pages 1–5. IEEE, 2020.

[126] D. J. Thomson and N. R. Chapman. A wide-angle split-step algorithm for the parabolic equation. *The Journal of the Acoustical Society of America*, 74(6):1848–1854, 1983.

[127] K. B. Smith. Convergence, stability, and variability of shallow water acoustic predictions using a split-step Fourier parabolic equation model. *Journal of Computational Acoustics*, 9(01):243–285, 2001.

[128] Y-T. Lin, T. F. Duda, and A. E. Newhall. Three-dimensional sound propagation models using the parabolic-equation approximation and the split-step Fourier method. *Journal of Computational Acoustics*, 21(01):1250018, 2013.

Abstract: The tropospheric long-range propagation of electromagnetic waves is a topic of major concern in many applications. The objective of this Ph.D. thesis is to develop a method to model the propagation in a realistic 3D domain. This method should be fast, accurate, and low in memory occupation. Three main milestones toward this objective are achieved. First, a 2D wavelet-based method has been improved. Second, a theoretical bound for the accuracy has been proposed. Lastly, a wavelet-based 3D propagation method has been developped. In the context of long-range propagation, the split-step Fourier method is widely used. For large domain propagation and 3D, the time and memory occupation become a major issue. Therefore, a matrix split-step wavelet (mSSW) method has been developed. Using compression and the fast wavelet transform, this method is fast. Compression is used to increase the efficiency of the method, but it introduces an accumulation of error throughout the propagation. We propose a formula for setting the compression thresholds in order to obtain a chosen accuracy in a given domain. Numerical tests have shown that the memory size of the propagator becomes an issue for large domains. Using wavelet properties, a local method of SSW (lSSW) has been proposed to reduce this requirement while keeping the computation time low. It is based on the computation of a minimal set of wavelet propagations, for which only the essential information is stored. Numerical tests have shown that this method is lower than mSSW in terms of memory occupation. Using the 2D wavelet representation, a 3D lSSW method has been proposed. Numerical tests have been performed to show validate the method on canonical scenarios. Finally, propagation over islands has been studied. We have shown that the discrete mixed Fourier transform, widely used in case of impedance ground, is not valid in this case.

Résumé La propagation troposphèrique des ondes électromagnétiques sur de longues distances est un sujet important pour de nombreuses applications. L'objectif de cette thèse est de développer une méthode rapide et précise pour la propagation dans des domaines 3D. Celle-ci devra être efficace en taille mémoire. Dans le but d'atteindre cet objectif, trois contributions majeures ont été obtenues. Une méthode locale de propagation basée sur les ondelettes a été introduite en 2D. Une borne pour l'erreur de compression de cette méthode a été démontrée. Enfin, la méthode locale a été étendue à la 3D. La méthode split-step Fourier est très utilisée dans le contexte de la propagation sur de longues distances. Cependant, dans le cas de la 3D elle n'est pas très efficace en taille mémoire et temps de calcul. Une méthode basée sur les ondelettes en 2D, matrix split-step wavelet (mSSW), a été introduite pour réduire le temps de calcul. En effet à l'aide de la transformée rapide en ondelettes et de la compression, la méthode est très efficace en temps de calcul. La compression introduit une erreur qui s'accumule pendant la propagation. C'est pourquoi nous avons proposé une formule qui permet de choisir les seuils de compression pour une erreur donnée. La taille mémoire est un problème majeur pour le passage à la 3D de mSSW. Une méthode locale SSW (lSSW) a été développée pour la réduire. Dans celle-ci, seule l'information essentielle à la propagation est stockée réduisant l'occupation mémoire au minimum. Des tests numériques ont montré que cette méthode est plus efficace que mSSW en taille mémoire. Cette méthode a donc été étendue à la 3D. Des tests numériques dans des cas canoniques ont montré l'efficacité de cette méthode. Le problème de la propagation au dessus d'île en 3D a été étudié. Nous avons montré que dans ce cas la discrete-mixed Fourier transform, largement utilisée pour les sols impédants, n'est pas valide dans ce cas.

Key words: Split-step, Wavelets, Parabolic wave equation, Atmospheric long-range propagation, Computational electromagnetics.