

Rapport de stage

---

# Développement d'un code de propagation électromagnétique dans l'ionosphère

---

*Auteur :*  
Gonzalo J. Carracedo Carballal

*Directeurs de stage :*  
Pauline Lafitte  
Sebastian Minjeaud

20 juin 2013



École Centrale Paris



Master Modélisation et Simulation

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Définition des exigences</b>	<b>6</b>
2.1	Exigences fonctionnelles . . . . .	6
2.2	Exigences non fonctionnelles . . . . .	7
<b>3</b>	<b>Théorie de l'optique géométrique</b>	<b>8</b>
3.1	Dérivation de l'optique géométrique . . . . .	8
3.2	Description et mise en œuvre du traceur de rayons . . . . .	13
3.3	Limitations de l'optique géométrique . . . . .	16
<b>4</b>	<b>L'approximation de Rytov</b>	<b>17</b>
4.1	Dérivation de l'approximation de Rytov . . . . .	18
4.1.1	La solution de Rytov au premier ordre . . . . .	19
4.1.2	Les solutions d'ordre supérieur . . . . .	20
4.1.3	Mise en œuvre de la méthode de Rytov . . . . .	20
4.1.4	Parallélisation . . . . .	21
<b>5</b>	<b>Post-processus : calcul des paramètres de scintillation</b>	<b>23</b>
5.1	Le coefficient de variation de l'intensité : $S_4$ . . . . .	23
5.2	L'écart type de la phase : $\sigma_\varphi$ . . . . .	23
<b>6</b>	<b>Architecture interne de waveprop</b>	<b>24</b>
<b>7</b>	<b>Résultats</b>	<b>27</b>
<b>8</b>	<b>Conclusion</b>	<b>30</b>
<b>A</b>	<b>Conventions de repère</b>	<b>32</b>
A.1	Caractéristiques partagées par tous les systèmes de coordonnées . . . . .	32
A.2	Coordonnées cartésiennes en espace inertiel . . . . .	33
A.3	Coordonnées sphériques cartographiques . . . . .	34
A.4	Système de potentiels d'Euler . . . . .	36
<b>B</b>	<b>La constellation Walker delta</b>	<b>38</b>
<b>C</b>	<b>Détails techniques de la source de données eulerds</b>	<b>40</b>
C.1	Arrangement spatiale de barycentres avec arbres K-d . . . . .	41
C.2	Rastérisation de tétraèdres . . . . .	41
C.3	Cellules euleriennes . . . . .	43



# Table des figures

3.1	Système de coordonnées du plan de l'antenne. En vert : aire couverte par les points d'arrivée. . . . .	14
3.2	Aire d'arrivée, aire de référence et distance de référence. . . . .	15
4.1	Parallélisation de l'intégrant optimisée. . . . .	22
4.2	Parallélisation de l'évaluation optimisée. . . . .	22
6.1	Diagramme UML des composants de <b>waveprop</b> . . . . .	24
7.1	Satellites visibles du plan orbital 1. . . . .	28
7.2	Satellites visibles du plan orbital 2. . . . .	28
7.3	Satellites visibles du plan orbital 3. . . . .	29
B.1	Constellation Galileo. . . . .	38
B.2	Éléments orbitaux. . . . .	39
C.1	Arbre K-d, trois dimensions. . . . .	41
C.2	Défauts générés par la rastérisation de tétraèdres. Remarquez les colonnes absentes. . . . .	42

# Chapitre 1

## Introduction

Le système Galileo est un système de positionnement par satellites destiné à supprimer la dépendance en matière spatiale par rapport au système américain GPS, dont la justification se trouve dans les restrictions sur la précision du positionnement (autour de 20 mètres pour le signal gratuit), sur la fiabilité ou la continuité car le système de positionnement peut être impossible dans certaines zones du globe et certains moments.

Le projet IODISSEE<sup>1</sup>, financé par l'ANR, a pour but d'aider à la conception d'un réseau de stations de surveillance en permettant d'atteindre les buts de la mission Galileo<sup>2</sup> et quantifier les performances du système. Jusqu'à présent, les logiciels disponibles pour aider les ingénieurs dans cette tâche se basent sur des modèles phénoménologiques ou statistiques. Le projet IODISSEE propose la construction d'un modèle ionosphérique complet couplé à un module de propagation, complémentaire à l'existant. Certes on peut le faire avec des modélisations traditionnelles, mais expériences récentes et la connaissance obtenus des satellites GPS et de leurs communications avec les stations au sol ont permis de confirmer et d'améliorer les modèles avec des données réelles. Pour atteindre ce but, le projet est divisé en plusieurs tâches. Principalement :

1. Définition des besoins. Le but de cette tâche est de détailler les besoins en termes de modélisation des phénomènes de scintillation nécessaires pour mettre en œuvre l'évaluation du système Galileo, de caractériser les limitations du modèle actuel GISM en termes de représentativité ionosphérique et de dériver un ensemble de spécifications avec l'objectif de répondre aux besoins du système Galileo.
2. Amélioration des modèles par comparaison avec des mesures expérimentales sur des événements rares de dépression du plasma.
3. Amélioration du modèle physique en utilisant des données GPS.
4. Développement d'un module de propagation électromagnétique dans une ionosphère perturbée.
5. Développement du modèle principal et des schémas numériques.

Dans le cadre de la tâche 4, le but de ce stage était de concevoir et de mettre en œuvre un module de propagation électromagnétique, appelé **waveprop**, qui soit facilement utilisable, maintenable et extensible. En disposant en entrée des données appropriées (soit des échantillons de densité électronique, soit des échantillons de permittivité diélectrique), d'une constellation des satellites

---

1. <http://iodissee.math.cnrs.fr/>

2. <http://www.gsa.europa.eu/galileo-0>

et des coordonnées d'une station au sol, on obtient des valeurs des paramètres de scintillation pour faire des mesures de la qualité de la communication satellite-Terre.

Pour ce faire, on va calculer la perturbation du signal induite par des perturbations ionosphériques grâce à deux approximations différentes : dans un premier temps, on obtient des résultats de perturbation de la phase grâce à l'optique géométrique. Toutefois, ces premiers résultats ne sont pas valables pour les perturbations de l'intensité à cause des limitations de l'optique. Dans un second temps, on obtient des résultats plus précis (mais aussi beaucoup plus coûteux) grâce à l'approximation de Rytov, valables cette fois aussi bien pour l'intensité que pour la phase, et on les compare avec les résultats de l'optique géométrique et des mesures réelles.

Enfin, on fournit une documentation orienté vers l'utilisateur et le programmeur qui voudrait modifier le code ultérieurement, ainsi qu'une liste des possibles détails à améliorer et à optimiser.

## Chapitre 2

# Définition des exigences

L'objectif de ce projet est la conception et la mise en œuvre d'un logiciel appelé **waveprop** capable de prévoir la perturbation du signal du satellite du fait des phénomènes de scintillation ionosphérique. Plus précisément, on va définir les exigences dans deux catégories :

### 2.1 Exigences fonctionnelles

**waveprop** doit être capable de :

- Obtenir les données de densité électronique / indice de réfraction à partir de plusieurs formats de fichier. Le code doit prévoir l'existence d'une diversité de formats afin de qu'on ne soit pas restreint à une seule structure de données.
- Pouvoir définir plusieurs constellations de satellites. Jusqu'à présent<sup>1</sup>, les satellites Galileo qui sont opérationnels ne sont que des satellites de validation en orbite<sup>2</sup> (IOV, In-Orbit Validation), donc on ne peut pas être sûr de la configuration définitive de la constellation Galileo. **waveprop** doit pouvoir d'être configuré pour travailler avec plusieurs types d'orbites à partir de leurs paramètres orbitaux.
- Pouvoir spécifier les coordonnées des stations au sol. Puisqu'il y a plusieurs stations au tour du monde dont on a les mesures empiriques, il doit être possible de configurer plusieurs points de vue afin de comparer les sorties du code avec ces observations-là.
- Calculer la perturbation du signal avec deux méthodes différentes : l'Optique Géométrique (dont on suppose la fréquence infinie et aucun effet diffractif, basé sur le tracé de rayons) et l'approximation de Rytov (méthode plus complète qui tient compte de la totalité du domaine) . Ceci a une finalité double : d'un côté on l'utilise pour valider les résultats d'une méthode avec les résultats de l'autre. De l'autre côté, on peut définir des conditions expérimentaux où il est plus pratique d'utiliser une méthode que l'autre.
- Exprimer la perturbation du signal grâce aux paramètres de scintillation  $S_4$  (une mesure de variance de l'intensité du signal le long d'un intervalle de temps) et  $\sigma_\varphi$  (écart type des variations de la phase du signal) est le but central du code. **waveprop** doit être aussi capable

---

1. à date du 3 juin 2013

2. Pour l'instant il n'y a que deux satellites dans le premier plan orbital et autres deux dans le deuxième plan. Le but de cette constellation réduite est de faire une évaluation préliminaire du système. Voir [http://download.esa.int/docs/Galileo\\_IOV\\_Launch/Galileo\\_factsheet\\_2012.pdf](http://download.esa.int/docs/Galileo_IOV_Launch/Galileo_factsheet_2012.pdf)

de sauvegarder ces données sous plusieurs formats. Principalement RINEX<sup>3</sup>, mais aussi sous d'autres comme MATLAB afin de pouvoir travailler avec ces sorties d'une forme plus pratique.

## 2.2 Exigences non fonctionnelles

En plus des préalables fonctionnels, il est aussi intéressant que le code puisse atteindre les buts suivants :

- Être facilement configurable. Du fait qu'il y a beaucoup de paramètres dont on doit tenir compte il, faut qu'ils soient bien structurés dans le fichier de configuration.
- Être optimal. On ne se contente pas de la transcription littéral des formules de la physique, mais on fait des optimisations pour atteindre de meilleures performances.
- Être parallèle. Particulièrement, les opérations réalisées par le solveur de Rytov sont spécialement coûteuses (mais aussi extrêmement parallélisables) donc il faut profiter de la nature multi-processeur des nouveaux calculateurs scientifiques.
- Être extensible. On veut qu'il ne soit pas nécessaire de modifier `waveprop` directement et de le compiler. Il est beaucoup plus pratique d'avoir un système de *plug-ins* de sorte qu'on puisse écrire nos propres sources de données / formats de sortie / solveurs sans alourdir le code central de `waveprop`.
- Être maintenable. Le code est conçu de sorte qu'il soit facile à comprendre et à améliorer. Les fonctions et interfaces du code doivent être systématiquement commentés.

Ces considérations vont justifier la conception finale du code, laquelle sera développée le long des chapitres suivants.

---

3. RINEX est le format traditionnel pour exprimer ce type d'information, on a choisi ce format pour raisons de compatibilité avec les systèmes existentes. Voir <ftp://igscb.jpl.nasa.gov/pub/data/format/rinex210.txt>

## Chapitre 3

# Théorie de l'optique géométrique

Le premier solveur de `waveprop` (nommé `goptics`) utilise les équations de l'optique géométrique. Le signal des satellites est approché par un ensemble des rayons qui arrivent à la station au sol, perturbés le long de leur chemin à cause des variations de l'indice de réfraction de l'ionosphère (lequel dépend directement de la valeur ponctuelle de la densité électronique).

### 3.1 Dérivation de l'optique géométrique

On ne présente ici qu'un résumé de la dérivation du modèle, une dérivation complète peut par exemple être trouvée dans Wheelon[2]. Tout d'abord, on doit partir des équations qui décrivent les phénomènes électromagnétiques, c'est-à-dire, les équations de Maxwell. Afin de simplifier les calculs suivants, on les exprimera dans le système CGS :

$$\vec{\nabla} \times \vec{E} = -\frac{1}{c} \frac{\partial \vec{B}}{\partial t}, \quad (3.1)$$

$$\vec{\nabla} \times \vec{H} = \frac{1}{c} \frac{\partial \vec{D}}{\partial t} + \frac{4\pi}{c} \vec{J}, \quad (3.2)$$

$$\nabla \cdot \vec{D} = 4\pi \rho_e, \quad (3.3)$$

$$\nabla \cdot \vec{B} = 0 \quad (3.4)$$

avec  $\vec{J}$  la densité de courant et  $\rho_e$  la densité de charge. En appliquant la divergence à l'équation d'Ampère(3.2) et en utilisant l'équation de Gauss (3.3), on établit l'équation de conservation de la densité de charge :

$$\nabla \cdot \vec{J} + \frac{\partial \rho_e}{\partial t} = 0 \quad (3.5)$$

Dans la pratique, ce bilan s'applique dans l'antenne du satellite. Hors de l'antenne, on peut ignorer le courant et la charge et la divergence du champ de déplacement électrique  $\vec{D}$  devient également nulle :

$$\nabla \cdot \vec{D} = 0 \quad (3.6)$$

À ces équations on doit ajouter les équations constitutives qui caractérisent le milieu de propagation, qui vont établir des rapports entre  $\vec{D}$  et  $\vec{E}$  et entre  $\vec{B}$  et  $\vec{H}$ . Dans l'atmosphère

terrestre, le champ magnétique est pratiquement égal à l'induction magnétique, donc on peut considérer :

$$\vec{B} = \vec{H} \quad (3.7)$$

Le rapport le plus important dans l'optique géométrique est celui qui connecte le champ électrique avec le déplacement électrique :

$$\vec{D} = \varepsilon(\vec{x}, t) \vec{E} \quad (3.8)$$

La permittivité diélectrique  $\varepsilon$  contient toute l'information nécessaire pour décrire la propagation dans l'ionosphère. Nous écrivons cette quantité comme suit :

$$\varepsilon(\vec{x}, t) = \varepsilon_0 + \Delta\varepsilon(\vec{x}, t) \quad (3.9)$$

où  $\varepsilon_0$  est la valeur moyenne de  $\varepsilon$  qu'on identifie avec la valeur de la constante diélectrique dans le vide. Dans la basse atmosphère, la différence de la valeur  $\varepsilon_0$  par rapport à l'unité est de 300 parties par million. Donc on peut poser  $\varepsilon_0 = 1$ . Pour arriver à l'équation d'ondes on doit appliquer le rotationnel à l'équation de l'induction (3.1) et utiliser l'équation d'Ampère (3.2) pour exprimer le rotationnel de  $\vec{H}$  par rapport aux vecteurs de déplacement et courant électriques :

$$\vec{\nabla} \times \vec{\nabla} \times \vec{E} = -\frac{1}{c^2} \frac{\partial^2}{\partial t^2} (\varepsilon \vec{E}) - \frac{4\pi}{c} \frac{\partial \vec{J}}{\partial t} \quad (3.10)$$

On utilise (3.6) pour exprimer la divergence de  $\vec{D}$  par rapport au gradient de la permittivité diélectrique :

$$\nabla \cdot \vec{D} = \nabla \cdot (\varepsilon \vec{E}) = \vec{E} \cdot \vec{\nabla} \varepsilon + \varepsilon \nabla \cdot \vec{E} = 0 \quad (3.11)$$

alors :

$$\nabla \cdot \vec{E} = -\vec{E} \cdot \vec{\nabla} (\log \varepsilon). \quad (3.12)$$

En simplifiant le double rotationnel, on arrive finalement à l'équation d'ondes :

$$\nabla^2 \vec{E} - \frac{1}{c^2} \frac{\partial^2}{\partial t^2} [(1 + \Delta\varepsilon) \vec{E}] = \frac{4\pi}{c} \frac{\partial \vec{J}}{\partial t} - \vec{\nabla} \{ \vec{E} \cdot \vec{\nabla} [\log(1 + \Delta\varepsilon)] \} \quad (3.13)$$

Le dernier terme de gradient décrit les changements de polarisation qui sont négligeables pour la propagation atmosphérique. Si les fluctuations de la permittivité diélectrique ne dépendent pas du temps, on peut effectuer une décomposition spectrale de Fourier. Cette hypothèse est justifiée car  $\Delta\varepsilon$  change très lentement par rapport au champ électrique dans les milieux atmosphériques. Si on considère que la source émet à une fréquence particulière on a :

$$\vec{J}(\vec{x}, t) = \vec{J}(\vec{x}) e^{-i\omega t} \quad (3.14)$$

$$\vec{E}(\vec{x}, t) = \vec{E}(\vec{x}) e^{-i\omega t} \quad (3.15)$$

qui est une solution à variables séparées. En dérivant les termes exponentiels par rapport au temps, on arrive à une équation de Helmholtz 3D qui dépend d'une fréquence de l'onde porteuse  $\omega = 2\pi f$  :

$$\nabla^2 \vec{E}(\vec{x}) + k^2[1 + \Delta\varepsilon(\vec{x}, t)]\vec{E}(\vec{x}) = -i4\pi k \vec{J}(\vec{x}) \quad (3.16)$$

avec  $k = \frac{\omega}{c}$ . Puisque les composantes du vecteur du champ électrique ne sont pas mêlées, il n'est pas nécessaire d'utiliser la notation vectorielle pour décrire l'amplitude du champ électrique :

$$\nabla^2 E(\vec{x}) + k^2[1 + \delta\varepsilon(\vec{x}, t)] E(\vec{x}) = -i4\pi k J(\vec{x}) \quad (3.17)$$

Cette équation est le point de départ des deux approximations (Rytov et optique géométrique) que nous utilisons. Si l'émetteur était très éloigné des points où on veut calculer le champ, et si  $\varepsilon(\vec{x})$  était constant la solution serait une onde plane :

$$E = E_0 e^{ikz} \quad (3.18)$$

Ce résultat justifie la prise de l'Ansatz suivant, où on écrit la solution comme le produit d'une amplitude par un terme de phase exponentielle :

$$E(\vec{x}) = A(\vec{x}) e^{ik\Psi(\vec{x})} \quad (3.19)$$

La fonction  $\Psi(\vec{x})$ , à partir de laquelle on peut obtenir la phase en multipliant par  $k$ , s'appelle *eikonale*. L'hypothèse basique de l'optique géométrique considère que les conditions aux bords et la permittivité électrique changent très lentement par rapport à la longueur d'onde. En utilisant cette hypothèse, on fait un développement asymptotique du terme d'amplitude en puissances inverses du nombre d'onde :

$$A(\vec{x}) = A_1(\vec{x}) + \frac{A_2(\vec{x})}{ik} + \frac{A_3(\vec{x})}{(ik)^2} + \dots \quad (3.20)$$

En injectant cet Ansatz avec le développement asymptotique dans l'équation de Helmholtz, on arrive à :

$$\begin{aligned} [\nabla^2 + k^2\varepsilon(\vec{x})] E &= e^{ik\Psi} \{k^2[\varepsilon(\vec{x}) - (\vec{\nabla}\Psi)^2] \\ &+ ik[2\vec{\nabla}\Psi \cdot \vec{\nabla}A_1 + A_1\nabla^2\Psi] + [2\vec{\nabla}\Psi \cdot \vec{\nabla}A_2 + A_2\nabla^2\Psi + \nabla^2A_1] \\ &+ \frac{i}{k}[2\vec{\nabla}\Psi \cdot \vec{\nabla}A_3 + A_3\nabla^2\Psi + \nabla^2A_2]\} = source \end{aligned} \quad (3.21)$$

Les coefficients doivent disparaître individuellement car cet équation doit se vérifier pour toutes les valeurs de la fréquence. Le nombre d'onde est suffisamment grand, donc on peut se permettre ne travailler qu'avec les deux premiers termes. Ceci donne à deux équations différentielles dont la solution détermine  $A_1$  et  $\Psi$ .

L'équation de l'eikonale apparaît lorsque on fait annuler le terme en  $k^2$  :

$$(\vec{\nabla}\Psi)^2 = \varepsilon \quad (3.22)$$

On note que l'eikonale ne dépend que de la permittivité électrique et du type d'onde transmise. L'amplitude du signal peut s'obtenir en annulant le coefficient du terme en  $k$  et en résolvant l'équation de transport, qui dépend justement de l'eikonale :

$$2\vec{\nabla}\Psi \cdot \vec{\nabla}A_1 + A_1 \nabla^2\Psi = 0 \quad (3.23)$$

Ces équations ne donnent pas information sur la longueur d'onde, donc la solution ne peut pas expliquer les effets diffractifs de l'amplitude.

La première tâche est d'obtenir  $\Psi$ . On peut le faire grâce à la méthode des caractéristiques, qui vont donner les chemins parcourus par les signaux sous forme de rayons émis par le satellite. On arrive à une première expression pour  $\Psi$ , qui est donnée comme l'intégrale de la racine carrée de  $\varepsilon(\vec{x})$  le long de la trajectoire d'un rayon :

$$\Psi(R) = \int_T^R \sqrt{\varepsilon[\vec{x}(s)]} ds \quad (3.24)$$

avec  $s$  l'abscisse curviligne le long de la courbe du rayon,  $R$  la valeur de l'abscisse curviligne correspondant au point du récepteur,  $T$  la valeur pour le point d'émission (c'est-à-dire, le satellite). Cette expression (qui donne la longueur du chemin optique) vérifie l'équation :

$$\vec{\nabla}\Psi = \vec{t} \frac{\partial\Psi}{\partial R} = \vec{t} \sqrt{\varepsilon[\vec{x}(R)]} \quad (3.25)$$

où  $\vec{t}$  représente le vecteur unitaire tangent au rayon. Le problème se réduit alors à calculer les trajectoires des rayons. En calculant le carré de cette expression, on retrouve l'équation eikonale. Le vecteur tangent peut s'exprimer simplement comme la dérivée de la trajectoire du rayon par rapport à l'abscisse curviligne :

$$\vec{t} = \frac{d\vec{x}}{ds} \quad (3.26)$$

En remplaçant cette égalité dans l'équation du gradient de l'eikonale on obtient :

$$\vec{\nabla}\Psi = \frac{d\vec{x}}{ds} \sqrt{\varepsilon(\vec{x})} \quad (3.27)$$

En dérivant encore une fois par rapport à l'abscisse :

$$\begin{aligned} \frac{d}{ds} \left( \sqrt{\varepsilon(\vec{x})} \frac{d\vec{x}}{ds} \right) &= \frac{d}{ds} (\vec{\nabla}\Psi) \\ &= \left( \frac{d\vec{x}}{ds} \cdot \nabla \right) (\vec{\nabla}\Psi) \\ &= \frac{1}{\sqrt{\varepsilon(\vec{x})}} (\vec{\nabla}\Psi \cdot \vec{\nabla}) (\vec{\nabla}\Psi) \end{aligned} \quad (3.28)$$

Compte-tenu l'identité vectorielle :

$$(\vec{\nabla}\Psi \cdot \vec{\nabla}) (\vec{\nabla}\Psi) = \frac{1}{2} \vec{\nabla}(\vec{\nabla}\Psi)^2 \quad (3.29)$$

et en sachant que le gradient de  $\varepsilon$  peut s'écrire comme :

$$\vec{\nabla}(\sqrt{\varepsilon}) = \frac{1}{2} \frac{1}{\sqrt{\varepsilon}} \vec{\nabla}\varepsilon \quad (3.30)$$

on arrive à l'équation différentielle dont la solution donne la trajectoire d'un rayon :

$$\frac{d}{ds} \left( \sqrt{\varepsilon(\vec{x})} \frac{d\vec{x}}{ds} \right) = \vec{\nabla} \left[ \sqrt{\varepsilon(\vec{x})} \right] \quad (3.31)$$

Cette équation peut s'exprimer aussi par rapport à l'indice de réfraction  $n$ . On sait que si le milieu n'est pas magnétique alors  $\frac{\varepsilon}{\varepsilon_0} = n^2$ . En remplaçant  $\sqrt{\varepsilon}$  par  $n\sqrt{\varepsilon_0}$  on a :

$$\frac{d}{ds} \left[ n(\vec{x}) \frac{d\vec{x}}{ds} \right] = \vec{\nabla} [n(\vec{x})] \quad (3.32)$$

car les facteurs constants  $\sqrt{\varepsilon_0}$  se simplifient. Cette équation es vraiment un système de trois équations différentielles vectorielles qu'il faut résoudre pour chaque composante spatiale.

Pour résoudre, il faut donner des conditions initiales pour  $\vec{x}$  (qui correspond aux coordonnées du point émetteur du rayon) et  $\vec{t} = \frac{d\vec{x}}{ds}$  (qui correspond à la tangente initiale du rayon). Une fois qu'on a trouvé la trajectoire du rayon, l'eikonale (et donc la phase) peut s'obtenir à partir de (3.24).

Du fait que le satellite va se déplacer très vite dans l'orbite et que la phase  $S$  dépend de la longitude de la trajectoire parcourue, on a besoin de la séparer en deux termes :

$$S = k\Psi = S_0 + \varphi \quad (3.33)$$

où le terme  $S_0$  ne dépend que de la distance (et qui correspond à la phase à l'arrivée d'un rayon se propageant dans le vide, dont la valeur peut être obtenue analytiquement) et  $\varphi$  qui est dû aux irrégularités de  $\varepsilon$ . Pour les calculs de la variance de la phase, on va supprimer le terme  $S_0$  de la phase calculée, qui pour une distance  $d$  entre émetteur et récepteur vaut  $k\varepsilon_0 d = kd$ . En plus, même si on ne peut donner une direction initiale aux rayons avec les conditions initiales de (3.31), on peut pas prévoir le point d'arrivée car les perturbations de la permittivité électrique vont dévier les trajectoires. Dans la pratique, on lance plusieurs rayons dirigés vers les environs du récepteur et on fait la moyenne des phases des rayons qui tombent dans une certaine distance du récepteur.

Pour le calcul de l'amplitude, on doit utiliser l'équation du transport (3.23). Puisque le gradient de l'eikonale est toujours aligné avec la tangente des rayons, on peut la réécrire comme :

$$A_1 \nabla^2 \Psi = -2\vec{\nabla}\Psi \cdot \vec{\nabla}A_1 = -2 \left( \vec{t} \frac{\partial \Psi}{\partial s} \right) \cdot \vec{\nabla}A_1 = -2\sqrt{\varepsilon[\vec{x}(s)]} \frac{\partial A}{\partial s} \quad (3.34)$$

Cette équation aux dérivées partielles peut se résoudre avec l'intégrale de ligne :

$$A_1(R) = A_1(T) e^{-\frac{1}{2} \int_T^R \frac{\nabla^2 \Psi}{\sqrt{\varepsilon[\vec{x}(s)]}} ds} \quad (3.35)$$

En remplaçant  $\Psi$  par l'intégrale (3.24) :

$$A_1(R) = A_1(T) e^{-\frac{1}{2} \int_T^R \nabla^2 \left( \int_T^s \sqrt{\varepsilon[\vec{x}(s')] ds'} \right) \frac{ds}{\sqrt{\varepsilon[\vec{x}(s)]}}} \quad (3.36)$$

Cette expression peut se simplifier en explicitant le laplacien (séparation des dérivées prises parallèlement et orthogonalement à la trajectoire du rayon :

$$\nabla^2 = \frac{\partial^2}{\partial s^2} + \nabla_{\perp}^2 \quad (3.37)$$

Alors :

$$A_1(R) = A_1(T) e^{-\frac{1}{2} \frac{\partial^2}{\partial s^2} \int_T^R (\int_T^s \sqrt{\varepsilon[\vec{x}(s')]} ds') \frac{ds}{\sqrt{\varepsilon[\vec{x}(s)]}}} \quad (3.38)$$

$$\times e^{-\frac{1}{2} \int_T^R \nabla_{\perp}^2 (\int_T^s \sqrt{\varepsilon[\vec{x}(s')]} ds') \frac{ds}{\sqrt{\varepsilon[\vec{x}(s)]}}} \quad (3.39)$$

Du fait que  $\nabla_{\perp}^2 \Psi = \nabla \cdot_{\perp} (\vec{\nabla}_{\perp} \Psi(s))$ , la deuxième exponentielle est égale à 1 car le gradient est toujours orienté vers la tangente du rayon. En résolvant l'intégrale de la première exponentielle, on a :

$$A_1(R) = A_1(T) e^{-\frac{1}{4} \ln \left\{ \frac{\varepsilon[\vec{x}(R)]}{\varepsilon[\vec{x}(T)]} \right\}} \quad (3.40)$$

$$A_1(R) \{\varepsilon[\vec{x}(R)]\}^{\frac{1}{4}} = A_1(T) \{\varepsilon[\vec{x}(T)]\}^{\frac{1}{4}} \quad (3.41)$$

La conséquence la plus importante de cette expression est qu'on peut récrire l'équation du transport (3.23) comme :

$$\nabla \cdot (I \vec{\nabla} \Psi) = 0 \quad (3.42)$$

où on définit l'intensité  $I$  du signal comme le carré de l'amplitude  $A_1$ . En remplaçant le gradient de  $\Psi$  par  $\vec{t} \sqrt{\varepsilon}$ , en intégrant le long d'un tube de rayons et en appliquant le théorème de la divergence de Gauss, on déduit que :

$$I(s) \sqrt{\varepsilon[\vec{x}(s)]} \mathbf{A} = \text{const.} \quad (3.43)$$

avec  $\mathbf{A}$  l'aire de la section du tube de rayons. C'est-à-dire, l'énergie dans un front d'ondes se conserve. Grâce à ce rapport on peut obtenir l'amplitude à l'arrivée en connaissant l'amplitude dans un point quelconque du tube de rayons (dont l'aire de la section soit aussi connue).

## 3.2 Description et mise en œuvre du traceur de rayons

L'algorithme central de la mise en œuvre de l'optique géométrique est le traceur de rayons (qui se trouve dans les fichiers `src/go/goptics/goptics.c` et `src/go/raycast.c`). Il résout numériquement l'équation (3.32). Puisqu'il s'agit d'un système d'équations de deuxième ordre, il est intéressant de le découpler en deux systèmes de premier ordre :

$$\frac{d\vec{u}}{ds} = \vec{\nabla} n(\vec{x}) \quad (3.44)$$

$$\frac{d\vec{x}}{ds} = \frac{\vec{u}}{n(\vec{x})} \quad (3.45)$$

De plus, puisque l'eikonale s'obtient en intégrant le long de la trajectoire, on peut la résoudre en même temps en ajoutant cette équation scalaire à l'ensemble d'équations précédent :

$$\frac{d\Psi}{ds} = n \quad (3.46)$$

Le problème est donc de résoudre un système de sept équations différentielles ordinaires de premier ordre dont la variable indépendante est l'abscisse curviligne. Les conditions initiales valent :

$$\vec{x}(0) : \text{point d'origine du rayon.} \quad (3.47)$$

$$\vec{u}(0) : \text{tangente à l'origine multiplié par } n[\vec{x}(0)]. \quad (3.48)$$

$$\Psi(0) = 0 \quad (3.49)$$

$$s_0 = 0 \quad (3.50)$$

Ce système est résolu en coordonnées cartésiennes (`src/go/goptics/goptics.c`) grâce à la méthode de Runge-Kutta 4 (`src/go/rkutta4/rkutta4.c`).

Un des buts du module traceur est de lancer un rayon vers un point des environs du récepteur. Pour expliciter ce domaine géométrique, on utilise un système de coordonnées (appelé *système de coordonnées de l'antenne*) qui a par origine la position du récepteur.

Dans ce système, on définit une base vectorielle tridimensionnelle, dont le vecteur  $\hat{k}$  (qui correspond à la composante  $z$  de ce système de coordonnées) est aligné avec la ligne de vue récepteur-satellite, et dirigé vers le satellite. Le vecteur  $\hat{i}$  (qui correspond à la composante  $x$  de ce système de coordonnées) se calcule en normalisant le produit vectoriel  $\hat{z} \times \hat{k}^1$ , avec  $\hat{z}$  la direction Nord de l'axe de rotation de la Terre. La troisième composante  $\hat{j}$  (la composante  $y$ ) se calcule en normalisant  $\hat{k} \times \hat{i}$ . Pour ce système,  $\hat{i}$  est orienté vers l'Est,  $\hat{j}$  vers le Nord.

À partir de ce système, on peut se déplacer orthogonalement à la ligne de vue grâce aux vecteurs  $\hat{i}$  et  $\hat{j}$ . On prend un déplacement aléatoire dans ce plan orthogonal à  $\hat{k}$  et à partir de ce point on calcule le vecteur pointant le satellite. Ceci définit la tangente initiale du rayon.

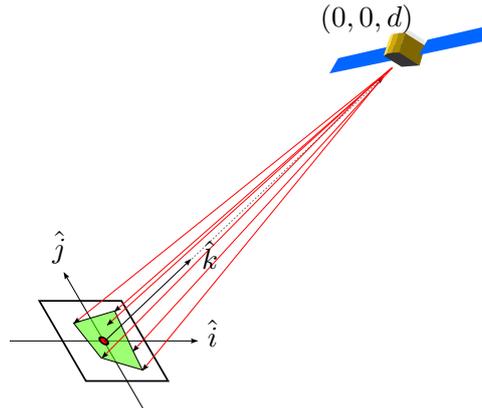


FIGURE 3.1 – Système de coordonnées du plan de l'antenne. En vert : aire couverte par les points d'arrivée.

1. Ceci ne fonctionne que pour des satellites dont la ligne de vue n'est pas parallèle à l'axe de rotation de la Terre (axe  $z$ ). Heureusement, pour la constellation du système Galileo, les orbites sont tellement éloignées que cette configuration n'arrivera jamais

Le critère d'arrêt utilisé par le traceur vérifie si le rayon traverse le plan du récepteur. Soit  $s_m$  la première valeur de l'abscisse curviligne telle que la composante  $z$  de  $\vec{x}(s_m)$  par rapport au système de coordonnées de l'antenne est négative (c'est-à-dire, le premier point de la trajectoire qui est derrière le plan de l'antenne). Le traceur s'arrête une fois qu'on est arrivé à ce point  $\vec{x}(s_m)$ . Cependant, dans la plupart des cas, cette valeur de  $\vec{x}$  est très éloigné du récepteur, avec une distance très grande par rapport à la longueur d'onde. Si on suppose la solution discrète aux équations de l'optique comme constantes par morceaux, il ne faut que calculer l'intersection du segment de droite  $[\vec{x}(s_{m-1})\vec{x}(s_m)]$  avec le plan de l'antenne.

Soit  $\vec{o}$  les coordonnées du récepteur. Tout point  $\vec{x}_p$  de ce plan vérifie :

$$\hat{k} \cdot (\vec{x}_p - \vec{o}) = 0 \quad (3.51)$$

$$\hat{k} \cdot \vec{x}_p = \hat{k} \cdot \vec{o} \quad (3.52)$$

Le segment de droite  $[\vec{x}(s_{m-1})\vec{x}(s_m)]$  peut s'exprimer paramétriquement comme :

$$\vec{x} = \vec{x}(s_{m-1}) + t\Delta\vec{x} \text{ (avec } \Delta\vec{x} = \vec{x}(s_m) - \vec{x}(s_{m-1})) \quad (3.53)$$

Alors, pour trouver  $t$  tel que le point  $\vec{x}$  du segment se trouve aussi dans le plan de l'antenne, il faut combiner (3.51) avec (3.53) :

$$\hat{k} \cdot (\vec{x}(s_{m-1}) + t\Delta\vec{x}) = \hat{k} \cdot \vec{o} \quad (3.54)$$

$$\hat{k} \cdot \vec{x}(s_{m-1}) + t\hat{k} \cdot \Delta\vec{x} = \hat{k} \cdot \vec{o} \quad (3.55)$$

$$t = \frac{\hat{k} \cdot [\vec{o} - \vec{x}(s_{m-1})]}{\hat{k} \cdot \Delta\vec{x}} \quad (3.56)$$

On utilise aussi cette valeur  $t$  pour trouver la valeur de l'eikonale dans le récepteur comme :

$$\Psi = \Psi(s_{m-1}) + t\Delta\Psi \text{ (avec } \Delta\Psi = \Psi(s_m) - \Psi(s_{m-1})) \quad (3.57)$$

Enfin, après avoir lancé plusieurs rayons et calculé les intersections par rapport au système de coordonnées de l'antenne, on réalise un parcours de Graham 2-D sur les points d'arrivée et on obtient une mesure de l'aire couverte par le faisceau de rayons. On utilise cette aire et l'aire de la section du faisceau de rayons à certaine distance du satellite dans le vide.

Pour le calcul de l'aire de référence, il suffit de réaliser le parcours de Graham des points d'arrivée des rayons dans le vide, calculer l'aire et la diviser par  $\left(\frac{d-d_{ref}}{d}\right)^2$ , avec  $d_{ref}$  la distance de référence et  $d$  la distance entre la station au sol et le satellite.

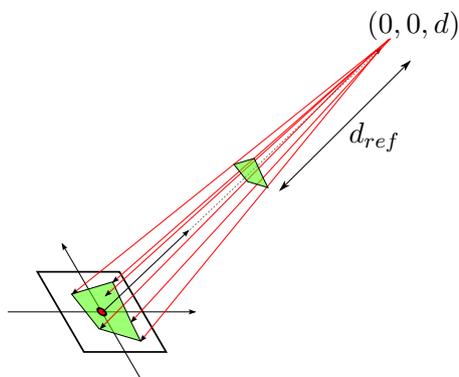


FIGURE 3.2 – Aire d'arrivée, aire de référence et distance de référence.

Pour la phase, il faut calculer  $\varphi = k(\Psi - \Psi_{vide}) = k(\Psi - d)$  pour chaque rayon et calculer la moyenne de  $\varphi$ .

### 3.3 Limitations de l'optique géométrique

Néanmoins, l'optique géométrique est restreinte par trois conditions de validité[2]. La première condition, appelée **condition de milieu lisse** restreint la petite échelle des turbulences du plasma :

$$\lambda \ll \ell_0 \tag{3.58}$$

Heureusement, Galileo travaille autour de la fréquence de 1.6 GHz et du fait que  $\lambda = 0.18m \ll 2m < \ell_0 < 5m$ [3], la condition de milieu lisse est vérifiée.

Lorsqu'on fait le calcul de l'amplitude, il faut travailler avec un ensemble de rayons qui traversent une surface. L'amplitude se décrit en termes de proximité des rayons entre eux. La **condition de caustique** impose des restrictions sur la vitesse de convergence (ou divergence) des rayons. Si l'espacement entre rayons varie très rapidement le long d'une longueur d'onde, l'optique géométrique n'est plus valable pour calculer les perturbations de l'amplitude. Cette condition peut s'écrire aussi comme  $\langle \chi \rangle^2 \ll 1$ , avec  $\chi$  la fluctuation logarithmique de l'amplitude, c'est-à-dire : l'optique géométrique n'est pas capable de prédire les variations de l'amplitude dans les situations de forte scintillation.

Et finalement, les **conditions de réfraction** donnent des conditions par rapport à la longueur de Fresnel  $\sqrt{\lambda R}$ , avec  $R$  la distance de la turbulence. Du fait que la couche F de l'ionosphère (où se trouvent la plupart des perturbations qui affectent les signaux) se trouve à une hauteur maximale de 500 km, la longueur de Fresnel vaut autour de 300 mètres. Pour les calculs de variances de phase, la condition de réfraction est  $\sqrt{\lambda R} \ll L_0$  avec  $L_0$  la grande échelle de turbulence. Pour ce type de problèmes,  $5km < L_0 < 20km$  donc l'optique géométrique est valable pour les calculs de phase. Toutefois, pour l'amplitude la condition de réfraction est beaucoup plus restrictive :

$$\sqrt{\lambda R} \ll \ell_0 \tag{3.59}$$

Et cette condition ne va jamais se vérifier pour des problèmes réels.

Ces conditions indiquent que l'approximation de l'optique géométrique donne des résultats très précis pour les calculs de phase, mais elle sera inutile pour l'amplitude. Cette restriction motive la recherche d'une méthode alternative (l'approximation de Rytov) qui puisse compléter cette carence.

## Chapitre 4

# L'approximation de Rytov

L'approximation de Rytov est capable de suppléer à la carence de l'optique géométrique au moment d'obtenir des bonnes mesures de la fluctuation de l'amplitude. Pour comprendre cette approximation, il faut expliquer tout d'abord l'approximation de Born dont l'approximation de Rytov se déduit.

L'approximation de Born est une technique qui a été utilisée pour résoudre des problèmes de la mécanique quantique. Son application à la propagation électromagnétique se justifie par de la petitesse des fluctuations de la permittivité diélectrique relative :

$$\delta\varepsilon_r \ll 1 \tag{4.1}$$

Ceci permet de faire un développement du champ électrique selon les puissances de  $\delta\varepsilon_r$  :

$$E = E_0 + E_1 + E_2 + E_3 + \dots \tag{4.2}$$

où le terme  $E_0$  représente le champ électrique obtenu pour un milieu non perturbé,  $E_1$  le champ électrique dû à la dispersion simple du fait des perturbations de la permittivité,  $E_2$  le champ dû à la dispersion double des ondes de  $E_1$  qui retrouvent les mêmes perturbations, etc.

Les expressions de cet approche sont beaucoup plus complexes que celles de l'optique géométrique car les intégrales de ligne vont être remplacées par des intégrales de volume. En plus, la validité de cette approximation est restreinte à :

$$\langle \varphi^2 \rangle + \langle \chi^2 \rangle < 1 \tag{4.3}$$

qui s'appelle condition de Born, où  $\varphi$  est la phase et  $\chi$  est la variation logarithmique de l'amplitude. Si on considère le cas de dispersion faible, cette restriction se rapporte surtout à la phase, dont les fluctuations vont croître linéairement avec la fréquence. Rytov affaiblit cette condition en faisant un développement du champ électrique qui est fondamentalement une généralisation de l'optique géométrique qui prend une eikonale complexe, appelée cette fois **fonction de substitution** (ou *surrogate function* selon la littérature en anglais[4]).

De même, cette fonction de substitution s'exprime encore une fois à partir d'un développement en puissances de  $\delta\varepsilon_r$ . Les termes de cette série peuvent s'exprimer en combinant algébriquement les termes de la série de Born, d'où l'importance de cette série. Cette fois, la condition de validité pour Rytov se réduit à :

$$\langle \chi^2 \rangle < 1 \tag{4.4}$$

## 4.1 Dérivation de l'approximation de Rytov

Le point de départ est encore une fois l'équation (3.17). La difficulté de la résolution de cette équation pour une permittivité diélectrique qui n'est pas constante est due au fait que  $\delta\varepsilon$  multiplie la fonction qu'on veut trouver. Rytov simplifie cette difficulté en transformant les perturbations de la permittivité en un terme source additionnel. L'Ansatz proposé par Rytov exprime le champ électrique comme le produit d'un champ électrique non perturbé par un terme exponentiel complexe de la fonction de substitution  $\psi$  :

$$E(\vec{x}) = E_0(\vec{x})e^{\psi(\vec{x})} \quad (4.5)$$

En injectant cette Ansatz dans (3.17) et en utilisant l'identité :

$$\nabla^2(FG) = F\nabla^2G + 2\vec{\nabla}F \cdot \vec{\nabla}G + G\nabla^2F \quad (4.6)$$

on montre que :

$$E_0\nabla^2(e^\psi) + 2\vec{\nabla}E_0 \cdot \vec{\nabla}(e^\psi) + k^2\delta\varepsilon E_0 e^\psi = -4i\pi k \vec{J}(\vec{x}) - e^\psi \nabla^2 E_0 - k^2 E_0 e^\psi \quad (4.7)$$

On remarque que les termes à droite  $-e^\psi \nabla^2 E_0 - k^2 E_0 e^\psi$  sont proportionnels au terme source de l'équation des ondes non perturbée. Ceci nous permet de les remplacer et alors :

$$\begin{aligned} E_0\nabla^2(e^\psi) + 2\vec{\nabla}E_0 \cdot \vec{\nabla}(e^\psi) + k^2\delta\varepsilon E_0 e^\psi &= -4i\pi k \vec{J}(\vec{x}) + e^\psi 4i\pi k \vec{J}(\vec{x}) \\ E_0\nabla^2(e^\psi) + 2\vec{\nabla}E_0 \cdot \vec{\nabla}(e^\psi) + k^2\delta\varepsilon E_0 e^\psi &= -4i\pi k \vec{J}(\vec{x}) \left(1 - e^\psi\right) \end{aligned} \quad (4.8)$$

En utilisant les identités :

$$\vec{\nabla}e^\psi = e^\psi \vec{\nabla}\psi, \quad (4.9)$$

$$\nabla^2 e^\psi = \nabla \cdot (\vec{\nabla}e^\psi) = \vec{\nabla}e^\psi \cdot \vec{\nabla}\psi + e^\psi \nabla^2\psi = e^\psi (\vec{\nabla}\psi)^2 + e^\psi \nabla^2\psi \quad (4.10)$$

on arrive à :

$$\begin{aligned} e^\psi \{E_0[(\vec{\nabla}\psi)^2 + \nabla^2\psi] + 2\vec{\nabla}E_0 \cdot \vec{\nabla}\psi + k^2\delta\varepsilon E_0\} \\ = -4i\pi k \vec{J}(\vec{x}) \left(1 - e^\psi\right), \\ E_0[(\vec{\nabla}\psi)^2 + \nabla^2\psi] + 2\vec{\nabla}E_0 \cdot \vec{\nabla}\psi + k^2\delta\varepsilon E_0 = -4i\pi k \vec{J}(\vec{x}) \left(1 - e^\psi\right), \\ (\vec{\nabla}\psi)^2 + \nabla^2\psi + 2\vec{\nabla}(\ln E_0) \cdot \vec{\nabla}\psi + k^2\delta\varepsilon = -4i\pi k \vec{J}(\vec{x}) \left(\frac{e^{-\psi} - 1}{E_0}\right) \end{aligned} \quad (4.11)$$

Puisqu'on se place loin de l'émetteur, on peut ignorer le terme source et on arrive a une équation non linéaire pour  $\psi$  :

$$\nabla^2\psi + (\vec{\nabla}\psi)^2 + 2\vec{\nabla}(\ln E_0) \cdot \vec{\nabla}\psi = -k^2\delta\varepsilon_r(\vec{x}, t) \quad (4.12)$$

qui permet de faire intervenir la permittivité seulement comme un terme source. Afin de supprimer le logarithme, il est plus pratique d'exprimer  $E_0$  comme  $e^{\psi_0(\vec{x})}$ . Avec cette formulation, le champ électrique s'exprime alors comme  $E(\vec{x}) = e^{\psi_0+\psi}$  et l'équation (4.12) devient :

$$\nabla^2\psi + (\vec{\nabla}\psi)^2 + 2\vec{\nabla}\psi_0 \cdot \vec{\nabla}\psi = -k^2\delta\varepsilon_r(\vec{x}, t) \quad (4.13)$$

On développe la solution de (4.13) en puissances croissantes de  $\delta\varepsilon_r$  :

$$\psi(\vec{x}) = \psi_1(\vec{x}) + \psi_2(\vec{x}) + \psi_3(\vec{x}) + \psi_4(\vec{x}) + \dots \quad (4.14)$$

En injectant ce développement dans l'équation (4.13) et en séparant les termes selon les puissances croissantes de  $\delta\varepsilon_r$ , on arrive à l'ensemble d'équations suivant :

$$\nabla^2\psi_0 + (\vec{\nabla}\psi_0)^2 + k^2 = 0 \quad (4.15)$$

$$\nabla^2\psi_1 + 2\vec{\nabla}\psi_0 \cdot \vec{\nabla}\psi_1 + k^2\delta\varepsilon = 0 \quad (4.16)$$

$$\nabla^2\psi_2 + 2\vec{\nabla}\psi_0 \cdot \vec{\nabla}\psi_2 + (\vec{\nabla}\psi_1)^2 = 0 \quad (4.17)$$

$$\nabla^2\psi_3 + 2\vec{\nabla}\psi_0 \cdot \vec{\nabla}\psi_3 + 2\psi_1 \cdot \vec{\nabla}\psi_2 = 0 \quad (4.18)$$

$$\nabla^2\psi_4 + 2\vec{\nabla}\psi_0 \cdot \vec{\nabla}\psi_4 + 2\psi_1 \cdot \vec{\nabla}\psi_3 + (\vec{\nabla}\psi_2)^2 = 0 \quad (4.19)$$

$$\dots \quad (4.20)$$

Certes dans la plupart des cas il suffit considérer seulement le premier terme, mais Wheelon a montré[4] que le calcul de la variance des fluctuations d'intensité ne peut se décrire correctement qu'en utilisant l'approximation de quatrième ordre. Le but est alors d'obtenir des expressions pour  $\psi_n$   $n = [0, 4]$ .

#### 4.1.1 La solution de Rytov au premier ordre

La solution au premier ordre s'obtient à partir du terme  $\psi_1$  en résolvant l'équation (4.16). Si on fait la substitution :

$$\psi_1(\vec{x}) = Q(\vec{x})e^{\psi_0(\vec{x})} \quad (4.21)$$

on peut transformer l'équation (4.16) en une équation de Helmholtz pour  $Q$  :

$$\nabla^2Q + Q[-\nabla^2\psi_0 - (\vec{\nabla}\psi_0)^2] = -k^2e^{\psi_0}\delta\varepsilon_r(\vec{x}, t) \quad (4.22)$$

car le terme entre crochets est simplement  $k^2$  selon . La solution de cette équation peut s'exprimer avec l'aide de sa solution élémentaire : la fonction Green de l'équation de Helmholtz 3-D :

$$Q(\vec{x}) = -k^2 \iiint G(\vec{x}, \vec{r})\delta\varepsilon_r(\vec{r}, t)e^{\psi_0(\vec{r})}d^3r \quad (4.23)$$

où la fonction de Green vaut :

$$G(\vec{x}, \vec{y}) = \frac{e^{ik\|\vec{x}-\vec{y}\|}}{4\pi\|\vec{x}-\vec{y}\|} \quad (4.24)$$

En remplaçant l'exponentielle de  $\psi_0$  par  $E_0$  on arrive finalement à une expression pour  $\psi_1$  :

$$\psi_1 = -k^2 \iiint G(\vec{x}, \vec{r})\delta\varepsilon_r(\vec{r}, t)\frac{E_0(\vec{r})}{E_0(\vec{x})}d^3r \quad (4.25)$$

Dans ce cas, le terme  $\psi_1$  est équivalent au premier terme de la série de Born (qu'on appelle  $B_1$ ), mais cette identité n'est plus vraie pour les termes d'ordre supérieur. Le champ électrique comme solution de Rytov au premier ordre s'exprime alors comme :

$$E_1(\vec{x}) = E_0(\vec{x})e^{B_1} \quad (4.26)$$

### 4.1.2 Les solutions d'ordre supérieur

C'est à partir de l'ordre 2 que les approximations de Rytov et de Born commencent à diverger. Toutefois, les deux approximations peuvent être liées de la façon suivante[4] :

$$\psi_1 = B_1 \quad (4.27)$$

$$\psi_2 = B_2 - \frac{1}{2}B_1^2 \quad (4.28)$$

$$\psi_3 = B_3 + \frac{1}{3}B_1^3 - B_1B_2 \quad (4.29)$$

$$\psi_4 = B_4 - B_1B_3 + B_2(B_1^2 - \frac{1}{2}B_2) - \frac{1}{2}B_1^4 \quad (4.30)$$

Ce rapport est important puisque le terme  $B_n, n > 1$  peut s'obtenir à partir du terme précédent en évaluant l'intégrale :

$$B_n(\vec{x}) = -k^2 \iiint G(\vec{x}, \vec{r}) \delta \varepsilon_r(\vec{r}, t) B_{n-1}(\vec{r}) \frac{E_0(\vec{r})}{E_0(\vec{x})} d^3r \quad (4.31)$$

La solution de quatrième ordre peut s'écrire comme :

$$E_4 = E_0 e^{\psi_1 + \psi_2 + \psi_3 + \psi_4} \quad (4.32)$$

On remarque que pour évaluer  $E_4$  en un point  $\vec{x}$ , il faut seulement évaluer les intégrales  $B_1$  à  $B_4$  en un seul point  $\vec{x}$ . Néanmoins, les intégrales  $B_1$  à  $B_3$  doivent aussi être évaluées pour tout  $\vec{x}$  tel que la valeur de la variation de la permittivité ne soit pas nulle, car  $B_2$  à  $B_4$  dépendent globalement de  $B_1$  à  $B_3$ .

### 4.1.3 Mise en œuvre de la méthode de Rytov

Puisqu'évaluer des intégrales triples en chaque point du domaine est une opération très coûteuse, il faut optimiser l'évaluation de ces intégrales pour chaque type de système de coordonnées. La mise en œuvre actuelle ne travaille qu'avec le système de potentiels d'Euler (voir page 36), mais l'approche suivie pour ce cas est facilement généralisable.

Lorsqu'on fait une intégration volumique dans un système curviligne quelconque, il faut reformuler le volume élémentaire par rapport à ce nouveau système de coordonnées :

$$d^3r = h_\alpha d\alpha h_\beta d\beta h_\gamma d\gamma \quad (4.33)$$

Pour le système de potentiels d'Euler, les facteurs de proportionnalité valent :

$$h_\alpha = r \cos(\theta) \quad (4.34)$$

$$h_\beta = \frac{1}{r \cos(\theta) |B|} \quad (4.35)$$

$$h_\gamma = \frac{1}{|B|} \quad (4.36)$$

alors :

$$d^3r = \frac{d\alpha d\beta d\gamma}{|B|^2} \quad (4.37)$$

On réécrit les formules (4.25) et (4.31) par ce nouveau volume élémentaire et on les multiplie par  $E_0(\vec{x})$  afin de sortir le facteur constant  $\frac{1}{E_0(\vec{x})}$  hors de l'intégrale :

$$E_0(\vec{x})B_1(\vec{x}) = -k^2 \iiint G(\vec{x}, \vec{r}) \delta\varepsilon(\vec{r}, t) E_0(\vec{r}) \frac{d\alpha d\beta d\gamma}{|B|^2} \quad (4.38)$$

$$E_0(\vec{x})B_n(\vec{x}) = -k^2 \iiint G(\vec{x}, \vec{r}) \delta\varepsilon(\vec{r}, t) E_0(\vec{r}) B_{n-1}(\vec{r}) \frac{d\alpha d\beta d\gamma}{|B|^2} \quad (4.39)$$

Cette formulation permet d'obtenir tous les termes  $B_1 \cdots B_4$  en évaluant un seul type d'intégrale :

$$I_n(\vec{x}) = -k^2 \iiint G(\vec{x}, \vec{r}) \delta\varepsilon_r(\vec{r}, t) I_{n-1}(\vec{r}) \frac{d\alpha d\beta d\gamma}{|B|^2} \quad (4.40)$$

avec  $I_0 = E_0(\vec{x})$  qui peut être évaluée analytiquement. On modélise le satellite comme une source ponctuelle, donc le champ non perturbé  $E_0$  est la fonction de Green centrée sur le satellite avec une amplitude obtenue à partir d'une distance et d'une amplitude de référence.

Pour réaliser ce calcul, le code doit réserver de la mémoire pour les fonctions  $I_0 \cdots I_3$  et les obtenir une après l'autre pour chaque point du domaine. Ceci permet d'obtenir la valeur de  $I_0(\vec{R}) \cdots I_4(\vec{R})$  dans le récepteur, puis de  $B_n(R) = \frac{I_n}{I_0}$ . Après avoir obtenu les valeurs de la fonction de substitution à partir de (4.27) on arrive finalement à la valeur de  $E_4(R)$  grâce à (4.32). Ces opérations ont été mises en œuvre dans le fichier `src/rytov/integrate.c`

#### 4.1.4 Parallélisation

Le calcul des intégrales  $I_1 \cdots I_3$  doit se faire pour chaque point du domaine, donc la complexité est de l'ordre de  $O(n^2)$  avec  $n$  le nombre de cellules du maillage. Même si la complexité est trop grande pour un maillage de 50000 cellules ( $2.5 \times 10^9$  évaluations de l'intégrand), heureusement le calcul est extrêmement parallélisable. Pour la parallélisation, on a utilisé la bibliothèque d'échange de messages MPI d'Intel.

Pour cette parallélisation on peut suivre deux approches différentes. Soit  $n$  le nombre de cellules,  $m$  le nombre de processus et  $s = \frac{n}{m}$  les cellules par processus :

##### Paralléliser l'intégrand

Chaque processus évalue, pour chaque cellule du domaine, l'intégrand dans un sous-domaine  $\Omega_p \subset \Omega$  qui dépend du rang  $p$  du processus, et après ils échangent les résultats de ces sous-intégrales pour faire la réduction (somme) et arriver à l'intégrale complète. Le nombre de communications que le code doit faire pour cet échange est de  $m \times (m - 1) \times n$  multiplié par la taille d'un scalaire complexe  $S_{\mathbb{C}}$ . Puisqu'il est possible de saturer le réseau du calculateur avec beaucoup de petits messages, on attend d'avoir calculé toutes les sous-intégrales pour les échanger toutes à la fin. Si chaque processus les envoie à un processus racine (lequel fait la réduction) et après il renvoie le résultat au reste des processus, on peut réduire le nombre de communications à  $2(m - 1)$  multipliée par  $n S_{\mathbb{C}}$

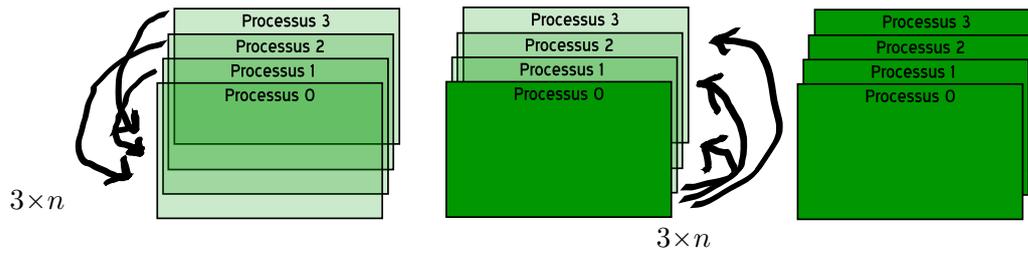


FIGURE 4.1 – Parallélisation de l'intégrant optimisée.

### Paralléliser l'évaluation

L'autre approche consiste en répartir le travail de façon à ce que chaque processus évalue l'intégrale complète sur chaque cellule d'un sous-domaine  $\Omega_p \subset \Omega$ . Si on utilise l'opération MPI `Allgather`, le nombre de communications est de  $m \times (m - 1)$  multipliée par  $s S_C$ . On peut optimiser cette communication en envoyant les résultats partiels à un processus racine qui fait l'assemblage des sous-domaines, et après en renvoyant le domaine complet au reste des processus. Le nombre de communications se réduit à  $m - 1$  multiplié par  $s S_C$  et  $m - 1$  multiplié par  $n S_C$ , avec  $s < n$ .

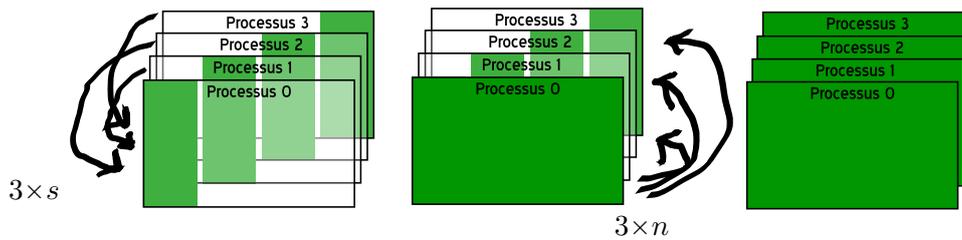


FIGURE 4.2 – Parallélisation de l'évaluation optimisée.

Puisque le nombre de communications nécessaire dans la seconde approche est légèrement plus petit que celui nécessaire dans la première approche, c'est la seconde qui a été retenue dans `waveprop`.

## Chapitre 5

# Post-processus : calcul des paramètres de scintillation

Après avoir obtenu un ensemble d'échantillons de phase et d'amplitude, on évalue la qualité du signal. Cette évaluation se base sur deux paramètres qui mesurent l'écart type de la phase ( $\sigma_\varphi$ ) et l'intensité ( $S_4$ ) sur l'ensemble de échantillons dans un intervalle qui est habituellement d'une minute[5][6]. Néanmoins, `waveprop` permet de changer la taille de cet intervalle de calcul.

### 5.1 Le coefficient de variation de l'intensité : $S_4$

Le paramètre  $S_4$  exprime la variation normalisée de l'intensité du signal, donc il ne dépend pas de l'ordre de grandeur de l'amplitude, mais des variations relatives entre plusieurs échantillons. Le paramètre  $S_4$  se définit comme suit :

$$S_4 = \sqrt{\frac{\langle I^2 \rangle - \langle I \rangle^2}{\langle I \rangle^2}} \quad (5.1)$$

Avec  $I$  l'intensité exprimée comme le carré de l'amplitude, et  $\langle \cdot \rangle$  la moyenne arithmétique calculée sur l'ensemble des échantillons. Grâce a  $S_4$  on peut faire une classification des événements de scintillation : pour  $S_4 < 0.3$  on se place dans le cas de scintillation faible, pour  $0.3 < S_4 < 0.7$  on peut parler des scintillations moyennes et pour  $S_4 > 0.7$  on a des scintillations fortes[6].

### 5.2 L'écart type de la phase : $\sigma_\varphi$

Le paramètre  $\sigma_\varphi$  exprime une mesure de l'écart type des variations de phase, en radians :

$$\sigma_\varphi = \sqrt{\langle \varphi^2 \rangle - \langle \varphi \rangle^2} \quad (5.2)$$

À partir des valeurs de  $\sigma_\varphi$  de l'ordre de 1 rad on peut parler de scintillations fortes[7]. Le lien entre hautes valeurs de  $\sigma_\varphi$  et *cycle slips* (glissement de cycle, c'est-à-dire, une discontinuité dans la phase du signal) est bien connu[8], donc ce paramètre peut aider à prévoir les pertes du verrouillage de phase des ondes porteuses dans les récepteurs.

## Chapitre 6

# Architecture interne de waveprop

waveprop est un logiciel complexe qui fournit beaucoup de fonctionnalités, donc pour mieux comprendre son fonctionnement il faut connaître quelques concepts relatifs à sa structure interne.

waveprop s'inspire de la programmation orientée composant (POC), laquelle consiste à concevoir le programme comme un ensemble des sous-modules (appelés composants) qui offrent un certain nombre de fonctionnalités. Ces composants échangent des informations grâce à une interface de programmation (API). La POC permet aussi de simplifier l'étape de *Testing* du développement du code puisque les composants sont faiblement couplés.

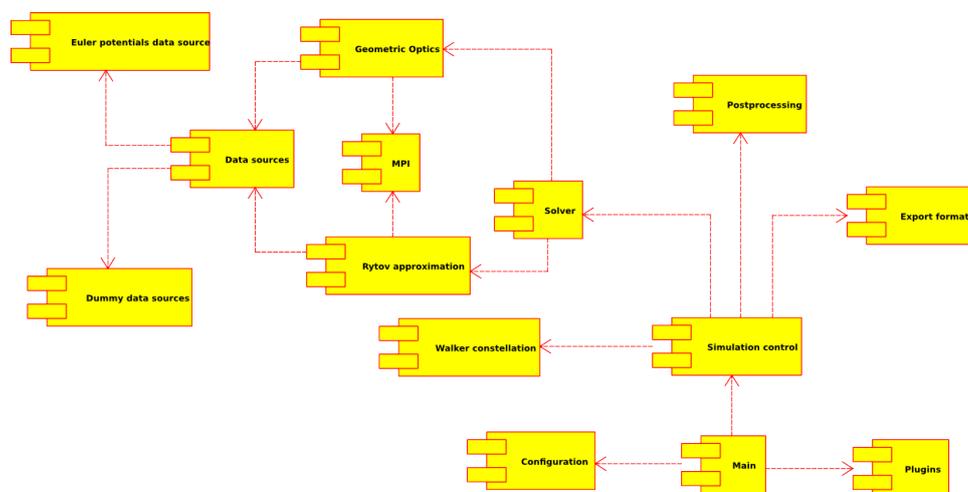


FIGURE 6.1 – Diagramme UML des composants de waveprop .

On peut organiser les composants selon leur appartenance au noyau du code. Les composants inclus dans le fichier exécutable principal de waveprop sont :

**Main** C'est le composant principal et il a comme seul objectif d'appeler les fonctions d'initialisation de chacun des autres composants. C'est le point d'entrée et il doit faire ces appels dans un ordre précis, en respectant les relations de dépendance entre composants. Son code source se trouve dans `src/main.c`

**Configuration** Ce composant permet de lire le fichier de configuration. Son code source se trouve dans les fichiers `src/config.c` et `src/confload.c`.

**Plugins** Composant de charge des plugins sous forme des bibliothèques dynamiques partagées (fichiers \*.so. Son code source se trouve dans `src/plugin.c`).

**Constellation** Ce composant offre une interface pour travailler avec les orbites des satellites en espace inertiel. Son code peut être trouvé dans `src/constellation.c`

**Simulation control** Composant de contrôle de la simulation. Ce composant parcourt tous les satellites à chaque pas de temps et détermine ceux qui sont visibles du point de vue de la station au sol. Pour chaque satellite visible à chaque pas de temps, ce composant appelle le solveur correspondant. Il maintient aussi une liste avec les sorties de chaque appel au solveur. Le code source se trouve dans `src/simulation.c`

**Solver** Composant abstrait qui offre une interface pour appeler a un solveur quelconque d'une manière générale. Ce composant maintient une liste des solveurs concrets et des opérations qu'ils effectuent. Son code source se trouve dans `src/solver.c`.

**Postprocessing** Ce composant permet de calculer les paramètres de scintillation  $S_4$  et  $\sigma_\varphi$ . Le code source se trouve dans le fichier `src/params.c`

**Export formats** C'est le composant d'exportation des données. Ce composant offre une variété de formats de sortie (texte, image, RINEX...) pour sauvegarder les résultats de la simulation. Son code source est réparti entre plusieurs fichiers dans le répertoire `export/`.

**MPI** Ce composant est en fait une couche de compatibilité pour abstraire le parallélisme. Grâce à cette couche, le code peut être compilé même s'il n'y a pas de bibliothèque MPI installée dans le système. Dans ce cas là, le code se comporte comme s'il y a un seul processus. Son code source se trouve dans le fichier `src/mpi.c`.

**Data sources** Ce composant offre une abstraction des données d'entrée sous forme d'une fonction qui fournit la valeur de l'indice de réfraction en un point donné et son gradient. Ce composant permet aussi de convertir les sources de données de type "densité électronique" en "indice de réfraction" suivant la formule :

$$n = \sqrt{1 - \rho_e \frac{e^2}{4\pi m_e \varepsilon_0 f^2}} \quad (6.1)$$

où  $n$  est l'indice de réfraction,  $\rho_e$  la densité électronique,  $e$  la charge de l'électron,  $m_e$  la masse de l'électron,  $\varepsilon_0$  la permittivité diélectrique dans le vide et  $f$  la fréquence de l'onde porteuse. Ces valeurs s'expriment en unités MKS.

Ce code transforme aussi les gradients de densité électronique en gradients de l'indice de réfraction comme suit :

$$\frac{dn}{d\xi} = \frac{dn}{d\rho_e} \frac{d\rho_e}{d\xi} = -\frac{e^2}{8\pi m_e \varepsilon_0 f^2} \left(1 - \rho_e \frac{e^2}{4\pi m_e \varepsilon_0 f^2}\right)^{-\frac{1}{2}} \frac{d\rho_e}{d\xi} \quad (6.2)$$

avec  $\xi$  la direction (en coordonnées cartésiennes) selon laquelle on calcule le gradient.

Puisque les sources de données peuvent être exprimées en plusieurs systèmes de coordonnées, ce composant offre aussi les fonctions de transformation appropriées. Son code source se trouve dans `src/datasource.c`.

**Geometric Optics et Rytov approximation** Ces composants sont les mises en œuvre des solveurs de l'optique géométrique et de l'approximation de Rytov. Leur code source se trouve dans les répertoires `src/go` et `src/rytov` respectivement.

**Dummy datasources** Pseudo-sources de données pour des raisons de *testing*. Les codes de l'optique géométrique et Rytov sont validés grâce à ces pseudo-sources. Son code source se trouve dans `src/dummy.c`.

D'autre part, il y a deux composants qui pour des raisons d'incertitude de leurs spécifications ont été placés hors de l'exécutable principal. Ces composants son :

**Walker constellation** Ce composant fournit les positions des satellites d'une constellation. Néanmoins la constellation actuelle de Galileo n'est pas complète, donc on ne peut pas être absolument sûr de sa fiabilité. Son code se trouve dans le répertoire `plugins/walker`.

**Euler potentials data source** Ce composant offre une source de données en coordonnées des potentiels d'Euler. Cependant, il faudra écrire un nouveau composant lorsque les caractéristiques des sorties du modèle physique du projet IODISSEE seront définitivement fixées. Le code peut être trouvé dans le répertoire `plugins/eulerds`.

## Chapitre 7

# Résultats

On est en attente des résultats numériques. les calculs ont été lancés dans le Mesocentre de calcul de l'ECP. Pour l'instant on dispose seulement des évaluations de l'optique géométrique dans le vide. Ces résultats ont été obtenus en simulant entre les dates 21 :52 :46 31/05/2013 et 22 :52 :46 31/05/2013 (UTC), dans les coordonnées 0°N, 69.4391°E, avec une fréquence de 1.6 GHz, en lançant 12 rayons par satellite. On établit une amplitude du champ électrique de  $E_0 = 10^3 N/C$  à 1 kilomètre du satellite. Le pas d'intégration du rayon est de 10 km.

Dans les graphiques suivants, on présente 4 colonnes où on montre la phase, l'intensité,  $\sigma_\varphi$  et  $S_4$  respectivement, pour les satellites visibles de chaque plan orbital. La valeur non nulle de  $\sigma_\varphi$  est expliquée par des erreurs de calcul. Heureusement, cette *fausse scintillation* est très faible par rapport au seuil de la scintillation moyenne. Par contre, on perçoit une variation de l'intensité due surtout aux variations de la distance du satellite à la station au sol. Même si la variation est très grande (centaines de parties par million), elle est encore très petite par rapport au seuil de la scintillation moyenne.

Il y a aussi des défauts (satellites B7, C4) dus au calcul de l'aire couverte par la section du faisceau de rayons à l'arrivée. Si on ne lance pas beaucoup de rayons cette section peut devenir très dégénérée et ajouter des imprécisions au calcul de l'aire et donc de l'intensité.

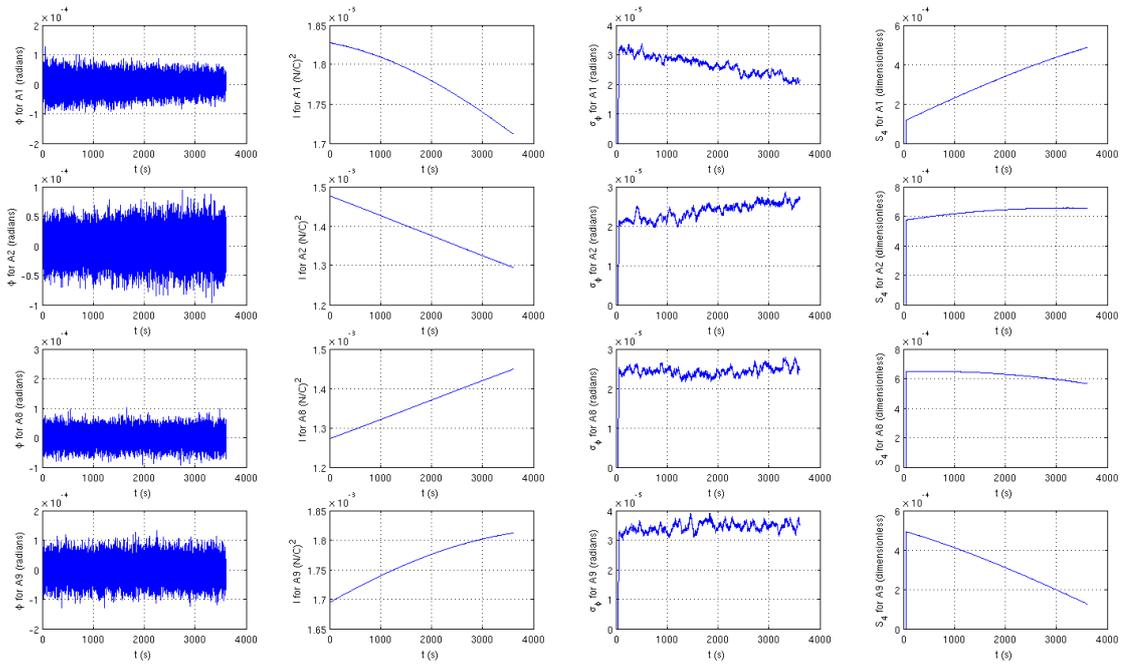


FIGURE 7.1 – Satellites visibles du plan orbital 1.

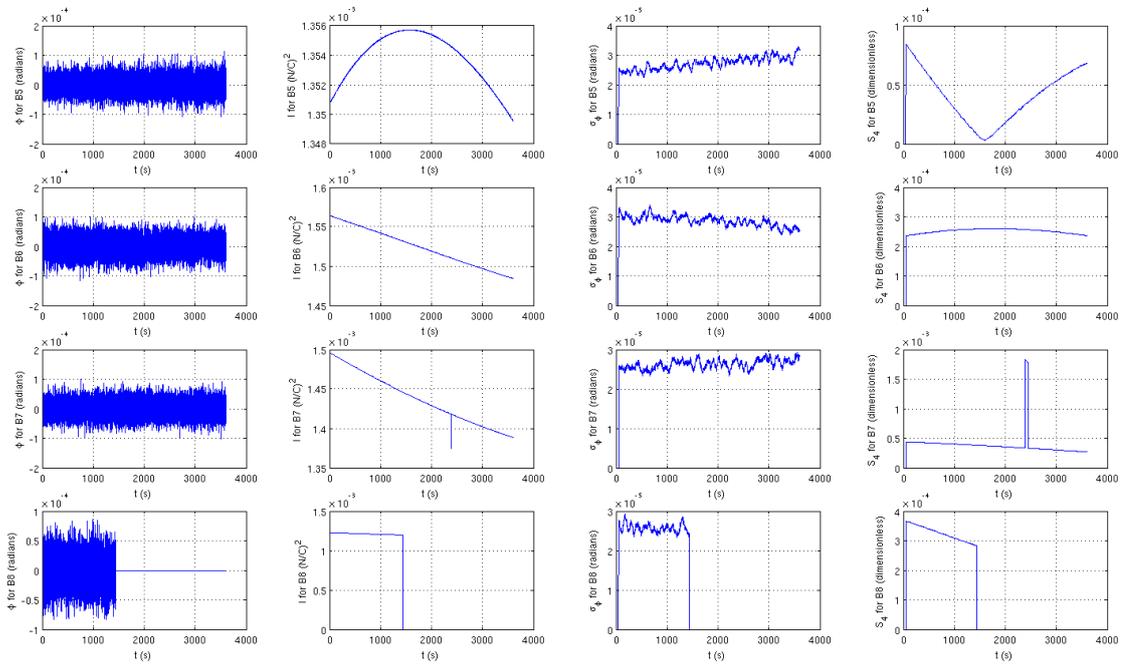


FIGURE 7.2 – Satellites visibles du plan orbital 2.

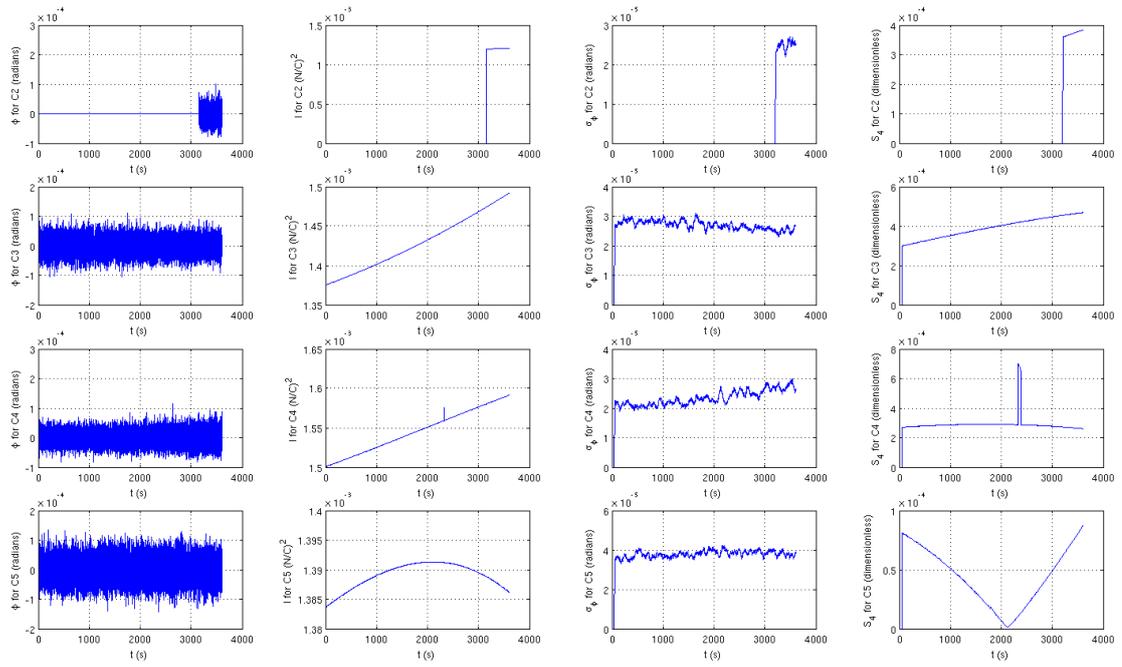


FIGURE 7.3 – Satellites visibles du plan orbital 3.

# Chapitre 8

## Conclusion

Même s'il est encore tôt pour écrire une conclusion par rapport aux résultats, on espère pouvoir utiliser l'optique géométrique pour mesurer la variance de phase et l'approximation de Rytov si on a besoin de mesures de variance d'intensité. En plus, le long du cycle de développement on a fait une liste d'améliorations et d'optimisations potentielles qu'on peut ajouter à `waveprop` :

**Écrire une constellation basée sur le format TLE** : Le format TLE<sup>1</sup> est plus adéquat pour décrire les orbites d'un satellite quelconque. La constellation Walker incluse dans `waveprop` peut aider à prévoir les futures positions des satellites Galileo, mais elle sera très inexacte par rapport aux orbites définitives.

**Écrire une source de données SAO** : SAO est un format de fichier texte clair développé par l'Université de Massachussets-Lowell<sup>2</sup> qui décrit l'état de l'ionosphère (y compris la densité électronique à plusieurs hauteurs) grâce aux sondages de haute fréquence par des stations au sol spéciales, appelées *ionosondes*. Puisque le format SAO est largement utilisé par l'ensemble d'ionosondes réparties autour du monde, il peut être intéressant de faire `waveprop` compatible avec ce format.

**Améliorer les tests** : `waveprop` inclut aussi un système d'évaluation des fonctionnalités (`src/tests/main.c`), mais il n'est pas complet. Ajouter plus de tests est une bonne idée.

**Réécrire une source de type maillage partitionné** : La source de données `eulerds` travaille avec le format `*.rgr`, qui est un format ad-hoc conçu pour le stockage d'information de maillages petits. Si l'information à lire est trop grande pour la stocker en mémoire principale, on peut avoir besoin d'utiliser un maillage partitionné avec la bibliothèque `pdom`, initialement conçue pour la rasterisation de maillages non structurés.

**Améliorer le contrôle de la simulation** : On devrait pouvoir arrêter la simulation quand on veut et la continuer sur autre ordinateur si nécessaire, ou connaître l'état de la simulation elle-même si on n'a pas accès aux messages d'état du logiciel.

**Généraliser l'approximation de Rytov** : L'approximation de Rytov utilise le système de coordonnées eulériennes. Il est intéressant de la rendre compatible avec les autres types de systèmes de coordonnées.

**Paralléliser des boucles** : Par exemple, essayer d'utiliser OpenMP pour paralléliser la boucle de la simulation.

---

1. *Two-Line Elements* est un format texte clair de description d'orbites keplériennes, c'est le standard de facto pour décrire les orbites d'un objet quelconque qui orbite au tour de la Terre à partir d'une époque de référence.

2. <http://ulcar.uml.edu/~iag/SAO-4.3.htm>

**Optimiser le tracé des rayons :** Les rayons ne vont pas dévier avant d'arriver à l'ionosphère ni après en être sorti. Limiter le calcul de l'optique géométrique à cette zone du domaine.

**Corriger les effets de la distance sur l'intensité reçue :** Les résultats ont montré que la valeur de  $S_4$  est affectée par la variation d'intensité due à la distance. On peut se débarrasser de cet effet en utilisant la loi des carrés inverses.

**Décider des stratégies de lancement de rayons :** L'optique géométrique lance des rayons vers le voisinage du récepteur, mais les points de ce voisinage sont générés aléatoirement. Cette approche peut avoir des problèmes si on ne lance pas beaucoup de rayons. Améliorer cette stratégie.

**Généraliser la conversion entre coordonnées :** Les fonctions de transformation entre systèmes de coordonnées sont fixes. Si on veut travailler avec des autres systèmes de coordonnées on a besoin de simplifier cette partie du code.

# Annexe A

## Conventions de repère

Dans ce document, et particulièrement dans le logiciel `waveprop`, on considère 4 systèmes de coordonnées différents en ayant toujours le centre de la Terre comme origine. L'idée est de suivre plus facilement la géométrie des sources de données. En plus, puisque le solveur de l'optique géométrique doit travailler avec les gradients des données, il faut définir aussi les gradients et les formules de conversion entre gradients.

### A.1 Caractéristiques partagées par tous les systèmes de coordonnées

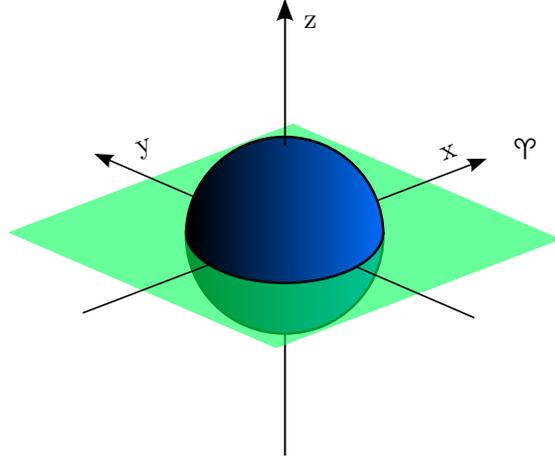
`waveprop` a été conçu pour pouvoir étendre la liste de systèmes de coordonnées initialement supportés par son noyau. Néanmoins, chaque système de coordonnées doit remplir les conditions suivantes :

- Être un système de coordonnées orthogonales.
- Être défini (au moins) dans le volume de la sphère creuse d'épaisseur la distance entre la surface de la Terre et le satellite le plus éloigné dans son apogée.
- Avoir un gradient défini dans tout point de son domaine de définition.
- Considérer le temps comme une composante supplémentaire.

Le temps doit toujours être fourni avec l'ensemble des coordonnées spatiales, car certains systèmes de coordonnées peuvent changer par rapport au temps et même les sources de données peuvent évoluer le long d'un intervalle. Les quatre systèmes de coordonnées initialement supportés sont : les coordonnées cartésiennes en espace inertiel, les coordonnées sphériques cartographiques (absolues et relatives) et le système de potentiels d'Euler.

## A.2 Coordonnées cartésiennes en espace inertiel

Le système de coordonnées cartésiennes en espace inertiel est le système utilisé principalement par le solveur de l'optique géométrique. Du fait que les plans orbitaux sont fixes par rapport à l'espace inertiel (même s'il y a une petite dérive sous forme de précession du nœud ascendant) et que, a priori, il n'y a pas de directions privilégiées pour les rayons issus des satellites, il est intéressant d'utiliser un repère cartésien avec son origine au centre de la Terre.



Dans ce système on considère le sens direct, avec l'axe  $X$  orienté vers le point vernal de l'hémisphère Nord mesuré dans l'époque de référence  $t_{\varphi}$ <sup>1</sup>. L'équateur terrestre est contenu par le plan  $OXY$ , donc l'axe  $OZ$  de ce repère coïncide avec l'axe de rotation de la Terre, et il reste orienté vers le Nord.

Les points dans cet espace s'expriment comme :

$$\vec{p}_{in} = (x, y, z, t) \quad (\text{A.1})$$

où  $x$ ,  $y$  et  $z$  sont exprimées en mètres, et le temps  $t$  en secondes (voir section suivante). Dans ce système de coordonnées, tous les facteurs de proportionnalité valent 1 et alors, le gradient d'un champ scalaire s'exprime simplement comme :

$$\vec{\nabla}_{in} = \frac{\partial}{\partial x} \hat{x} + \frac{\partial}{\partial y} \hat{y} + \frac{\partial}{\partial z} \hat{z} \quad (\text{A.2})$$

---

1. Le point vernal n'est pas un point fixe dans l'espace inertiel, il va changer à cause de la précession des équinoxes. C'est pourquoi il faut toujours le définir par rapport à une époque de référence. En fait, on choisit des équinoxes parce qu'ils sont des événements enregistrés avec une grande précision où on connaît très bien l'orientation de la Terre.

### A.3 Coordonnées sphériques cartographiques

Du point de vue des sources de données, il est peu pratique de repérer points de l'ionosphère par rapport à l'espace inertiel car l'ionosphère suit la rotation de la Terre. En plus, l'ionosphère est munie d'une symétrie sphérique dont on peut profiter. C'est pourquoi le noyau de `waveprop` offre aussi deux systèmes de coordonnées sphériques, notamment un système de coordonnées sphériques cartographiques dont le rayon a comme origine le centre de la Terre (coordonnées sphériques absolues) et un système relatif à la surface de la Terre (coordonnées sphériques relatives) où on remplace le rayon par l'hauteur du point sur le sol.

Puisque le système dépend du temps, il faut ajouter une origine (c'est-à-dire, une *époque*). Pour de raisons de compatibilité avec les systèmes Unix, on a pris comme "temps zéro" le 1<sup>er</sup> janvier 1970 00 :00 :00 UTC. Les points en coordonnées sphériques cartographiques absolues s'expriment comme :

$$\vec{p}_{abs} = (\theta, \phi, \rho_a, t) \quad (\text{A.3})$$

tandis que les points en coordonnées relatives s'expriment comme :

$$\vec{p}_{rel} = (\theta, \phi, \rho_r, t) \quad (\text{A.4})$$

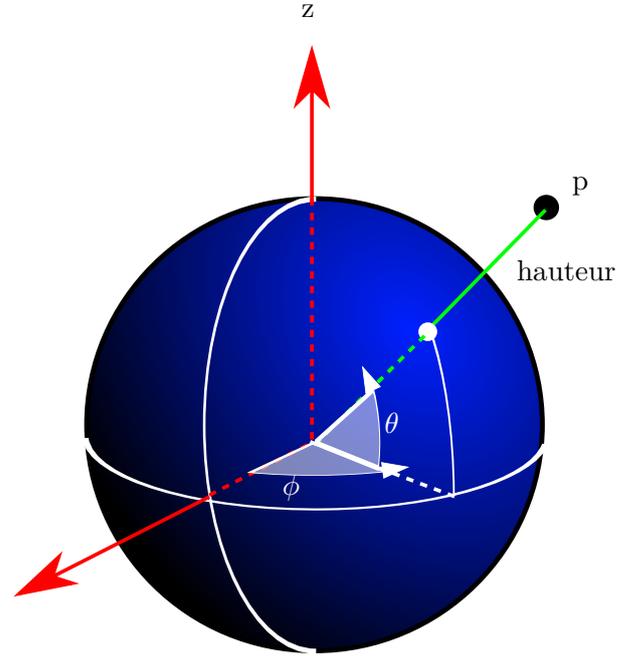
où  $\theta$  représente la latitude nord en degrés par rapport à l'équateur (pôle Nord :  $90^\circ$ , pôle Sud :  $-90^\circ$ ),  $\phi$  est la longitude par rapport au méridien de Greenwich (positive en direction est, negative en direction ouest),  $\rho_a$  la distance du point par rapport au centre de la Terre et  $\rho_r$  la hauteur du point par rapport à la surface de la Terre. On remarque que  $\rho_a = \rho_r + R_\oplus$ , avec  $R_\oplus$  le rayon de la Terre (au peu près  $6371000$  mètres<sup>2</sup>).

On note le rapport avec le système de coordonnées cartésiennes en espace inertiel :

$$x = \rho_a \cos(\theta) \cos[\phi + \psi(t)] \quad (\text{A.5})$$

$$y = \rho_a \cos(\theta) \sin[\phi + \psi(t)] \quad (\text{A.6})$$

$$z = \rho_a \sin(\theta) \quad (\text{A.7})$$



2. Obtenu de <http://nssdc.gsfc.nasa.gov/planetary/factsheet/earthfact.html>. Pour l'instant il n'y a pas besoin d'une mesure plus précise de ce rayon car le but de ce système de coordonnées est d'aider à la représentation des données. Une amélioration de `waveprop` pour obtenir des valeurs réalistes du rayon de la Terre (afin de travailler avec des mesures vraiment réalisées par rapport à la surface) devrait se baser sur systèmes géodésiques plus précis comme WGS 84.

$\psi(t)$  est un terme d'angle horaire appelé *Greenwich Mean Sidereal Time* ou GMST, qu'on peut définir comme l'angle entre le méridien de Greenwich et le direction de référence  $\mathcal{V}$ . Plus formellement,  $\psi(t)$  est la fonction affine :

$$\psi(t) = \Omega(t - t_{\mathcal{V}}) + \psi_0 \quad (\text{A.8})$$

avec  $\Omega$  la vitesse moyenne de rotation de la Terre mesurée dans un référentiel d'inertie local,  $4.178074137 \times 10^{-3} / s$  (ou  $7.2921150(1) \times 10^{-5} rad/s^3$ ),  $t_{\mathcal{V}}$  l'époque de référence (par convention, on va choisir l'équinoxe de mars de 2013, qui a eu lieu le 20 mars 2013, à 11 :02 :00 UTC) et  $\psi_0$  la valeur du GMST dans l'époque  $t_{\mathcal{V}}$ .

Inversement :

$$\rho_a = \sqrt{x^2 + y^2 + z^2} \quad (\text{A.9})$$

$$\theta = \arcsin\left(\frac{z}{\rho_a}\right) \quad (\text{A.10})$$

$$\phi = -\psi(t) + \begin{cases} \arctan\left(\frac{y}{x}\right) & x > 0 \text{ et } y > 0 \\ 360 + \arctan\left(\frac{y}{x}\right) & x > 0 \text{ et } y < 0 \\ 90 \operatorname{sgn}(y) & x = 0 \\ 180 + \arctan\left(\frac{y}{x}\right) & x < 0 \end{cases} \quad (\text{A.11})$$

Pour la définition du gradient (nécessaire pour l'optique géométrique) il faut connaître la base curviligne covariante, laquelle peut être obtenue en normalisant à son tour le gradient de chaque coordonnée sphérique. Alors on arrive à :

$$\hat{\rho}_a = \hat{\rho}_r = \rho_a^{-1}(x \hat{x} + y \hat{y} + z \hat{z}) + (\rho_r + R_{\oplus})^{-1}(x \hat{x} + y \hat{y} + z \hat{z}) \quad (\text{A.12})$$

$$\hat{\theta} = -\cos[\phi + \psi(t)] \sin(\theta) \hat{x} - \sin[\phi + \psi(t)] \sin(\theta) \hat{y} + \cos(\theta) \hat{z} \quad (\text{A.13})$$

$$\hat{\phi} = -\sin[\phi + \psi(t)] \hat{x} + \cos[\phi + \psi(t)] \hat{y} \quad (\text{A.14})$$

Pour ce système, les facteurs de proportionnalité prennent la forme :

$$h_{\rho_a} = h_{\rho_r} = 1 \quad (\text{A.15})$$

$$h_{\theta} = \rho_a \quad (\text{A.16})$$

$$h_{\phi} = \rho_a \cos(\theta) = (\rho_r + R_{\oplus}) \cos(\theta) \quad (\text{A.17})$$

Et on arrive finalement aux expressions du gradient :

$$\vec{\nabla}_{abs} = \frac{\partial}{\partial \rho_a} \hat{\rho}_a + \frac{1}{\rho_a} \frac{\partial}{\partial \theta} \hat{\theta} + \frac{1}{\rho_a \cos(\theta)} \frac{\partial}{\partial \phi} \hat{\phi} \quad (\text{A.18})$$

$$\vec{\nabla}_{rel} = \frac{\partial}{\partial \rho_r} \hat{\rho}_r + \frac{1}{\rho_r + R_{\oplus}} \frac{\partial}{\partial \theta} \hat{\theta} + \frac{1}{(\rho_r + R_{\oplus}) \cos(\theta)} \frac{\partial}{\partial \phi} \hat{\phi} \quad (\text{A.19})$$

---

3. Cette valeur a une incertitude relative de l'ordre de 0.014ppm à cause des perturbations naturelles de la rotation de la Terre. Voir <http://hpiers.obspm.fr/eop-pc/models/constants.html>

## A.4 Système de potentiels d'Euler

Dans la pratique, les modèles numériques travaillent sur un domaine aligné vers le champ magnétique terrestre. Afin de faciliter la mise en œuvre de ces modèles numériques, il est approprié de travailler sur un système de coordonnées également aligné vers ce champ magnétique. Pour profiter de la symétrie de ces domaines, `waveprop` offre aussi le système de coordonnées de Potentiels d'Euler proposé par Yang[1, p. 77]. Néanmoins, on ne définira ce système que par rapport aux coordonnées sphériques expliqués dans la section précédente. Le passage aux coordonnées cartésiennes en espace inertiel nécessitera d'une conversion en deux étapes.

La définition de ce système se base sur l'approximation dipolaire axisymétrique du champ magnétique terrestre  $\vec{B}$  (aligné autour de l'axe de rotation de la Terre), qui peut être décrit en coordonnées sphériques comme :

$$\vec{B} = B_{\rho_a} \hat{\rho}_a + B_{\theta} \hat{\theta} = -\frac{\mu_0}{4\pi} \frac{M}{\rho_a^3} [2 \sin(\theta) \hat{\rho}_a - \cos(\theta) \hat{\theta}] \quad (\text{A.20})$$

Yang a montré qu'il existe un potentiel scalaire  $\gamma$  et un potentiel vecteur  $\mathbf{B}$  tels que  $\vec{B} = -\vec{\nabla} \gamma$  et  $\vec{B} = \vec{\nabla} \times \mathbf{B}$  où on pose la composante  $\phi$  de  $\mathbf{B}$  égale à  $\frac{\beta}{\rho_a \cos(\phi)}$  [1, p. 79]. Ceci lui a permis de déduire un système de coordonnées curvilignes en composantes  $(\alpha, \beta, \gamma)$  où  $\alpha$  est la longitude,  $\gamma$  la composante qui parcourt chaque ligne de champ et  $\beta$  la troisième composante perpendiculaire à  $\alpha$  et  $\gamma$ . Dans ce repère, la composante  $\beta$  vaut zéro dans les latitudes polaires et devient infinie dans l'équateur. Par contre, la composante  $\gamma$  vaut zéro dans l'équateur, moins infini dans le pôle Nord et plus infini dans le pôle Sud.

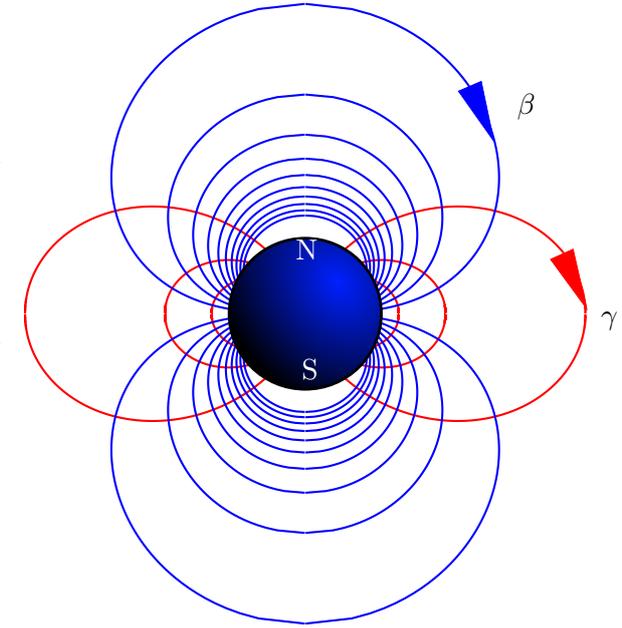
Yang aussi a déduit les formules explicites pour ces composantes par rapport aux composantes sphériques<sup>4</sup>[1, p. 81] :

$$\alpha = \phi \quad (\text{A.21})$$

$$\beta = C \frac{\cos^2(\theta)}{\rho_a} \quad (\text{A.22})$$

$$\gamma = -C \frac{\sin(\theta)}{\rho_a^2} \quad (\text{A.23})$$

4. Yang utilise une latitude ( $\varphi$ ) qui a par origine le pôle Nord, tandis que `waveprop` utilise la latitude géographique ( $\theta$ ). Puisque  $\theta = \frac{\pi}{2} - \varphi$ ,  $\sin(\theta) = \cos(\varphi)$  et  $\cos(\theta) = \sin(\varphi)$ .



où  $C$  est un facteur d'adimensionalisation qui pour le système MKS vaut :

$$C = \frac{\mu_0 M}{4\pi|B|(R_{\oplus})} = 8.05 \times 10^5 \quad (\text{A.24})$$

Inversement, on peut retrouver les coordonnées sphériques par rapport au système de Potentiels d'Euler en normalisant  $\bar{\beta} = \frac{\beta}{C}$  et  $\bar{\gamma} = \frac{\gamma}{C}$ . Le rayon peut se trouver directement :

$$\rho_a = \frac{1}{2} \sqrt{\tau} + \frac{1}{2} \sqrt{-\tau - 2\sqrt{t^2 + 4b}}$$

$$\text{avec } \tau = \sqrt[3]{\frac{1}{2}a + \sqrt{\frac{1}{4}a^4 + \left(\frac{4}{3}b\right)^3}}, a = \frac{\bar{\beta}}{\bar{\gamma}^2}, b = \frac{1}{\bar{\gamma}^2} \quad (\text{A.25})$$

ou par la méthode de Newton, en utilisant la formule itérative avec une tolérance de  $10^{-5}$ [1] :

$$\rho_{am+1} = \rho_{am} - \frac{\bar{\gamma}^2 \rho_{am}^4 + \bar{\beta} \rho_{am} - 1}{4\bar{\gamma}^2 \rho_{am}^3 + \bar{\beta}} \quad (\text{A.26})$$

Dans la pratique et après quelques optimisations, l'itération de Newton est de l'ordre de 120% plus rapide que l'expression directe. Le reste des coordonnées se définissent comme suit :

$$\theta = \arctan\left(-\frac{\rho_a^2 \bar{\gamma}}{\sqrt{\rho_a \bar{\beta}}}\right) \quad (\text{A.27})$$

$$\phi = \alpha \quad (\text{A.28})$$

La base curviligne covariante pour ce système s'écrit comme[1, p. 80] :

$$\hat{\alpha} = \hat{\phi} \quad (\text{A.29})$$

$$\hat{\beta} = \frac{1}{|B|} (-B_{\theta} \hat{\rho}_a + B_{\rho_a} \hat{\theta}) \quad (\text{A.30})$$

$$\hat{\gamma} = -\frac{1}{|B|} (B_{\rho_a} \hat{\rho}_a + B_{\theta} \hat{\theta}) \quad (\text{A.31})$$

Et les facteurs de proportionnalité pour ce système sont[1, p. 79] :

$$h_{\alpha} = \rho_a \cos(\theta) \quad (\text{A.32})$$

$$h_{\beta} = \frac{1}{\rho_a \cos(\theta) |B|} \quad (\text{A.33})$$

$$h_{\gamma} = \frac{1}{|B|} \quad (\text{A.34})$$

Donc le gradient s'écrit comme :

$$\vec{\nabla}_{eu} = \frac{1}{\rho_a \cos(\theta)} \frac{\partial}{\partial \alpha} \hat{\alpha} + \rho_a \cos(\theta) |B| \frac{\partial}{\partial \beta} \hat{\beta} + |B| \frac{\partial}{\partial \gamma} \hat{\gamma} \quad (\text{A.35})$$

## Annexe B

# La constellation Walker delta

Le système Galileo se composera d'un ensemble de 30 satellites dans l'orbite terrestre moyenne (MEO), dont 3 satellites sont de réserve[9]. Ces satellites se trouveront dans une configuration orbitale particulière appelée *Walker Delta 56 : 27/3/1*.

Les constellations Walker Delta recouvrent une classe d'orbites circulaires réparties dans plusieurs plans orbitaux régulièrement espacés, sur chaque orbite les satellites sont aussi régulièrement espacés. Une configuration Walker Delta se décrit grâce a 4 paramètres habituellement exprimés avec la notation :

$i: t/p/f$

où  $i$  indique l'inclinaison de chaque plan,  $t$  le nombre des satellites (dans tous les plans),  $p$  le nombre de plans orbitaux et  $f$  l'angle entre satellites de plans adjacents. C'est-à-dire, pour le système Galileo on a trois plans orbitaux, avec 9 satellites chacun.

La différence d'anomalie vraie entre satellites équivalents de plans voisins est de  $2\pi \frac{f}{t}$  radians. En ajoutant un rayon orbital  $R$  (ou une altitude), les orbites sont complètement définies sauf deux paramètres de phase (l'angle du premier plan par rapport à une direction de référence et l'angle du premier satellite du premier plan dans l'orbite par rapport à certains points de référence). Le code permettant d'obtenir les positions des satellites de la constellation dans l'espace inertiel se trouve dans le fichier `src/plugins/walker/orbit.c`.

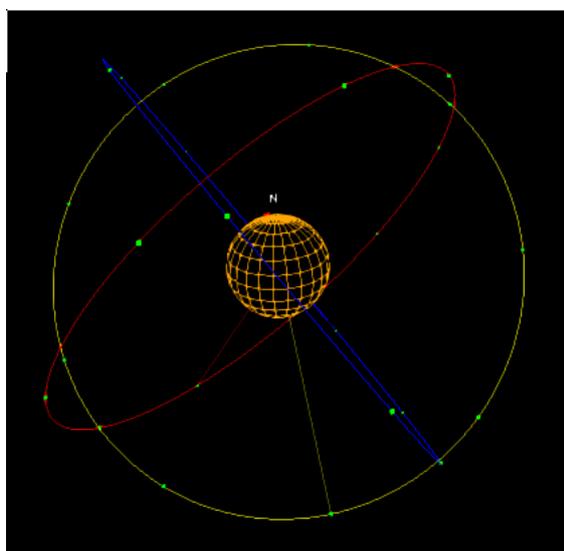


FIGURE B.1 – Constellation Galileo.

---

**Algorithm 1:** Calcul des positions des satellites dans une configuration Walker Delta quelconque  $i$ :  $\mathbf{t}/\mathbf{p}/\mathbf{f}$ .  $\mathbb{R}_x(\phi)$  représente la matrice de rotation en sens direct d'angle  $\phi$  autour de l'axe  $x$ .

---

**Data:**  $t_a$  temps (secondes),  $n$  identifiant ( $[1, \frac{t}{p}]$ ) et  $m$  plan ( $[1, p]$ ).

**Result:**  $\vec{s}$  coordonnées du satellite en espace inertiel.

Calculer  $t_r = t_a - t_0$  (avec  $t_0$  l'époque);

Calculer la période orbitale  $T = 2\pi\sqrt{\frac{R^3}{GM_\oplus}}$  ( $M_\oplus$  : la masse de la Terre);

Calculer l'angle du satellite dans le plan :  $\alpha = 2\pi\frac{t_r}{T} + 2\pi\frac{p}{T}(n-1) + 2\pi\frac{f}{T}(m-1) + \phi_s$ ;

Obtenir les coordonnées du satellite dans le plan :  $\vec{x}_o = R(\cos\alpha, \sin\alpha, 0)$ ;

Tourner autour de la ligne de nœuds :  $\vec{x}_p = \mathbb{R}_x(i)\vec{x}_o$ ;

Calculer la rotation du plan autour de l'axe  $z$  :  $\beta = 2\pi\frac{m-1}{p} + \phi_p$ ;

**return**  $\vec{s} = \mathbb{R}_z(\beta)\vec{x}_p$

---

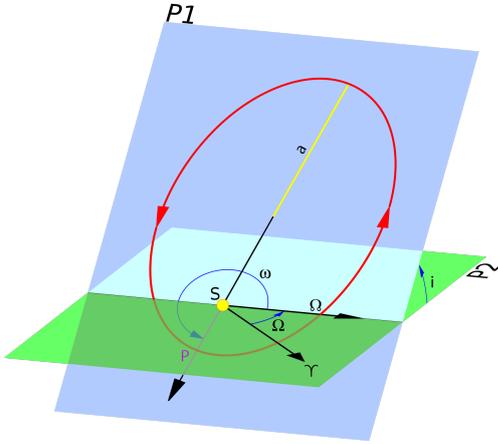


FIGURE B.2 – Éléments orbitaux.

Théoriquement, pour la mise en œuvre de l'algorithme 1, il suffit de définir l'époque  $t_0$  et les angles  $\phi_s, \phi_p$ . Néanmoins, la réalité est bien plus complexe : normalement, on ne connaît la position du premier satellite (et de son orbite) que par ses éléments orbitaux (ascension droite du nœud ascendant, argument du périégée, etc). Donc le code doit exprimer  $\phi_s, \phi_p$  par rapport à ces autres paramètres :

$$\phi_s = \omega + M \quad (\text{B.1})$$

$$\phi_p = \Omega \quad (\text{B.2})$$

où  $\Omega$  est l'ascension droite du nœud ascendant,  $\omega$  l'argument du périégée et  $M$  l'anomalie moyenne. Même si ces paramètres décrivent une orbite elliptique, on suppose que l'excentricité de l'ellipse

est suffisamment petite pour l'approcher par un cercle. Dans le cas idéal, on travaillerait avec les vrais éléments orbitaux de chaque satellite, mais ceci n'est pas possible du fait que la constellation Galileo n'est pas complète.

En plus, l'angle  $\phi_p$  va subir une dérive constante à cause du phénomène de la précession du nœud ascendant dû à l'aplatissement de la Terre, qui implique que le plan va tourner contre la direction de révolution de l'orbite avec une vitesse angulaire  $\omega_p = -7.44 \times 10^{-7}$  rad/s, à peu près 9.45 degrés par an. La valeur de  $\phi_p$  doit se corriger par :

$$\phi_p = \Omega - \omega_p t_r \quad (\text{B.3})$$

Les résultats de cette approche ont été vérifiés avec le logiciel Orbitron (<http://www.stoff.pl/>) et avec l'outil web de Jens T. Satre à <http://www.satellite-calculations.com/TLETracker/SatTracker.htm>.

## Annexe C

# Détails techniques de la source de données eulerds

La partie la plus complexe du cycle de développement de `waveprop` a été sans doute l'écriture de la première source de données.

Initialement, le seul ensemble de données disponible a été un fichier VTK texte clair en format non structuré. L'information fournie par ce fichier est une liste unidimensionnelle de cellules hexaédriques dont les sommets sont donnés en coordonnées cartésiennes avec un échantillon de densité électronique par cellule. Le premier pas a évidemment été d'écrire une bibliothèque de lecture des fichiers VTK (`plugins/eulerds/vtk`), cependant on ne peut pas se permettre d'accéder à ces données directement sans les structurer préalablement : parcourir la liste de cellules en cherchant celle qui contient le point où on veut évaluer la densité est trop coûteux (ordre  $O(n)$ , avec  $n$  le nombre de cellules) et en plus, il s'agit d'une opération qu'il faut réaliser plusieurs fois ( $N$  fois par rayon pour l'optique géométrique, avec  $N$  le nombre de pas d'intégration, ou même plus que  $3n^2$  fois pour Rytov).

Le but est de structurer ces données. Pour cette structuration, on a essayé plusieurs méthodes :

## C.1 Arrangement spatiale de barycentres avec arbres K-d

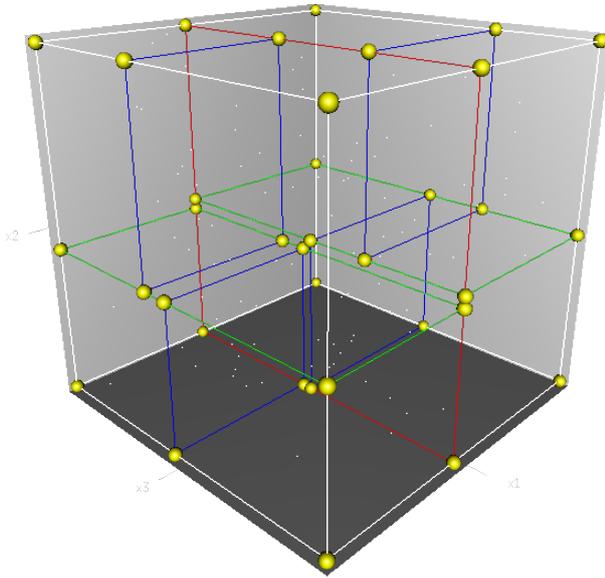


FIGURE C.1 – Arbre K-d, trois dimensions.

L'arbre K-d est une structure de données de type arbre binaire de recherche qui range plusieurs points d'un espace  $K$ -dimensionnel en divisant cet espace récursivement avec l'aide des plans orientés vers certaine dimension. Ces arbres de recherche sont largement utilisés dans quelques techniques de rendu en synthèse d'image comme le *lancer de rayons* afin de trouver les objets les plus proches des rayons lancés[10]. Pour trouver l'élément de l'arbre le plus proche d'un point, il faut décider pour chaque nœud quel est le sous-arbre dans lequel on peut trouver l'élément désiré, en remontant si nécessaire.

L'ordre de complexité de la construction de cet arbre est de  $O(n \log n)$ . La construction n'est à effectuer qu'une seule fois. La recherche est ensuite beaucoup plus rapide :  $O(\log n)$  dans la plupart des cas[11]. Dans le pire des cas, la recherche devient  $O(n)$ [12].

La première approche utilisait cette structure en rangeant les barycentres  $\vec{C}_j$ . La définition de ces barycentres est :

$$\vec{C}_j = \frac{1}{8} \sum_{i=1}^8 \vec{v}_i^j \quad (\text{C.1})$$

où  $\vec{C}_j$  sont les coordonnées cartésiennes du barycentre de la cellule  $j$ , et  $\vec{v}_i^j$  les coordonnées du sommet  $i$  de la cellule  $j$ .

Malheureusement, dans la plupart des évaluations (plus précisément, celles qui sont faites hors du domaine) la recherche est aussi complexe que la recherche linéaire. Cette situation a été confirmée par deux mises en œuvre différentes de l'arbre K-d : une mise en œuvre propre et la bibliothèque des arbres K-d de John Tsiombikas<sup>1</sup>.

## C.2 Rastérisation de tétraèdres

La seconde alternative a été d'essayer de rastériser les cellules dans un maillage régulier. Ceci a pour avantage de simplifier la recherche d'une cellule (complexité  $O(1)$  car il faut seulement convertir les coordonnées réelles de l'espace en coordonnées entières du maillage). Les calculs des gradients sont également simplifiés. Par contre, le processus de rastérisation d'un hexaèdre quelconque n'est pas trivial, il faut tenir compte des problèmes de décomposition des hexaèdres en figures plus simples, des problèmes de crénelage, etc.

1. <http://code.google.com/p/kdtree/>

Le maillage au format VTK représente un tube de champ discrétisé en  $50 \times 50 \times 20$  cellules bornées dans une boîte de  $356.79km \times 52.3542km \times 2439.6km$ . Chaque cellule a une épaisseur d'à peu près 1 km. Si on veut les discrétiser dans un maillage régulier, on peut donner une borne supérieure du pas de discrétisation d'un kilomètre. Avec cette configuration, on arrive à un maillage de  $357 \times 53 \times 2440$  cellules. Si chaque cellule stocke une valeur à virgule flottante de 64 bits, le maillage complet occupe  $357 \times 53 \times 2440 \times 8 = 352Mio$ . En pratique, cette taille pourrait être même plus grande pour éviter les problèmes de crénelage (on rappelle que les cellules ne sont pas alignées vers les coordonnées cartésiennes).

Ces considérations impliquent deux problèmes de nature différente : d'une part, on a un problème algorithmique où on a besoin d'une méthode de rastérisation d'hexaèdres ; d'autre part, on a un problème d'ingénierie parce qu'on doit accéder à un maillage trop grand pour le stocker en mémoire principale.

Le problème de stockage du maillage a été résolu en utilisant un maillage partitionné (nommé *pdom*, répertoire *pdom/*) qui travaille avec une mémoire cache de partitions : on apporte en mémoire principale que les morceaux du maillage complet dont on a besoin.

Le problème de la rastérisation des hexaèdres peut se résoudre en les découpant en 5 tétraèdres et en utilisant un algorithme de rastérisation de tétraèdres. L'algorithme mis en œuvre, présenté dans l'article *Voxelization of solids using simplicial coverings*[13], a montré de très bonnes performances pour des tétraèdres beaucoup plus grands que la taille du voxel. Néanmoins, pour des tétraèdres très petits ou dégénérés, on peut trouver des défauts sous forme de cellules qui ne vont jamais se rastériser.

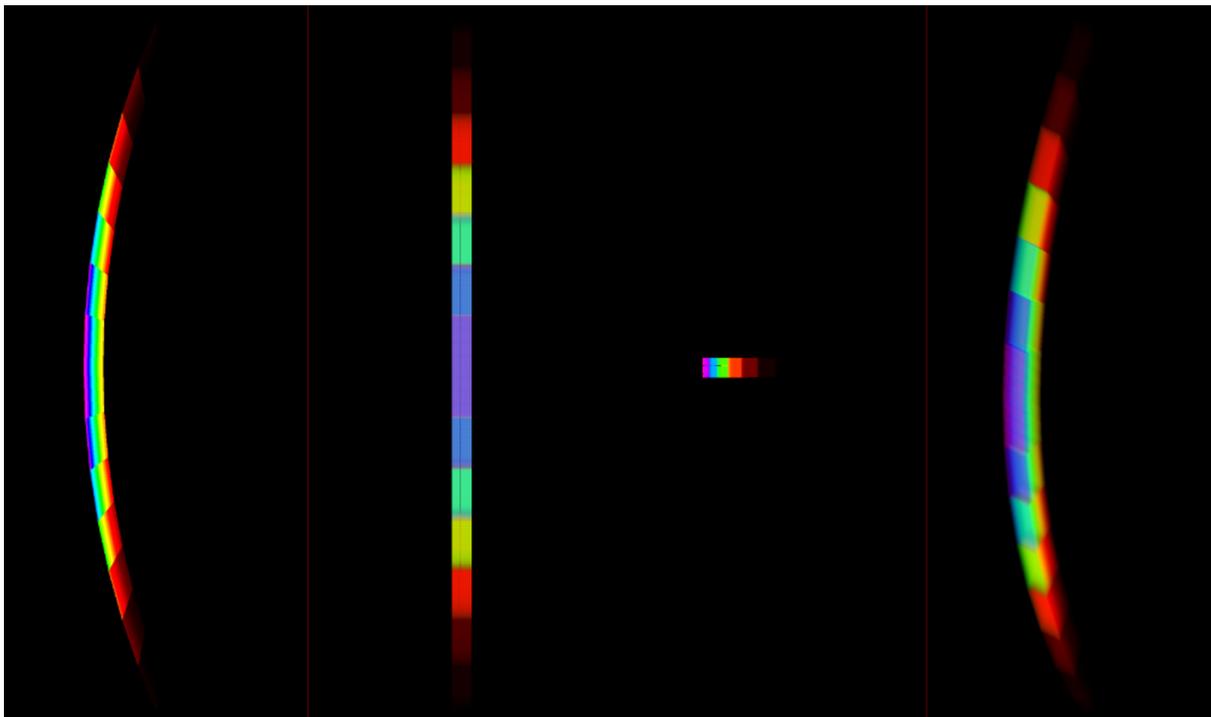


FIGURE C.2 – Défauts générés par la rastérisation de tétraèdres. Remarquez les colonnes absentes.

Puisque ceci est un problème algorithmique, on a décidé de chercher une alternative à cette méthode.

### C.3 Cellules eulériennes

L'approche finalement retenue dans `waveprop` a été de travailler directement avec des coordonnées eulériennes (ceci a été ce qui a motivé une fonctionnalité de conversion entre systèmes de coordonnées).

Sous ce système de coordonnées, les cellules du maillage sont alignées avec les coordonnées, et évaluer la densité en un point est une opération de complexité  $O(1)$ .

`waveprop` offre un outil appelé `vtk2rgr` qui peut convertir un maillage non structuré en coordonnées eulériennes au format VTK en un fichier `*.rgr` (voir répertoire `rectgrid/`) qui stocke une matrice 3-D d'échantillons de densité électronique.

Le plugin `eulerds` fournit une source de données aussi appelée `eulerds` qui, en lisant un fichier `*.rgr` et quelques paramètres de localisation et périodicité de la configuration principale, permet d'obtenir les valeurs de densité avec un temps d'accès très petit.

## Annexe D

# Compilation et installation de waveprop

waveprop est un logiciel standard compilé avec l'aide des `autotools` pour des systèmes Linux (même si théoriquement il pourrait être aussi compilé sous n'importe quelle variante d'Unix compatible avec POSIX). Tout d'abord, avant le compiler, il faut appeler le script `configure` :

```
% ./configure
```

En option, on peut compiler sans MPI :

```
% ./configure --with-mpi=no
```

Une fois que le script a fini, on compile le projet avec `make` :

```
% make
```

et après on l'installe comme `root` :

```
# make install
```

Enfin, en ayant le fichier de configuration, pour commencer la simulation il faut exécuter (avec MPI) :

```
% mpirun -np N waveprop fichier-de-configuration.ini
```

avec `N` le nombre de processus à utiliser avec MPI. On peut aussi l'exécuter en mode séquentiel :

```
% waveprop fichier-de-configuration.ini
```

Par défaut, les résultats de la simulation seront écrits dans le même répertoire où on appelle `waveprop` , mais ce comportement peut être modifié dans le fichier de configuration. Un fichier de configuration d'exemple a été inclus avec la distribution des sources de `waveprop` .

# Bibliographie

- [1] Chang YANG, *Analyse et mise en œuvre des schémas numériques pour la physique des plasmas ionosphériques et de tokamaks*. Thèse. Université Lille 1 Sciences et Technologies 2011.
- [2] Wheelon Albert D., *Electromagnetic Scintillation, volume 1 : Geometrical Optics*. Cambridge University Press 2004.
- [3] Minjeaud Sebastian, *Propagation d'ondes dans l'ionosphère perturbée*. Présentation (rencontre ANR IODISSEE 11 janvier 2013).
- [4] Wheelon Albert D. *Electromagnetic Scintillation, volume 2 : Weak Scattering*. Cambridge University Press 2003.
- [5] Meggs Robert W., N. Mitchell Cathryn and Smith Andrew M. *An Investigation into the Relationship between Ionospheric Scintillation and Loss of Lock in GNSS Receivers*. Department of Electronic and Electrical Engineering, University of Bath 2006.
- [6] Béniguel Yannick, Adam J-P, Prieto-Cerdeira R., Arbesser-Rastburg B., *Ionospheric Scintillations at L & C Bands*. IEEA, ESA/ESTEC 2009.
- [7] *Weathering the Storm - GNSS and the Solar Maximum. Next Generation GNSS Ionospheric Scintillation and TEC Monitoring*. NovAtel White Paper - March 2012.
- [8] Meggs Robert W., N. Mitchell Cathryn, Honary Farideh *GPS scintillation over the European Arctic during the November 2004 storms*. Department of Electronic and Electrical Engineering, University of Bath, 2008.
- [9] *Galileo fact sheet*. esa.int. 15 février 2012. [http://download.esa.int/docs/Galileo\\_IOV\\_Launch/Galileo\\_IOV\\_factsheet\\_2012.pdf](http://download.esa.int/docs/Galileo_IOV_Launch/Galileo_IOV_factsheet_2012.pdf)
- [10] Wald Ingo, Havran Vlastimil, *On building fast kd-Trees for Ray Tracing, and on doing that in  $O(N \log N)$* . SCI Institute (University of Utah). Czech Technical University in Prague. 2006.
- [11] Ladner Richard, *K-D Trees (CSE 373, Data Structures, Lecture 22)*. University of Washington. 2002.
- [12] Moore Andrew W. *Efficient Memory based Learning for Robot Control*. PhD. Thesis. Technical Report No. 209 Computer Laboratory, University of Cambridge. 1991.
- [13] Rueda Antonio J., Segura Rafael J., Feito Francisco R., Ruiz de Miras Juan, Orgáyar Carlos. *Voxelization of solids using simplicial coverings* Escuela Politécnica Superior, Universidad de Jaén. 2003.