



Étude de méthodes ensemblistes robustes pour une localisation multisensorielle intégrée. Application à la navigation des véhicules en milieu urbain.

Vincent Drevelle

► To cite this version:

Vincent Drevelle. Étude de méthodes ensemblistes robustes pour une localisation multisensorielle intégrée. Application à la navigation des véhicules en milieu urbain.. Automatique. Université de Technologie de Compiègne, 2011. Français. NNT: . tel-00679502v2

HAL Id: tel-00679502

<https://theses.hal.science/tel-00679502v2>

Submitted on 3 Jul 2012

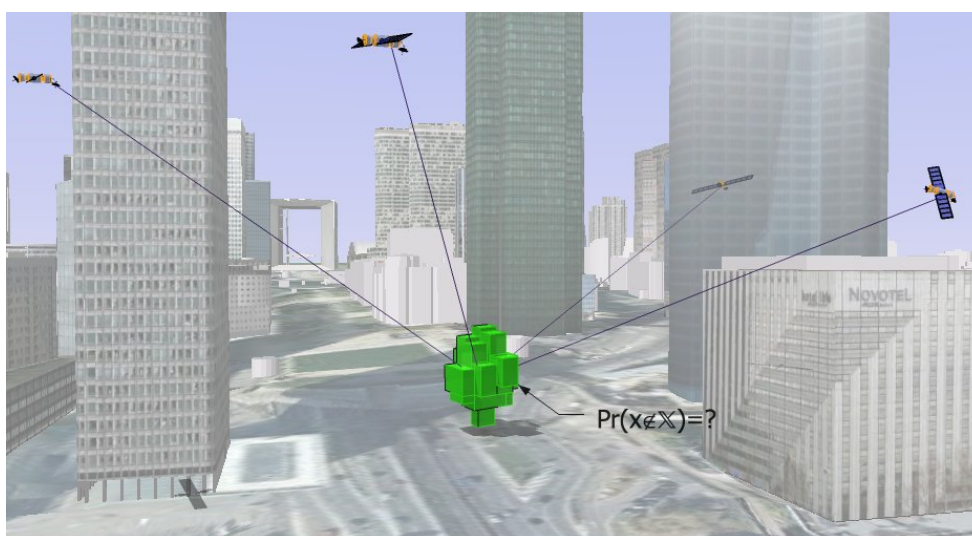
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Étude de méthodes ensemblistes robustes pour une localisation multisensorielle intégrée

Application à la navigation des véhicules en milieu urbain

Vincent Drevelle



Thèse soutenue le 1^{er} décembre 2011 devant le jury composé de

M. Philippe Bonnifait	<i>directeur de thèse</i>
M. Luc Jaulin	<i>rapporteur</i>
M. Dominique Meizel	<i>rapporteur</i>
Mme Véronique Cherfaoui	<i>examinatrice</i>
M. Michel Kieffer	<i>examineur</i>
M. François Peyret	<i>examineur</i>



Remerciements

Afin de marquer mon attachement à ma région natale, je tiens en premier lieu à remercier le *Conseil Régional de Picardie* qui a financé cette thèse dans le cadre de *l'Appui à l'Émergence*.

Je remercie Ali CHARARA, directeur de l'unité mixte de recherche CNRS 6599 *Heu-DiaSyc*, de m'avoir accueilli dans son laboratoire, au sein de l'équipe *Automatique, Systèmes Embarqués, Robotique*.

Un grand merci à Philippe BONNIFAIT, mon directeur de thèse, pour m'avoir suivi et supporté. Ses qualités, tant scientifiques qu'humaines, ont été la source de discussions toutes plus riches et intéressantes durant ces trois années d'accompagnement.

Mes remerciements vont également à Luc JAULIN et Dominique MEIZEL, pour le temps passé à étudier ce travail en tant que rapporteurs, ainsi qu'à tous les membres du jury pour la grande pertinence de leurs questions et l'intérêt de la discussion que j'ai pu avoir avec eux.

Je remercie tous les amis et collègues du laboratoire, qui ont été l'âme du cadre sympathique et de la bonne ambiance dans laquelle j'ai pu évoluer durant cette thèse. Mes sentiments vont particulièrement à mes amis doctorants et chercheurs de tous horizons qui entretiennent une ambiance multiculturelle et colorée dans ce pays grisailant, ainsi qu'aux secrétaires pour leur gentillesse et leur efficacité.

Merci à tous mes amis, qui ont été la source d'évasion musicale, culturelle et festive nécessaire pour ne pas entamer une métamorphose en GPS ensembliste.

Merci encore à mes parents, pour leur soutien et leur confiance, et à mon petit frère qui, m'ayant suivi dans cette voie, nous permettra bientôt d'ouvrir un cabinet de docteurs à la maison.

Enfin, merci à la chaise qui, écrasant consciencieusement mon orteil, me donna le privilège de pouvoir soutenir cette thèse en accordant costume, chemise et cravate aux plus belles chaussettes et sandales de ma garde-robe.

Table des matières

1	Introduction générale	1
1.1	Introduction	1
1.2	Hypothèses de travail et objectifs	2
1.3	Plan de la thèse	3
2	Positionnement Global pour la Navigation des Véhicules Intelligents	5
2.1	Introduction	5
2.2	Les systèmes de positionnement par satellites	7
2.2.1	NAVSTAR GPS	7
2.2.1.1	Historique	7
2.2.1.2	Référentiels associés au GPS	8
2.2.1.3	Constitution du système GPS	8
2.2.1.4	Structure du signal GPS	10
2.2.1.5	Évolutions futures	12
2.2.2	GLONASS	13
2.2.3	Galileo	14
2.2.3.1	Historique	14
2.2.3.2	Description du système	14
2.2.4	Les systèmes d'augmentation	15
2.3	Observables GPS et calcul d'une position	17
2.3.1	Équation d'observation associée au code	17
2.3.1.1	Principe de la mesure	17
2.3.1.2	Prise en compte des référentiels de temps et d'espace	18
2.3.1.3	Phénomènes dégradant la mesure de code	20
2.3.1.4	Correction des mesures avec les données de navigation	23
2.3.1.5	Autres phénomènes induisant des imprécisions sur la mesure	23
2.3.2	Calcul d'une solution de localisation	25
2.3.2.1	Mise en équations	25
2.3.2.2	Résolution classique	26
2.4	Qualité de service d'un système de positionnement	27
2.4.1	Définitions générales	27
2.4.2	Métriques utilisées en aéronautique	28
2.5	Calcul classique d'intégrité autonome	30
2.5.1	Surveillance autonome d'intégrité	30
2.5.1.1	Impact d'un défaut sur le calcul aux moindres carrés	33

2.5.1.2	Détection de faute	34
2.5.1.3	Identification et adaptation	36
2.5.2	Quantification de l'intégrité	38
2.5.2.1	Intégrité interne	38
2.5.2.2	Intégrité externe	39
2.5.2.3	Lien entre la détection de défaut et l'intégrité externe	42
2.5.2.4	Choix du seuil de détection	43
2.5.3	Planification de mission en fonction de l'intégrité externe . . .	46
2.6	Conclusion	47
3	Estimation non-linéaire ensembliste robuste	49
3.1	Introduction	49
3.2	Approche ensembliste	50
3.3	Analyse par intervalles	52
3.3.1	Intervalles	53
3.3.2	Calculs sur les intervalles	53
3.3.3	Généralisation des intervalles	55
3.3.3.1	Intervalles vectoriels	55
3.3.3.2	Matrices intervalles	56
3.3.4	Fonctions d'inclusion	56
3.3.4.1	Propriétés spécifiques de fonctions d'inclusion	57
3.3.4.2	Fonctions d'inclusion naturelles	58
3.3.5	Contracteurs	59
3.3.5.1	Problèmes de satisfaction de contraintes	59
3.3.5.2	Contracteur par propagation-rétropropagation	60
3.3.5.3	Programmation linéaire	63
3.3.5.4	Linéarisation parallèle	66
3.4	Inversion ensembliste par intervalles (SIVIA)	67
3.4.1	Sous pavage	67
3.4.2	SIVIA	68
3.4.3	SIVIA avec contracteur	71
3.5	Inversion ensembliste robuste	71
3.5.1	Inversion ensembliste q -relaxée	74
3.5.1.1	Intersection q -relaxée	74
3.5.1.2	Inversion ensembliste robuste par intervalles	74
3.5.1.3	Contracteur rapide pour l'intersection q -relaxée . . .	78
3.5.2	GOMNE	81
3.5.3	Comparaison entre les méthodes d'estimation robustes et l'ap- proche DIA	83
3.6	Détection et identification de mesures aberrantes	85
3.7	Conclusion	87
4	Estimation à risque quantifié d'un domaine de localisation	89
4.1	Introduction	89

4.2	Contrainte liée aux mesures de pseudodistances	90
4.2.1	Modèle d'observation	90
4.2.2	Contracteur par propagation de contraintes	92
4.2.3	Robustesse	93
4.3	Contrainte d'altitude	95
4.3.1	Modèle numérique de terrain	95
4.3.2	Contracteur d'altitude	96
4.4	Calcul du risque associé à la zone d'incertitude et choix des bornes . .	98
4.4.1	Principe	98
4.4.2	Calcul du risque associé au domaine de localisation	98
4.4.3	Choix des bornes pour un risque donné	100
4.5	Mise en œuvre et comparaison des résultats avec une approche standard	104
4.5.1	Estimation non contrainte	104
4.5.2	Ajout d'une contrainte d'altitude	108
4.5.3	Essai avec un récepteur haute sensibilité	112
4.6	Implémentation en temps-réel	113
4.6.1	Calcul du sous-pavage solution	115
4.6.1.1	Calcul interruptible	115
4.6.1.2	Parallélisation	116
4.6.2	Visualisation	116
4.7	Discussion par rapport à l'approche classique	117
4.8	Conclusion	118
5	Positionnement intègre contraint en milieu urbain	119
5.1	Introduction	119
5.2	Positionnement contraint par la route	120
5.2.1	Représentation 3D surfacique de la route	120
5.2.2	Contrainte de route	122
5.2.2.1	Contrainte liée à une facette	122
5.2.2.2	Contraction par clipping de Sutherland-Hodgman . .	122
5.2.2.3	Contrainte liée à l'espace roulant	123
5.2.2.4	Sélection des facettes candidates	125
5.2.3	Synthèse	126
5.3	Contrainte liée à la dérive d'horloge du récepteur	127
5.3.1	Prédiction du décalage d'horloge	127
5.3.2	Gestion des sauts d'horloge	128
5.4	Résultats	130
5.5	Conclusion	136
6	Estimation robuste de la pose sur un horizon de données	137
6.1	Introduction	137
6.2	Modèle cinématique 2D	138
6.3	Gestion de l'historique de données	139
6.3.1	Gestion de l'historique des positions	140

6.3.2	Gestion de l'historique des mesures proprioceptives	141
6.3.2.1	Suppression des données redondantes à l'arrêt	142
6.3.2.2	Ré-échantillonnage des données sur l'historique	142
6.4	Contracteur robuste de pose sur l'horizon	143
6.4.1	Contraintes liées aux positions de l'historique	143
6.4.2	Contraintes associant historique de position et route	145
6.5	Calcul du risque	147
6.6	Implémentation en temps-réel	148
6.7	Résultats	150
6.8	Conclusion	156
7	Conclusion et perspectives	159
7.1	Conclusion	159
7.2	Contributions	161
7.3	Perspectives	161
Annexes		165
A	Méthodes classiques d'estimation robuste	167
A.1	M-estimateurs	167
A.2	Least Median of Squares	169
A.3	RANSAC	170
B	Contracteur par linéarisation parallèle du calcul GPS	171
C	Jeux de données utilisés	177
C.1	Essai à Compiègne	177
C.1.1	Parcours	177
C.1.2	Vérité terrain	177
C.1.3	Données	179
C.2	Essai à La Défense	179
C.2.1	Parcours	179
C.2.2	Vérité terrain	179
C.2.3	Données	179
C.3	Essai à Paris, Mairie du XII ^e arrondissement	181
C.3.1	Parcours	181
C.3.2	Vérité terrain	181
C.3.3	Données	181
D	Génération d'une carte 3D de l'espace roulant	183
D.1	Représentation de l'information	183
D.2	Création d'une carte 3D	184
D.2.1	Saisie des bords de voie	184

D.2.2	Génération des facettes	188
Bibliographie		191
Liste des symboles		197

Table des figures

2.1	Véhicules autonomes	5
2.2	Constellation des satellites du système GPS	9
2.3	Stations de contrôle du système GPS	10
2.4	Récepteurs GPS militaire, géodésique et grand public	10
2.5	Structure et génération du signal GPS	12
2.6	Constellation Galileo (<i>Source : ESA</i>)	15
2.7	Fonctionnement d'un système SBAS	16
2.8	Principe de la mesure de pseudodistance	18
2.9	Temps d'un satellite et du récepteur	19
2.10	Retards dégradant la mesure de code	21
2.11	Multi-trajets	24
2.12	Modes de navigation au cours d'un vol (<i>RTCA/DO-229D</i>)	29
2.13	Algorithme de RAIM	31
2.14	Algorithme de FDE	32
2.15	Exemple de test avec les distributions centrée χ^2_{m-4} et non centrée $\chi^2_{m-4,\delta}$ pour $m = 11$	35
2.16	Projection dans le plan le plus explicatif des vecteurs résidus correspondant à chaque défaut unitaire, avec 6 satellites.	37
2.17	Relation linéaire entre le paramètre de non centralité $\sqrt{\delta}$ et l'erreur de position horizontale, en supposant que le défaut ne touche qu'un seul satellite.	40
2.18	Relations entre HUL, HAL et HPL et résultat du RAIM	43
2.19	Seuil minimal $T_{\min}$ et seuil maximal $T_{\max}$ du test du χ^2 vérifiant les RNP (p_{FA} , p_{MD} et HAL)	45
2.20	Prévision de la disponibilité du RAIM par AUGUR pour une approche à l'aérodrome de Compiègne-Margny.	46
2.21	Prédiction de la disponibilité du RAIM par AC90-100A pour la phase d'approche sur le territoire des USA	47
3.1	Localisation d'un robot en extérieur à l'aide de trois balises radio, et dans une pièce à l'aide de deux balises.	50
3.2	Localisation ensembliste avec 3 balises sous hypothèse d'erreurs bornées	51
3.3	Différentes représentations ensemblistes de la solution du problème de localisation à trois balises.	52
3.4	Représentation du pavé $[1, 3] \times [1, 2]$	55

3.5	Images d'une boîte par la fonction $\mathbf{f}$, par une fonction d'inclusion $[\mathbf{f}]$ et par la fonction d'inclusion minimale $[\mathbf{f}]^*$	56
3.6	Décomposition en contraintes primitives de la contrainte associée à la mesure de distance d'une balise B_i	62
3.7	Graphe des contraintes primitives liées à deux balises B_i et B_j	64
3.8	Application du contracteur par propagation-rétropropagation sur le problème de localisation à 3 balises	64
3.9	Encadrement d'un ensemble entre deux sous-pavages	68
3.10	Test d'inclusion pour l'inversion ensembliste	69
3.11	Résultat de l'algorithme SIVIA sur un problème de localisation avec 4 balises (données simulées)	70
3.12	Localisation ensembliste avec trois balises, en présence d'une mesure aberrante	73
3.13	Intersection q -relaxée de trois ensembles, avec $q \in \{0, 1, 2\}$	74
3.14	Déroulement de RSIVIAEXT sur un problème de localisation à 3 balises, avec $q = 1$	77
3.15	Résultat de l'algorithme RSIVIA sur un problème de localisation avec 4 balises (données simulées)	79
3.16	Une itération du contracteur rapide pour l'intersection q -relaxée, avec 3 contraintes	80
3.17	Résultat de l'algorithme GOMNE (sans contraction) sur un problème de localisation à 4 balises, en présence de données aberrantes (données simulées)	82
3.18	Résultat de l'algorithme GOMNE sur un problème de localisation à 4 balises, en présence d'une mesure aberrante non détectable	84
3.19	Détection et identification de mesure aberrante lors du calcul d'une solution 1-relaxée avec trois mesures	86
4.1	Décomposition en contraintes primitives de la contrainte liée à une pseudodistance	93
4.2	Multitrajets et propagation sans ligne de vue directe (NLoS) en environnement urbain	94
4.3	Modèle numérique de terrain des environs de Compiègne, IGN.	96
4.4	Étapes de la contraction d'une boîte de position par un MNT, et influence de l'effet d'enveloppe.	97
4.5	Contraction optimale par un modèle de terrain en fonction du système de coordonnées choisi.	98
4.6	Risque associé au sous-pavage solution	99
4.7	Problème du choix des bornes d'erreur	101
4.8	Logarithme décimal du risque $(1 - p)$ lié à chaque mesure pour que le risque sur la solution soit de 10^{-7}	101
4.9	Répartition du risque lors du choix des bornes à $\pm\alpha\sigma$ pour une erreur de mesure gaussienne.	103

4.10	Vue d'ensemble des résultats : boîtes englobantes des solutions à chaque époque.	105
4.11	Bornes du domaine de localisation, relativement à la vraie position. Une erreur de type rampe est simulée sur le satellite n°1.	106
4.12	Projection sur le plan horizontal de l'ensemble solution 1-relaxé calculé en présence d'un biais sur un satellite.	107
4.13	Seuils de détection et d'exclusion de faute sur chaque satellite avec l'inversion ensembliste 1-relaxée à $t = 10$ s.	108
4.14	Vue 3D ^{1/2} des sous-pavages solutions du problème de localisation avec 6 pseudodistances. Les couleurs représentent le décalage d'horloge. . .	109
4.15	Rayon de la solution 1-relaxée avec (<i>lignes vertes épaisses</i>) et sans (<i>lignes bleues</i>) la contrainte d'altitude du MNT.	110
4.16	Bornes du domaine d'incertitude de localisation, relativement à la vraie position. Un biais croissant est appliqué sur la pseudodistance du satellite n°1. La contrainte d'altitude du MNT est appliquée. . .	111
4.17	Influence de la contrainte d'altitude sur la détection et l'exclusion de faute avec l'algorithme SIVIA 1-relaxé	112
4.18	Vues <i>Google Earth</i> à La Défense	113
4.19	Détection et identification de mesures aberrantes (en rouge) à La Défense, avec une solution 2-relaxée.	114
4.20	Véhicule CARMEN sur la piste d'essais lors de la journée démonstrations de la conférence IV2011 à Baden-Baden (Allemagne)	115
4.21	Visualisation 3D des zones de localisation calculées sur les Quais de Seine à Paris et à La Défense. À droite, une mesure aberrante a été identifiée et représentée en rouge.	117
5.1	Modélisation 3D d'un quartier du XII ^e arrondissement de Paris pour le projet CityVIP, incluant une représentation surfacique de la route en bleu clair. (<i>IGN</i>)	121
5.2	Vue « fil de fer » en perspective de l'espace roulant représenté sous forme de maillage triangulaire.	121
5.3	Facette ABC et normales associées à chaque arête.	122
5.4	Contraction du domaine de position $[x]$ avec la facette triangulaire ABC, en utilisant l'algorithme de Sutherland-Hodgman. $[x_c]$ est le domaine contracté.	123
5.5	Contraction d'une boîte $[\mathbf{x}_0]$ avec l'espace roulant $\mathcal{M}$	124
5.6	Utilisation de la topologie de la route pour limiter l'ambiguïté lors de la sélection des facettes candidates	125
5.7	Algorithme de positionnement contraint par l'espace roulant.	127
5.8	Dérive d'horloge (en $\mu\text{s/s}$) sur 18 h de fonctionnement d'un récepteur <i>ublox LEA-4T</i>	128
5.9	Estimation de la dérive d'horloge et prédiction	129
5.10	Évolution du décalage d'horloge (en μs) d'un récepteur <i>ublox LEA-4T</i> sur 18 h de mesures. On observe très nettement les sauts d'horloge. .	129

5.11	Influence des sauts d'horloge sur les pseudodistances mesurées (en m) du satellite 24.	130
5.12	Trajectoire de l'essai autour de la mairie du XII ^e arrondissement de Paris. <i>Vue Google Earth</i>	131
5.13	Histogramme du nombre de satellites visibles avec le récepteur <i>PolarX</i> sur une boucle autour de la mairie.	132
5.14	Algorithme de positionnement contraint par l'espace roulant et le décalage d'horloge prédit	132
5.15	Ensembles solutions calculés avec la contrainte d'espace roulant . . .	134
5.16	Bornes et centre de gravité du sous-pavage solution.	135
5.17	Fonctions de répartition empiriques de l'erreur de position 2D et du rayon du domaine de confiance de position.	136
6.1	Modèle de véhicule type « char »	139
6.2	Règles d'ajout de positions dans l'historique L_p	140
6.3	Facteur de densité spatiale dans l'historique L_p	141
6.4	Prédiction de la pose au temps t à partir de deux positions. L'ensemble des poses compatibles avec une position a une forme d'hélice.	143
6.5	Estimation de la pose courante (au temps t) à partir de deux positions et de données proprioceptives.	144
6.6	Estimation de la pose courante à partir de deux positions, de mesures du déplacement et d'une carte. Le déplacement est ici supposé linéaire.	146
6.7	Propagation de contraintes sur la pose $[\mathbf{x}](t)$ avec une position $[\mathbf{p}](t_k)$ et les mesures proprioceptives. La contrainte cartographique est appliquée à chaque pas de temps.	146
6.8	Implémentation en temps-réel du système d'estimation robuste de pose.	149
6.9	Visibilité satellitaire après seuillage sur le rapport signal bruit $C/N_0 \geq 35\text{dBHz}$	151
6.10	Centre de gravité des positions calculées époque par époque par fusion des pseudodistances GPS avec l'espace roulant.	152
6.11	Sous-pavage solution représenté sur la carte OpenStreetMap du quartier de la mairie du XII ^e arrondissement.	153
6.12	Trajectoire 2D du centre de gravité du sous-pavage solution.	154
6.13	Erreurs d'estimation de position et de cap par le centre de gravité du sous-pavage et bornes de la zone de confiance.	155
6.14	Fonction de répartition empirique de l'erreur de position horizontale (HPE).	156
A.1	Exemples de fonctions ρ	168
A.2	Exemples de fonctions ψ	169
B.1	Itérations de contraction d'une boîte de $1100 \times 1100 \times 11 \times 2000\text{ km}$ avec les pseudodistances de 5 satellites, sous Matlab.	173

B.2	Itérations de contraction d'une boîte de $11000 \times 11000 \times 110 \times 20000$ km avec les pseudodistances de 5 satellites, sous Matlab.	175
C.1	Parcours de l'essai à Compiègne	178
C.2	Parcours de l'essai à La Défense	180
C.3	Parcours de l'essai à Paris (mairie du XII ^e arrondissement)	182
D.1	Vue 3D de l'espace roulant autour de la mairie du XII ^e arrondissement de Paris	183
D.2	Saisie des bords de voie par relevé de points GPS	186
D.3	Saisie des bords de voie à partir d'une orthoimage et d'un MNT sous QGIS	187
D.4	Génération des bords de voie à partir d'une carte 2D sous QGIS . . .	188
D.5	Création de l'espace roulant sous QGIS à partir des points de bord de voie	189

Chapitre 1

Introduction générale

1.1 Introduction

La localisation est une fonction essentielle pour la navigation autonome ou supervisée en robotique mobile, aux côtés de la planification de trajectoire, de l'évitement d'obstacle et de la commande. Un robot mobile doit en effet être capable de se situer dans une représentation de son environnement. Cette représentation peut être de multiples formes en fonction de la complexité du robot, de ses moyens de perception et de la tâche à accomplir, de la plus basique telle que le contact avec un mur ou un autre robot, à la localisation sur une carte précise de l'environnement et des obstacles, en passant par la position dans une base d'images apprises (mémoire visuelle).

On dispose la plupart du temps de plusieurs sources d'information pour se localiser. Il peut s'agir de capteurs extéroceptifs tels que les télémètres et les caméras, fournissant des données à partir de l'environnement du robot, mais aussi de capteurs proprioceptifs mesurant des données propres au robot telles que la vitesse de ses roues ou son accélération. Connaissant les caractéristiques des différents capteurs, ainsi qu'un modèle de l'évolution du robot, on peut fusionner l'ensemble de ces données afin de calculer une position.

Un cas particulier est la navigation de véhicules routiers, dont l'évolution est contrainte sur le sol, et la plupart du temps sur les routes. La cartographie numérique des routes, accessible au sein de systèmes d'information géographique, fournit ainsi une information précieuse pour la navigation. Elle est la plupart du temps donnée dans un référentiel global, et nécessite donc un moyen de positionnement absolu pour être exploitée. Les systèmes de positionnement par satellites répondent à ces critères, en fournissant une position référencée par rapport à un repère global lié à la Terre, et une couverture quasi totale de la planète. Il est parfois préférable pour la navigation de se localiser non pas de façon métrique dans l'espace (en coordonnées géographiques ou cartésiennes) mais de façon symbolique sur le réseau routier : savoir sur quel tronçon de route on se trouve et à quelle position sur ce tronçon. Le *map-matching* consiste ainsi à se positionner sur la carte à partir d'une position globale. Lorsque cette étape est réalisée, tous les attributs associés à la base de données navigable deviennent accessibles à la machine intelligente.

Alors que les systèmes de localisation par satellites sont fiables et précis en environnement ouvert, la localisation par satellites de véhicules en milieu urbain est plus problématique. En effet, les signaux des satellites sont souvent masqués et reflétés par les bâtiments, ce qui fausse les mesures et biaise la position calculée. Le nombre de satellites visibles peut même devenir insuffisant pour pouvoir se localiser. L'utilisation conjointe d'autres capteurs et de données cartographiques est ainsi vitale pour le positionnement en milieu urbain.

La navigation est une activité pouvant mettre en jeu la sécurité des personnes, qu'il s'agisse de guider un avion à l'atterrissage ou de faire évoluer un véhicule autonome au milieu du trafic routier. Dans ce cas, la localisation doit offrir suffisamment de garanties pour que son utilisation ait un niveau de risque acceptable. La première information nécessaire est une quantification de l'erreur maximale de position commise par le système. Un avion n'utilisera un système de guidage par satellites pour l'atterrissage que si l'erreur d'altitude est bornée. De même, une voiture autonome doit avoir une erreur de position d'au plus quelques décimètres pour respecter les voies de circulation lorsque l'information de pose est utilisée directement dans une contre-réaction de type « retour d'état » ou de quelques mètres lorsque la localisation métrique est utilisée pour sélectionner le couloir de navigation en complément d'autres capteurs. La seconde information nécessaire est une quantification du risque de dépassement des bornes d'erreur. Un *mécanisme d'intégrité* doit donc être mis en place pour s'assurer que la position respecte ces critères. La notion d'intégrité du positionnement est particulièrement développée en aéronautique, où elle répond à des spécifications précises. Dans le domaine des véhicules autonomes ou de la robotique mobile, cette problématique en est encore à ses débuts ; plusieurs travaux ont été initiés à la fin des années 90, mais aucun principe méthodologique ne domine à ce jour.

Dans ce contexte de quantification de l'erreur de position, l'étude de *méthodes ensemblistes* pour la localisation se justifie par le fait qu'elles permettent de décrire l'ensemble des solutions compatibles avec les mesures, compte tenu de leurs incertitudes. C'est donc un moyen de caractériser le domaine d'incertitude en position en fonction des erreurs de mesure. Les méthodes basées sur l'analyse par intervalles permettent de plus d'obtenir des résultats garantis, même dans le cas non-linéaire. Lors du positionnement sur carte, on trouve en particulier les non-linéarités aux intersections. Enfin, des méthodes robustes par intervalles ont été développées récemment, ce qui permet de prendre en compte la présence de mesures aberrantes, fréquentes lors du positionnement par satellites en milieu urbain.

1.2 Hypothèses de travail et objectifs

L'utilisation de méthodes ensemblistes par intervalles nous impose que les erreurs de mesure soient bornées, afin de pouvoir être représentées par des intervalles. L'intervalle de mesure ainsi obtenu doit contenir la vraie valeur. Cette contrainte, qui

permet de réaliser un calcul garanti, est très forte, et souvent impossible à mettre en place. En effet, la définition de bornes garanties de l'erreur de mesure peut conduire à des intervalles très conservatifs, voir infinis si l'erreur n'est pas à support borné.

Nous supposons donc que les erreurs de mesure sont indépendantes et que leur fonction de répartition est connue, afin de pouvoir déterminer des bornes d'erreurs associées à une confiance. L'hypothèse d'indépendance permettra de déterminer le risque d'intégrité de la solution à partir de la confiance en chaque mesure.

Pour le calcul d'un domaine de localisation, nous supposons que le véhicule est équipé d'un récepteur GPS permettant l'accès aux données brutes (mesures liées à chaque satellite) ainsi que de l'accès à la vitesse des roues et à un gyromètre de lacet. Enfin, nous supposons qu'une cartographie précise en trois dimensions de la surface de la route est disponible.

Une exigence importante liée à ces travaux sera l'applicabilité en temps-réel de la méthode, qui doit pouvoir être embarquée dans le véhicule expérimental du laboratoire pour en démontrer les performances et la faisabilité. Les calculs de position devront pouvoir être effectués à la cadence des capteurs et si possible à une cadence plus élevée déterminée par l'utilisateur du système.

1.3 Plan de la thèse

Le premier chapitre de cette thèse sera consacré au positionnement global des véhicules par satellites. On y présentera tout d'abord les différents systèmes de positionnement par satellites en service et en cours de développement, et plus particulièrement le système GPS. Ensuite seront détaillés le principe de la mesure des données brutes et les erreurs associées, ainsi que la méthode classique de calcul de position à partir de ces observables. Les critères de qualité et d'intégrité d'un service de positionnement seront alors définis, d'après les normes aéronautiques. Enfin, nous décrirons une méthode classique de surveillance autonome et de quantification de l'intégrité du positionnement par satellites.

Nous présenterons ensuite des méthodes d'estimation ensembliste dans un deuxième chapitre. Elles seront illustrées par un exemple non-linéaire de localisation. Les outils de l'analyse par intervalles y seront exposés, ainsi que la notion de contracteur. Nous y présenterons ensuite la méthode d'inversion ensembliste par intervalles, puis son extension robuste aux mesures aberrantes. Enfin, la question de la détection et de l'identification des anomalies dans les mesures sera abordée.

Dans le troisième chapitre, nous traiterons du calcul d'un domaine de localisation à partir de données GPS brutes et à l'aide des méthodes robustes par intervalles présentées auparavant. Nous y présenterons la contrainte liée aux mesures GPS et le contracteur associé, puis une seconde contrainte permettant d'intégrer l'information d'altitude d'un modèle numérique de terrain. La méthode de calcul du risque associé

au domaine d'incertitude de localisation et une méthode de choix des bornes d'erreur de mesure seront ensuite détaillées. La méthode sera alors appliquée sur un jeu de données réelles, et les performances de détection de défauts comparées à l'approche standard. Enfin, nous évoquerons les détails de l'implémentation en temps-réel.

Le quatrième chapitre abordera la localisation en milieu urbain, en introduisant deux contraintes permettant de pallier le manque d'observables GPS. Nous y présenterons tout d'abord une contrainte de route, basée sur une représentation surfacique de l'espace roulant en trois dimensions. Ensuite, une seconde contrainte liée à la dérive d'horloge du récepteur sera introduite. Ces deux contraintes seront introduites dans l'algorithme de calcul du domaine de localisation présenté dans le chapitre précédent, et testées sur un jeu de données réelles acquises à Paris.

Enfin, disposant d'une méthode de calcul d'un domaine de localisation, nous étudierons dans le dernier chapitre l'estimation robuste de la pose d'un véhicule à partir d'un historique fini de positions et de mesures proprioceptives. Nous y présenterons le modèle d'évolution du véhicule, puis une méthodologie de gestion de l'historique de données permettant d'optimiser le calcul de pose. Un contracteur robuste pour la pose du véhicule sera ensuite proposé. Nous détaillerons enfin l'implémentation en temps réel et des résultats en milieu urbain contraint.

Chapitre 2

Positionnement Global pour la Navigation des Véhicules Intelligents

2.1 Introduction

La localisation est une tâche primordiale pour la navigation de véhicules intelligents. Qu'il s'agisse d'informer l'utilisateur de sa position en vue de le guider ou de contrôler un véhicule de façon autonome, le véhicule doit pouvoir se localiser dans son environnement.

Nombre de tâches robotiques peuvent se contenter d'une information de position relative, qu'elle soit dans l'espace de mesure des capteurs (pour les méthodes d'asservissement visuel par exemple), ou dans un référentiel lié à la mission courante (c'est le cas pour le SLAM¹). Ce dernier référentiel peut éventuellement ne pas être métrique (cas de la navigation dans une base d'amers visuels). Le repère peut également être robocentré, dans le cas où l'on souhaite éviter des obstacles ou suivre une cible.

1. Simultaneous Localization And Mapping : technique consistant pour un robot à simultanément construire une carte d'un environnement inconnu et s'y localiser.



Figure 2.1: Véhicules autonomes. *Boss*, de l'équipe Tartan Racing de la Carnegie Mellon University (Darpa Urban Challenge) et *Caroline* de la Technische Universität Braunschweig (projet Stadtpilot).

Cependant, dès qu'il est nécessaire d'utiliser des données géoréférencées (telles que des points d'intérêt, des points de navigation ou des données cartographiques) ou de communiquer son information de position avec d'autres systèmes, il est nécessaire d'avoir une référence commune. Dans le cas du SLAM, il est possible à plusieurs robots évoluant dans le même environnement d'échanger leurs cartes, afin de partager un référentiel spatial commun, mais cela ne permet pas d'intégrer des données cartographiques a priori. Ainsi, on a besoin d'une méthode de positionnement global, fournissant une localisation référencée par rapport à la Terre. La cohérence des informations de positionnement global et des bases de données géographiques repose sur l'utilisation d'un système géodésique de référence commun. L'utilisation d'un système positionnement global permet ainsi l'identification de la route courante sur une carte, afin par exemple de respecter les règles de circulation en vigueur : le système d'information géographique (SIG) peut stocker des limitations de vitesse pour chaque tronçon de route). Cela permet également la navigation topologique sur une carte navigable.

Il existe de nombreuses techniques de positionnement absolu fondées sur une infrastructure terrestre. Parmi les plus répandues, citons la localisation des téléphones GSM, connaissant la position des antennes relais avec lesquelles ils sont en contact. La géolocalisation à partir de bases de données des points d'accès WiFi est également une méthode largement déployée dans les appareils communicants, car peu coûteuse. Le désavantage de ces deux dernières méthodes est que leur zone de couverture est limitée aux zones urbaines. Le système LORAN², basé sur la différence de temps de propagation (TDoA³) entre plusieurs émetteurs grandes ondes, couvrait la majeure partie de l'hémisphère nord. Les émetteurs américains ont toutefois cessé d'émettre en 2010. Les émetteurs européens sont maintenus en activité, comme solution de secours en cas de défaillance du GPS.

La géolocalisation par satellites offre l'avantage d'une couverture globale avec une constellation de satellites en orbite moyenne (bien que certains systèmes, basés sur des satellites géostationnaires, n'ont qu'une couverture régionale). Depuis le premier système satellitaire global TRANSIT mis en service en 1967, d'autres systèmes ont été développés. Les systèmes NAVSTAR GPS et GLONASS sont ainsi actuellement opérationnels. Ces systèmes fournissent un service de positionnement absolu avec une précision d'une dizaine de mètres, pour un coût raisonnable.

Chaque tâche de navigation a des exigences différentes concernant la précision du système de localisation. La connaissance de l'incertitude de localisation est donc primordiale pour l'utilisation de cette donnée. De plus, dans le cas où une erreur pourrait avoir un impact sur des vies humaines, ou des conséquences juridiques et financières sévères, il est nécessaire de mettre en place un mécanisme garantissant l'intégrité de la géolocalisation. C'est notamment le cas dans l'aéronautique, où les récepteurs GPS implémentent une méthode assurant que l'erreur de position ne

2. LOnG RAnge Navigation

3. Time Difference of Arrival

dépasse pas une valeur critique.

Dans ce chapitre, nous allons tout d'abord présenter les systèmes de positionnement par satellites, et en particulier le GPS. Le calcul d'une solution de géolocalisation par GPS sera ensuite expliqué. Nous discuterons ensuite des différents critères de qualité de service d'un système de positionnement. Pour finir, la méthode classique d'intégrité de localisation implantée dans les récepteurs GPS sera présentée.

2.2 Les systèmes de positionnement par satellites

Les systèmes de positionnement par satellites se basent sur une constellation de satellites de navigation pour déterminer la position de l'utilisateur. On appelle GNSS (*Global Navigation Satellite System*) un système de navigation par satellites fournissant un service de géolocalisation couvrant le monde entier.

Actuellement, deux systèmes sont opérationnels : le système NAVSTAR GPS, développé par les États-Unis d'Amérique, et le système russe GLONASS qui est de nouveau opérationnel suite à sa restauration après une période d'abandon. Deux autres systèmes globaux de navigation par satellites sont en cours de développement : le système chinois Beidou-2, aussi appelé Compass, et le système européen Galileo.

2.2.1 NAVSTAR GPS

Le système américain NAVSTAR⁴ GPS, ou plus simplement GPS (*Global Positioning System*) par abus de langage, est aujourd'hui le plus développé des systèmes de positionnement par satellites. Permettant la mesure de position, de vitesse et du temps, il est utilisé dans le monde entier pour des applications aussi diverses que l'aide à la navigation, la gestion de flotte, le guidage d'engins, le relevé topographique ou encore la synchronisation d'horloges et d'émetteurs.

2.2.1.1 Historique

Le système GPS a été développé à partir des années 1970 par le Ministère de la Défense (DoD) américain en remplacement de l'ancien système *Transit* qui offrait une couverture et une précision insuffisante pour la navigation. Les premiers satellites ont été mis en orbite à partir de 1978 et le GPS a été déclaré pleinement opérationnel le 27 avril 1995.

Conçu à la base pour des applications exclusivement militaires, le congrès américain a cependant fait pression pour que des applications civiles soit possibles. La précision

4. NAVigation System by Timing And Ranging

des applications civiles était limitée à une centaine de mètres par la mise en place de la *Selective Availability* (SA), une dégradation volontaire de la qualité des signaux civils. La désactivation de la SA au cours de l'an 2000 a permis de faire passer la précision à une dizaine de mètres.

Le système GPS fournit ainsi deux types de services aux utilisateurs pour la localisation et la mesure de temps :

- SPS (Standard Positioning Service) : Ce service a été mis à disposition des utilisateurs civils, et permet une précision en position de l'ordre d'une dizaine de mètres.
- PPS (Precise Positioning Service) : Ce service est réservé aux militaires américains et à certains utilisateurs autorisés par le département de la défense. Le signal permettant la localisation par le PPS est crypté.

2.2.1.2 Référentiels associés au GPS

L'utilisation d'un système de positionnement global nécessite le choix de référentiels communs de temps et d'espace, définis sur toute la surface du globe.

Le référentiel de temps pour les calculs GPS est le temps GPS. Le temps GPS a été calé sur le Temps universel coordonné (UTC) en 1980. En raison des secondes intercalaires ajoutées régulièrement pour garder l'UTC synchronisé avec la rotation de la Terre, en 2011 le temps GPS a un écart de 15 s par rapport à l'UTC.

Le système géodésique associé au GPS est le WGS 84 (*World Geodetic System 1984*), ou Système géodésique mondial, révision de 1984, qui s'est rapidement imposé comme une référence pour la cartographie. L'origine de WGS 84 se trouve au centre de masse de la Terre, avec une erreur supposée inférieure à 2 cm. Il définit également les paramètres d'un ellipsoïde de révolution aplati aux pôles approximant le géoïde terrestre⁵, ainsi qu'un méridien de longitude zéro situé 102,5 m à l'est du méridien de Greenwich.

2.2.1.3 Constitution du système GPS

Le système NAVSTAR GPS se décompose en trois parties nommées « segments » : le segment spatial, le segment de contrôle et le segment utilisateur.

Segment spatial Le segment spatial consiste, dans sa vision initiale, en un ensemble de 24 satellites NAVSTAR répartis sur 6 plans orbitaux (Figure 2.2). Actuellement, la constellation comporte 31 satellites. Ces satellites évoluent à une altitude d'environ 20 000 km et mettent un demi-jour sidéral, soit 11 h 58 min 2 s pour effectuer une rotation autour de la terre.

5. Le géoïde correspond à une équipotentielle du champ de gravité terrestre choisie de manière à coller au plus près de la « surface réelle »

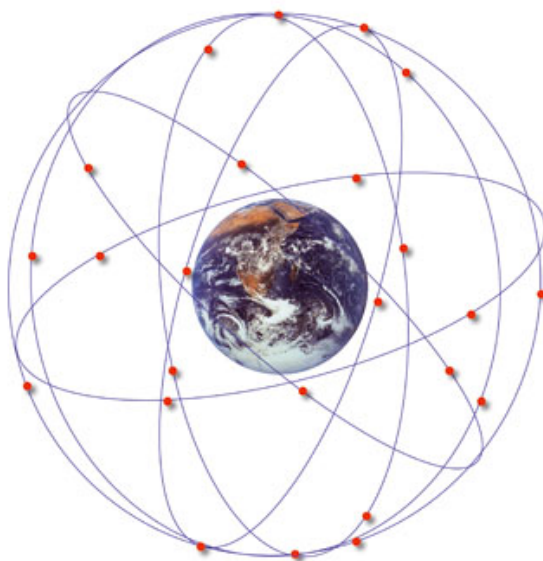


Figure 2.2: Constellation des satellites du système GPS

Chaque satellite possède plusieurs horloges atomiques, qui asservissent un oscillateur à une fréquence fondamentale de 10,23 MHz. L'émetteur génère deux porteuses (L1 et L2) de fréquences respectives 1575,42 MHz et 1227,60 MHz.

Chaque satellite émet :

- des signaux horaires (codes) sur les porteuses L1 et L2,
- des almanachs (données grossières sur les orbites de l'ensemble des satellites de la constellation),
- et des éphémérides fournissant une description fine de l'orbite du satellite, son état, le décalage d'horloge, etc.

Segment de contrôle Le segment de contrôle, qui permet de piloter le système, est composée de cinq stations au sol réparties le long de l'équateur. En septembre 2005, 6 stations supplémentaires gérées par la NGA ⁶ ont été ajoutées afin de pouvoir surveiller tous les satellites sur au moins deux stations en simultané (Figure 2.3).

Chaque station transmet ainsi les observations à la station principale située à Colorado Springs qui se charge de l'estimation des éphémérides et des paramètres d'horloge. Les nouveaux paramètres sont ensuite transmis aux satellites via les stations de transmission (Ascension, Diego Garcia, Kwajalein).

Segment utilisateur Le segment utilisateur comprend l'ensemble des récepteurs civils et militaires qui ne font que recevoir les informations des satellites. Le nombre d'utilisateurs est ainsi illimité. Pour déterminer la position, la vitesse et le temps, la réception simultanée des signaux de quatre satellites est nécessaire.

6. National Geospatial-Intelligence Agency



Figure 2.3: Stations de contrôle du système GPS



Figure 2.4: Récepteurs GPS militaire, géodésique et grand public

On trouve une large variété de récepteurs, allant du récepteur mono-fréquence bon marché au récepteur dit de géodésie, beaucoup plus précis et réservé aux applications de topographie. Le type de récepteur employé dépend fortement de l'application qui exploite le GPS. Ainsi, pour une application dans le domaine de l'automobile, les constructeurs se contentent généralement de récepteurs mono-fréquence (L1C) à faible coût.

2.2.1.4 Structure du signal GPS

La structure du signal s'articule autour de la fréquence fondamentale $f_0 = 10.23$ MHz générée par l'horloge atomique du satellite. La grande stabilité de l'horloge embarquée permet une très faible dérive de la fréquence dans le temps. A partir cette fréquence fondamentale sont générées les deux porteuses L1 et L2 de telle sorte que :

$$\begin{cases} f_{L1} &= 154 \cdot f_0 = 1575.42 \text{ MHz} \\ f_{L2} &= 120 \cdot f_0 = 1227.60 \text{ MHz} \end{cases} \quad (2.1)$$

Ces deux porteuses constituent la base du signal GPS et sont modulées par le message de navigation ainsi que des séquences pseudo-aléatoires différentes pour chaque

satellite appelées *codes PRN*. Chacun des codes est transmis à des débits différents [Kaplan and Hegarty, 2006] :

- **Code C/A (coarse/acquisition)**. Séquence pseudo-aléatoire de 1023 bits émise à la fréquence de 1,023 MHz. La séquence se répète à l'identique toutes les millisecondes. La rapidité d'émission de ce code permet une acquisition rapide du signal par le récepteur.
- **Code P(Y) (precise)**. Ce code est réservé au service *PPS*. Il est constitué d'une séquence unique de 2.3547×10^{14} bits émis à la fréquence de 10,23 MHz, soit une période de 266,41 jours. En réalité chaque satellite émet un tronçon particulier de ce code, équivalent à 7 jours d'émission. Du fait de sa longueur, l'acquisition du code P est difficile, mais présente l'avantage de ne pas être ambiguë. L'utilisation du code P(Y) repose ainsi sur l'acquisition préalable du code C/A qui, une fois « accroché », permet de basculer sur le code P(Y).
- **Message de navigation**. Le message de navigation est transmis à la vitesse de 50 bits par seconde et se décompose en cinq sous-messages (*subframes*) de 300 bits. Les deux premiers mots de chaque portion sont identiques : ils contiennent un préambule fixe, le *time-of-week*, qui permet de passer du code C/A au code P, et un bit d'alerte indiquant un fonctionnement erroné. Le message de navigation contient des informations sur le satellite telles que son éphéméride (qui contient les paramètres képlériens de l'orbite), les paramètres d'horloge, des indicateurs sur la santé du satellite, etc. Les sous-messages 4 et 5 contenant des informations de moindre importance (en particulier les almanachs, qui décrivent de façon grossière les orbites de tous les satellites de la constellation), ils sont découpés en *pages* qui sont transmises alternativement de telle sorte que la réception complète du message de navigation nécessite 25 cycles, soit 12 minutes et 30 secondes [IS-GPS-200D, 2004].

Les codes et le message de navigation sont combinés par addition binaire et sont utilisés pour moduler la fréquence porteuse par inversion de phase (modulation BPSK⁷) comme le montre la figure 2.5. Il est à noter que sur L1 sont simultanément transmis les codes C/A et P, en quadrature de phase afin de simplifier la démodulation.

Dans le cas d'un récepteur mono-fréquence, il est possible d'extraire la mesure de code C/A, la mesure de phase du signal, ainsi que le décalage Doppler de la fréquence mesurée par rapport à la fréquence L1.

L'utilisation d'un récepteur bi-fréquence permet d'accéder à des observables supplémentaires : la mesure de phase sur L2, le décalage Doppler sur L2 ainsi que la mesure de code P(Y) si le récepteur sait la décoder. Les récepteurs bi-fréquences sont généralement des récepteurs haut de gamme orientés vers des applications nécessitant un positionnement précis. En effet, le bi-fréquence fournit une redondance d'information permettant de réaliser des combinaisons d'observables de façon à éliminer l'allongement ionosphérique par exemple [Duquenne et al., 2005, p.35]. Ceci permet d'augmenter sensiblement la qualité de la solution calculée, la résolution

7. Binary Phase-Shift Keying

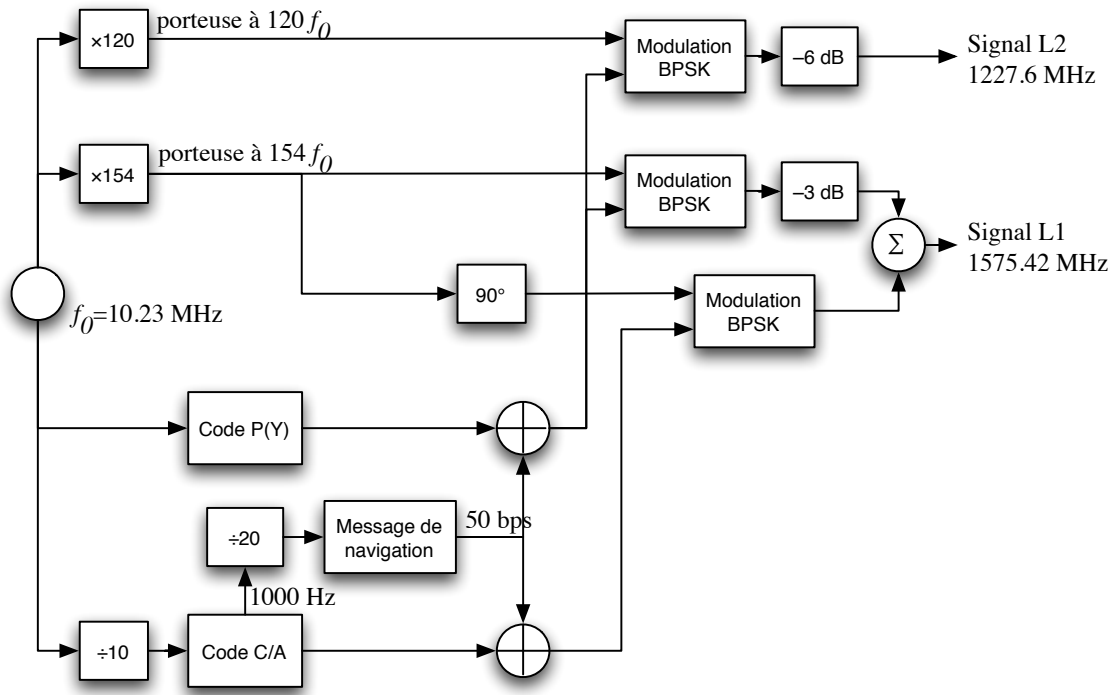


Figure 2.5: Structure et génération du signal GPS

n'est possible cependant que si les mesures sur L1 et L2 sont disponibles à l'instant du calcul.

2.2.1.5 Évolutions futures

La désactivation de la SA décidée au cours de l'année 2000 par le gouvernement américain peut être considérée comme la première étape de la modernisation du système GPS. Ces améliorations visent un double objectif. Dans un premier temps, il s'agit d'améliorer la précision et l'intégrité pour les utilisateurs civils. En parallèle, il s'agit d'améliorer la robustesse du signal GPS face au brouillage pour les utilisateurs militaires.

Le principal effort porte sur une modification des signaux. Les satellites les plus récents ont été modernisés à partir de mai 2005 pour permettre l'émission d'un code civil additionnel sur la porteuse L2. Ce code additionnel (L2C) permettra aux utilisateurs du code civil de corriger les effets ionosphériques par combinaison bi-fréquence.

Parallèlement, une troisième fréquence porteuse (L5 – 1176,45 MHz) sera mise en place, notamment pour des applications de type *Safety-of-life*. Le *Federal Navigation Plan* paru en 2005 prévoit ainsi que 24 satellites émettant le code L2C seront en orbite en 2013 et que 24 satellites émettant la fréquence L5 seront disponibles en 2015.

La modernisation du service dédié aux applications militaires est également prévue. Le code M va ainsi être mis en place en remplacement du code P. Ce code, émis à la fois sur les porteuses L1 et L2 vise à augmenter la robustesse du GPS au brouillage intentionnel.

Toutes ces améliorations conduiront à une augmentation de la précision pour les utilisateurs. Elles sont cependant conditionnées par le renouvellement de la constellation. D'après [Duquenne et al., 2005], une précision métrique pourra être atteinte pour les applications basées sur la mesure de code, à condition d'effectuer une combinaison des codes C/A et L2C pour corriger les effets ionosphériques. L'ajout d'une troisième fréquence permettra également de faciliter la résolution des ambiguïtés pour le positionnement par la phase.

2.2.2 GLONASS

Le système GLONASS⁸ est l'équivalent russe du GPS américain. Déclaré opérationnel en 1997, il a souffert de dégradation et d'interruptions de service en raison de l'absence d'entretien de la constellation par les autorités russes, au point de n'avoir plus que 6 satellites fonctionnels en 2000. De nouveaux satellites ont été mis en orbite depuis le début des années 2000 afin de restaurer le fonctionnement du système.

Comme pour le GPS, le système GLONASS se décompose en trois segments. Le segment spatial compte actuellement 27 satellites dont 22 sont opérationnels. Les satellites sont répartis sur trois plans orbitaux inclinés à environ 65° pour une altitude de 19 000 km. Cette inclinaison plus importante des plans orbitaux permet d'assurer une couverture de meilleure qualité pour les zones à proximité des pôles, la constellation étant optimisée pour la couverture du territoire russe. Contrairement au GPS où chaque satellite émet un code propre sur deux fréquences communes (transmission de type CDMA⁹), les satellites GLONASS émettent un code unique mais sur des fréquences propres (transmission de type FDMA¹⁰). Les fréquences de transmission pour les satellites GLONASS sont déterminées comme suit [Kaplan and Hegarty, 2006] :

$$f = \left(178.0 + \frac{K}{16}\right) \cdot Z \text{ (MHz)} \quad (2.2)$$

où K est un entier compris entre -7 et 4 dépendant du satellite (les satellites situés aux antipodes partagent le même nombre K), et $Z = 9$ pour L1 et $Z = 7$ pour L2.

Contrairement au système américain qui utilise des stations de contrôle réparties tout autour du globe, le segment de contrôle du système GLONASS ne repose que sur des stations basées sur le territoire russe. D'où une dégradation de la précision des

8. GLObalnaya NAVigatsionnaya Spoutnikovaya Sistema

9. Code-Division Multiple Access

10. Frequency-Division Multiple Access

éphémérides en dehors de cette zone et une latence importante en cas de problème sur un satellite puisque le message de navigation de ce dernier ne peut être modifié que lors du survol du territoire russe.

2.2.3 Galileo

2.2.3.1 Historique

Galileo, le système GNSS européen, part du constat que les systèmes de positionnement par satellites ont pris une part importante dans de nombreuses activités et que l'utilisation du GPS est soumise au bon vouloir du ministère de la défense américain. Sur cette base, l'Union Européenne envisage en 1994 la création d'un système GNSS propre à la communauté Européenne et géré par un organisme civil. Le développement du programme Galileo a été officiellement lancé le 26 mars 2002 sur décision de la Commission Européenne et, le 10 décembre 2004, le déploiement opérationnel de Galileo est autorisé. Le programme Galileo devait à l'origine être financé pour un tiers par l'UE et l'Agence Spatiale Européenne (ESA), les deux tiers restants étant financés par des consortiums privés. Le projet a pris beaucoup de retard suite aux rivalités entre états, à la difficulté de choix des consortiums. Le 23 avril 2008, le Parlement Européen a finalement voté le financement entièrement public de Galiléo, avec un objectif de finalisation en 2014. Après les deux satellites expérimentaux Giove-A et Giove-B respectivement mis en orbite le 28 décembre 2005 et le 27 avril 2008, le lancement des deux premiers satellites de la constellation a eu lieu le 21 octobre 2011 à Kourou. La mise en place du segment spatial devrait ensuite s'effectuer au rythme de 4 satellites par an.

2.2.3.2 Description du système

Cinq services de positionnement complémentaires ont été prévus pour Galileo, offrant des niveaux de performance et d'intégrité adaptés aux besoins des différents utilisateurs [Kaplan and Hegarty, 2006] :

- **Open service (OS)** : service ouvert et gratuit pour le positionnement en mode naturel, équivalent au service SPS fournis par le système GPS. Ce service est principalement destiné à des applications grand public et ne disposera pas de dispositif de contrôle d'intégrité.
- **Commercial Service (CS)** : service payant, géré par des fournisseurs d'accès privé, offrant des performances accrues par l'adjonction de deux signaux supplémentaires cryptés, notamment en terme d'intégrité et de garantie de service.
- **Public Regulated Service (PRS)** : service gouvernemental crypté destiné à couvrir les besoins des institutions publiques de la Communauté Européenne en matière de protection civile et de sécurité nationale.
- **Safety of Life (SoL)** : service de haute intégrité pour des applications spécifiques mettant en jeu la sécurité des personnes, tel que le transport aérien ou maritime.

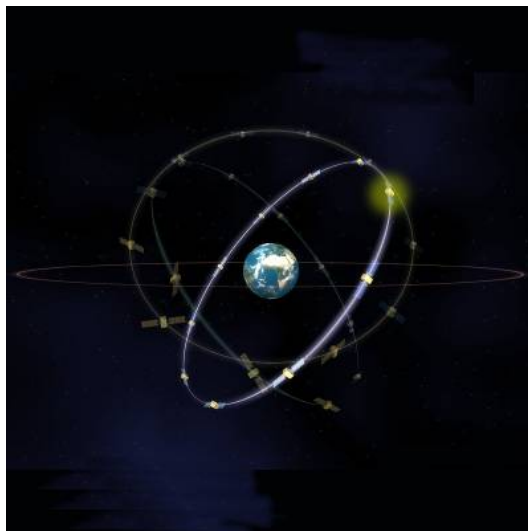


Figure 2.6: Constellation Galileo (Source : ESA)

- **Search And Rescue (SAR)** : service contribuant au système de transmission de message de détresse COSPAR-SARSAT. Il proposera ainsi un canal de transmission permettant la réception et l'émission de messages émis par une balise de détresse en quasi temps-réel, ainsi que l'amélioration de la précision de la localisation de la balise.

Les trois premiers services proposés sont semblables à ceux proposés par le système GPS et le système GLONASS. Cependant le service *SoL* et le service *SAR* présentent une avancée considérable dans le domaine des systèmes de positionnement par satellites.

Lorsque que Galileo sera pleinement opérationnel, le système sera, comme ses homologues russe et américain, composé de trois segments :

- **Segment spatial** : L'ESA envisage le déploiement d'une constellation de 30 satellites répartis sur trois plans orbitaux inclinés à 56° et à une altitude de $23.500km$ environ (Figure 2.6), afin d'assurer une couverture maximale des latitudes les plus élevées. Les signaux sont diffusés sur 3 bandes (L1, E5, E6). La diffusion sur L1 permet une interopérabilité avec le système GPS.
- **Segment de contrôle** : le segment de contrôle de Galileo sera articulé autour de deux centres de calcul principaux situés en Europe et de trente stations de base réparties autour du globe.
- **Segment utilisateurs** : l'ensemble des utilisateurs utilisant les différents services proposés par Galileo.

2.2.4 Les systèmes d'augmentation

Les applications actuelles, notamment en aéronautique, nécessitent plus de précision et d'intégrité dans l'information fournie que le GPS seul ne peut fournir, en parti-

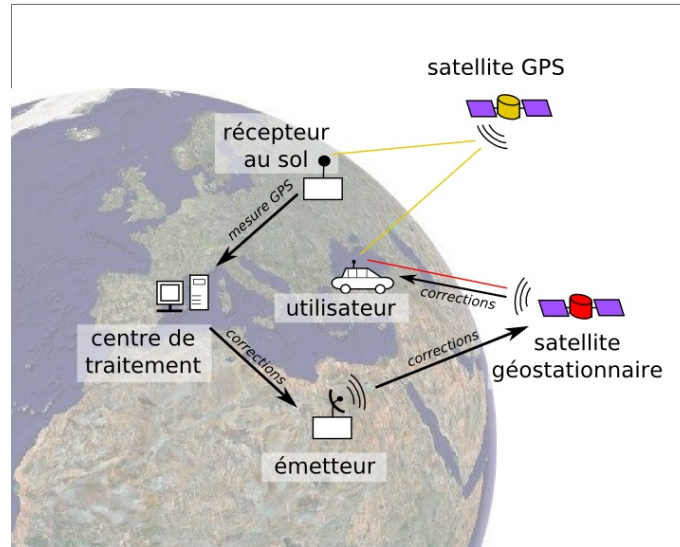


Figure 2.7: Fonctionnement d'un système SBAS

culier pour les phases finales de vol. Ce constat a conduit à la création de *systèmes d'augmentation* permettant l'extension du système GPS. On distingue ainsi les systèmes GBAS (*Ground-Based Augmentation System*), qui reposent uniquement sur une station au sol et dont la portée est limitée, et les systèmes SBAS (*Satellite-Based Augmentation System*), qui reposent sur un ensemble de stations au sol assurant le calcul des paramètres de correction et d'intégrité qui sont ensuite transmis via des satellites géostationnaires.

Il existe à ce jour trois systèmes d'augmentation distincts mais interopérables car ils reposent sur la même norme [RTCA SC-159, 2006]. On distingue ainsi le système EGNOS (*European Geostationary Navigation Overlay Service*) européen, le système WAAS (*Wide Area Augmentation System*) américain et le système MSAS (*MTSAT¹¹-Based Satellite Augmentation System*) japonais. En parallèle, il existe également des services d'augmentation payants, tels que le système *OmniSTAR* par exemple, dédiés aux applications de précision.

Ces systèmes fonctionnent sur le principe du positionnement en mode différentiel (Figure 2.7). Un ensemble de stations au sol assurent le suivi des satellites GPS et fournit l'ensemble des mesures au centre de traitement. La position des stations au sol étant connue, le centre de traitement peut estimer les différentes corrections à appliquer aux mesures et les transmet au récepteur via les satellites géostationnaires. En plus des paramètres de correction, un ensemble de paramètres permettant d'estimer la fiabilité de la solution est transmis et un observable additionnel est créé pour les satellites géostationnaires.

11. Multifunctional Transport Satellites

2.3 Observables GPS et calcul d'une position

Dans cette section, nous présentons le calcul d'une position à partir du système GPS. Le positionnement à l'aide de GLONASS ou du futur Galileo utilise des méthodes similaires.

Sur chaque porteuse (L1 ou L2) les récepteurs GPS peuvent fournir trois types d'observables : la mesure de pseudodistance, qui est une mesure de temps de vol obtenue à partir des codes C/A ou P(Y) ; la mesure de Doppler, la fréquence des signaux reçus varie en fonction de la vitesse relative entre les satellites et le récepteur ; et enfin la mesure de phase, qui compte le nombre de cycles de la porteuse (à une ambiguïté inconnue près).

Nous allons nous concentrer sur le calcul d'une position à partir des mesures de pseudodistance. Les méthodes de positionnement à partir des mesures de phase et de Doppler, et par combinaison d'observables sur les fréquences L1 et L2 sont détaillées dans [Leick, 2004, Kaplan and Hegarty, 2006].

2.3.1 Équation d'observation associée au code

2.3.1.1 Principe de la mesure

On mesure le temps de propagation de l'onde, ou temps de vol $\Delta t = t_r - t_e$, en mesurant son instant d'arrivée t_r sachant que l'on connaît parfaitement son instant d'émission t_e (dans le système de temps GPS).

Les séquences PRN générées par les satellites dépendent du temps et de l'identifiant du satellite selon les codes C/A et P(Y) connus. Les séquences PRN sont des séquences pseudo-aléatoires telles que leur autocorrélation est une série de triangles larges de ± 1 bit avec une période de 1 ms pour le code C/A et une période d'une semaine pour le code P(Y). De plus, les séquences des différents satellites sont orthogonales, si bien que leurs intercorrélations sont nulles.

Comme le code est connu, le récepteur peut générer localement les mêmes séquences que celles des satellites. La mesure du temps de vol se fait alors par corrélation entre le signal reçu et différentes répliques locales du signal décalées dans le temps, correspondant aux différentes mesures possibles de temps de vol (Figure 2.8).

Pour l'acquisition, le récepteur GPS génère une réplique des codes pseudo-aléatoires des différents satellites, avec les 1023 décalages possibles. Comme la fréquence du signal est décalée en raison du mouvement relatif entre satellite et récepteur (effet Doppler), la corrélation de signaux ne peut se faire que si le décalage en fréquence est connu. L'acquisition d'un satellite se fait donc en explorant l'ensemble des combinaisons « phase de code (temps de vol), décalage de fréquence » à la recherche d'un pic de corrélation.

Une fois le signal acquis, le récepteur passe en mode suivi, où seules les corrélations avec les valeurs voisines de la valeur courante sont calculées.

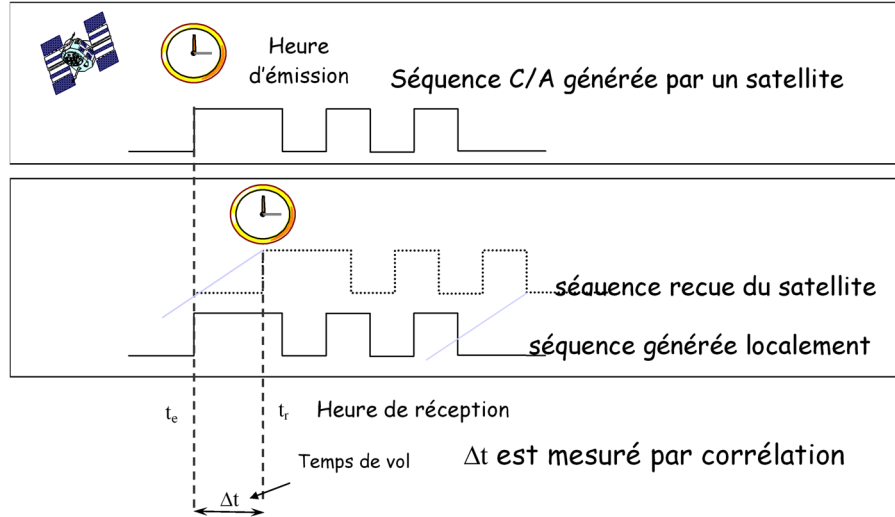


Figure 2.8: Principe de la mesure de pseudodistance

2.3.1.2 Prise en compte des référentiels de temps et d'espace

Référentiel de temps Le référentiel de temps pour les calculs GPS est le temps GPS. En pratique, comme les horloges dérivent, ni le récepteur ni le satellite ne sont parfaitement à l'heure GPS : ils présentent tous les deux un décalage respectif dt_u et dt_s par rapport au temps GPS (Figure 2.9).

A un instant t donné, on a donc

$$\begin{aligned} t_u(t) &= t + dt_u(t) \text{ (heure dans l'échelle de temps du récepteur)} \\ t_s(t) &= t + dt_s(t) \text{ (heure dans l'échelle de temps du satellite)} \end{aligned}$$

On note $t_s(t_e)$ la date à laquelle a été émise la séquence dans l'échelle de temps du satellite i , et $t_u(t_r)$ la date à laquelle a été reçue la séquence dans l'échelle de temps du récepteur.

Le récepteur met en correspondance ces deux séquences pour obtenir une mesure de temps de vol, notée Δt_m ("m" pour mesuré).

$$\Delta t_m = t_u(t_r) - t_s(t_e)$$

Comme $t_u(t_r) = t_r + dt_u(t_r)$ et $t_s(t_e) = t_e + dt_s(t_e)$,

$$\Delta t_m = t_r - t_e + dt_u(t_r) - dt_s(t_e).$$

On suppose que les ondes des signaux se propagent à la vitesse de la lumière (notée c). On définit la *pseudodistance* ρ (exprimée en mètres) entre le satellite et le récepteur

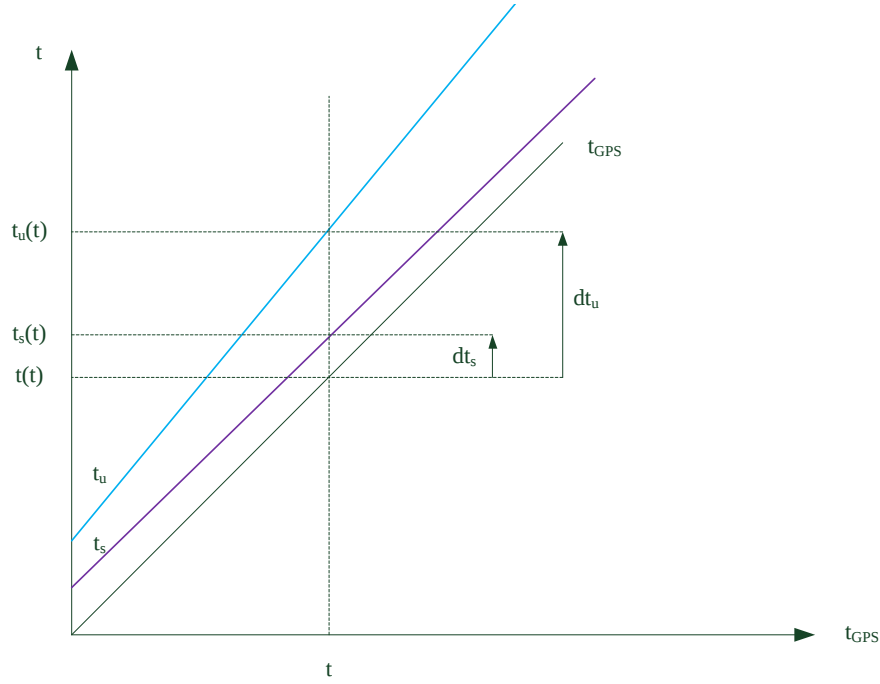


Figure 2.9: Temps d'un satellite et du récepteur

par la relation

$$\begin{aligned}
 \rho &= c.\Delta t_m \\
 &= c(t_r - t_e) + c.dt_u(t_r) - c.dt_s(t_e) \\
 &= R + c.dt_u(t_r) - c.dt_s(t_e)
 \end{aligned} \tag{2.3}$$

où R est la distance géométrique entre le satellite et le récepteur.

On parle de mesure de pseudodistance car l'estimation de la distance fournie par le récepteur est entachée des erreurs d'horloge à la fois du satellite et du récepteur.

Repère spatial Le repère dans lequel se fait le calcul est un repère qui tourne avec la terre. On le note ECEF comme *Earth Centered Earth Fixed*. Ainsi, la distance géométrique est donnée par

$$R = \left\| \mathbf{x}^{ECEF(t_r)}(t_r) - \mathbf{x}_s^{ECEF(t_r)}(t_e) \right\|$$

où $\mathbf{x}$ représente le vecteur position du récepteur dans le repère ECEF lié à l'instant de réception et $\mathbf{x}_s$ le vecteur position du satellite dans le repère ECEF lié également à l'instant de réception¹².

En pratique, les satellites émettent les paramètres de leur orbites en temps GPS. Le

12. Dans ce document, nous adopterons la convention de noter les vecteurs en gras.

récepteur connaissant à peu près le temps de vol peut reconstruire la position du satellite dans le repère $ECEF(t_e)$ correspondant à son instant d'émission. Il peut ensuite en déduire la position du satellite dans repère $ECEF(t_r)$ correspondant à l'instant de réception en appliquant une rotation correspondant à l'angle dont a tourné la terre pendant le voyage de l'onde¹³. On a donc

$$\mathbf{x}_s^{ECEF(t_r)}(t_e) = \mathbf{R}_z(\omega_{\text{earth}}(t_r - t_e)) \cdot \mathbf{x}_s^{ECEF(t_e)}(t_e)$$

où $\mathbf{R}_z(\alpha)$ définit la matrice de rotation d'angle α autour de l'axe z du repère. Cette correction est communément appelée *Earth Rotation correction*.

Afin de simplifier la lecture, nous utiliserons la convention d'écriture suivante, considérant par défaut le repère ECEF lié à l'instant de réception, les positions de satellites à l'instant d'émission et la position du récepteur à l'instant de réception :

$$\begin{cases} \mathbf{x}_s &= \mathbf{x}_s^{ECEF(t_r)}(t_e) \\ \mathbf{x} &= \mathbf{x}^{ECEF(t_r)}(t_r) \end{cases}$$

La distance géométrique devient alors

$$R = \|\mathbf{x}_u - \mathbf{x}_s\|.$$

2.3.1.3 Phénomènes dégradant la mesure de code

La relation entre pseudodistance, distance géométrique et décalage d'horloge de l'équation (2.3) n'est vraie que dans le cas d'un signal évoluant dans le vide. Ce n'est pas le cas des signaux GPS, qui doivent traverser l'atmosphère terrestre. Ils subissent donc un ensemble de retards, se traduisant en allongements des pseudodistances. Nous allons maintenant décrire ces retards, détaillés dans [Durand, 2003, Kaplan and Hegarty, 2006, Leick, 2004, Fouque, 2010].

Retards électroniques Le signal généré par le satellite doit franchir un ensemble de circuits électroniques et de câbles avant d'être émis par l'antenne. Le même phénomène de retard a lieu aussi de l'autre côté de la chaîne, au niveau du récepteur.

Le retard du côté du satellite (noté δ_{TGD} comme *Total Group Delay*) est monitoré par le système de contrôle GPS (depuis 1999). Physiquement, ce retard de groupe correspond au retard entre la génération du signal de code et son émission physique au centre de phase de l'antenne du satellite. Il est propre à chaque satellite et dépend de la configuration matérielle de ce dernier.

Pour le récepteur, tous les signaux reçus passent dans le même câble et sont donc tous retardés de la même valeur. Ce retard vient donc perturber l'estimation du décalage d'horloge du récepteur, mais pas le calcul de position.

13. de l'ordre de 70 ms typiquement

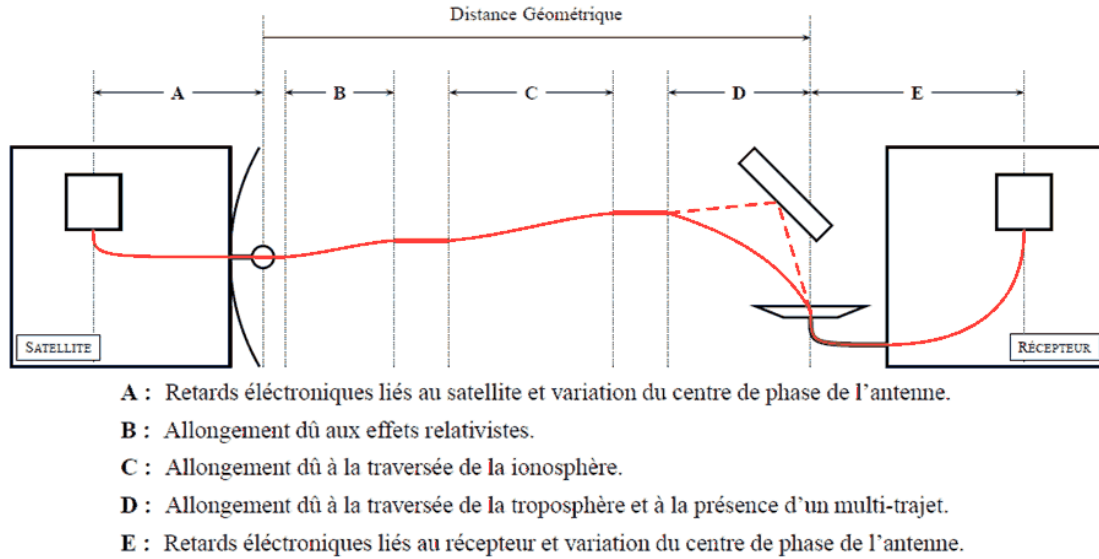


Figure 2.10: Retards dégradant la mesure de code

Effets atmosphériques L'atmosphère est un milieu dispersif, ce qui signifie que sa traversée impacte la vitesse de propagation du signal. Deux couches affectent principalement les signaux GPS : l'ionosphère, de 50 km à 1000 km, et la troposphère, de 0 km à $\simeq 10$ km. De nombreux travaux ont été réalisés sur les effets liés à la traversée de l'atmosphère. Nous nous contenterons donc de rappeler les conséquences de ces effets et nous invitons le lecteur intéressé à se reporter aux différents ouvrages précédemment cités.

Effet de l'ionosphère Située entre 50 et quelques centaines de kilomètres d'altitude, l'ionosphère est une couche haute de l'atmosphère ionisée par l'action des radiations solaires. L'activité de l'ionosphère dépend de la latitude, de l'activité solaire, de la période de l'année, du jour et de la nuit.

Lorsque les signaux GPS traversent ce milieu, tout se passe comme si la distance mesurée (en mesure de pseudodistance) était plus longue que la vraie distance géométrique satellite-récepteur. Ceci est dû au fait que les ondes électromagnétiques traversant le milieu sont retardées par un ralentissement de la vitesse de groupe. Si le récepteur est bi-fréquence, il est possible d'éliminer les effets ionosphériques par combinaison d'observables car l'effet de l'ionosphère varie avec la fréquence du signal.

L'effet sur la mesure induit un allongement δ_{iono} qui peut varier de 0 à 50 m.

Effet de la troposphère Située entre le sol et 8 à 17 kilomètres d'altitude, la troposphère est la couche basse de l'atmosphère.

Elle dépend de la température, de la pression, de l'humidité. Son effet sur la mesure dépend de l'élévation du satellite mais pas de la fréquence. On ne peut recourir qu'à des modèles pour réduire cette perturbation. On peut ainsi utiliser un modèle saisonnier dépendant de la latitude et de la date, ou utiliser un relevé météorologique pour paramétrer un modèle avec des valeurs précises de température, de pression et d'humidité.

Les retards du signal sont dus à la combinaison de deux phénomènes : le trajet des ondes passant dans ce milieu est courbé (et donc la longueur du trajet est augmentée) et la vitesse de propagation décroît.

L'effet sur la mesure (noté δ_{tropo}) peut varier de 2 à 30 m suivant l'élévation du satellite.

Effets relativistes L'évolution à haute vitesse des satellites dans le champ gravitationnel terrestre fait que l'on doit tenir compte des effets liés à la théorie de la relativité. Le système GPS tient compte d'une partie de ces effets en étalonnant les horloges des satellites à la fréquence de 10,22999999543 MHz de façon à obtenir une fréquence apparente au sol de 10,23 MHz. Cependant, l'excentricité des orbites ajoute une composante non fixe sur les horloges qui doit être ajoutée à l'expression du décalage d'horloge. Il en résulte un ralentissement de l'horloge du satellite au périhélie de sa trajectoire, et inversement lorsqu'il passe à l'apogée.

Soit δ_{rel} le décalage d'horloge dû à la relativité. Selon [Kaplan and Hegarty, 2006], il s'exprime sous la forme

$$\delta_{rel}(t) = F \cdot e \cdot \sqrt{a} \cdot \sin E(t) \quad (2.4)$$

avec : $F = -4,442807633 \times 10^{-10} \text{ s} \cdot \text{m}^{-\frac{1}{2}}$, e l'excentricité de l'orbite, a le demi-grand-axe de l'orbite, et $E(t)$ l'anomalie excentrique à l'instant considéré.

[Durand, 2003] donne une valeur maximum de l'ordre de 50 ns, ce qui correspond à un allongement de 15 m environ. Cet effet affecte également l'horloge du récepteur, mais est considéré comme négligeable.

Un autre effet relativiste concernant le système GPS est lié à la rotation de la Terre : c'est l'effet Sagnac. On trouve dans [Kaplan and Hegarty, 2006] une méthode basée sur des changements de repères pour éliminer cet effet.

Le bruit de mesure Le signal provenant du satellite étant fortement atténué, le récepteur est obligé de l'amplifier. Malheureusement le bruit aussi est amplifié. On considère que le bruit de la mesure induit une erreur minimum de 1% de la longueur d'onde du signal sur lequel on fait la mesure ; par exemple, pour le code C/A de longueur d'onde 300 m, on a une erreur de 1 à 3 m. Ce bruit affecte la mesure du retard temporel et donc le calcul de la pseudodistance. On le note β .

Modèle complet de la mesure de code En prenant en compte les diverses dégradations que nous venons d'évoquer, on obtient :

$$\rho = R + c.dt_u - c.dt_s + \delta_{TGD} + \delta_{rel} + \delta_{tropo} + \delta_{iono} + \beta$$

2.3.1.4 Correction des mesures avec les données de navigation

Nous avons vu précédemment que les données du message de navigation fournissent des indications telles que l'état de santé du satellite, son identification, son éphéméride, sa correction d'horloge, la correction ionosphérique, etc.

Grâce au message de navigation, on peut connaître directement le retard TGD $\hat{\delta}_{TGD}$ que l'on considère être celui du code C/A. On peut calculer :

- le décalage d'horloge du satellite $\hat{dt}_s$
- l'allongement ionosphérique $\hat{\delta}_{iono}$
- l'allongement troposphérique $\hat{\delta}_{tropo}$
- les erreurs relativistes $\hat{\delta}_{rel}$

La bibliothèque logicielle GPSTk [Tolman et al., 2004] de l'Université du Texas contient de nombreuses fonctions qui permettent d'estimer ces paramètres. On peut noter que les allongements ionosphérique et troposphérique dépendent de la position du satellite par rapport au récepteur. Ces corrections ne peuvent donc être appliquées qu'une fois qu'une position grossière a été déterminée.

Quand on applique ces corrections, on obtient la pseudodistance corrigée

$$\begin{aligned} \rho &= R + c.dt_u(t_r) - c.dt_s(t_e) + \delta_{TGD} + \delta_{Rel} + \delta_{Tropo} + \delta_{Iono} + \beta \\ &\quad + c.\hat{dt}_s(t_e) - \hat{\delta}_{TGD} - \hat{\delta}_{Rel} - \hat{\delta}_{Tropo} - \hat{\delta}_{Iono} \\ &= R + c.dt_u(t_r) + \beta + \text{erreurs résiduelles} \end{aligned}$$

2.3.1.5 Autres phénomènes induisant des imprécisions sur la mesure

Les multi-trajets La présence d'obstacles près du récepteur (bâtiments, végétation, ...) peut provoquer des réflexions du signal et éventuellement bloquer la propagation du signal direct. Ainsi, une erreur de multi-trajet se produit quand le signal GPS arrive au récepteur après plus d'un trajet à cause de réflexions. Cela se traduit par des retards δ_M , le trajet étant plus long. Ce phénomène n'est pas modélisable car deux mesures faites à des intervalles de temps différents ne donnent pas le même résultat. Cet effet ne dépend que des obstacles proches du récepteur.

L'effet sur les mesures peut atteindre quelques dizaines de mètres (souvent 10 à 20 m) pour les pseudodistances.

Pour réduire ce phénomène, il faut :

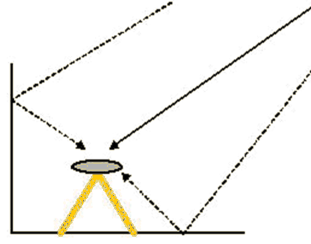


Figure 2.11: Multi-trajets

- Utiliser une antenne polarisée : Les signaux GPS sont émis avec une polarisation circulaire, et la polarisation des signaux est inversée après une réflexion. Une antenne polarisée atténue ainsi fortement les signaux réfléchis, au prix d'un coût et d'un encombrement plus élevés qu'une antenne basique de type « patch ».
- Placer les stations dans des endroits sans obstacle. Ce n'est malheureusement pas applicable pour les véhicules évoluant en milieu urbain.

Ce phénomène pose surtout problème pour les calculs en temps réel avec une durée d'observation courte comme c'est le cas pour une station mobile. En effet, le satellite se déplaçant rapidement, un multi-trajet sur un satellite ne dure pas plus de quelques minutes.

Mauvaise connaissance du centre de phase de l'antenne La position du satellite transmise via les éphémérides radio-diffusées correspond à la position du centre de l'antenne émettrice. Le même phénomène a lieu de l'autre côté de la chaîne, au niveau du récepteur, puisque l'on va calculer la position du centre de l'antenne du récepteur.

De plus, la position mesurée correspond donc en réalité au centre de phase de l'antenne, c'est-à-dire le point immatériel dans l'antenne où le signal semble se matérialiser. Par construction, ce point est proche du centre géométrique de l'antenne. Cependant, sa position est sujette à des variations suivant la puissance du signal reçu, ou émis, la fréquence utilisée, ainsi que l'élévation et l'azimut du satellite émetteur dans le cas du centre de phase de l'antenne réceptrice. Cette variation est également dépendante de la qualité de l'antenne utilisée. D'après [Leick, 2004], la variation moyenne de la position du centre de phase est centimétrique. Au regard de la précision de la mesure de pseudodistance sur le code C/A, métrique environ, on peut négliger ces effets dans l'expression du modèle d'observation pour la mesure de pseudodistance.

Mauvaise connaissance de la position des satellites Les paramètres d'éphéméride qui permettent de retrouver les coordonnées des satellites $\mathbf{x}_s^{ECEF}$ sont estimés à partir d'observations réalisées par les stations permanentes liées au segment de

contrôle. Ces valeurs sont envoyées toutes les deux heures aux satellites qui les répètent dans leurs données de navigation.

Lorsque l'on fait du calcul en temps réel, le récepteur doit donc extrapoler ces paramètres pour retrouver la position du satellite. Il existe ainsi une erreur résiduelle, notée δ_{X_S} , qui affecte la mesure de pseudodistance. D'après [Kaplan and Hegarty, 2006], δ_{X_S} est de l'ordre du mètre à 1σ .

On distingue trois composantes pour l'erreur en position du satellite : l'erreur radiale, l'erreur en position le long de la trajectoire (*along-track*) et perpendiculairement à cette dernière (*cross-track*). L'erreur radiale est la composante ayant l'influence la plus importante sur l'erreur de mesure mais c'est celle qui peut être la plus aisément observée et corrigée par le centre de contrôle. Inversement, les erreurs transversale et le long de la trajectoire ont un impact faible sur l'erreur en position.

L'utilisation des éphémérides précises permet de réduire l'influence de ces termes d'erreurs. Elles ne sont pas disponibles en temps-réel mais peuvent être utilisées en traitement différé.

2.3.2 Calcul d'une solution de localisation

Nous allons dans cette section présenter le calcul d'une solution de position à l'aide des mesures de pseudodistance. L'utilisation d'autres observables, tels que les mesures de Doppler permettant également le calcul de la vitesse du mobile ou la phase pour un positionnement précis sont disponibles dans [Kaplan and Hegarty, 2006, Leick, 2004].

2.3.2.1 Mise en équations

Supposons qu'à un instant donné, m satellites soient visibles et on suppose $m \geq 4$. On se place à un instant de réception noté t_r . Les satellites sont à des distances différentes et donc les instants d'émission $t_{e,i}$ de leur code C/A sont différents les uns des autres. Comme les satellites diffusent des paramètres d'orbites qui permettent de reconstruire leur position, on note $[x_i(t_{e,i}), y_i(t_{e,i}), z_i(t_{e,i})]^T$ leurs coordonnées aux instants d'émission (et on suppose qu'elles sont connues sans erreur).

Notons $d(t_r) = c \cdot dt_u(t_r)$. Le système à résoudre s'écrit

$$\begin{cases} \rho_1(t_r) &= \sqrt{(x(t_r) - x_1(t_{e,1}))^2 + (y(t_r) - y_1(t_{e,1}))^2 + (z(t_r) - z_1(t_{e,1}))^2} + d(t_r) \\ \dots & \\ \rho_i(t_r) &= \sqrt{(x(t_r) - x_i(t_{e,i}))^2 + (y(t_r) - y_i(t_{e,i}))^2 + (z(t_r) - z_i(t_{e,i}))^2} + d(t_r) \\ \dots & \\ \rho_m(t_r) &= \sqrt{(x(t_r) - x_m(t_{e,m}))^2 + (y(t_r) - y_m(t_{e,m}))^2 + (z(t_r) - z_m(t_{e,m}))^2} + d(t_r) \end{cases}$$

Pour simplifier la notation, on ré-écrit ce modèle comme

$$\begin{cases} \rho_1 &= \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2} + d \\ \dots & \\ \rho_i &= \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} + d \\ \dots & \\ \rho_m &= \sqrt{(x - x_m)^2 + (y - y_m)^2 + (z - z_m)^2} + d \end{cases}$$

En notant $\mathbf{x} = [x, y, z, d]^T$ et $\rho = [\rho_1, \dots, \rho_m]^T$, on obtient sous forme vectorielle

$$\rho = \mathbf{g}(\mathbf{x}) \quad (2.5)$$

Avec $m > 4$, on est amené à résoudre un système non-linéaire redondant. Une méthode classique de résolution est la méthode des moindres carrés.

2.3.2.2 Résolution classique

Réécrivons (2.5) en tenant compte de l'erreur $\mathbf{e}_\rho$ sur les mesures de pseudodistances, soit

$$\mathbf{g}(\mathbf{x}) = \rho + \mathbf{e}_\rho, \quad (2.6)$$

en supposant que l'erreur sur les pseudodistances $\mathbf{e}_\rho$ suit une loi normale centrée $\mathbf{e}_\rho \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_\rho)$ de variance $\mathbf{Q}_\rho$.

Si le nombre de mesures m est supérieur ou égal à 4, nous pouvons résoudre le problème de localisation (2.6) en utilisant une méthode de moindres carrés pondérés non-linéaire, basée sur la méthode de Gauss-Newton. La matrice de pondération utilisée est l'inverse $\mathbf{Q}_\rho^{-1}$ de la matrice de variance/covariance de l'erreur d'observation.

Connaissant une solution approchée $\mathbf{x}_0$, on peut linéariser (2.5) autour de $\mathbf{x}_0$,

$$\mathbf{g}(\mathbf{x}_0) + \mathbf{M}_k(\mathbf{x} - \mathbf{x}_0) = \rho$$

où $\mathbf{M}_k$ est la jacobienne de $\mathbf{g}$ en $\mathbf{x}_k$:

$$\mathbf{M}_k = \mathbf{g}'(\mathbf{x}_k) = \begin{bmatrix} \frac{\partial g_1(\mathbf{x}_k)}{\partial x_1} & \dots & \frac{\partial g_1(\mathbf{x}_k)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_m(\mathbf{x}_k)}{\partial x_1} & \dots & \frac{\partial g_m(\mathbf{x}_k)}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \frac{x-x_1}{R_1} & \frac{y-y_1}{R_1} & \frac{z-z_1}{R_1} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x-x_m}{R_m} & \dots & \frac{z-z_m}{R_m} & 1 \end{bmatrix} \quad (2.7)$$

En posant $\Delta \mathbf{x} = \mathbf{x} - \mathbf{x}_0$ et $\Delta \rho = \rho - \mathbf{g}(\mathbf{x}_0)$, on a

$$\mathbf{M}_k \Delta \mathbf{x} = \Delta \rho.$$

Quand $m > 4$, la matrice $\mathbf{M}_k$ n'est pas carrée, on ne peut donc pas l'inverser pour déterminer $\Delta \mathbf{x}$. On forme donc un problème d'estimation aux moindres carrés pondérés, consistant à inverser la matrice $(\mathbf{M}_k^T \mathbf{Q}^{-1} \mathbf{M}_k)$:

$$(\mathbf{M}_k^T \mathbf{Q}^{-1} \mathbf{M}_k) \Delta \mathbf{x} = \mathbf{M}_k^T \mathbf{Q}^{-1} \Delta \rho.$$

L'algorithme de Gauss-Newton consiste, en partant d'une solution initiale $\mathbf{x}_0$, à résoudre le problème d'estimation linéarisé en $\mathbf{x}_0$, puis d'itérativement utiliser la solution obtenue comme nouveau point de linéarisation pour la résolution du problème. Pour la $k^{ième}$ itération de l'algorithme, les équations permettant de converger vers la solution sont définies par

$$\Delta \mathbf{x}_k = (\mathbf{M}^T \mathbf{Q}_\rho^{-1} \mathbf{M})^{-1} \mathbf{M}^T \mathbf{Q}_\rho^{-1} \Delta \rho_k \quad (2.8)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x}_k \quad (2.9)$$

avec

$$\Delta \rho_k = \rho - \mathbf{g}(\mathbf{x}_k) \quad (2.10)$$

Les équations (2.10, 2.7, 2.8, 2.9) permettent ainsi de calculer l'estimation de la position de façon itérative.

En pratique, seule la localisation initiale peut nécessiter plusieurs itérations de la méthode de Gauss-Newton. En effet, comme les satellites sont situés à plus de 20 000 km du récepteur, le problème de localisation par mesures de pseudodistance se linéarise parfaitement à proximité de la solution. La position précédente est donc utilisée comme solution initiale de l'algorithme, qui converge alors en une seule itération.

Dans le cas où la position initiale est inconnue, on peut recourir à une solution analytique telle que la méthode de Bancroft [Bancroft, 1985] afin d'initialiser la méthode de résolution aux moindres carrés.

2.4 Qualité de service d'un système de positionnement

2.4.1 Définitions générales

La disponibilité (availability) d'un système est sa capacité à fournir la fonction demandée avec les performances requises au moment où l'on souhaite commencer

une manœuvre. Elle indique la capacité du système à fournir un service utilisable, à l'intérieur de sa zone de couverture.

La continuité (continuity) d'un système est la capacité pour l'ensemble du système de réaliser sa fonction sans interruption pendant la manœuvre considérée. Plus particulièrement, la continuité est la probabilité que les performances spécifiées soient maintenues pendant la durée d'une phase d'opération, en supposant que le système était disponible au début de la phase d'opération et qu'il devait opérer pendant toute l'opération.

2.4.2 Métriques utilisées en aéronautique

En aéronautique, la localisation de l'avion est une information critique, en particulier lors des phases d'approche. Les critères requis pour l'utilisation aéronautique du GPS sont définis dans les *Minimal Operational Performance Standards* (MOPS) [RTCA SC-159, 2006].

Modes de navigation et performance requises Les performances attendues d'un système de localisation dépendent de la tâche à accomplir. En aéronautique, le vol est décomposé en différentes phases (*En route*, *Terminal*, *Approach*, ...), pour lesquelles les critères de performances requis sont spécifiés. On parle ainsi de *Required Navigation Performance* (RNP) défini pour des modes de navigation donnés.

La figure 2.12 présente l'enchaînement des différents modes de navigation au cours d'un vol. Sont ainsi définis la précision horizontale et/ou verticale requise selon la phase de vol, les limites d'alertes sur l'erreur commise en position, ainsi que des critères concernant la continuité, la disponibilité et l'intégrité.

Limites d'alerte On définit pour un système de positionnement des limites sur l'erreur de position commise. Ces limites font partie des spécifications sur la performance de navigation requise, et sont déterminées en fonction de l'application et du type de manœuvre à réaliser.

La limite d'alerte horizontale (HAL) est le rayon d'un cercle dans le plan horizontal (le plan localement tangent à l'ellipsoïde WGS-84) ayant pour centre la vraie position, et qui délimite la région qui doit contenir la position indiquée avec la probabilité requise pour un mode de navigation particulier (par exemple, pour la phase « en route », la probabilité de dépassement de la HAL est spécifiée à 10^{-7} par heure de vol).

On définit de la même façon la limite d'alerte verticale (VAL) comme la demi-longueur d'un segment sur l'axe vertical (orthogonal au plan horizontal de l'ellipsoïde WGS-84) ayant pour centre la vraie position, et qui décrit la région qui doit contenir la position indiquée avec une probabilité spécifiée pour un mode de navigation particulier.

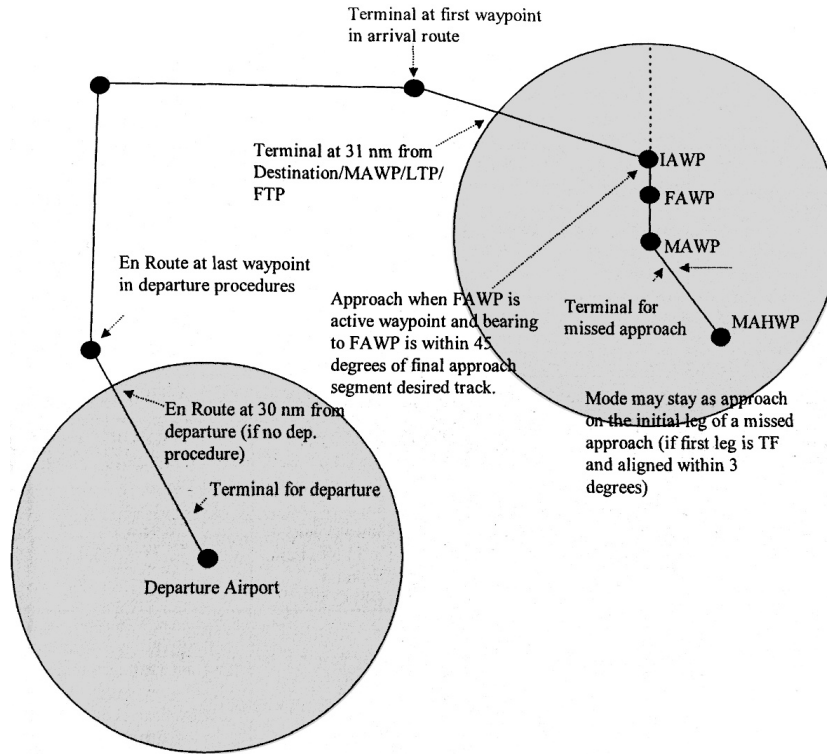


Figure 2.12: Modes de navigation au cours d'un vol (RTCA/DO-229D)

Dans le cas où l'erreur de position dépasse une des limites d'alertes, le système de navigation doit déclencher une alerte et indiquer son incapacité à fournir une position conforme aux performances requises. Le but des méthodes d'intégrité pour le positionnement par satellites est ainsi de détecter et éventuellement exclure les défauts qui pourraient perturber la solution de position au delà de la limite d'alerte spécifiée.

Afin de s'assurer que la solution de navigation fournie est à l'intérieur du volume défini par les limites d'alertes, un mécanisme d'intégrité est nécessaire. La section suivante présente le mécanisme classique d'intégrité embarqué dans les récepteurs GPS utilisés en aéronautique.

Intégrité L'intégrité est définie par la combinaison des paramètres suivants :

- Le *time to alert*, qui correspond au temps entre l'occurrence d'une condition d'alerte et le moment où l'alerte est reçue par l'utilisateur ;
- La *limite d'alerte*, qui correspond à l'erreur tolérable maximale dans la solution de position avant qu'une alerte ne soit signalée dans le *time to alert* spécifié ;
- Le *risque d'intégrité*, qui est la probabilité que, pendant une période de fonctionnement continue, l'erreur de position horizontale ou verticale dépasse la limite d'alerte correspondante sans que l'utilisateur n'en soit informé dans le *time to alert*.

2.5 Calcul classique d'intégrité autonome

Lorsqu'un système de positionnement est utilisée dans une application mettant en jeu la sécurité des personnes ou la responsabilité légale d'une société, il faut pouvoir garantir (à un risque près) l'intégrité de la position calculée à chaque instant. Il peut, de plus, être nécessaire d'être capable de prédire si le système pourra satisfaire les RNP pour accomplir une mission.

L'intégrité peut être fournie par un système externe, surveillant la constellation et calculant des paramètres de correction, d'intégrité des satellites, ainsi que des indicateurs permettant de quantifier l'erreur de mesure commise. C'est le cas des systèmes d'augmentation par satellites SBAS tels que les système américain WAAS, le système européen EGNOS et le système japonais MSAS. Ces systèmes permettent de détecter et éventuellement corriger les erreurs liées aux satellites (horloge et orbite) ainsi qu'à la propagation du signal dans l'atmosphère. Cependant, ils ne permettent pas de protéger contre les erreurs liées à l'environnement proche du récepteur (multitrajets, interférences). Un mécanisme de surveillance de l'intégrité par le récepteur est donc nécessaire. Ce mécanisme qui correspond à la vérification autonome de l'intégrité fait l'objet de cette section. On renvoie le lecteur intéressé par l'intégrité SBAS à l'ouvrage de référence [RTCA SC-159, 2006].

2.5.1 Surveillance autonome d'intégrité

Les récepteurs GPS peuvent embarquer un algorithme de contrôle d'intégrité autonome, appelé *Receiver Autonomous Integrity Monitoring* (RAIM) [Brown, 1987]. Le RAIM consiste à vérifier l'intégrité de la solution de localisation en profitant de la redondance des mesures.

Le principe général des méthodes d'intégrité repose sur une méthodologie en trois étapes nommées *détection, identification et adaptation* (DIA) [Teunissen, 1990], dont un cas particulier est la *détection et l'exclusion de défauts* (*Fault Detection and Exclusion* FDE dans la littérature). Les méthodes de détection de défauts sont issus de domaines du diagnostic et du traitement du signal, tel que la détection de changements abrupts dans des signaux bruités [Basseville and Nikiforov, 1993]. Les algorithmes de RAIM les plus simples implémentent au minimum une étape de détection, qui consiste à effectuer un test statistique lorsqu'il y a une redondance de mesures¹⁴. Ce test, appelé test global, est appliqué à l'ensemble du modèle pour vérifier si un problème d'intégrité est présent. Lorsqu'il n'y a pas de redondance, on dit que ce dernier n'est pas possible (Figure 2.13).

Dans le cas où la présence d'une faute a été détectée, l'identification consiste en des tests sur chacune des hypothèses de défaut. En RAIM, les hypothèses de défaut

14. Dans la norme DO229D, le terme RAIM signifie uniquement détection de défaut. Dans la pratique, il est d'usage de parler de RAIM pour désigner les algorithmes de FDE et de calcul de niveau de protection. C'est l'usage qui sera fait de ce terme dans ce manuscrit.

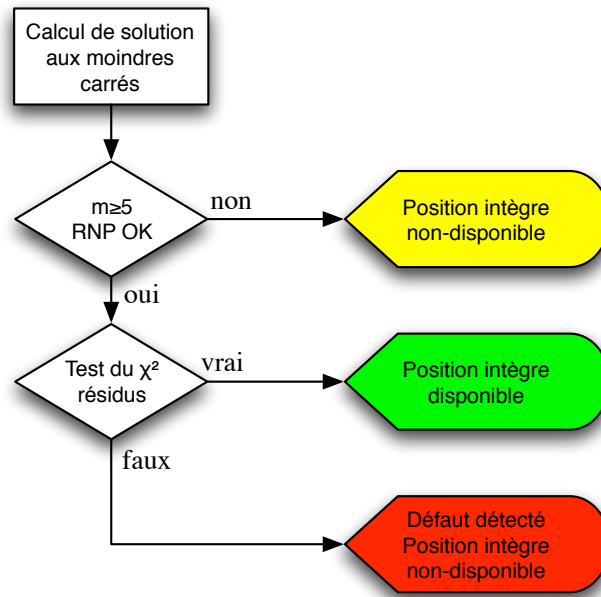


Figure 2.13: Algorithme de RAIM

correspondent en général à des biais sur les pseudodistances et l'identification est tentée sur chaque mesure. Enfin, si la phase d'identification a abouti, l'adaptation consiste à prendre en compte la connaissance de la présence de défaut pour modifier le modèle d'observation comme intégrer l'estimation d'un biais par exemple. En RAIM, les mécanismes d'adaptation consistent souvent à évincer les mesures marquées comme aberrantes du calcul de solution afin de restaurer l'intégrité du positionnement. Ainsi, la continuité du positionnement intègre est améliorée (Figure 2.14).

En suivant les étapes classiques de RAIM, la position calculée à chaque époque est indiquée comme intègre dans deux cas [RTCA SC-159, 2006] :

- Le test global est possible et a réussi ;
- Le test global a échoué, mais la ou les mesures aberrantes ont été identifiées et écartées avec succès (c'est-à-dire qu'après suppression des mesures identifiées comme aberrantes, un nouveau test global est encore possible et passé avec succès). La solution fournie est alors la position calculée après suppression des données aberrantes.

Au contraire, la perte d'intégrité est signalée dans deux cas :

- Le test global a échoué, et il n'a pas été possible d'identifier avec succès les mesures aberrantes ;
- Le test global est impossible car il n'y a pas de redondance ou bien, après l'étape de FDE, la suppression des mesures aberrantes fait qu'il ne reste plus suffisamment de mesures pour vérifier l'intégrité de la solution.

Le récepteur Septentrio PolaRx2 intègre une méthode de RAIM basée sur ce principe, et est capable de rejeter plusieurs mesures aberrantes simultanées [Septentrio, 2005].

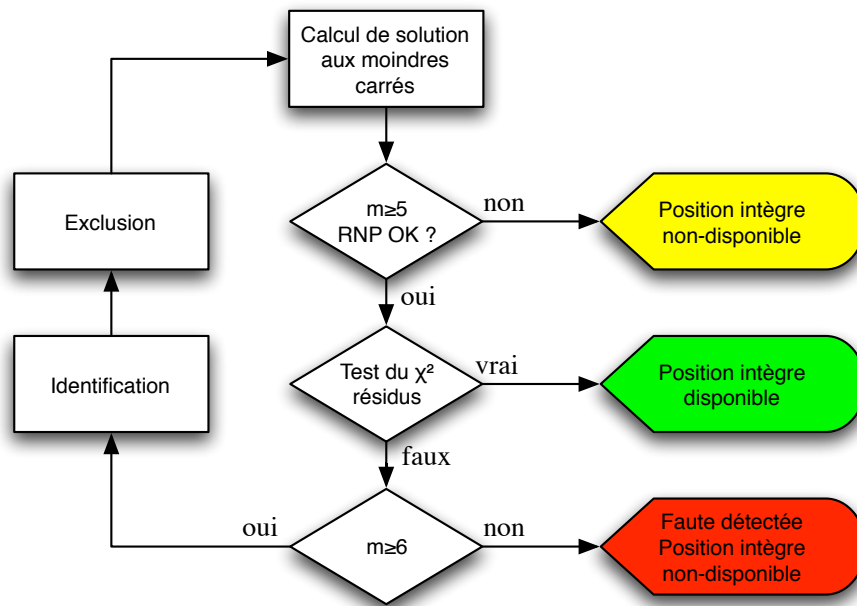


Figure 2.14: Algorithme de FDE

Table 2.1: Résultats du test d'hypothèse

	Pas de faute	Mesure aberrante
Faute détectée	Fausse alarme (Erreur de type I)	Correct
Faute non détectée	Correct	Non détection (Erreur de type II)

Le RAIM est basé sur des tests statistiques, qui supposent la connaissance d'un modèle a priori de la distribution de l'erreur de mesure. On peut classer le résultat du test en quatre cas classiques, présentés dans le tableau 2.1.

L'étape de détection de défaut du RAIM produit une décision correcte dans deux cas : une mesure aberrante dans les données est détectée avec succès ; aucune mesure aberrante n'est détectée quand aucune n'est présente. Cependant, quand aucune mesure aberrante n'est présente et que le RAIM déclare la présence d'une erreur, une *fausse alarme*¹⁵ est déclenchée. Quand une mesure aberrante n'est pas détectée, on parle de *non détection*.

On présente dans la suite de ce chapitre, les principes classiques des méthodes de RAIM utilisant des méthodes linéaires basées sur les résidus, leur seuillage et une décision logique associée. Dans le cas linéaire, on peut montrer que cette méthode est équivalente aux méthodes basées sur l'espace de parité ou la séparation de solutions

15. Une alerte s'adresse à un utilisateur tandis qu'une alarme est un événement interne qui peut conduire à déclencher une alerte le cas échéant.

[Brown, 1992].

2.5.1.1 Impact d'un défaut sur le calcul aux moindres carrés

Soit x l'état du système à estimer (ici, la position et le décalage d'horloge). Supposons que le modèle d'observation faisant intervenir une mesure y soit linéaire ou bien linéarisé autour d'un point avec une bonne approximation,

$$y = Hx + b + e_y$$

où b est le vecteur des défauts et $e_y \sim \mathcal{N}(0, Q_y)$ un bruit aléatoire de distribution normale centrée et de covariance Q_y . On a ainsi

$$y \sim \mathcal{N}(\mu_y, Q_y) \text{ avec } \mu_y = Hx + b$$

L'estimation à variance minimale est donnée par [Aitken, 1934]

$$\begin{aligned} \hat{x} &= \left(H^T Q_y^{-1} H \right)^{-1} H^T Q_y^{-1} y \\ &= H^+ y \end{aligned}$$

où $H^+ = \left(H^T Q_y^{-1} H \right)^{-1} H^T Q_y^{-1}$ est la pseudo-inverse pondérée de la matrice H .

Notons r le vecteur des résidus. On a

$$\begin{aligned} r &= y - H\hat{x} \\ &= y - HH^+ y \\ &= \left(I - HH^+ \right) y \\ &= Sy \end{aligned} \tag{2.11}$$

Comme y est une variable aléatoire gaussienne, le vecteur de résidus r suit également une loi gaussienne. Ainsi, on a $r \sim \mathcal{N}(\mu_r, Q_r)$, avec $\mu_r = S\mu_y$ et $Q_r = SQ_y S^T$.

Dans le cas où le vecteur des défauts b est nul, on a

$$\begin{aligned} \mu_r &= S\mu_y = S(Hx + b) \\ &= SHx \\ &= \left(I - HH^+ \right) Hx \\ &= Hx - Hx \\ &= 0. \end{aligned}$$

La moyenne du vecteur des résidus est donc nulle en l'absence de défaut.

En revanche, en présence de défauts, la moyenne du vecteur de résidus est donnée par

$$\begin{aligned}\mu_r &= S\mu_y \\ &= S(Hx + b) \\ &= Sb.\end{aligned}$$

La moyenne des résidus n'est ainsi impactée que par les défauts, au travers de la matrice S . Cette propriété est à la base du test d'hypothèse de la section suivante.

2.5.1.2 Détection de faute

On formule l'hypothèse statistique suivante pour déterminer la présence d'une faute impliquant une erreur sur une pseudodistance mesurée. On définit ainsi l'hypothèse nulle H_0 « aucune faute ne se produit » et l'hypothèse alternative H_1 « présence d'une faute ».

$$\begin{aligned}H_0 &: E\{y\} = Hx, \text{Var}\{y\} = Q_y \\ H_1 &: E\{y\} = Hx + b_y, \text{Var}\{y\} = Q_y\end{aligned}$$

L'algorithme de RAIM réalise un test statistique pour déterminer si l'hypothèse H_0 doit être rejetée ou si elle peut être acceptée. Le test consiste en une comparaison entre une variable de décision et un seuil T . La variable de décision souvent choisie est la somme des carrés des résidus normalisés par la variance σ_i^2 de chaque mesure correspondante (on se place dans le cas général où les mesures peuvent avoir des variances différentes). On la note $NSSE$ et elle est définie par

$$NSSE = \sum_{i=1}^m \left(\frac{r_i}{\sigma_i} \right)^2.$$

Avec un vecteur y de m mesures indépendantes, $NSSE$ suit une loi du χ^2 à $m - n$ degrés de liberté. Dans le cas du calcul d'une position GPS, où le vecteur de paramètres est de dimension 4, la somme des carrés des résidus normalisés suit une loi du χ^2 à $m - 4$ degrés de liberté. La détection de mesures aberrantes nécessite donc au moins 5 mesures. Sous l'hypothèse H_0 de l'absence de défaut, le vecteur de résidus suit une loi normale centrée, aussi $NSSE$ suit une loi du χ^2 centrée. Au contraire dans l'hypothèse H_1 de présence d'un défaut, $NSSE$ suit une loi du χ^2 de paramètre de non-centralité δ . Comme on l'a déjà vu, les défauts biaisent les résidus et on suppose que les variances sont connues.

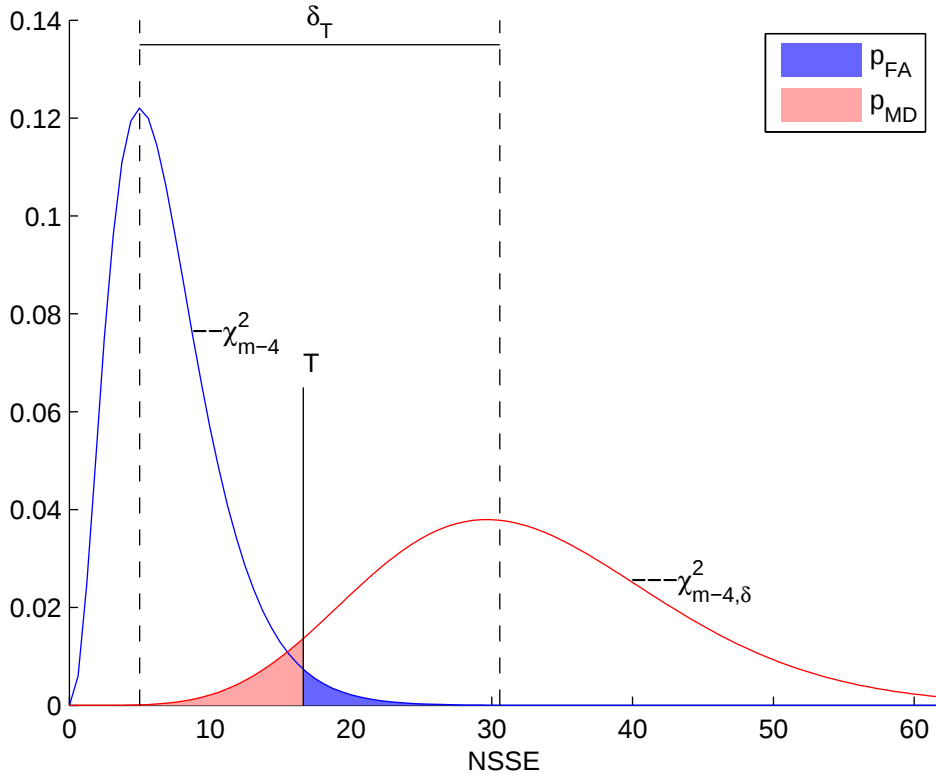


Figure 2.15: Exemple de test avec les distributions centrée χ^2_{m-4} et non centrée $\chi^2_{m-4,\delta}$ pour $m = 11$.

On peut ainsi reformuler les hypothèses H_0 et H_1 :

$$\begin{aligned} H_0 &: NSSE \sim \chi^2_{m-4} \\ H_1 &: NSSE \sim \chi^2_{m-4,\delta} \end{aligned}$$

On définit un seuil de décision T , que l'on compare à la somme des carrés pondérés des résidus afin de choisir l'hypothèse retenue :

- si $NSSE < T$, H_0 est acceptée
- si $NSSE \geq T$, H_0 est rejetée, on accepte l'alternative H_1

Le test peut conduire à deux types d'erreurs : la non-détection d'une faute et la fausse alarme. Les probabilités respectives p_{MD} et p_{FA} de ces deux événements sont les suivantes :

$$\begin{aligned} p_{MD} &= P(NSSE < T \mid H_1) \\ p_{FA} &= P(NSSE > T \mid H_0) \end{aligned}$$

La densité de probabilité d'une distribution du χ^2 à v degrés de liberté est

$$f(x) = \frac{x^{(v-2)/2} e^{-x/2}}{2^{v/2} \Gamma(v/2)}$$

où $\Gamma(m) = \int_0^\infty x^{m-1} e^{-x} dx$.

On peut ainsi exprimer la probabilité de fausse alarme

$$p_{FA} = 1 - \int_0^T \frac{x^{(v-2)/2} e^{-x/2}}{2^{v/2} \Gamma(v/2)} dx.$$

De la même façon, la probabilité de non détection peut être déterminée par une loi du χ^2 non centrée, soit

$$p_{MD} = \int_0^T \frac{e^{-(x+\delta)/2}}{2^{v/2}} \sum_{j=0}^{\infty} \frac{x^{j-1+v/2} \delta^j}{\Gamma(j + (v/2)) 2^{2j} j!} dx. \quad (2.12)$$

En pratique, les probabilités de fausse alarme et de non-détection sont spécifiées, et il est nécessaire de déterminer le seuil T et le paramètre de non-centralité δ de la loi du χ^2 représentant la distribution en présence d'erreur. La stratégie usuelle est de définir le seuil T en fonction de p_{FA} . Une fois ce seuil défini, on peut alors déterminer δ en fonction de p_{MD} , en « calant à gauche » la distribution du χ^2 non-centrée sur le seuil (cf. figure 2.15).

2.5.1.3 Identification et adaptation

Si le test sur la somme pondérée des résidus au carré conclut à la présence d'une erreur, il peut être nécessaire d'identifier la ou les mesures affectées, en particulier si une adaptation du modèle d'observation aux mesures aberrantes est effectuée (la plus courante étant l'exclusion des mesures aberrantes).

On admet communément, dans le cas du GPS, l'hypothèse qu'un seul défaut peut survenir à la fois¹⁶. L'identification du défaut se base alors sur une mesure de corrélation entre le vecteur des résidus courant et le vecteur des résidus qui aurait été obtenu en présence d'un défaut unitaire $b_i = [0_1 \dots 1_i \dots 0_m]^T$ sur la $i^{ème}$ mesure :

$$corr_i = \frac{(Sb_i)^T \cdot r}{\|Sb_i\| \|r\|}.$$

16. Cette hypothèse, souvent retenue dans les applications aéronautiques, n'est cependant valable qu'en environnement dégagé. En effet, il n'est pas rare de rencontrer des défauts sur plusieurs mesures simultanément lors de la navigation en milieu urbain [Le Marchand et al., 2009].

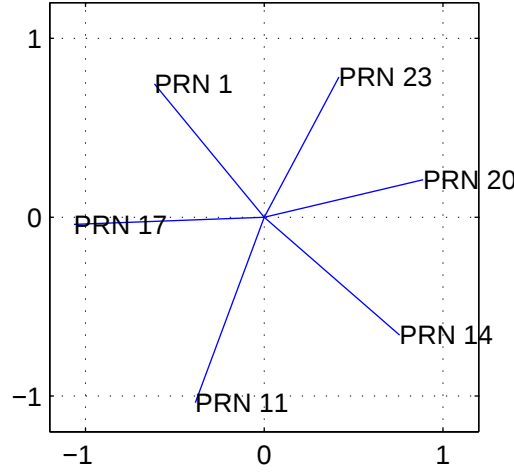


Figure 2.16: Projection dans le plan le plus explicatif des vecteurs résidus correspondant à chaque défaut unitaire, avec 6 satellites.

On détermine le défaut i_{fault} en prenant la plus grande valeur absolue de corr_i , soit

$$i_{\text{fault}} = \arg \max_{i=1\dots m} |\text{corr}_i|.$$

Un calcul plus simple et équivalent, en considérant qu'un défaut ne touche qu'une mesure à la fois, est le « w -test » [Baarda, 1968], consistant à trouver le maximum sur i de la valeur absolue de w_i , avec

$$w_i = \frac{r_i}{\sigma_i \sqrt{S_{i,i}}},$$

$$i_{\text{fault}} = \arg \max_{i=1\dots m} |w_i|.$$

Un minimum de six mesures de pseudo-distance est nécessaire pour pouvoir effectuer l'identification. On démontre que ces tests permettent d'identifier le défaut en l'absence de bruit dans le cas où un seul défaut est présent (voir la thèse d'Olivier LE MARCHAND [Le Marchand, 2010]). Cependant, si les vecteurs Sb_i ne sont pas suffisamment décorrélés en raison de la géométrie du problème, l'identification peut échouer en présence de bruit. On parle de *séparabilité* pour quantifier la difficulté à distinguer les différentes hypothèses de défaut. Cette séparabilité est liée à l'angle entre les vecteurs Sb_i : elle est maximale entre deux défauts dont les vecteurs de résidus sont orthogonaux, et impossible quand les vecteurs sont colinéaires. La figure 2.16 représente, projetés dans le plan le plus explicatif, les vecteurs résidus correspondant à chaque défaut pour un exemple à 6 satellites.

Le tableau 2.2 montre, sur le même exemple, les corrélations entre les vecteurs de

Table 2.2: Corrélations entre les vecteurs de résidus générés par des hypothèses de défaut unitaire. Plus la valeur absolue est faible, plus les hypothèses de défaut sont séparables.

PRN	1	11	14	17	20	23
1	1.0000	-0.6159	-0.9271	0.5904	-0.2730	0.4662
11		1.0000	0.2757	0.2721	-0.5897	-0.9841
14			1.0000	-0.8500	0.6138	-0.1006
17				1.0000	-0.9376	-0.4387
20					1.0000	0.7238
23						1.0000

résidus générés en présence d'un défaut unitaire sur chaque pseudo-distance. Grâce à la dernière colonne du tableau, on remarque par exemple que la géométrie rend très difficile la distinction d'un défaut sur le satellite 11 d'un défaut sur le satellite 23, même en présence d'un faible bruit. Cela était également visible sur la figure 2.16 où les vecteurs de résidus associés aux défauts sur ces deux satellites sont presque colinéaires.

2.5.2 Quantification de l'intégrité

Nous venons de décrire une méthode de détection et d'exclusion de faute, pour laquelle les probabilités de fausse alarme et de non-détection peuvent être spécifiées. Une méthode d'intégrité définie par p_{FA} et p_{MD} est sans intérêt pour la localisation intègre si on n'a pas une mesure des défauts et des erreurs dont elle nous protège. Aussi, nous allons maintenant nous attacher à quantifier la taille des plus petits défauts détectables (intégrité interne), ainsi que leur influence sur la solution calculée (intégrité externe).

2.5.2.1 Intégrité interne

Pour chaque mesure de pseudodistance, on peut calculer le *biais d'amplitude minimale détectable* par l'algorithme de détection de faute. Ce biais est appelé *Minimal Detectable Bias* (MDB).

Le MDB décrit l'intégrité interne du test sur le modèle global, en fournissant une mesure de l'erreur minimale de pseudodistance qui peut être détectée avec une probabilité de non-détection donnée. Sous hypothèse d'un seul défaut à la fois, on peut

montrer facilement que, pour chaque satellite, le MDB est le suivant ¹⁷ :

$$MDB_i = \nabla_i = \frac{\sigma_i \sqrt{\delta}}{\sqrt{S_{i,i}}} \quad (2.13)$$

Pour le test d'hypothèse associée à la détection de faute, le paramètre de non-centralité δ est associé à une probabilité de non-détection p_{MD} et à un seuil de détection T .

Le MDB peut être calculé de deux manières, dépendant de la valeur de δ utilisée. On peut utiliser la valeur de δ correspondant au seuil de détection T choisi afin de satisfaire la probabilité de fausse alarme du test global. C'est la définition classique du MDB, qui ne dépend pas des mesures courantes.

On peut également définir un MDB_{obs} basé sur les observations courantes, c'est-à-dire sur la valeur actuelle de la NSSE. Dans ce cas, le paramètre de non-centralité δ_{obs} est déterminé à partir de la probabilité de non-détection et de la NSSE.

Il est important de noter que le calcul de δ à partir d'un seuil et de p_{MD} nécessite d'inverser l'équation 2.12. Cela rend le calcul de δ_{obs} (et par conséquent de MDB_{obs}) plus difficilement applicable dans des applications embarquées. On a en effet, d'après l'équation 2.12,

$$p_{MD} = \int_0^{NSSE} \frac{e^{-(x+\delta)/2}}{2^{v/2}} \sum_{j=0}^{\infty} \frac{x^{j-1+v/2} \delta_{obs}^j}{\Gamma(j + (v/2)) 2^{2j} j!} dx.$$

2.5.2.2 Intégrité externe

L'intégrité externe vise à quantifier l'impact des défauts non détectés sur l'estimation de position. Pour la quantifier de façon simple, on formule l'hypothèse qu'un seul défaut n'a pas été détecté. La méthode de calcul consiste ensuite à considérer le pire des cas, c'est-à-dire que le défaut apparaît sur la mesure ayant le plus d'impact sur la solution de position.

L'intégrité externe quantifie ainsi l'impact de chaque MDB_i sur l'erreur de position dans le pire cas en passant par la pseudo-inverse :

$$\|x - \hat{x}\| < \max_i \sqrt{\sum_{j=1}^4 (H_{j,i}^+)^2} \cdot MDB_{obs,i}.$$

17. Une démonstration en est faite en annexe de la thèse d'Olivier LE MARCHAND [Le Marchand, 2010]

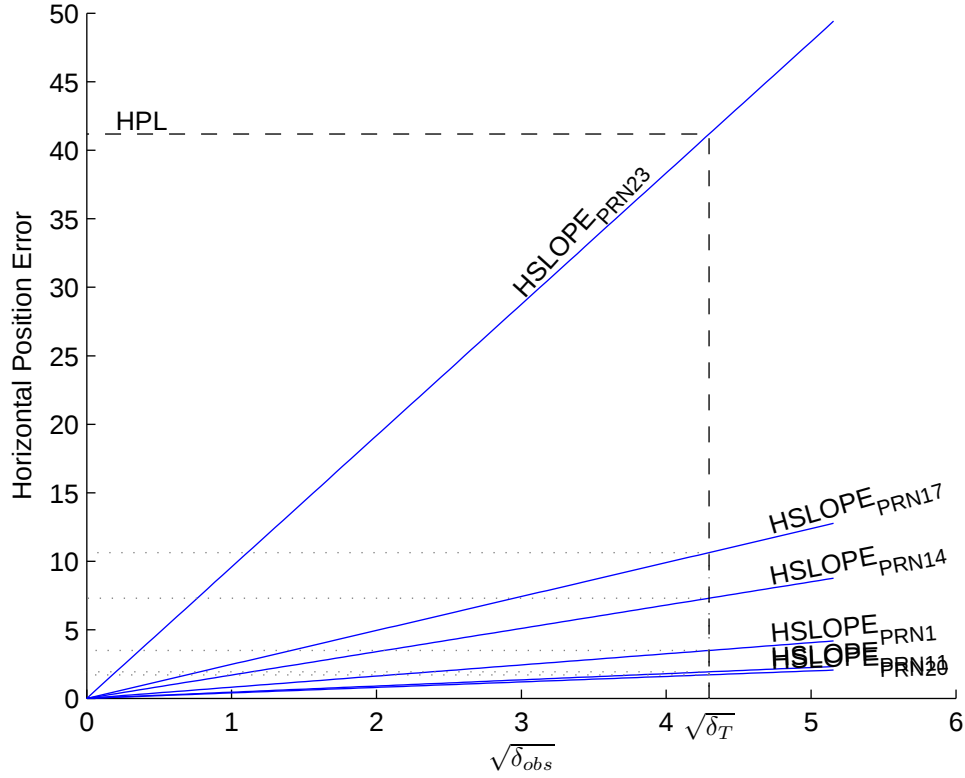


Figure 2.17: Relation linéaire entre le paramètre de non centralité $\sqrt{\delta}$ et l'erreur de position horizontale, en supposant que le défaut ne touche qu'un seul satellite. L'intégrité externe est déterminée à partir du défaut de pente HSLOPE maximale (le préfixe « H » signifie que l'on s'intéresse à l'erreur dans le plan tangent à la terre, plan qualifié d'horizontal).

On peut aussi écrire la formule comme suit

$$\|x - \hat{x}\| < \max_i \sqrt{\frac{\sum_{j=1}^4 (H_{j,i}^+)^2 \sigma_i^2}{S_{i,i}}} \sqrt{\delta_{obs}}$$

$$\|x - \hat{x}\| < \max_i SLOPE_i \sqrt{\delta_{obs}}$$

ce qui permet facilement de calculer l'intégrité externe comme l'illustre la figure 2.17.

Niveau d'incertitude Le calcul précédent permet de caractériser l'erreur maximum en position à partir de la géométrie, du bruit de mesure et de la somme pondérée des carrés des résidus du calcul de position. Cette erreur maximum, qui dépend des données réelles, est appelée *niveau d'incertitude*, ou Uncertainty Level (UL).

Le niveau d'incertitude quantifie l'influence maximale sur la solution du MDB_{obs} déterminée à partir des résidus. Ainsi et avec les hypothèses formulées, le risque que l'erreur de position dépasse le niveau d'incertitude est égal à la probabilité de non détection p_{MD} utilisée pour le calcul de δ_{obs} .

En général, on souhaite considérer l'impact du défaut uniquement sur certaines composantes de la position comme dans le plan horizontal ou encore l'altitude pour les avions par exemple. On peut ainsi calculer un *Horizontal Uncertainty Level* (HUL) comme suit, si les deux premières composantes de x correspondent au plan horizontal du repère local tangent à l'ellipsoïde de référence¹⁸ :

$$\begin{aligned} HUL &= \max_i \sqrt{(H_{1,i}^+)^2 + (H_{2,i}^+)^2} \cdot MDB_{obs,i} \\ HUL &= \max_i HSLOPE_i \sqrt{\delta_{obs}} \end{aligned}$$

$$\text{où } HSLOPE_i = \sqrt{\frac{((H_{1,i}^+)^2 + (H_{2,i}^+)^2) \sigma_i^2}{S_{i,i}}}.$$

Niveau de protection Il est parfois nécessaire de prédire l'erreur maximum en position pour un instant et une position donnés, sans connaître les mesures actuelles. C'est notamment le cas si l'on souhaite planifier une trajectoire de façon à optimiser la qualité de la localisation. Dans ce cas, on parle de niveau de protection (*PL*). Le niveau de protection est une borne supérieure de l'erreur de position qui ne doit pas être dépassée sans qu'une faute ne soit détectée. Il quantifie ainsi la capacité du RAIM à protéger l'utilisateur d'une erreur de position.

Le calcul d'un niveau de protection est identique au calcul d'un niveau d'incertitude, mais la valeur de NSSE est remplacée par le seuil T du test global. La valeur du niveau de protection ne dépend pas des mesures, et se base sur le paramètre de non-centralité δ de la distribution du χ^2 utilisé pour le test global, ou sur le MDB associé. On définit les niveaux de protection sur certaines composantes de la position, les plus courants étant le Horizontal Protection Level (HPL) et le Vertical Protection Level (VPL) définis comme suit :

$$\begin{aligned} HPL &= \max_i HSLOPE_i \sqrt{\delta} = \max_i \sqrt{(H_{1,i}^+)^2 + (H_{2,i}^+)^2} \cdot MDB_i \\ VPL &= \max_i VSLOPE_i \sqrt{\delta} = \max_i |H_{3,i}^+| \cdot MDB_i, \end{aligned}$$

$$\text{où } VSLOPE_i = \frac{|H_{3,i}^+| \sigma_i}{\sqrt{S_{i,i}}}.$$

Les niveaux de protection sont particulièrement adaptés à la quantification de l'intégrité externe dans les systèmes embarqués. En effet, contrairement au niveau d'in-

18. Convention que nous conserverons tout au long du manuscrit

certitude, le paramètre δ ne dépend pas des mesures et peut être précalculé pour un couple p_{FA}, p_{MD} donné. Ainsi, le calcul du niveau de protection peut être calculé en tout point du globe à un instant donné, en se basant sur les éphémérides GPS et sous hypothèse qu'aucun obstacle n'entrave la réception des signaux.

2.5.2.3 Lien entre la détection de défaut et l'intégrité externe

Les RNP spécifient la continuité et l'intégrité requises, ainsi que les limites d'alerte. Une alerte doit être déclenchée quand l'erreur de position HPE est supérieure à la limite d'alerte, soit $HPE > HAL$ quand la phase de vol est connue. Quand la phase de vol est inconnue du récepteur, on ne connaît pas le HAL , aussi une alerte doit être déclenchée si l'erreur de position dépasse le niveau de protection, soit $HPE > HPL$.

L'erreur de position est inconnue, mais le HUL constitue une borne de cette erreur estimée à partir du $NSSE$. De même, le niveau de protection HPL borne l'erreur maximale en position occasionnée par un biais non détectable. Le test de détection de défaut basé sur la somme des carrés des résidus est équivalent à comparer le HUL avec le HPL :

$$\begin{aligned} NSSE < T &\Leftrightarrow \delta_{obs} < \delta \\ &\Leftrightarrow HUL < HPL. \end{aligned}$$

Quand la limite d'alerte HAL est connue, l'erreur de position doit être maintenue sous cette limite. Aussi, la première étape d'un algorithme de RAIM consiste à vérifier qu'il peut protéger l'utilisateur d'une erreur supérieure à HAL .

Quand le test d'intégrité interne est vérifié, nous venons de voir que $HUL < HPL$. La méthode consiste à comparer HPL avec HAL :

- Si $HPL < HAL$ alors $HUL < HAL$ et donc le contrôle d'intégrité respecte les RNP.
- Si $HPL > HAL$ alors on ne peut plus garantir que $HUL < HAL$, car il est possible d'avoir $HAL < HUL < HPL$. Dans ce cas, un niveau d'incertitude HUL supérieur à la limite d'alerte HAL peut ne pas déclencher d'alarme. En d'autres termes, l'impact en position de l'erreur minimale détectable est supérieur à l'erreur tolérable maximale en position (HAL).

Le RAIM est donc déclaré indisponible quand $HPL > HAL$.

La figure 2.18 présente plusieurs cas de fonctionnement d'un algorithme de RAIM en fonction des relations entre HUL , HPL et HAL :

- $HPL > HAL$: Comme dit précédemment, le RAIM est indisponible.
- $HUL < HPL < HAL$: On est dans le cas de fonctionnement normal, avec $NSSE < T$. La position est déclarée intègre.
- $HPL < HUL < HAL$: La présence d'un défaut est détectée, car $NSSE > T$. Cependant, l'impact du défaut sur l'erreur de position n'est pas critique pour

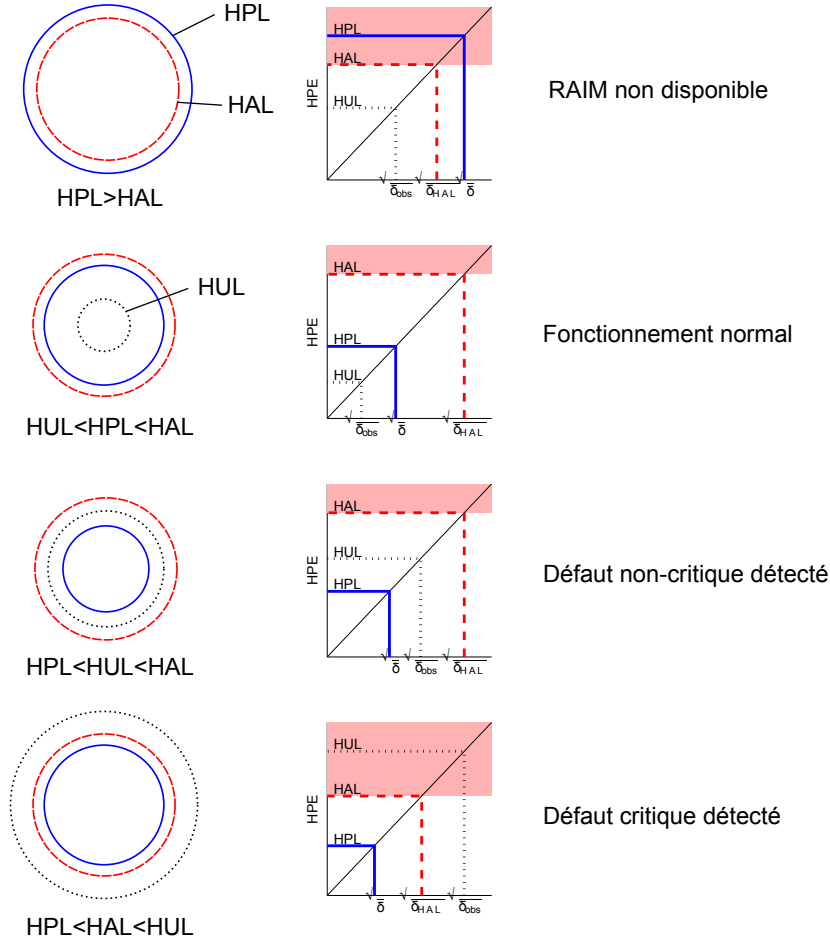


Figure 2.18: Relations entre HUL, HAL et HPL et résultat du RAIM

la phase de navigation courante. Un autre algorithme de RAIM réglé différemment, tout en restant conforme aux RNP, aurait pu ne pas détecter ce défaut (cf. section 2.5.2.4).

- $HPL < HAL < HUL$: On a $NSSE > T$, donc la présence d'un défaut est détectée. L'impact de ce défaut sur la position est critique pour la phase de navigation en cours ($HUL > HAL$), aussi, tout algorithme de RAIM respectant les RNP doit signaler la perte d'intégrité dans ce cas.

2.5.2.4 Choix du seuil de détection

La quantification de l'intégrité externe dépend du seuil T choisi pour le test global sur les résidus, ainsi que des probabilités de fausse alarme p_{FA} et de non-détection p_{MD} du test. Le seuil T doit être choisi de façon à respecter les RNP pour le mode de navigation courant. Nous présentons ici deux méthodes permettant de choisir T .

Méthode classique La méthode classique consiste à tout d'abord déterminer les probabilités de fausse alerte et de non détection du test en fonction des critères de continuité et d'intégrité des RNP. Ainsi, pour les modes de navigation « en route » et « terminal », d'après le standard [RTCA SC-159, 2006] la plus grande valeur admissible est¹⁹ $p_{MD} = 0.001$ et la probabilité de fausse alarme est définie au maximum à $p_{FA} = 3.33 \cdot 10^{-7}$ par échantillon. La méthode classique consiste à utiliser les valeurs limites spécifiées de p_{FA} et p_{MD} . On calcule alors le seuil T du test, défini comme le plus petit seuil vérifiant p_{FA} , comme sur la figure 2.15.

Pour statuer de la disponibilité du RAIM en fonction du HAL courant, on calcule le HPL à partir de T (ou du paramètre de non-centralité associé δ) et de la p_{MD} .

L'intérêt de cette méthode est que le seuil du test ne dépend que du nombre de satellites et de la probabilité de fausse alarme. Il peut donc être précalculé pour les valeurs admissibles du nombre de degrés de liberté du problème. On détermine en ligne le HPL en fonction de la géométrie courante et on décide de la disponibilité du RAIM en fonction du HAL défini pour la phase de mission courante.

Optimisation du seuil de détection en fonction du HAL courant Un inconvénient de la méthode classique est que le seuil est toujours défini le plus bas possible, même quand le HAL courant autorise la définition d'un seuil plus élevé. On privilégie ainsi arbitrairement la détection de faute (risque d'intégrité minimal) à la continuité de service. Pour le HAL courant, la probabilité de fausse alarme est fixée $p_{FA} = p_{FA}(RNP)$, tandis que la probabilité de non-détection est inférieure à celle spécifiée $p_{MD} < p_{MD}(RNP)$.

En règle générale, le choix d'un seuil T permettant de satisfaire les RNP n'est pas unique, mais tel que $T_{\min} \leq T \leq T_{\max}$ (Figure 2.19). $T_{\min}$ est le plus petit seuil vérifiant les probabilités de fausse alarme et de non détection. C'est le seuil de la méthode classique. $T_{\max}$ est le plus grand seuil permettant de satisfaire la probabilité de non-détection pour la limite d'alerte courante. $T_{\max}$ est lié au paramètre de non-centralité $\delta_{\max}$ calculé à partir du HAL courant et de la géométrie, tel que $\delta_{\max} = \left(\frac{HAL}{\max_i HSLOPE_i} \right)^2$.

Ainsi, toute valeur de T telle que $T_{\min} \leq T \leq T_{\max}$ permet de satisfaire les RNP. Dans le cas où $T_{\min} > T_{\max}$, le RAIM est déclaré indisponible. Une stratégie proposée dans [Feng et al., 2006] consiste à choisir, dans l'intervalle $[T_{\min}, T_{\max}]$ des valeurs permises par les RNP, le seuil de détection T qui minimise à la fois le risque d'intégrité et le risque de continuité pour le HAL courant. L'intérêt du choix d'un seuil supérieur au seuil minimal est de limiter les échecs et erreurs de la fonction d'exclusion de défaut, les défauts plus grands étant généralement plus faciles à identifier. Cette méthode est cependant plus difficile à embarquer dans un récepteur GPS, le seuil devant être recalculé à chaque époque en fonction du HAL et de la géométrie, ce qui nécessite l'inversion d'une loi du χ^2 non-centrée.

19. Afin d'assurer un risque de perte d'intégrité de 10^{-7} /h, sachant que la probabilité d'occurrence d'un défaut en GPS est spécifiée à 10^{-4} /h.

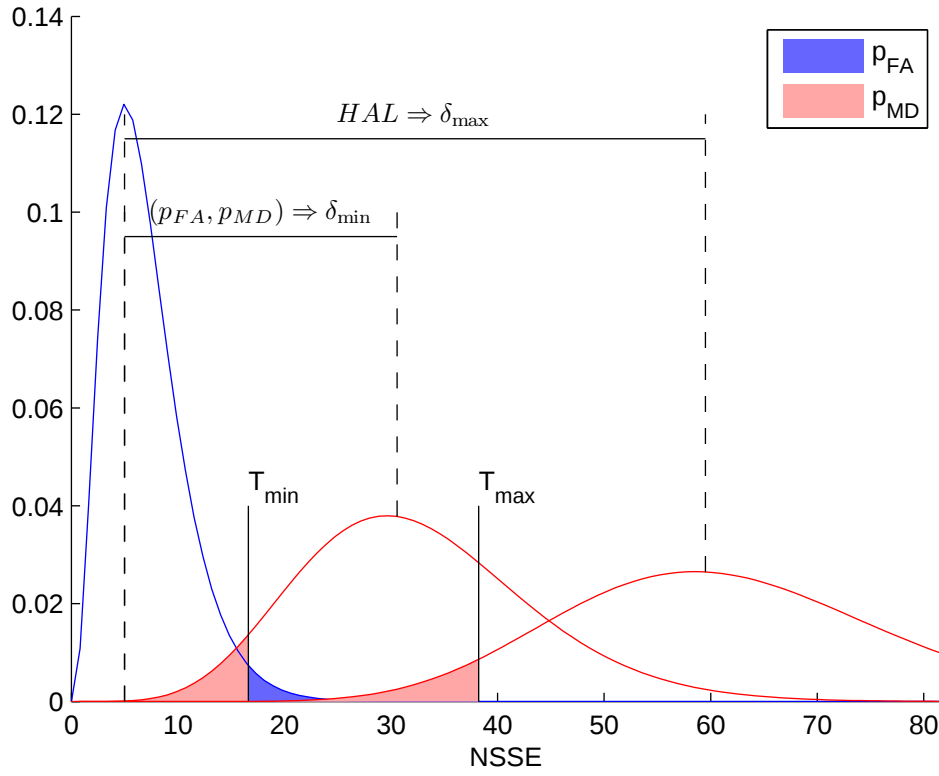


Figure 2.19: Seuil minimal T_{min} et seuil maximal T_{max} du test du χ^2 vérifiant les RNP (p_{FA} , p_{MD} et HAL)

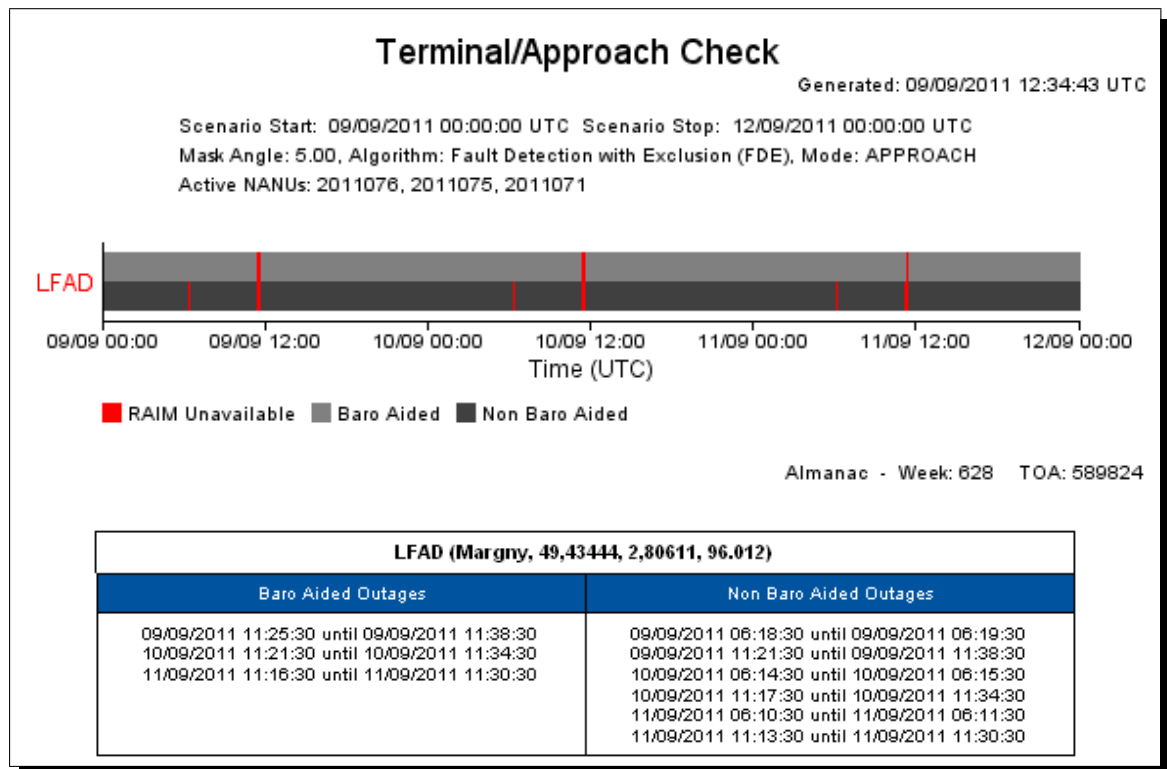


Figure 2.20: Prédiction de la disponibilité du RAIM par AUGUR pour une approche à l'aérodrome de Compiègne-Margny.

2.5.3 Planification de mission en fonction de l'intégrité externe

Nous avons vu que les niveaux de protection HPL et VPL ne dépendent pas des mesures courantes, ce qui les rend prédictibles à un lieu et un instant donnés si l'on connaît les positions des satellites. Comme le niveau de protection peut être prédit, il est possible de vérifier la disponibilité du positionnement intègre à l'avance, afin de planifier une mission exploitant la navigation GPS.

Par exemple, l'outil AUGUR²⁰ fourni par Eurocontrol permet de prédire la disponibilité du RAIM pour la phase d'approche aux différents aéroports, ou le long de la route parcourue entre différents points de contrôle. La figure 2.20 montre un résultat de prédiction sur trois jours pour une approche à Margny-les-Compiègne. On remarque par exemple qu'il est préférable de planifier son arrivée à Compiègne en dehors de la plage 11h25–11h39 le 9 septembre pour bénéficier d'une approche guidée par GPS.

De la même façon, la FAA²¹ fournit l'outil de prédiction de disponibilité du RAIM

20. <http://augur.ecacnav.com/augur/>

21. Federal Aviation Administration : agence gouvernementale chargée de la réglementation et des contrôles pour l'aviation civile aux États-Unis

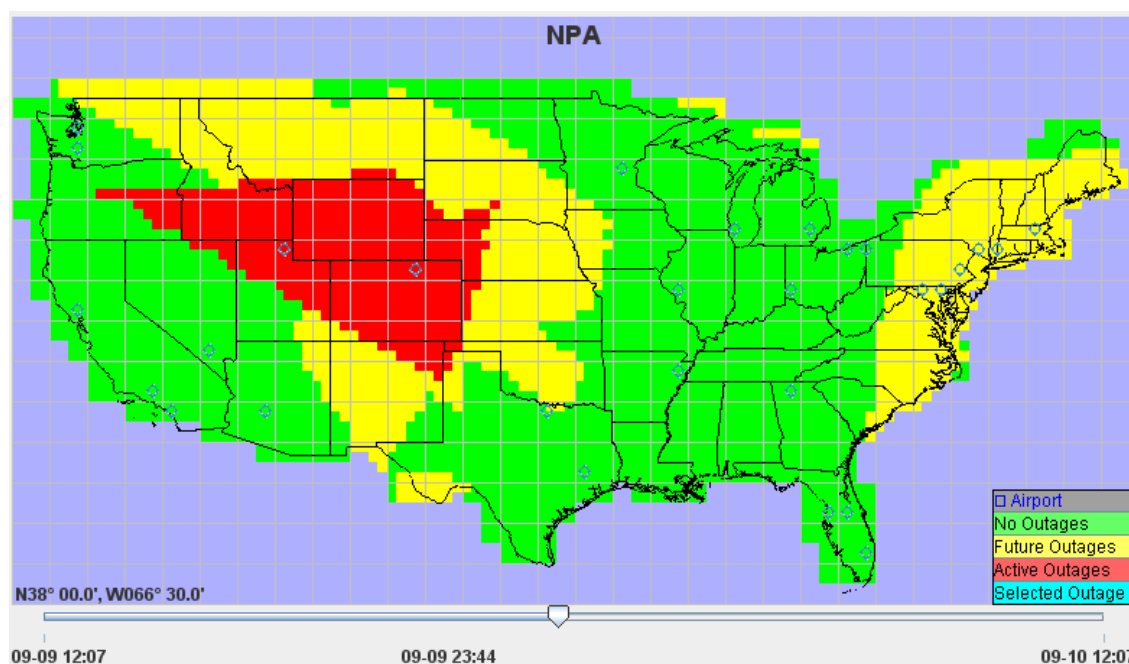


Figure 2.21: Prédiction de la disponibilité du RAIM par AC90-100A pour la phase d’approche sur le territoire des USA. Les zones où le RAIM n’est pas disponible pour la date choisie sont en rouge. Les zones en jaune correspondent aux zones d’indisponibilité dans le futur.

« AC90-100A »²² qui couvre le territoire des États Unis d’Amérique pour les différentes phases de vol. Les résultats sont fournis sous forme de carte interactive (Figure 2.21), ou à partir d’un plan de vol.

2.6 Conclusion

Un système de positionnement absolu est requis dans de nombreuses applications de robotique mobile, en particulier pour la navigation sur carte de véhicules routiers. De plus, face à la mondialisation des échanges, un système de positionnement global permettant la fourniture et la transmission d’une position en tout point de la planète est nécessaire. Les systèmes de positionnement par satellites sont une réponse à ce besoin, de par leur couverture globale et le faible coût des récepteurs. Cependant, la précision de ces systèmes est variable en fonction de la géométrie des satellites visibles (donc de l’heure et de la position de l’utilisateur). La dilution de précision permet de quantifier simplement cette imprécision en fonction de la variance de l’erreur de mesure.

Les GNSS peuvent être sujets aux fautes, qu’elles soient dues à une défaillance d’un satellite, au brouillage ou au trajet indirect du signal dans l’environnement. Il est

22. <http://www.raimprediction.net/ac90-100/>

nécessaire de disposer d'un mécanisme d'intégrité afin de protéger l'utilisateur de l'impact d'une telle faute dans les applications mettant en jeu la sécurité des personnes ou des biens. Cette information d'intégrité peut soit être déterminée par le récepteur lui-même (RAIM), ou alors provenir d'un réseau de stations de surveillance au sol et être transmise par satellites : c'est le cas des systèmes d'augmentation tels que WAAS et EGNOS. Dans ce dernier cas, la protection ne couvre alors pas les effets des multitrajets, qui sont des effets de l'environnement proche du récepteur sur la propagation du signal. Le RAIM consiste en un contrôle autonome d'intégrité par le récepteur, qui profite de la redondance de mesures pour vérifier par test d'hypothèse l'intégrité de la solution calculée et tenter d'identifier, le cas échéant, la ou les mesures aberrantes. Ces mécanismes d'intégrité sont associés à des méthodes de quantification de l'intégrité externe (niveaux d'incertitude et niveaux de protection), qui décrivent l'impact maximal d'une erreur non-détectée sur la solution. On peut ainsi déterminer pour l'instant courant, mais également prédire sur toute une mission, si le positionnement par satellite sera disponible en fonction de l'erreur maximale tolérable en position pour la mission (limites d'alerte).

Nous avons, dans ce chapitre, effectué un calcul ponctuel de position associé à un niveau d'incertitude et un niveau de protection, dans un cadre probabiliste où l'erreur de mesure est de variance connue et où le modèle d'observation se linéarise correctement autour de la solution. On détermine ainsi une zone de protection enveloppant la position calculée, sous réserve d'une seule faute affectant les mesures courantes. Dans le chapitre suivant, nous allons étudier l'estimation non-linéaire ensembliste robuste, dans un contexte d'erreurs bornées. Un exemple de localisation par balises, où les non-linéarités sont plus importantes qu'en GPS, y sera présenté afin d'illustrer le calcul d'un domaine de localisation en présence de mesures aberrantes.

Chapitre 3

Estimation non-linéaire ensembliste robuste

3.1 Introduction

Dans ce chapitre, on cherche à estimer un vecteur de paramètres à partir d'un ensemble de mesures et de connaissances *a priori*. On s'intéresse ici à des problèmes dits « statiques » c'est-à-dire des problèmes pour lesquels le temps n'intervient pas, comme par exemple l'estimation de la position d'un robot à partir de mesures synchrones acquises au même instant d'échantillonnage. On se place dans le cas général de problèmes mettant en œuvre des équations non linéaires, avec des observations éventuellement redondantes et aberrantes. Ce dernier qualificatif signifie que le modèle d'observation ne correspond pas à la mesure courante soit parce que le bruit de mesure est trop important, soit parce que la modélisation du phénomène de la mesure est en désaccord avec la réalité.

Pour illustrer ce chapitre, nous allons considérer un problème de localisation planaire dans lequel il s'agit d'estimer la position cartésienne d'un objet à partir de mesures de distances faites sur des balises de positions connues. Si les balises sont discernables alors l'association des mesures avec les balises est connue et le problème de localisation se ramène à un problème d'estimation. Considérons un système de radio-communication qui fournit des mesures de distance par une technique de double temps de vol comme le système utilisé par [Röhrig and Müller, 2009] pour localiser des robots mobiles de manutention dans des entrepôts industriels.

Sous hypothèse que la balise et l'objet sont en vue directe, le modèle d'observation s'écrit

$$\sqrt{(x - x_{Bi})^2 + (y - y_{Bi})^2} = d_{Bi}, i = 1 \dots n_B,$$

où (x, y) sont les coordonnées de la position de l'objet, (x_{Bi}, y_{Bi}) les coordonnées de la balise i et d_{Bi} la distance mesurée entre l'objet et la balise i . Ce problème est non linéaire mais la non-linéarité est d'autant plus importante que l'objet à localiser est proche des balises.

Des observations aberrantes peuvent d'abord se produire si l'erreur de mesure devient très grande comme par exemple à cause d'interférences. Mais, les hypothèses qui ont été formulées dans l'élaboration du modèle d'observation peuvent elles aussi conduire à des aberrations. Dans des environnements contraints par exemple, les signaux radios peuvent se réfléchir sur des murs et l'objet peut être confronté à une situation extrême dans laquelle seule une onde réfléchie est reçue. La distance mesurée ne correspond donc plus à la distance à vol d'oiseau.

Ce problème de localisation nécessite au moins trois balises non-alignées pour être résolu sans ambiguïté. Souvent, le nombre de mesures disponibles dépasse le nombre minimal requis. On dispose alors d'une redondance d'information qui peut être utilisée afin d'améliorer la précision, ou éventuellement de détecter et exclure les mesures aberrantes. Notons que si l'on sait a priori dans quel demi-plan se trouve le mobile, on peut effectuer la localisation sans ambiguïté avec seulement deux balises. C'est par exemple le cas quand on souhaite localiser un robot à l'intérieur d'une pièce à l'aide de deux balises situées le long du même mur (Figure 3.1).

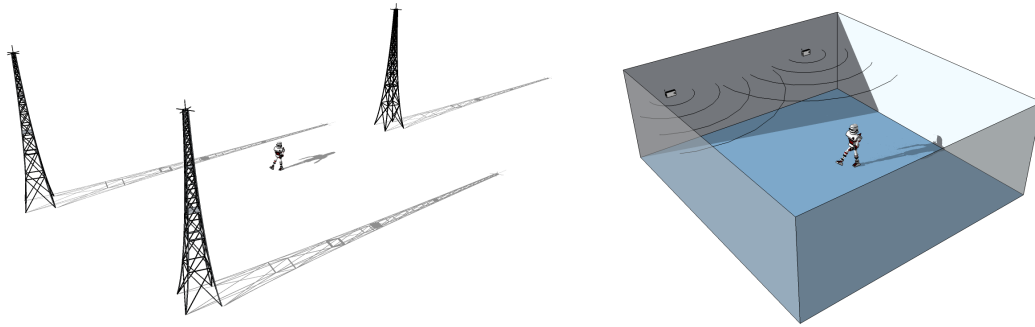


Figure 3.1: Localisation d'un robot en extérieur à l'aide de trois balises radio, et dans une pièce à l'aide de deux balises.

Ce chapitre est organisé comme suit. Après une brève présentation de l'approche ensembliste à travers un exemple simple de localisation, les outils d'analyse par intervalle sont présentés, avant d'être utilisés pour résoudre des problèmes d'inversion ensembliste. On introduit alors les méthodes d'inversion ensembliste robuste, permettant de tenir compte de la présence de mesures aberrantes. Pour finir, la question de la détection et de l'identification de ces mesures aberrantes est abordée.

3.2 Approche ensembliste

On s'intéresse particulièrement aux méthodes ensemblistes qui reposent sur l'hypothèse d'erreur bornée. L'approche ensembliste consiste alors à déterminer l'ensemble des paramètres compatibles avec les observations et leur intervalle d'erreur associé.

Les bornes des observations sont supposées connues. Le problème ne consiste donc pas à trouver une valeur du vecteur de paramètres minimisant un critère d'erreur, mais un ensemble de vecteurs de paramètres compatibles avec les observations.

Considérons que l'erreur de mesure de distance est bornée; on peut écrire pour chaque distance mesurée $d_{Bi} \in d_{Bi}^{meas} + [e_{Bi}]$. Ainsi, chaque mesure de distance contraint la position du mobile dans une couronne dont le centre est la balise concernée. L'objet se trouve ainsi dans l'intersection des couronnes correspondant à chaque mesure de distance (Figure 3.2).

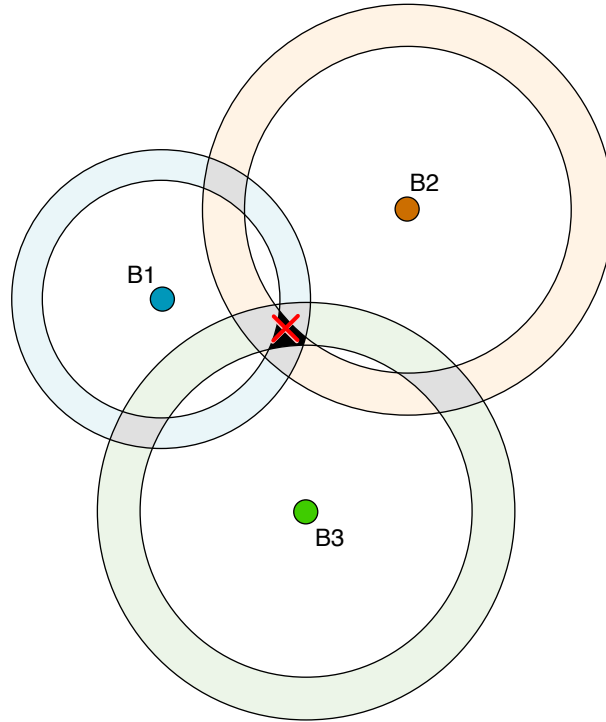


Figure 3.2: Localisation ensembliste avec 3 balises sous hypothèse d'erreurs bornées. La croix rouge matérialise la position inconnue. La zone noire représente la zone d'incertitude dans laquelle se trouve la solution.

L'ensemble solution étant généralement quelconque, diverses méthodes ensemblistes ont été développées, chacune reposant sur une représentation plus ou moins générique des observations et des solutions. On trouve ainsi dans la littérature des méthodes basées sur des boîtes, des ellipses, des polytopes ou encore des zonotopes. La figure 3.3 montre diverses représentations de la solution du problème de localisation à trois balises.

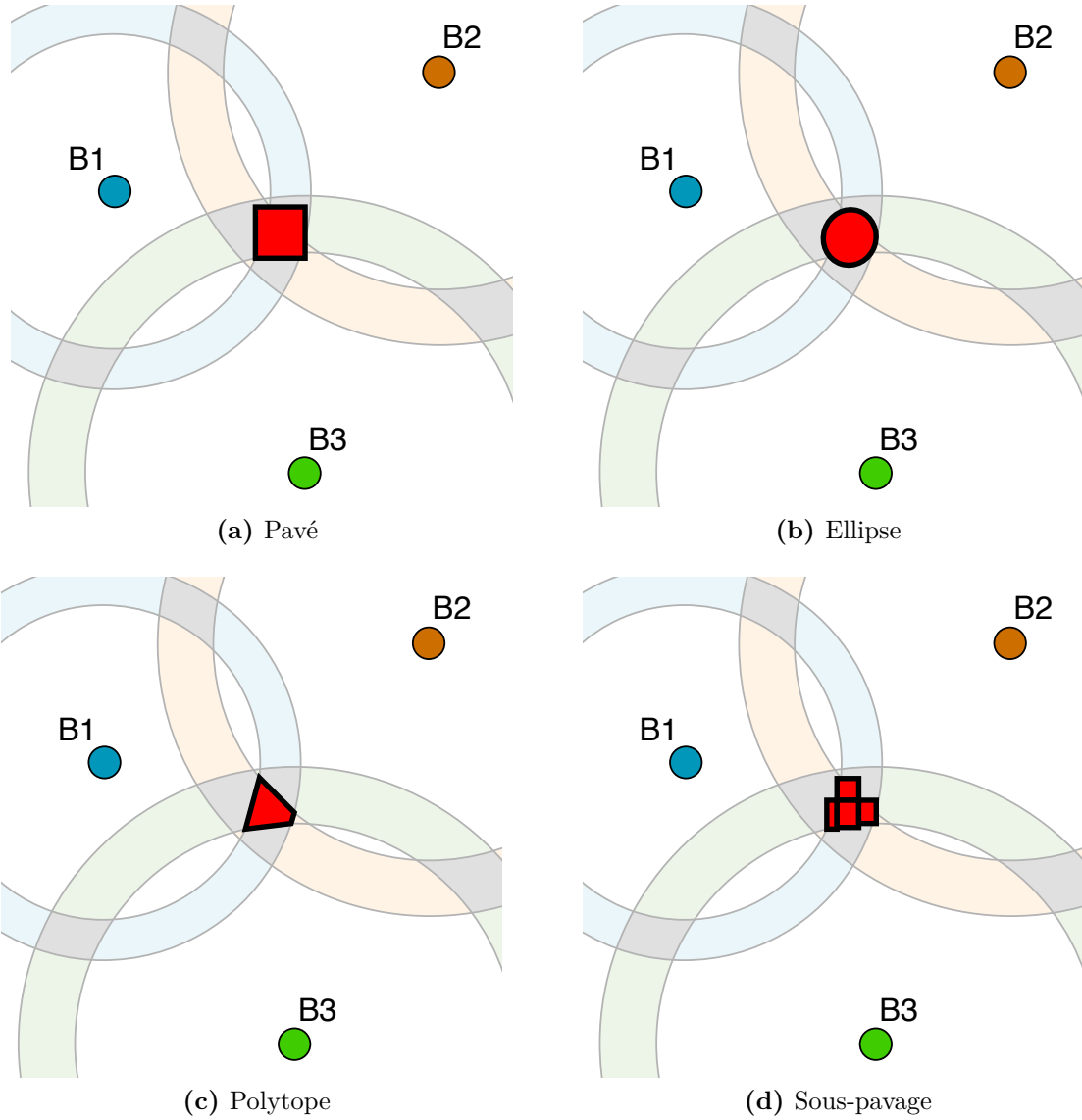


Figure 3.3: Différentes représentations ensemblistes de la solution du problème de localisation à trois balises.

3.3 Analyse par intervalles

L'analyse par intervalles consiste à manipuler des intervalles en lieu et place des nombres réels. Elle a été développée initialement pour quantifier l'erreur sur les résultats de calculs numériques [Moore, 1966]. En effet, les nombres sont usuellement représentés avec une précision finie par des flottants dans les calculateurs. Il existe de nombreux exemples pour lesquels des erreurs de représentation peuvent se propager de façon dramatique dans les traitements numériques. Dans ce travail de thèse, l'analyse par intervalles sert principalement à manipuler les incertitudes.

3.3.1 Intervalles

Un *intervalle* est un sous-ensemble connexe fermé de $\mathbb{R}$. On note $\mathbb{IR}$ l'ensemble des intervalles réels. Les grandeurs intervalles seront notées entre crochets afin de faciliter la lecture.

L'intervalle $[x]$ est défini comme l'ensemble des réels x compris entre sa *borne inférieure* $\underline{x}$ et sa *borne supérieure* $\bar{x}$:

$$[x] = [\underline{x}, \bar{x}] = \{x \in \mathbb{R}, \underline{x} \leq x \leq \bar{x}\}.$$

Le *centre* d'un intervalle non vide et borné est donné par

$$\text{mid}([x]) = \frac{\underline{x} + \bar{x}}{2}.$$

La *longueur* d'un intervalle non vide est définie par

$$w([x]) = \bar{x} - \underline{x}.$$

On note $\{x\} = [x, x]$ un intervalle ponctuel.

3.3.2 Calculs sur les intervalles

Dans le cas d'intervalles fermés, l'essentiel des opérations peut être fait en manipulant les bornes des intervalles.

L'union intervalle de deux intervalles fermés non vides $[x]$ et $[y]$ satisfait

$$\forall [x] \in \mathbb{IR}, \forall [y] \in \mathbb{IR}, [x] \sqcup [y] = [\min\{\underline{x}, \underline{y}\}, \max\{\bar{x}, \bar{y}\}]. \quad (3.1)$$

L'intersection de deux intervalles fermés non vides $[x]$ et $[y]$ satisfait

$$[x] \cap [y] = \begin{cases} [\max\{\underline{x}, \underline{y}\}, \min\{\bar{x}, \bar{y}\}] & \text{si } \max\{\underline{x}, \underline{y}\} \leq \min\{\bar{x}, \bar{y}\} \\ \emptyset & \text{sinon.} \end{cases} \quad (3.2)$$

Soit α un réel et $[x]$ un intervalle non vide, alors l'intervalle

$$\alpha[x] \triangleq \{\alpha x \mid x \in [x]\} \quad (3.3)$$

est donné par

$$\alpha[x] = [\alpha \underline{x}, \alpha \bar{x}] \text{ si } \alpha \geq 0 \quad (3.4)$$

$$= [\alpha \bar{x}, \alpha \underline{x}] \text{ si } \alpha < 0. \quad (3.5)$$

Pour deux intervalles $[x]$ et $[y]$ et un opérateur $\diamond \in \{+, -, *, /\}$, on définit $[x] \diamond [y]$ comme le plus petit intervalle contenant toutes les valeurs prises par $x \diamond y$ quand $x \in [x]$ et $y \in [y]$ [Jaulin et al., 2001], soit

$$[x] \diamond [y] = [\{x \diamond y \in \mathbb{R} \mid x \in [x], y \in [y]\}].$$

Dans le cas d'intervalles fermés, les calculs s'effectuent de la façon suivante :

$$[x] + [y] = [\underline{x} + \underline{y}, \bar{x} + \bar{y}], \quad (3.6)$$

$$[x] - [y] = [\underline{x} - \bar{y}, \bar{x} - \underline{y}], \quad (3.7)$$

$$[x] * [y] = [\min\{\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}\}, \max\{\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}\}]. \quad (3.8)$$

On notera indifféremment le produit de deux intervalles $[x] * [y]$ ou $[x][y]$.

Le calcul de l'inverse est donné par

$$1/[y] = \begin{cases} \emptyset & \text{si } [y] = [0, 0], \\ [1/\bar{y}, 1/\underline{y}] & \text{si } 0 \notin [y], \\ [1/\bar{y}, \infty[& \text{si } \underline{y} = 0 \text{ et } \bar{y} > 0, \\] - \infty, 1/\bar{y}] & \text{si } \underline{y} < 0 \text{ et } \bar{y} = 0, \\] - \infty, \infty[& \text{si } \underline{y} < 0 \text{ et } \bar{y} > 0, \end{cases} \quad (3.9)$$

et la division par

$$[x]/[y] = [x] * (1/[y]). \quad (3.10)$$

Ces règles se simplifient dans le cas d'intervalles ponctuels, et on retrouve alors les règles de l'arithmétique réelle. On peut ainsi considérer l'arithmétique intervalle comme une extension de l'arithmétique réelle.

L'image $f([x])$ d'un intervalle par une fonction f est

$$f([x]) = \{f(x) \mid x \in [x]\}.$$

Cette image n'est pas nécessairement un intervalle. En effet, si f n'est pas continue, $f([x])$ est une union d'intervalles.

L'extension intervalle d'une fonction est définie comme la fonction renvoyant l'enveloppe intervalle de $f([x])$:

$$[f]([x]) = [\{f(x) \mid x \in [x]\}].$$

L'extension intervalle des fonctions élémentaires peut être exprimée directement en fonction des bornes. Par exemple, pour un intervalle non vide $[x]$, la fonction exponentielle s'exprime

$$[\exp]([x]) = [\exp \underline{x}, \exp \bar{x}]. \quad (3.11)$$

Dans le cas de fonctions non monotones, la situation se complique. Ainsi, $[\cos]([-\pi, \pi]) = [-1, 1]$ diffère de l'intervalle $[\cos(-\pi), \cos(\pi)] = [-1, -1]$. Des algorithmes spécifiques doivent donc être construits pour l'évaluation intervalle de ces fonctions (voir par exemple [Bouron, 2002]).

3.3.3 Généralisation des intervalles

On peut généraliser la notion d'intervalles à tout ensemble admettant une structure de treillis. Ainsi, outre les intervalles de réels, on peut considérer des intervalles de vecteurs, de booléens, de fonctions ou encore d'ensembles [Jaulin, 2011a].

3.3.3.1 Intervalles vectoriels

Un intervalle vectoriel, aussi appelé pavé ou encore boîte $[\mathbf{x}]$ est un sous-ensemble de $\mathbb{R}^n$ qui peut être défini comme le produit cartésien de n intervalles réels

$$[\mathbf{x}] = [\underline{x}_1, \bar{x}_1] \times \cdots \times [\underline{x}_n, \bar{x}_n] = [x_1] \times \cdots \times [x_n].$$

C'est un vecteur dont les composantes sont des intervalles, noté

$$[\mathbf{x}] = ([x_1], \dots, [x_n])^T,$$

et la $i^{\text{ème}}$ composante intervalle $[x_i]$ est la projection de $[\mathbf{x}]$ sur le $i^{\text{ème}}$ axe. Ainsi, l'ensemble vide de $\mathbb{R}^n$ doit être noté $\emptyset \times \cdots \times \emptyset$ car toutes ses composantes intervalles sont vides. Une expression telle que $[\mathbf{x}] = [2, 3] \times \emptyset$ n'a pas de sens, car $[2, 3]$ n'est pas la projection de $[\mathbf{x}]$ sur le premier axe.

On note $\mathbb{IR}^n$ l'ensemble des pavés de $\mathbb{R}^n$. Un pavé non vide peut être représenté par un parallélotope aligné sur les axes (Figure 3.4).

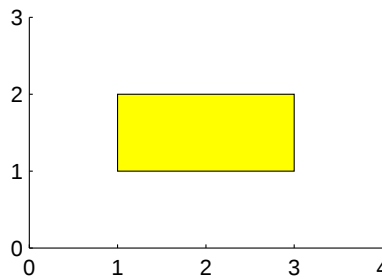


Figure 3.4: Représentation du pavé $[1, 3] \times [1, 2]$

La plupart des notions introduites pour les intervalles s'étendent facilement aux

pavés. Ainsi, la borne inférieure et la borne supérieure d'un pavé sont définies par

$$\underline{\mathbf{x}} = (\underline{x}_1, \dots, \underline{x}_n)^T,$$

$$\overline{\mathbf{x}} = (\overline{x}_1, \dots, \overline{x}_n)^T.$$

La longueur d'un pavé est définie comme la longueur de son côté le plus long :

$$w([\mathbf{x}]) = \max_{1 \leq i \leq n} w([x_i]).$$

Les opérations élémentaires vues sur les intervalles peuvent se généraliser aux intervalles vectoriels, en effectuant les calculs intervalles sur chacune des composantes.

3.3.3.2 Matrices intervalles

De la même façon que la notion d'intervalle peut se généraliser aux vecteurs, on peut définir des matrices intervalles, où chaque composante est un intervalle. Les calculs classiques d'algèbre linéaire peuvent ainsi être « intervalisés », ce qui permet par exemple de tenir compte de l'erreur de linéarisation en définissant une matrice incertaine. L'utilisation de matrices intervalles permet également de tenir compte de la précision de la représentation numérique en machine et des calculs numériques, pour résoudre des problèmes mal conditionnés.

3.3.4 Fonctions d'inclusion

L'image d'un pavé par une fonction n'est généralement pas un pavé, et il n'existe en général pas de représentation exacte facile à manipuler de l'ensemble image d'un pavé par une fonction. Nous allons donc utiliser des fonctions intervalles, nommées fonctions d'inclusion, qui renvoient un pavé contenant l'image d'une boîte par une fonction donnée (Figure 3.5).

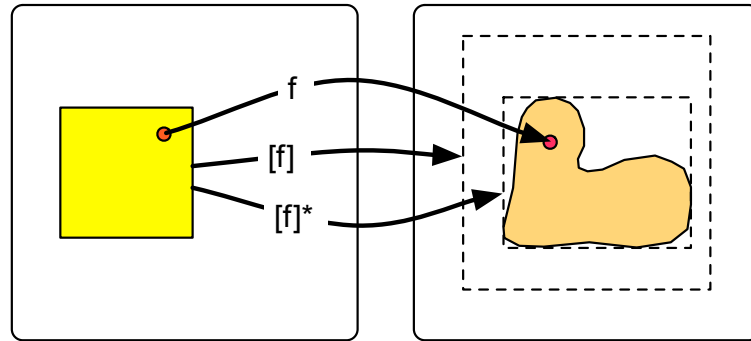


Figure 3.5: Images d'une boîte par la fonction $\mathbf{f}$, par une fonction d'inclusion $[\mathbf{f}]$ et par la fonction d'inclusion minimale $[\mathbf{f}]^*$.

La fonction intervalle $[\mathbf{f}] : \mathbb{R}^n \rightarrow \mathbb{IR}^m$ est une *fonction d'inclusion* de la fonction $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ si et seulement si

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]).$$

Ainsi, $[\mathbf{x}] \mapsto \mathbb{R}^m$ est une fonction d'inclusion pour toutes les fonctions de $\mathbb{R}^n$ dans $\mathbb{R}^m$. Quelque soit la forme de l'ensemble image $\mathbf{f}([\mathbf{x}])$, une fonction d'inclusion $[\mathbf{f}]$ de $\mathbf{f}$ permet de calculer un pavé $[\mathbf{f}]([\mathbf{x}])$ garanti de contenir l'image de $[\mathbf{x}]$ par $\mathbf{f}$.

3.3.4.1 Propriétés spécifiques de fonctions d'inclusion

Une fonction d'inclusion est dite *fine* si l'image d'un intervalle vectoriel ponctuel $\{\mathbf{x}\}$ est ponctuelle, c'est-à-dire $[\mathbf{f}](\{\mathbf{x}\}) = \{\mathbf{f}(\mathbf{x})\}$.

Une fonction d'inclusion $[\mathbf{f}]$ est dite *convergente* si, pour toute suite de boîtes $[\mathbf{x}](k)$,

$$\lim_{k \rightarrow \infty} w([\mathbf{x}](k)) = 0 \Rightarrow \lim_{k \rightarrow \infty} w([\mathbf{f}]([\mathbf{x}](k))) = 0.$$

Cette propriété de convergence est en général demandée pour la convergence des algorithmes utilisant l'analyse par intervalles.

On dit de $[\mathbf{f}]$ qu'elle est *monotone pour l'inclusion* si

$$[\mathbf{x}] \subset [\mathbf{y}] \Rightarrow [\mathbf{f}]([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{y}]).$$

Une fonction d'inclusion $[\mathbf{f}]$ est dite *minimale* si, pour tout $[\mathbf{x}]$, $[\mathbf{f}]([\mathbf{x}])$ est la plus petite boîte contenant $\mathbf{f}([\mathbf{x}])$. La fonction d'inclusion minimale de $\mathbf{f}$ est unique et notée $[\mathbf{f}]^*$ (Figure 3.5). Au contraire, une fonction d'inclusion dont l'image d'un intervalle n'est pas égale au plus petit intervalle contenant l'évaluation intervalle de la fonction réelle sur cet intervalle est dite *pessimiste*.

Soient m fonctions d'inclusion $[f_j] : \mathbb{R}^n \rightarrow \mathbb{IR}$, $j = 1, \dots, m$ associées aux fonctions coordonnées f_j d'une fonction vectorielle $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$. Une fonction d'inclusion pour $\mathbf{f}$ est donnée par

$$[\mathbf{f}]([\mathbf{x}]) = [f_1]([\mathbf{x}]) \times \dots \times [f_m]([\mathbf{x}]).$$

$[\mathbf{f}]$ est convergente (respectivement minimale) si toutes ses fonctions coordonnées $[f_i]$ sont convergentes (respectivement minimales). La construction de fonctions d'inclusion efficaces d'une fonction vectorielle $\mathbf{f}$ peut ainsi être ramenée à la construction de fonctions d'inclusion efficaces pour ses fonctions coordonnées. Nous allons donc nous intéresser plus particulièrement à l'élaboration de fonctions d'inclusions pour des fonctions à valeurs réelles.

3.3.4.2 Fonctions d'inclusion naturelles

Soit f une *fonction terme*, c'est-à-dire une fonction qui peut être construite par un nombre fini de compositions des fonctions élémentaires telles que $\sin$, $\cos$, $\exp$, $\sqrt{\cdot}$, sqr , $\max$, $\dots$ et des opérateurs $+$, $-$, $*$, $/$. La méthode la plus simple pour obtenir une fonction d'inclusion d'une fonction terme f consiste à remplacer les variables scalaires $(x, y, \dots)$ dans l'expression de f par leur contrepartie intervalle $([x], [y], \dots)$, et toutes les fonctions élémentaires réelles par leur extension intervalle. La fonction $[f]_n$ ainsi obtenue est la *fonction d'inclusion naturelle* de f . Elle est monotone pour l'inclusion et fine. De plus, si l'expression de f ne contient que des opérateurs et fonctions élémentaires continus, alors la fonction d'inclusion naturelle est convergente.

La fonction d'inclusion naturelle n'est pas minimale en général, en raison de la dépendance entre variables et de l'effet d'enveloppe. Toutefois, une fonction d'inclusion naturelle est minimale dans le cas où chaque variable n'apparaît qu'une seule fois dans l'expression de f et que tous les opérateurs et fonction élémentaires utilisés sont continus.

Dans l'exemple de localisation à partir de trois balises, la fonction d'observation est

$$\mathbf{g} : \mathbb{R}^2 \rightarrow \mathbb{R}^3 \quad (3.12)$$

$$\begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} \sqrt{(x - x_{B1})^2 + (y - y_{B1})^2} \\ \sqrt{(x - x_{B2})^2 + (y - y_{B2})^2} \\ \sqrt{(x - x_{B3})^2 + (y - y_{B3})^2} \end{pmatrix}.$$

La fonction d'inclusion naturelle $[\mathbf{g}]$ de $\mathbf{g}$ est ainsi donnée par

$$[\mathbf{g}] : \mathbb{IR}^2 \rightarrow \mathbb{IR}^3$$

$$\begin{pmatrix} [x] \\ [y] \end{pmatrix} \mapsto \begin{pmatrix} \sqrt{([x] - x_{B1})^2 + ([y] - y_{B1})^2} \\ \sqrt{([x] - x_{B2})^2 + ([y] - y_{B2})^2} \\ \sqrt{([x] - x_{B3})^2 + ([y] - y_{B3})^2} \end{pmatrix}.$$

Comme il n'est fait usage que d'opérateurs et fonctions continues, et puisque chaque variable n'apparaît qu'une seule fois dans chaque expression, alors $[\mathbf{g}]$ est la fonction d'inclusion minimale de $\mathbf{g}$.

3.3.5 Contracteurs

3.3.5.1 Problèmes de satisfaction de contraintes

Soit n_x variables réelles $x_i \in \mathbb{R}, i \in \{1, \dots, n_x\}$ reliées par n_f relations (ou contraintes) de la forme

$$f_j(x_1, x_2, \dots, x_{n_x}) = 0, j \in \{1, \dots, n_f\}. \quad (3.13)$$

On sait que chaque variable x_i appartient à un *domaine* $\mathbb{X}_i$. Nous considérerons pour simplifier que les domaines sont des intervalles, notés $[x_i]$. Définissons le vecteur $\mathbf{x}$ tel que

$$\mathbf{x} = (x_1, x_2, \dots, x_{n_x})^T \quad (3.14)$$

et le domaine a priori pour $\mathbf{x}$

$$[\mathbf{x}] = [x_1] \times \dots \times [x_{n_x}]. \quad (3.15)$$

Soit $\mathbf{f}$ la fonction dont les fonctions coordonnées sont les relations f_j . On peut ainsi écrire l'équation 3.13 sous forme vectorielle $\mathbf{f}(\mathbf{x}) = \mathbf{0}$. Nous avons alors un *problème de satisfaction de contraintes* ou CSP (Constraint Satisfaction Problem) $\mathcal{H}$ qui peut s'écrire

$$\mathcal{H} : (\mathbf{f}(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in [\mathbf{x}]). \quad (3.16)$$

Un CSP est donc constitué de variables, de domaines contenant ces variables et de contraintes.

L'ensemble solution $\mathbb{S}$ de $\mathcal{H}$ est défini comme

$$\mathbb{S} = \{\mathbf{x} \in [\mathbf{x}] \mid \mathbf{f}(\mathbf{x}) = \mathbf{0}\}. \quad (3.17)$$

Les problèmes de satisfaction de contraintes peuvent mettre en jeu des contraintes d'égalité, mais également des inégalités. Pour définir un ensemble de contraintes dans le cadre de travail des CSP, les inégalités sont transformées en égalités en introduisant des variables positives nommées *variables d'écart*. Ainsi, l'ensemble de contraintes suivant

$$\begin{cases} x_1 - \tan(x_2) \leq 0, \\ x_1 + 2x_2 = 5, \end{cases} \quad (3.18)$$

se traduit en CSP en introduisant la variable d'écart x_3 . On obtient l'ensemble de

contraintes

$$\begin{cases} x_1 - \tan(x_2) + x_3 = 0, \\ x_1 + 2x_2 - 5 = 0, \end{cases} \quad (3.19)$$

où les domaines des variables sont $[x_1] = [x_2] =]-\infty, \infty[$ et $[x_3] = [0, \infty[$. Les fonctions coordonnées sont $f_1(\mathbf{x}) = x_1 - \tan(x_2) + x_3$ et $f_2(\mathbf{x}) = x_1 + 2x_2 - 5$.

Contracter un CSP $\mathcal{H}$ consiste à remplacer le domaine $[\mathbf{x}]$ par un plus petit domaine $[\mathbf{x}']$ sans changer l'ensemble solution. On a ainsi $\mathbb{S} \subset [\mathbf{x}'] \subset [\mathbf{x}]$. On définit la contraction *optimale* de $\mathcal{H}$ comme l'opération remplaçant $[\mathbf{x}]$ par la plus petite boîte contenant $\mathbb{S}$. Un opérateur permettant de contracter $\mathcal{H}$ est appelé un *contracteur*.

Différents contracteurs de base permettent de la contraction de CSP [Chabert and Jaulin, 2009]. En fonction de la classe du problème à résoudre, plusieurs méthodes peuvent être utilisées, notamment l'élimination de Gauss, l'algorithme de Gauss-Seidel, la méthode de Krawczyk et l'algorithme de Newton, détaillés dans [Jaulin et al., 2001]. Nous présentons ici les contracteurs qui seront utilisés par la suite : un contracteur par propagation-rétropropagation de contraintes $\mathcal{C}_{\downarrow\uparrow}$, un contracteur par programmation linéaire $\mathcal{C}_{\text{PL}}$ adapté aux contraintes linéaires, mais qui sera étendu à une classe plus générale de problèmes par la linéarisation parallèle et son contracteur $\mathcal{C}_{\parallel}$.

3.3.5.2 Contracteur par propagation-rétropropagation

Le contracteur par propagation-rétropropagation $\mathcal{C}_{\downarrow\uparrow}$, également appelé « *forward-backward* » dans la littérature [Benhamou et al., 1999, Jaulin, 2000], est basé sur la propagation de contraintes [Waltz, 1972]. Ce contracteur réduit les domaines du CSP $\mathcal{H} : (\mathbf{f}(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in [\mathbf{x}])$ en traitant de façon isolée chacune des contraintes.

On suppose que chaque contrainte est de la forme $f_i(x_1, \dots, x_{n_x}) = 0$, et que la fonction f_i peut se décomposer en une suite d'opérations mettant en œuvre des opérateurs et fonctions élémentaires telles que $(+, -, *, /, \sin, \cos \dots)$. On peut alors décomposer la contrainte en *contraintes primitives*.

Dans l'exemple de localisation par mesure de distances avec trois balises, d'après la fonction d'observation (3.12), la contrainte associée à une balise B_i peut s'écrire

$$d_{B_i} = \sqrt{(x - x_{B_i})^2 + (y - y_{B_i})^2},$$

et peut ainsi se décomposer en contraintes primitives en introduisant des variables

intermédiaires :

$$\begin{cases} i_1 = -x_{B_i} \\ i_2 = x + i_1 \\ i_3 = i_2^2 \\ i_4 = -y_{B_i} \\ i_5 = y + i_4 \\ i_6 = i_5^2 \\ i_7 = i_3 + i_6 \\ d_{B_i} = \sqrt{i_7} \end{cases}$$

Les domaines associés aux variables intermédiaires i_k introduites dans la décomposition sont $] -\infty, \infty[$. Une méthode pour contracter $\mathcal{H}$ avec la contrainte consiste à contracter chaque contrainte primitive jusqu'à ce que le contracteur arrive à un point fixe. C'est le principe de la propagation de contraintes, introduite par Waltz [Waltz, 1972]. Pour les contraintes mettant en jeu deux variables et une fonction, telle que la racine carrée, deux étapes de contraction sont produites en reformulant la contrainte : une à partir de l'image directe par la fonction, l'autre à partir de l'image réciproque. Ainsi, dans notre exemple, la contrainte $d_{B_i} = \sqrt{i_7}$ peut être réécrite de deux façons :

$$\begin{cases} d_{B_i} = \sqrt{i_7} \\ i_7 = (d_{B_i})^2 \end{cases}$$

et on obtient les étapes de contraction suivantes :

$$\begin{aligned} [d_{B_i}] &:= \sqrt{[i_7]} \cap [d_{B_i}] \\ [i_7] &:= [d_{B_i}]^2 \cap [i_7]. \end{aligned}$$

Pour les contraintes liant trois variables par un opérateur binaire (tel que l'addition), trois réécritures sont possibles. La contrainte $i_7 = i_3 + i_6$ peut ainsi s'exprimer sous les formes

$$\begin{cases} i_3 = i_7 - i_6 \\ i_6 = i_7 - i_3 \\ i_7 = i_3 + i_6 \end{cases}$$

ce qui permet d'obtenir les étapes de contraction suivantes :

$$[i_3] := ([i_7] - [i_6]) \cap [i_3]$$

$$[i_6] := ([i_7] - [i_3]) \cap [i_6]$$

$$[i_7] := ([i_3] + [i_6]) \cap [i_7].$$

L'algorithme de Waltz consiste à appliquer l'ensemble des étapes de contraction, sans ordre défini, jusqu'à ce que toutes les contraintes soient inefficaces. Cette stratégie peut nécessiter d'effectuer plusieurs tours de contraction avant d'atteindre un point fixe.

La contraction par propagation et rétropropagation permet de choisir les contraintes primitives utilisées dans un ordre optimal afin de maximiser la contraction obtenue [Benhamou et al., 1999].

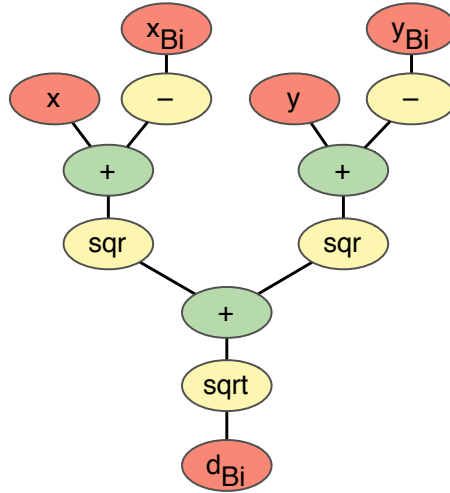


Figure 3.6: Décomposition en contraintes primitives de la contrainte associée à la mesure de distance d'une balise B_i .

Quand la décomposition en contraintes primitives mène à un graphe de contraintes acyclique (Figure 3.6), la contraction optimale par une contrainte $f_i = 0$ est obtenue en descendant puis remontant les nœuds de l'arbre. On peut ainsi écrire l'algorithme 3.1 du contracteur par propagation et rétropropagation pour la contrainte associée à la mesure de distance d'une balise B_i .

Le contracteur que nous venons de présenter ne traite qu'une contrainte à la fois. Quand il s'agit de contracter un domaine en tenant compte de toutes les contraintes du problème, on va appliquer successivement les contracteurs liés à chaque contrainte. Cependant, les dépendances entre contraintes impliquent que le contracteur n'arrive pas nécessairement à un point fixe dès sa première application. Le contracteur va donc être appelé plusieurs fois, jusqu'à obtention d'un point fixe.

Algorithme 3.1 Contracteur par propagation et rétropropagation pour une balise*Propagation**Rétropropagation*

$[i_1] := -[x_{B_i}]$	$[i_7] := [d_{B_i}]^2 \cap [i_7]$
$[i_2] := [x] + [i_1]$	$[i_3] := ([i_7] - [i_6]) \cap [i_3]$
$[i_3] := [i_2]^2$	$[i_6] := ([i_7] - [i_3]) \cap [i_6]$
$[i_4] := -[y_{B_i}]$	$[i_5] := (\text{sqr}^{-1}[i_6]) \cap [i_5]$
$[i_5] := [y] + [i_4]$	$[y] := ([i_5] - [i_4]) \cap [y]$
$[i_6] := [i_5]^2$	$[i_4] := ([i_5] - [y]) \cap [i_4]$
$[i_7] := [i_3] + [i_6]$	$[y_{B_i}] := -[i_4] \cap [y_{B_i}]$
$[d_{B_i}] := \sqrt{[i_7]} \cap [d_{B_i}]$	$[i_2] := (\text{sqr}^{-1}[i_3]) \cap [i_2]$
	$[x] := ([i_2] - [i_1]) \cap [x]$
	$[i_1] := ([i_2] - [x]) \cap [i_1]$
	$[x_{B_i}] := -[i_1] \cap [x_{B_i}]$

On peut voir dans notre exemple que le graphe liant les contraintes primitives correspondant à plusieurs balises comporte des cycles (Figure 3.7) qui préviennent la contraction optimale en une seule étape. On doit donc appliquer successivement les contracteurs correspondant à chaque balise jusqu'à obtention d'un point fixe.

La figure 3.8 montre les étapes de la contraction par propagation et rétropropagation appliquée au problème de localisation à trois balises. La boîte initiale est d'abord contractée par la contrainte liée à la balise B1, puis les contracteurs *forward-backward* associés à chaque balise sont successivement appelés, jusqu'à obtention d'un point fixe. L'interdépendance des variables fait que les contracteurs doivent être appelés plusieurs fois.

3.3.5.3 Programmation linéaire

Nous cherchons ici à contracter les domaines du CSP

$$\mathcal{H} : \left(\begin{array}{l} \mathbf{Ax} - \mathbf{b} = \mathbf{0} \\ \mathbf{A} \in [\mathbf{A}], \mathbf{b} \in [\mathbf{b}], \mathbf{x} \in [\mathbf{x}] \end{array} \right), \quad (3.20)$$

où la matrice $\mathbf{A}$ n'est pas supposée carrée. Dans le cas général, trouver la plus petite boîte $[\mathbf{x}]$ contenant tous les vecteurs $\mathbf{x}$ consistants avec $\mathcal{H}$ est un problème NP-difficile.

Sous certaines hypothèses, on peut transformer ce problème en $2n_{\mathbf{x}}$ problèmes de programmation linéaire. On peut alors construire le contracteur $\mathcal{C}_{\text{PL}}$, basé sur la résolution de problèmes de programmation linéaire.

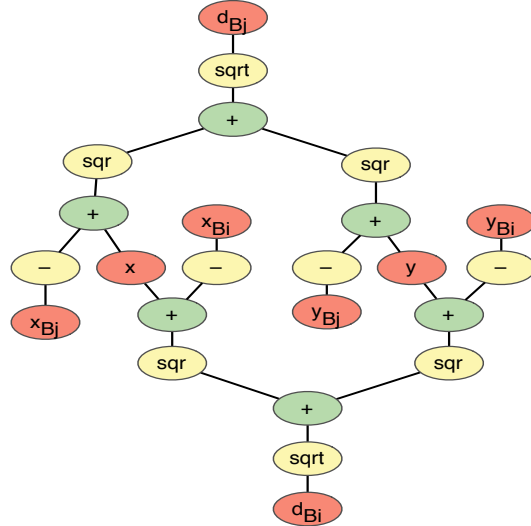


Figure 3.7: Graphe des contraintes primitives liées à deux balises B_i et B_j

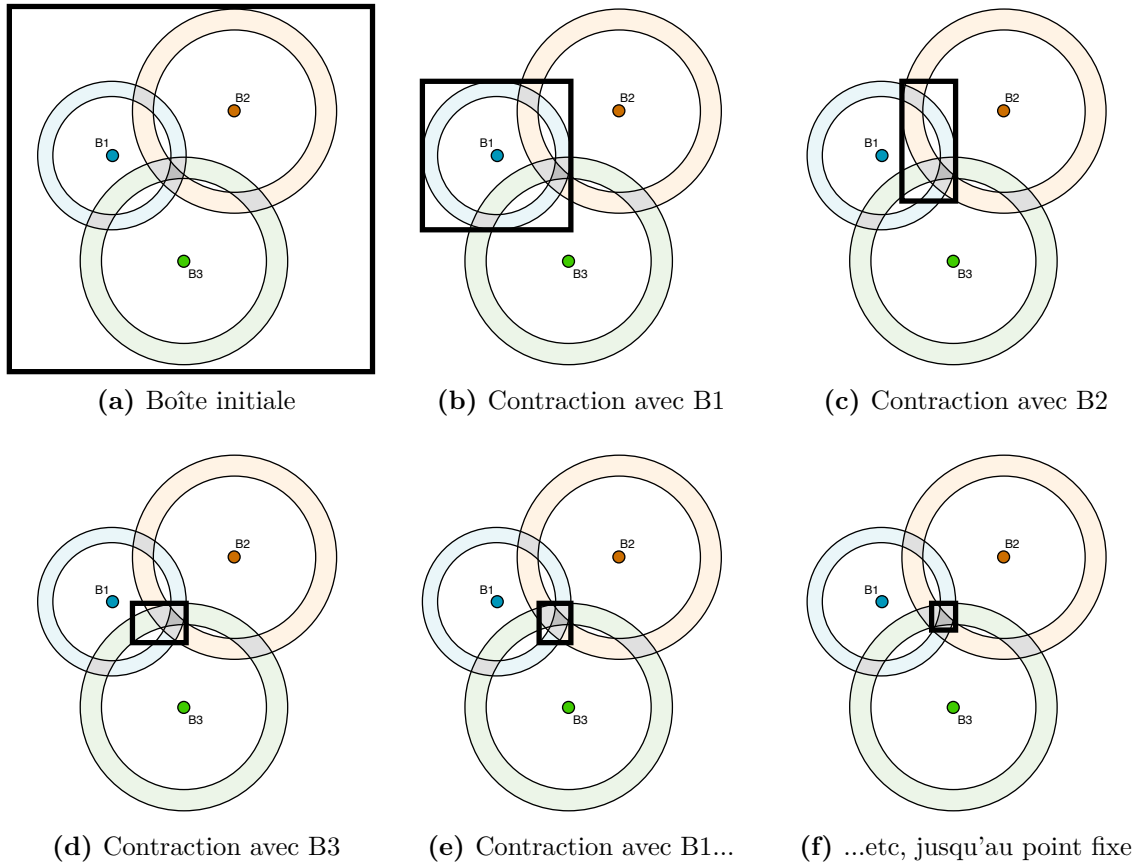


Figure 3.8: Application du contracteur par propagation-rétropropagation sur le problème de localisation à 3 balises

Étudions deux cas particuliers où le problème de la contraction du CSP $\mathcal{H}$ peut être ramené à la résolution de $2n_x$ problèmes d'optimisation linéaire [Jaulin et al., 2001].

Cas 1. Le vecteur $\mathbf{x}$ est supposé positif ($\mathbf{x} > \mathbf{0}$).

Le vecteur $\mathbf{x} \in [\mathbf{x}]$ est consistant avec $\mathcal{H}$ si et seulement si il existe $\mathbf{A} \in [\mathbf{A}]$ et il existe $\mathbf{b} \in [\mathbf{b}]$ tels que $\mathbf{Ax} - \mathbf{b} = \mathbf{0}$, c'est à dire,

$$\begin{aligned} & \exists \mathbf{A} \in [\mathbf{A}], \exists \mathbf{b} \in [\mathbf{b}] \mid \mathbf{Ax} - \mathbf{b} = \mathbf{0} \\ \Leftrightarrow & \exists \mathbf{A} \in [\mathbf{A}] \mid \mathbf{Ax} \in [\mathbf{b}] \\ \Leftrightarrow & \exists \mathbf{A} \in [\mathbf{A}] \mid \mathbf{Ax} \geq \underline{\mathbf{b}} \text{ et } \mathbf{Ax} \leq \overline{\mathbf{b}}. \end{aligned} \quad (3.21)$$

Or, $\mathbf{x} > \mathbf{0}$, on a donc les équivalences suivantes :

$$(\exists \mathbf{A} \in [\mathbf{A}] \mid \mathbf{Ax} \geq \underline{\mathbf{b}}) \Leftrightarrow \overline{\mathbf{Ax}} \geq \underline{\mathbf{b}}, \quad (3.22)$$

$$(\exists \mathbf{A} \in [\mathbf{A}] \mid \mathbf{Ax} \leq \overline{\mathbf{b}}) \Leftrightarrow \underline{\mathbf{Ax}} \leq \overline{\mathbf{b}}. \quad (3.23)$$

En conséquence, $\mathbf{x} \in [\mathbf{x}]$ est consistant avec $\mathcal{H}$ si et seulement si

$$\overline{\mathbf{Ax}} \geq \underline{\mathbf{b}} \text{ et } \underline{\mathbf{Ax}} \leq \overline{\mathbf{b}}. \quad (3.24)$$

La plus petite boîte $[\mathbf{q}]$ contenant tous les vecteurs $\mathbf{x}$ consistants avec $\mathcal{H}$ peut ainsi être calculée en résolvant $2n_x$ problèmes de programmation linéaire :

$$\begin{cases} \text{opt } x_i, \quad i = 1, \dots, n_x, \\ \begin{pmatrix} -\overline{\mathbf{A}} \\ \underline{\mathbf{A}} \end{pmatrix} \mathbf{x} \leq \begin{pmatrix} -\underline{\mathbf{b}} \\ \overline{\mathbf{b}} \end{pmatrix}, \\ \mathbf{x} \in [\mathbf{x}], \end{cases} \quad (3.25)$$

où opt représente successivement min et max pour calculer respectivement les coordonnées des bornes inférieures $\underline{\mathbf{q}}$ et supérieures $\overline{\mathbf{q}}$ du domaine contracté.

Cas 2. Le domaine $[\mathbf{A}]$ est supposé ponctuel ($\underline{\mathbf{A}} = \overline{\mathbf{A}}$).

Le vecteur $\mathbf{x} \in [\mathbf{x}]$ est consistant avec $\mathcal{H}$ si et seulement si il existe $\mathbf{b} \in [\mathbf{b}]$ tel que $\mathbf{Ax} - \mathbf{b} = \mathbf{0}$, c'est à dire,

$$\exists \mathbf{b} \in [\mathbf{b}] \mid \mathbf{Ax} - \mathbf{b} = \mathbf{0} \Leftrightarrow \mathbf{Ax} \in [\mathbf{b}] \Leftrightarrow \mathbf{Ax} \geq \underline{\mathbf{b}} \text{ et } \mathbf{Ax} \leq \overline{\mathbf{b}}. \quad (3.26)$$

Les coordonnées de la plus petite boîte contenant les vecteurs $\mathbf{x}$ consistants avec $\mathcal{H}$ peuvent ainsi être calculées par la résolution de $2n_x$ problèmes d'opti-

minimisation linéaire :

$$\begin{cases} \text{opt } x_i, i = 1, \dots, n_{\mathbf{x}}, \\ \begin{pmatrix} -\mathbf{A} \\ \mathbf{A} \end{pmatrix} \mathbf{x} \leq \begin{pmatrix} -\underline{\mathbf{b}} \\ \overline{\mathbf{b}} \end{pmatrix}, \\ \mathbf{x} \in [\mathbf{x}], \end{cases} \quad (3.27)$$

où opt représente successivement min et max pour obtenir les coordonnées des bornes inférieures et supérieures du domaine.

Le cas où $\mathbf{x} < \mathbf{0}$ peut facilement être reformulé de façon à être traité comme le cas 1. Le cas 2, où le domaine $[\mathbf{A}]$ est supposé ponctuel, sera utilisé pour mettre en œuvre le contracteur par linéarisation parallèle. Dans les deux cas, les $2n_{\mathbf{x}}$ problèmes d'optimisation obtenus sont résolus par une méthode classique, telle que l'algorithme du simplexe [Dantzig, 1963].

3.3.5.4 Linéarisation parallèle

On cherche à contracter le CSP $\mathcal{H} : (\mathbf{f}(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in [\mathbf{x}])$ dans le cas où le nombre de variables $n_{\mathbf{x}}$ et le nombre de contraintes $n_{\mathbf{f}}$ ne sont pas supposés identiques. Le contracteur $\mathcal{C}_{\parallel}$ permet de traiter toutes les contraintes simultanément, en utilisant une approche par *linéarisation parallèle* [Jaulin et al., 2001]. L'idée est de faire un encadrement de chaque $f_i(\mathbf{x})$ sur $[\mathbf{x}]$ entre deux hyperplans parallèles. On réalise ainsi un encadrement de la fonction $\mathbf{f}$ sur le domaine $[\mathbf{x}]$ tel que

$$\mathbf{A}\mathbf{x} + \underline{\mathbf{b}} \leq \mathbf{f}(\mathbf{x}) \leq \mathbf{A}\mathbf{x} + \overline{\mathbf{b}}. \quad (3.28)$$

Sous l'hypothèse que $\mathbf{f}$ est différentiable, on peut déterminer cet encadrement par le théorème des accroissements finis. Soit $\mathbf{x}_0$ un point de $[\mathbf{x}]$ (son centre par exemple), alors il existe $\xi \in [\mathbf{x}]$ tel que

$$\mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_0) + \mathbf{J}_{\mathbf{f}}(\xi)(\mathbf{x} - \mathbf{x}_0) \quad (3.29)$$

$$\Leftrightarrow \mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_0) + \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) + \mathbf{J}_{\mathbf{f}}(\xi)(\mathbf{x} - \mathbf{x}_0) - \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) \quad (3.30)$$

$$\Leftrightarrow \mathbf{f}(\mathbf{x}) = \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0)\mathbf{x} + \mathbf{f}(\mathbf{x}_0) - \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0)\mathbf{x}_0 + (\mathbf{J}_{\mathbf{f}}(\xi) - \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0))(\mathbf{x} - \mathbf{x}_0) \quad (3.31)$$

On peut alors, de façon équivalente, écrire $\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$, avec

$$\mathbf{A} = \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0) \quad (3.32)$$

et

$$\mathbf{b} = \mathbf{f}(\mathbf{x}_0) - \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0)\mathbf{x}_0 + (\mathbf{J}_{\mathbf{f}}(\xi) - \mathbf{J}_{\mathbf{f}}(\mathbf{x}_0))(\mathbf{x} - \mathbf{x}_0). \quad (3.33)$$

Une approximation externe de $\mathcal{H} : (\mathbf{f}(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in [\mathbf{x}])$ est ainsi obtenue

$$\left(\begin{array}{l} \mathbf{Ax} + \mathbf{b} = \mathbf{0} \\ \mathbf{A} = \mathbf{J}_f(\mathbf{x}_0) \\ \mathbf{b} = \mathbf{f}(\mathbf{x}_0) - \mathbf{Ax}_0 + (\mathbf{J}_f(\xi) - \mathbf{A})(\mathbf{x} - \mathbf{x}_0) \\ \mathbf{x} \in [\mathbf{x}], \xi \in [\mathbf{x}] \end{array} \right) \quad (3.34)$$

Le contracteur par linéarisation parallèle $\mathcal{C}_{\parallel}$ ainsi obtenu est décrit dans l'algorithme 3.2, où $\mathcal{C}_{\text{PL}}$ est le contracteur par programmation linéaire.

Algorithme 3.2 $\mathcal{C}_{\parallel}(\text{in} : \mathbf{f}, \text{inout} : [\mathbf{x}])$

- 1: $\mathbf{x}_0 := \text{mid}([\mathbf{x}])$
 - 2: $\mathbf{A} := \mathbf{J}_f(\mathbf{x}_0)$
 - 3: $[\mathbf{b}] := \mathbf{f}(\mathbf{x}_0) - \mathbf{Ax}_0 + ([\mathbf{J}_f]([\mathbf{x}]) - \mathbf{A})([\mathbf{x}] - \mathbf{x}_0)$
 - 4: $\mathcal{C}_{\text{PL}}(\mathbf{Ax} + \mathbf{b} = \mathbf{0}, \mathbf{x} \in [\mathbf{x}], \mathbf{b} \in [\mathbf{b}])$
-

3.4 Inversion ensembliste par intervalles (SIVIA)

Les intervalles et les boîtes sont des objets faciles à manipuler, grâce à l'arithmétique intervalle et aux fonctions d'inclusion. Les contracteurs définis précédemment permettent également de réduire un pavé en fonction d'une ou plusieurs contraintes.

En général, les solutions des problèmes traités par l'approche ensembliste sont des ensembles quelconques. La représentation par l'inclusion dans une boîte n'est, la plupart du temps, pas satisfaisante, en particulier lorsque l'ensemble solution n'est pas connexe.

3.4.1 Sous pavage

Afin de représenter un ensemble quelconque $\mathbb{X}$ tout en continuant à bénéficier des méthodes de calcul propres aux intervalles et aux boîtes, nous allons utiliser des *sous-pavages* de $\mathbb{R}^n$.

Un sous-pavage d'une boîte $[\mathbf{x}] \subset \mathbb{R}^n$ est l'union de boîtes de $[\mathbf{x}]$ qui ne se chevauchent pas, et de largeur non nulle. Deux boîtes d'un même sous-pavage peuvent avoir une intersection non vide si elles ont une frontière en commun, mais leurs intérieurs doivent être disjoints. Un sous-pavage de $[\mathbf{x}]$ qui couvre $[\mathbf{x}]$ est appelé un *pavage* de $[\mathbf{x}]$.

On peut, comme dans la figure 3.9, encadrer un ensemble $\mathbb{X}$ entre deux sous-pavages : une approximation intérieure $\underline{\mathbb{X}}$ et une approximation extérieure $\overline{\mathbb{X}} = \underline{\mathbb{X}} \cup \Delta\mathbb{X}$ (où $\Delta\mathbb{X}$ est le sous-pavage frontière) telles que

$$\underline{\mathbb{X}} \subset \mathbb{X} \subset \overline{\mathbb{X}}. \quad (3.35)$$

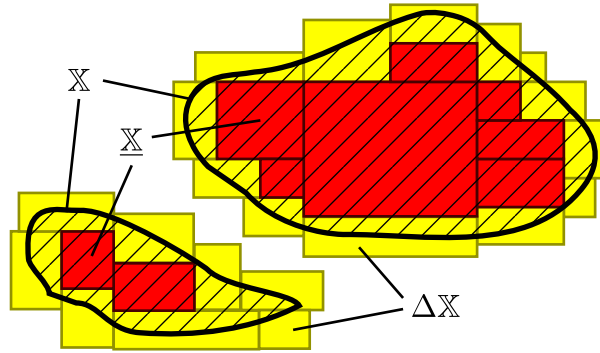


Figure 3.9: Encadrement de l'ensemble $\mathbb{X}$ entre deux sous-pavages $\underline{\mathbb{X}}$ et $\overline{\mathbb{X}} = \underline{\mathbb{X}} \cup \Delta\mathbb{X}$.

La connaissance du couple $\underline{\mathbb{X}}, \overline{\mathbb{X}}$ apporte de nombreuses informations sur l'ensemble $\mathbb{X}$. En particulier, si $\underline{\mathbb{X}}$ est non vide alors $\mathbb{X}$ est non vide. De même, si $\overline{\mathbb{X}}$ est vide, alors $\mathbb{X}$ est l'ensemble vide. Le volume du sous-pavage frontière $\Delta\mathbb{X}$ permet de caractériser la précision de l'approximation.

Lorsque chacune des boîtes d'un sous-pavage d'une boîte $[\mathbf{x}]$ peut être obtenue par bisections (division en deux boîtes selon un plan médian) et sélections successives à partir de $[\mathbf{x}]$, on parle de sous-pavage *régulier*. Les sous-pavages réguliers présentent l'avantage de pouvoir être représentés par des arbres binaires, ce qui limite leur occupation mémoire et permet d'implémenter facilement les opérations telles que l'union, l'intersection ou l'inclusion [Jaulin et al., 2001].

3.4.2 SIVIA

On cherche ici à caractériser l'ensemble $\mathbb{X}$ tel que $\mathbf{f}(\mathbb{X}) = \mathbb{Y}$. Autrement dit, il s'agit de déterminer l'image réciproque $\mathbb{X} = \mathbf{f}^{-1}(\mathbb{Y})$. En considérant que $\mathbb{Y}$ est un sous-pavage, l'algorithme d'inversion ensembliste par intervalles SIVIA (Set Inversion via Interval Analysis) permet de calculer un encadrement de $\mathbb{X}$ entre un sous-pavage intérieur et un sous-pavage extérieur tels que $\underline{\mathbb{X}} \subset \mathbb{X} \subset \overline{\mathbb{X}}$ [Jaulin and Walter, 1993]. Une boîte de recherche $[\mathbf{x}]$ dans laquelle l'ensemble solution doit être garanti de se trouver est nécessaire pour l'évaluation de SIVIA. Cette boîte peut éventuellement être choisie très grande si on n'a pas de connaissance a priori sur la solution. L'algorithme ne traitant que des pavés, une fonction d'inclusion convergente de $\mathbf{f}$ est également requise.

L'algorithme SIVIA est du type *branch and bound*. Il se déroule en deux étapes, successivement appelées par récursivité afin d'explorer l'espace des solutions. Tout d'abord, un test d'inclusion est effectué à l'aide de la fonction d'inclusion $[\mathbf{f}]$.

- Si l'image de la boîte $[\mathbf{x}]$ par $[\mathbf{f}]$ est disjointe de $\mathbb{Y}$, $[\mathbf{x}]$ est garanti de ne pas appartenir à l'ensemble solution. La boîte est donc oubliée.

- Si l'image de $[x]$ par $[f]$ est incluse dans l'ensemble $\mathbb{Y}$, alors, on a la garantie que $[x]$ appartient à l'ensemble solution $\mathbb{X}$. La boîte $[x]$ est donc ajoutée au sous-pavage intérieur $\underline{\mathbb{X}}$ et au sous-pavage extérieur $\overline{\mathbb{X}}$.
- Dans les autres cas, on ne peut pas conclure. On dit alors que $[x]$ est indéterminé. L'algorithme va alors consister à bissecter $[x]$ en deux boîtes $[x_1]$ et $[x_2]$, et appliquer récursivement l'inversion ensembliste sur les deux sous-boîtes obtenues.

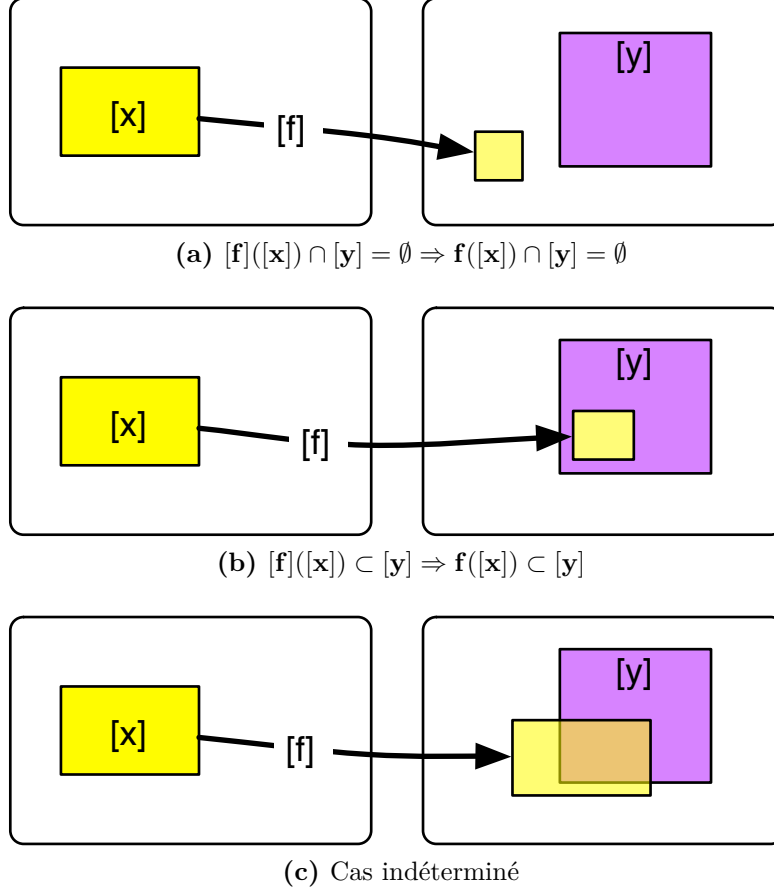


Figure 3.10: Test d'inclusion pour l'inversion ensembliste

Afin que l'algorithme termine, les boîtes indéterminées dont la largeur est inférieure à ε ne sont pas bissectées, mais ajoutées au sous pavage extérieur $\overline{\mathbb{X}}$. SIVIA est résumé dans l'algorithme 3.3, où les sous-pavages $\underline{\mathbb{X}}$ et $\overline{\mathbb{X}}$ sont initialement l'ensemble vide.

La figure 3.11 présente le résultat de l'algorithme SIVIA appliqué au problème de localisation d'un mobile avec quatre balises fixes. La partie droite de la figure montre la position des balises et du mobile dans le plan. Les arcs de cercles correspondent aux distances mesurées depuis chaque balise. Le résultat de l'inversion ensembliste est représenté à gauche de la figure : les pavés blancs ne font pas partie de la solution, les pavés rouges sont inclus dans l'ensemble solution et les pavés jaunes sont indéterminés.

Algorithme 3.3 SIVIA(in : $[\mathbf{x}]$, $\mathbf{f}$, $\mathbb{Y}$, ε ; inout : $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$)

```

1: if  $[\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} = \emptyset$  then
2:   return
3: end if
4: if  $[\mathbf{f}]([\mathbf{x}]) \subset \mathbb{Y}$  then
5:    $\underline{\mathbb{X}} := \underline{\mathbb{X}} \cup [\mathbf{x}]$ ;  $\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}]$ ; return
6: end if
7: if  $w([\mathbf{x}]) < \varepsilon$  then
8:    $\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}]$ ; return
9: end if
10: bissecter  $[\mathbf{x}]$  en  $[\mathbf{x}_1]$  et  $[\mathbf{x}_2]$ 
11: SIVIA( $[\mathbf{x}_1]$ ,  $\mathbf{f}$ ,  $\mathbb{Y}$ ,  $\varepsilon$ ,  $\underline{\mathbb{X}}$ ,  $\overline{\mathbb{X}}$ )
12: SIVIA( $[\mathbf{x}_2]$ ,  $\mathbf{f}$ ,  $\mathbb{Y}$ ,  $\varepsilon$ ,  $\underline{\mathbb{X}}$ ,  $\overline{\mathbb{X}}$ )

```

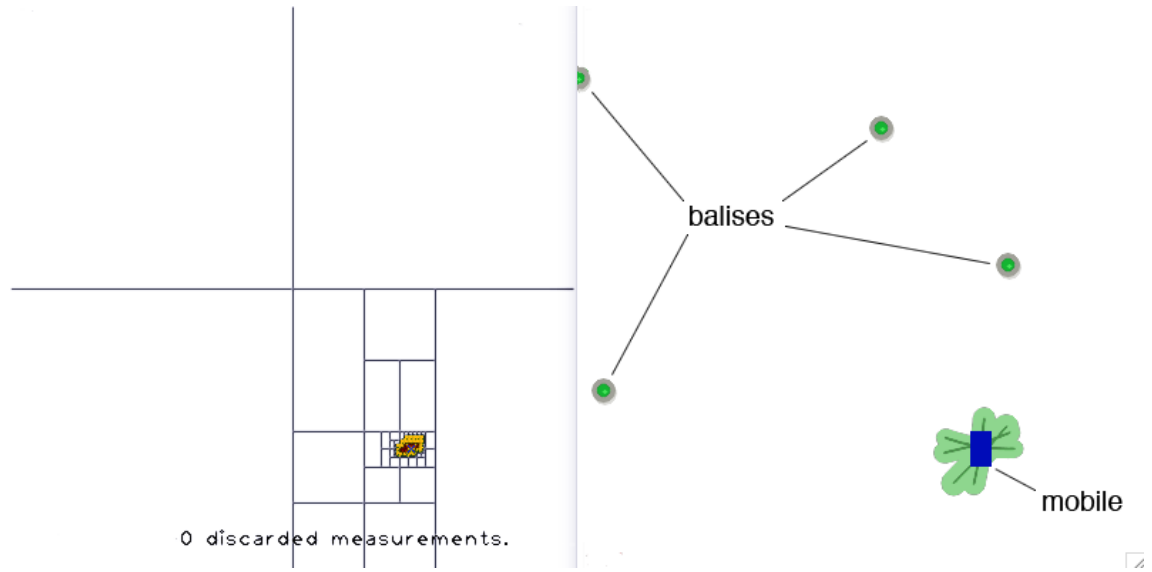


Figure 3.11: Résultat de l'algorithme SIVIA sur un problème de localisation avec 4 balises (données simulées)

L'implémentation de SIVIA présentée dans l'algorithme 3.3 utilise la récursivité. Les parties de l'espace à explorer sont ainsi implicitement stockées dans une structure de pile (la pile des appels de fonction dans ce cas précis), et la recherche se fait en profondeur d'abord. D'autres stratégies de recherche peuvent être préférées, en particulier quand le critère d'arrêt n'est plus seulement la taille des boîtes. On peut en effet souhaiter interrompre l'algorithme après un certain temps d'exécution ou un nombre de bisections. Dans ce cas, on préférera une stratégie d'exploration de l'espace des solutions en largeur d'abord. On utilisera donc une structure de file pour stocker les parties de l'espace à explorer [Bouron, 2002]. Nous reviendrons plus loin dans le chapitre sur cette stratégie qui permet de mieux explorer la zone de recherche lorsque l'algorithme est interrompu pour des contraintes de temps-réel par exemple.

3.4.3 SIVIA avec contracteur

Afin d'améliorer la qualité de la description de l'ensemble solution obtenu par inversion ensembliste pour un temps de calcul donné, on peut définir l'algorithme SIVIAP permettant l'estimation du paramètre $\mathbf{p}$ tel que $\mathbf{g}(\mathbf{p}) \in [\mathbf{y}]$ à l'aide d'un contracteur $\mathcal{C}_{\mathbb{S}_p}$. L'utilisation d'un contracteur permet la réduction du domaine de $\mathbf{p}$ avant l'opération de bisection. Pour un nombre de bisections donné, on obtient ainsi un sous-pavage solution plus précis. Le contracteur le plus simple pour $\mathbb{S}_p$ est le contracteur $\mathcal{C}_{\mathbb{S}_p}^0$ tel que

$$\mathcal{C}_{\mathbb{S}_p}^0 : \begin{cases} \mathbb{IR}^{n_p} & \rightarrow \mathbb{IR}^{n_p} \\ [\mathbf{p}] & \mapsto \begin{cases} \emptyset & \text{si } [\mathbf{g}]([\mathbf{p}]) \cap [\mathbf{y}] = \emptyset \\ [\mathbf{p}] & \text{sinon} \end{cases} \end{cases} .$$

L'utilisation du contracteur $\mathcal{C}_{\mathbb{S}_p}^0$ conduit au même résultat que l'algorithme SIVIA basé sur une fonction d'inclusion.

En pratique on va utiliser un contracteur plus efficace, qui peut être une combinaison de $\mathcal{C}_{\mathbb{S}_p}^0$ avec un ou plusieurs des contracteurs décrits dans la section précédente. La contrepartie de l'utilisation d'un contracteur est que le sous-pavage solution obtenu n'est plus nécessairement régulier.

3.5 Inversion ensembliste robuste

L'utilisation de méthodes robustes est fondamentale dès que les observations sont des données mesurées par des capteurs. Cela est d'autant plus vrai quand les capteurs utilisés sont de faible coût, les capteurs haut de gammes intégrant pour la plupart des mécanismes permettant d'éliminer les mesures aberrantes. C'est aussi le cas lorsque le modèle d'observation choisi n'explique qu'une partie des observations. Par exemple, dans [Jaulin, 2009] un robot sous-marin se localise avec un Sonar

Algorithme 3.4 SIVIAP(in : $[\mathbf{p}], \mathcal{C}_{\mathbb{S}_p}, \mathbf{g}, [\mathbf{y}], \varepsilon$; inout : $\mathbb{S}_p, \bar{\mathbb{S}}_p$)

```

1:  $[\mathbf{p}] := \mathcal{C}_{\mathbb{S}_p}([\mathbf{p}])$ 
2: if  $[\mathbf{p}] = \emptyset$  then
3:   return
4: end if
5: if  $[\mathbf{g}]([\mathbf{p}]) \subset [\mathbf{y}]$  then
6:    $\mathbb{S}_p := \mathbb{S}_p \cup [\mathbf{p}]; \bar{\mathbb{S}}_p := \bar{\mathbb{S}}_p \cup [\mathbf{p}];$  return
7: end if
8: if  $w([\mathbf{p}]) < \varepsilon$  then
9:    $\bar{\mathbb{S}}_p := \bar{\mathbb{S}}_p \cup [\mathbf{p}];$  return
10: end if
11: bissecter  $[\mathbf{p}]$  en  $[\mathbf{p}_1]$  et  $[\mathbf{p}_2]$ 
12: SIVIAP( $[\mathbf{p}_1], \mathcal{C}_{\mathbb{S}_p}, \mathbf{g}, [\mathbf{y}], \varepsilon, \mathbb{S}_p, \bar{\mathbb{S}}_p$ )
13: SIVIAP( $[\mathbf{p}_2], \mathcal{C}_{\mathbb{S}_p}, \mathbf{g}, [\mathbf{y}], \varepsilon, \mathbb{S}_p, \bar{\mathbb{S}}_p$ )
    
```

par rapport aux murs d'une piscine, mais la présence d'un filet dans le bassin qui renvoie une partie des ultra-sons n'est pas modélisée explicitement. Ces mesures sont considérées comme des mesures aberrantes.

De nombreuses méthodes ont été développées pour permettent l'estimation de paramètres à partir de données contenant des valeurs aberrantes. Les M-estimateurs permettent, par le choix d'une fonction à minimiser adaptée, de réduire ou négliger l'influence des valeurs aberrantes. La méthode RANSAC choisit par des tirages aléatoires un jeu de paramètres expliquant le maximum de données, en classifiant explicitement le jeu de données en l'ensemble des données expliquées par le modèle (inliers) et l'ensemble des mesures aberrantes (outliers). Citons encore l'estimateur Least Median of Squares considéré parmi les plus robustes. Ces méthodes classiques d'estimation robuste se basent sur les résidus et mettent en œuvre des mécanismes d'itération du fait de la non linéarité du système. Elles sont présentées dans l'annexe A.

Avant de présenter les méthodes ensemblistes robustes, étudions l'influence d'une mesure aberrante sur l'ensemble solution d'un problème à erreurs bornées. La figure 3.12 présente différents cas de figure généraux, illustrés par l'exemple de localisation à trois balises. Le premier cas (Figure 3.12a) correspond à l'absence de mesure aberrante : l'hypothèse d'erreur bornée sur le modèle d'observation est respectée, et l'ensemble solution contient la vraie valeur. Dans les trois autres cas, la mesure issue de la balise B3 est erronée. On peut alors obtenir une solution vide (Figure 3.12b), dans le cas où toutes les mesures sont incompatibles entre elles. Cependant, la présence d'une mesure aberrante n'est pas nécessairement source de solution vide (sur la figure (3.12b) la mesure est assez aberrante). L'erreur peut dépasser les bornes spécifiées, sans toutefois créer une inconstance avec l'ensemble des mesures. C'est le cas lorsque l'erreur est relativement petite (Figure 3.12c), où lorsque la non-linéarité du problème entre en jeu (Figure 3.12d - l'erreur sur B3 est

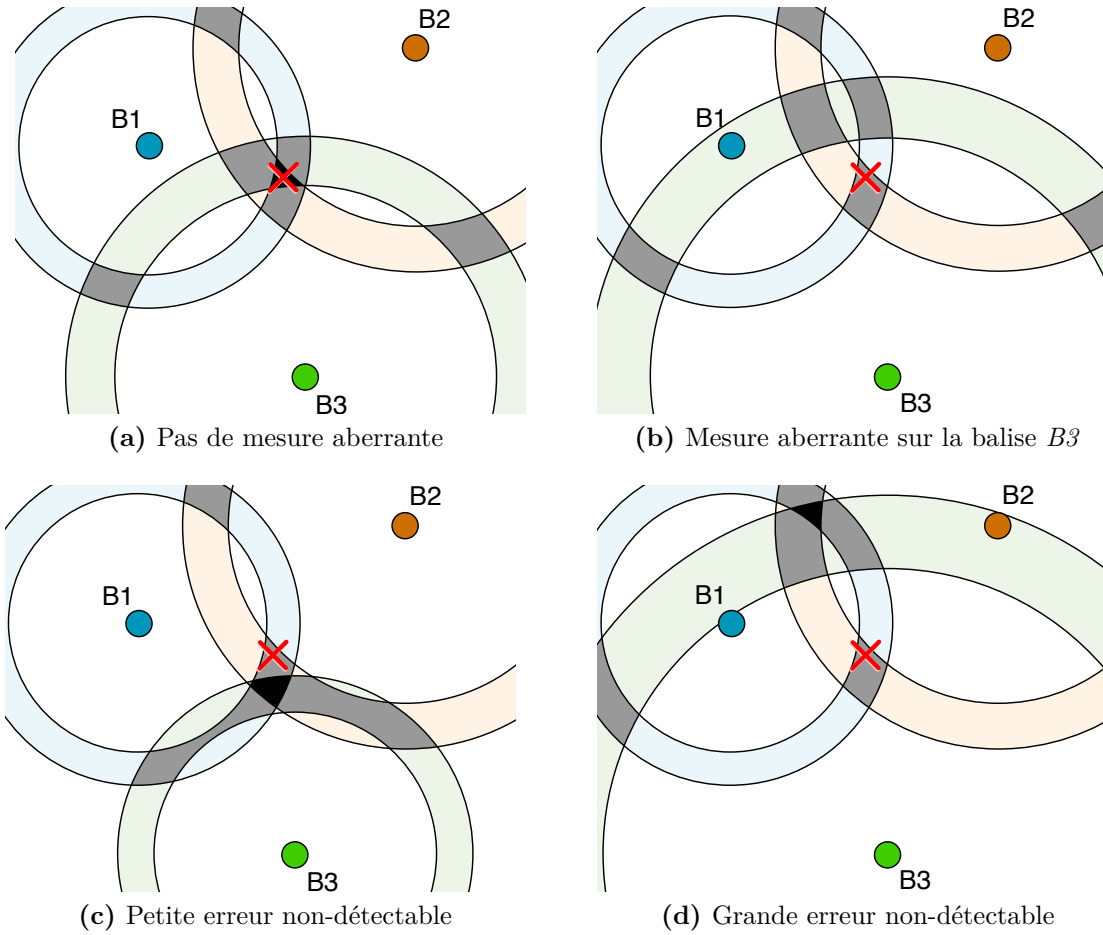


Figure 3.12: Localisation ensembliste avec trois balises, en présence d'une mesure aberrante. *L'ensemble solution est en noir. La solution robuste 1-relaxée est en gris. La croix rouge indique la position réelle du récepteur.*

tellement grande qu'elle crée une solution fantôme). Dans ces deux cas, on obtient un ensemble solution non-vide, mais qui ne contient pas la vraie valeur.

Intéressons-nous maintenant à la présentation de deux méthodes ensemblistes robustes.

3.5.1 Inversion ensembliste q -relaxée

Nous présentons ici une méthode d'estimation ensembliste robuste qui consiste à tolérer la présence d'un certain nombre q de mesures aberrantes parmi les m observations considérées. On réalise ainsi une *inversion ensembliste q -relaxée*, dont l'ensemble solution correspond à l'ensemble des paramètres compatibles avec au moins $m - q$ observations.

3.5.1.1 Intersection q -relaxée

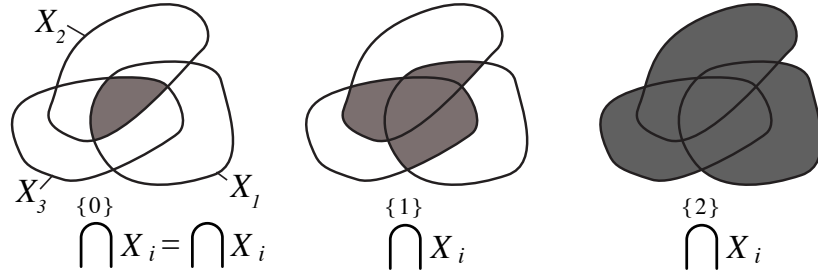


Figure 3.13: Intersection q -relaxée de trois ensembles, avec $q \in \{0, 1, 2\}$

Considérons m ensembles $X_1, \dots, X_m$ de $\mathbb{R}^n$. Nous pouvons définir leur *intersection q -relaxée*, notée $\bigcap_{\{q\}} X_i$, comme l'ensemble des points $x \in \mathbb{R}^n$ appartenant à au moins $m - q$ des ensembles X_i (Figure 3.13). En suivant cette définition, l'intersection 0-relaxée correspond à l'intersection au sens standard des ensembles. De même, l'intersection $(m - 1)$ -relaxée de m ensembles correspond à leur union.

3.5.1.2 Inversion ensembliste robuste par intervalles

L'algorithme d'inversion ensembliste robuste par intervalles RSIVIA (*Robust Set Inversion via Interval Analysis*) [Jaulin, 2009, Jaulin et al., 2001] permet le calcul garanti d'un ensemble solution q -relaxé au problème d'inversion ensembliste $X = \mathbf{f}^{-1}([\mathbf{y}])$. Il consiste à calculer l'intersection q -relaxée des ensembles solution $X^i = \mathbf{f}_i^{-1}([\mathbf{y}_i])$ correspondant à chaque composante de la fonction $\mathbf{f}$.

Sur chacune des sous-figures de la figure 3.12, la solution exacte 1-relaxée au problème de localisation est représentée par l'union des ensembles noirs et gris. On

peut observer que la vraie position est toujours à l'intérieure de l'ensemble solution 1-relaxé, même quand une mesure aberrante est présente. En revanche, par manque de redondance, l'ensemble solution 1-relaxé n'est pas connexe. La solution est donc ambiguë.

Si on connaît une fonction d'inclusion de la fonction $\mathbf{f}$ à inverser, on peut adapter l'algorithme SIVIA (Alg. 3.3) afin d'approximer la solution q -relaxée au problème d'inversion ensembliste entre un sous-pavage intérieur $\underline{\mathbb{X}}$ et un sous-pavage extérieur $\overline{\mathbb{X}}$. Pour élaborer un test d'inclusion pour l'appartenance d'une boîte à la solution q -relaxée, définissons le degré d'inclusion « incl » d'une boîte dans une autre et le degré de séparation « sep » entre deux boîtes [Jaulin et al., 2002], tels que :

$$\begin{aligned} \text{incl}([\mathbf{x}], [\mathbf{y}]) &= \sum_{i=1}^m \text{incl}([x_i], [y_i]) \\ \text{sep}([\mathbf{x}], [\mathbf{y}]) &= \sum_{i=1}^m \text{sep}([x_i], [y_i]), \end{aligned}$$

$$\text{où } \text{incl}([x], [y]) = \begin{cases} 1 & \text{si } [x] \subset [y] \\ 0 & \text{sinon} \end{cases} \text{ et } \text{sep}([x], [y]) = \begin{cases} 1 & \text{si } [x] \cap [y] = \emptyset \\ 0 & \text{sinon} \end{cases}.$$

On peut alors définir le test d'inclusion d'une boîte $[\mathbf{x}]$ pour la solution q -relaxée :

$$\begin{cases} \text{sep}([\mathbf{f}]([\mathbf{x}]), [\mathbf{y}]) > q & [\mathbf{x}] \text{ ne contient aucune solution} \\ \text{incl}([\mathbf{f}]([\mathbf{x}]), [\mathbf{y}]) \geq m - q & [\mathbf{x}] \text{ est inclus dans l'ensemble solution} \\ \text{sinon} & [\mathbf{x}] \text{ est indéterminé} \end{cases} \quad (3.36)$$

Ainsi, le degré d'inclusion représente le nombre de fonctions coordonnées f_i qui vérifient le test d'inclusion, et le degré d'exclusion représente le nombre de fonctions coordonnées f_i pour lesquelles le test d'inclusion échoue. L'algorithme 3.5 implémente l'inversion ensembliste q -relaxée. Il s'agit d'une adaptation de l'algorithme SIVIA (Alg. 3.3) mettant en œuvre le test d'inclusion q -relaxé que nous venons de présenter.

Si le problème est exprimé sous forme de contraintes, et que l'on dispose de contracteurs $\mathcal{C}_1, \dots, \mathcal{C}_m$ pour chacune des m contraintes en jeu, on peut définir un contracteur pour l'intersection q -relaxée de m contraintes, présenté dans l'algorithme 3.6. Ce contracteur prend en entrée m contracteurs, correspondant aux m contraintes parmi lesquelles q peuvent ne pas être satisfaites. Il applique chaque contracteur indépendamment sur la boîte initiale, et retourne la boîte englobante de l'intersection q -relaxée des m domaines contractés (l'opérateur $\square$ retourne l'enveloppe intervalle d'un ensemble). La principale difficulté réside dans le calcul effectif de l'intersection q -relaxée des m boîtes contractées. Une méthode rapide est présentée dans la section 3.5.1.3.

Le contracteur q -relaxé de m contraintes peut être utilisé dans l'algorithme d'inversion ensembliste avec contracteur (Alg. 3.4) conjointement avec le test d'inclusion

Algorithme 3.5 RSIVIA(in: $[\mathbf{x}]$, $\mathbf{f}$, $[\mathbf{y}]$, q ; inout: $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$)

```

1:  $incl := 0$ ;  $sep := 0$ 
2: for  $i = 1 \dots m$  do
3:   if  $[f_i]([\mathbf{x}]) \subset [y_i]$  then
4:      $incl := incl + 1$ 
5:   else if  $[f_i]([\mathbf{x}]) \cap [y_i] = \emptyset$  then
6:      $sep := sep + 1$ 
7:   end if
8: end for
9: if  $incl \geq m - q$  then
10:   $\underline{\mathbb{X}} := \underline{\mathbb{X}} \cup [\mathbf{x}]$ ,  $\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}]$ 
11: else if  $sep > q$  then
12:  return
13: else if  $w([\mathbf{x}]) < \varepsilon$  then
14:   $\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}]$ 
15: else
16:   $([\mathbf{x}_1], [\mathbf{x}_2]) := \text{bisect}([\mathbf{x}])$ 
17:  RSIVIA( $[\mathbf{x}_1]$ ,  $\mathbf{f}$ ,  $[\mathbf{y}]$ ,  $q$ ,  $\underline{\mathbb{X}}$ ,  $\overline{\mathbb{X}}$ )
18:  RSIVIA( $[\mathbf{x}_2]$ ,  $\mathbf{f}$ ,  $[\mathbf{y}]$ ,  $q$ ,  $\underline{\mathbb{X}}$ ,  $\overline{\mathbb{X}}$ )
19: end if
    
```

Algorithme 3.6 $\mathcal{C}_{qrel}$ (inout : $[\mathbf{x}]$; in : q , $\mathcal{C}_1, \dots, \mathcal{C}_m$)

 Contracteur q -relaxé de m contraintes.

```

1: for  $i = 1$  to  $m$  do
2:    $[\mathbf{x}^i] = \mathcal{C}_i([\mathbf{x}])$  contraction de  $[\mathbf{x}]$  avec la  $i^{\text{ème}}$  contrainte
3: end for
4:  $[\mathbf{x}] = \square \left( \bigcap_{i \in \{1, \dots, m\}}^{\{q\}} [\mathbf{x}^i] \right)$  enveloppe de l'intersection  $q$ -relaxée de  $m$  pavés
    
```

q -relaxé de l'équation (3.36) afin d'accélérer la résolution du problème d'inversion ensembliste robuste.

L'algorithme (3.7) présente une méthode d'inversion ensembliste robuste par intervalles avec un algorithme SIVIA ne calculant que le sous-pavage extérieur. Cette méthode permet de calculer un sous-pavage externe couvrant la solution q -relaxée du problème en n'utilisant que des contracteurs. Les appels récursifs de SIVIA après chaque bisection ont été remplacés par l'utilisation d'une liste de travail $\mathfrak{L}$ contenant les boîtes à traiter. Cela permet d'effectuer l'exploration de l'espace des solutions en largeur d'abord, au prix d'une plus grande occupation mémoire.

La figure 3.14 présente les étapes principales de l'inversion ensembliste robuste, appliquée au problème de localisation à 3 balises fixes. La boîte a priori $[\mathbf{x}]$ est tout d'abord contractée indépendamment avec la contrainte associée à chaque mesure.

Algorithm 3.7 RSIVIAExt(in : $[\mathbf{x}], \{\mathcal{C}_i, i = 1 \dots m\}, [\mathbf{y}], \varepsilon$; inout : $\bar{\mathbb{S}}_x$)

```

1: push( $[\mathbf{x}], \mathcal{L}$ )
2: while  $\mathcal{L} \neq \emptyset$  do
3:    $[\mathbf{x}] = \text{pull}(\mathcal{L})$ 
4:   for  $i = 1$  to  $m$  do
5:      $[\mathbf{x}^i] = \mathcal{C}_i([\mathbf{x}])$  contraction de  $[\mathbf{x}]$  avec la  $i^{\text{ème}}$  contrainte
6:   end for
7:    $[\mathbf{x}] = \square \left( \bigcap_{i \in \{1, \dots, m\}} [\mathbf{x}^i] \right)$ 
8:   if  $[\mathbf{x}] \neq \emptyset$  then
9:     if  $w([\mathbf{x}]) < \varepsilon$  then
10:       $\bar{\mathbb{S}}_x := \bar{\mathbb{S}}_x \cup [\mathbf{x}]$ 
11:     else
12:        $([\mathbf{x}_1], [\mathbf{x}_2]) = \text{bisect}([\mathbf{x}])$ 
13:       push( $[\mathbf{x}_1], \mathcal{L}$ ); push( $[\mathbf{x}_2], \mathcal{L}$ )
14:     end if
15:   end if
16: end while

```

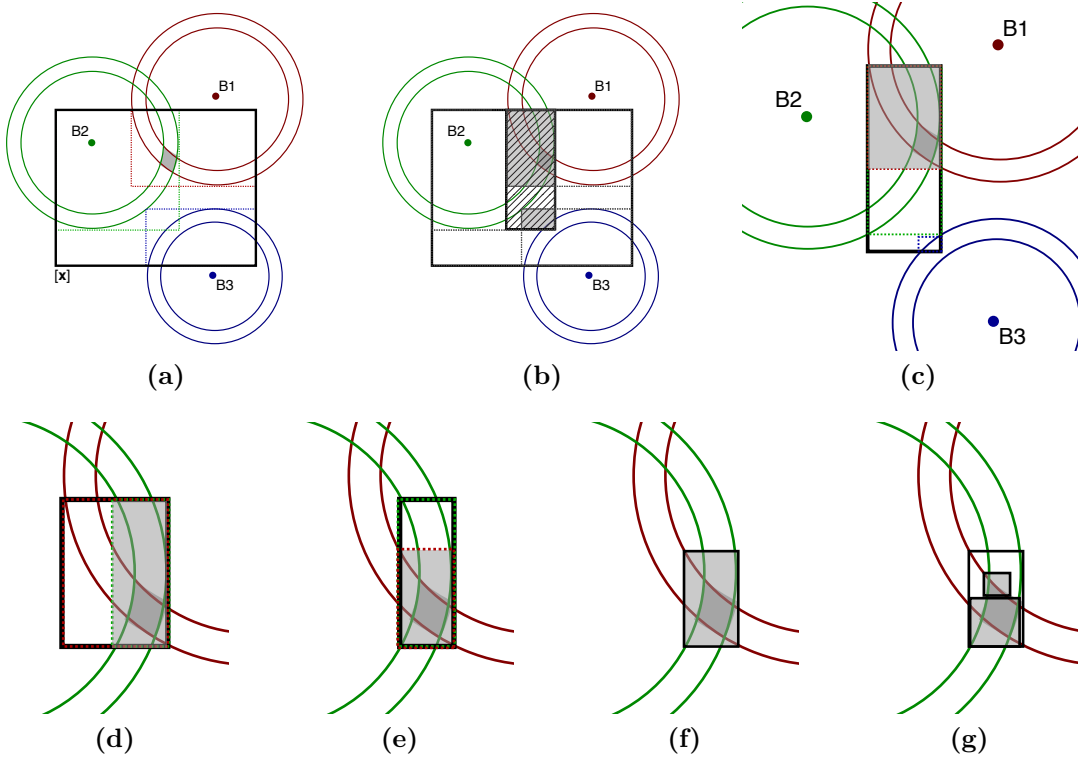


Figure 3.14: Déroulement de RSIVIAEXT sur un problème de localisation à 3 balises, avec $q = 1$.

On obtient ainsi trois boîtes $[\mathbf{x}^i]$ couvrant l'intersection de chaque contrainte avec la boîte a priori (Figure. 3.14a). Les zones en gris sur la figure 3.14b représentent l'intersection 1-relaxée des trois boîtes contractées. La boîte hachurée, qui correspond à l'union intervalle des boîtes grises, devient la nouvelle boîte initiale. Nous avons ainsi contracté le domaine a priori $[\mathbf{x}]$ sans perdre de solution. La contraction par rapport à chaque mesure est encore une fois effectuée à partir du nouveau domaine initial (Figure. 3.14c), puis la boîte est de nouveau réduite, de façon à être la boîte englobante de l'intersection q -relaxée des boîtes contractées. Ces opérations sont répétées (Figures. 3.14d et 3.14e) jusqu'à ce qu'aucune contraction ne puisse plus être effectuée (Figure. 3.14f). On effectue alors une bisection, et on réitère le même processus sur les deux sous-boîtes (Figure. 3.14g).

La figure 3.15 présente les résultats de l'inversion ensembliste robuste sur le problème de localisation avec quatre balises fixes, pour différentes valeurs de q . Le calcul avec $q = 0$ n'est pas représenté car il correspond à l'algorithme d'inversion ensembliste non-robuste déjà présenté sur la figure 3.11. Avec $q = 1$, le domaine d'incertitude s'est agrandi. Lors du calcul de la solution avec $q = 2$, on observe que l'ensemble solution n'est plus connexe. Cela est dû au fait qu'en relaxant deux contraintes, seules deux balises contraignent simultanément la position du mobile. Or, la localisation avec seulement deux balises présente une ambiguïté (deux solutions symétriques par rapport à l'axe joignant les deux balises). Seules deux composantes connexes de la solution correspondant à ces ambiguïtés sont visibles, les deux autres n'appartenant pas à la boîte initiale. Le calcul de la solution 3-relaxée fait apparaître les couronnes correspondant aux contraintes liées à chaque balise. En effet, l'intersection 3-relaxée de quatre ensembles correspond à leur union, comme nous l'avons déjà vu.

3.5.1.3 Contracteur rapide pour l'intersection q -relaxée

Le calcul de l'intersection q -relaxée de m boîtes de dimension n est l'étape centrale de l'algorithme d'inversion ensembliste robuste et du contracteur q -relaxé. Malheureusement, ce problème est de complexité $O(m^n)$ [Jaulin, 2009], ce qui complique sa mise en pratique sur des problèmes réels. Quand la dimension n des pavés est fixée, le problème est polynomial en le nombre de boîtes à intersecter. Cependant, comme le problème est de complexité exponentielle en n , le problème devient très rapidement incalculable quand la dimension de l'espace augmente. Il est ainsi illusoire de vouloir traiter des problèmes en dimension supérieure à 4 dans un temps raisonnable.

Afin d'accélérer le calcul de l'intersection q -relaxée de pavés, on peut définir un contracteur basé sur la projection des contraintes sur les axes. Chaque dimension est ainsi considérée séparément, afin de diminuer la complexité du problème.

La figure 3.16 illustre le principe de fonctionnement du contracteur présenté dans [Jaulin, 2011b]. La boîte initiale est contractée indépendamment avec les contraintes associées à chaque mesure. On obtient ainsi m boîtes contractées, dont on souhaite

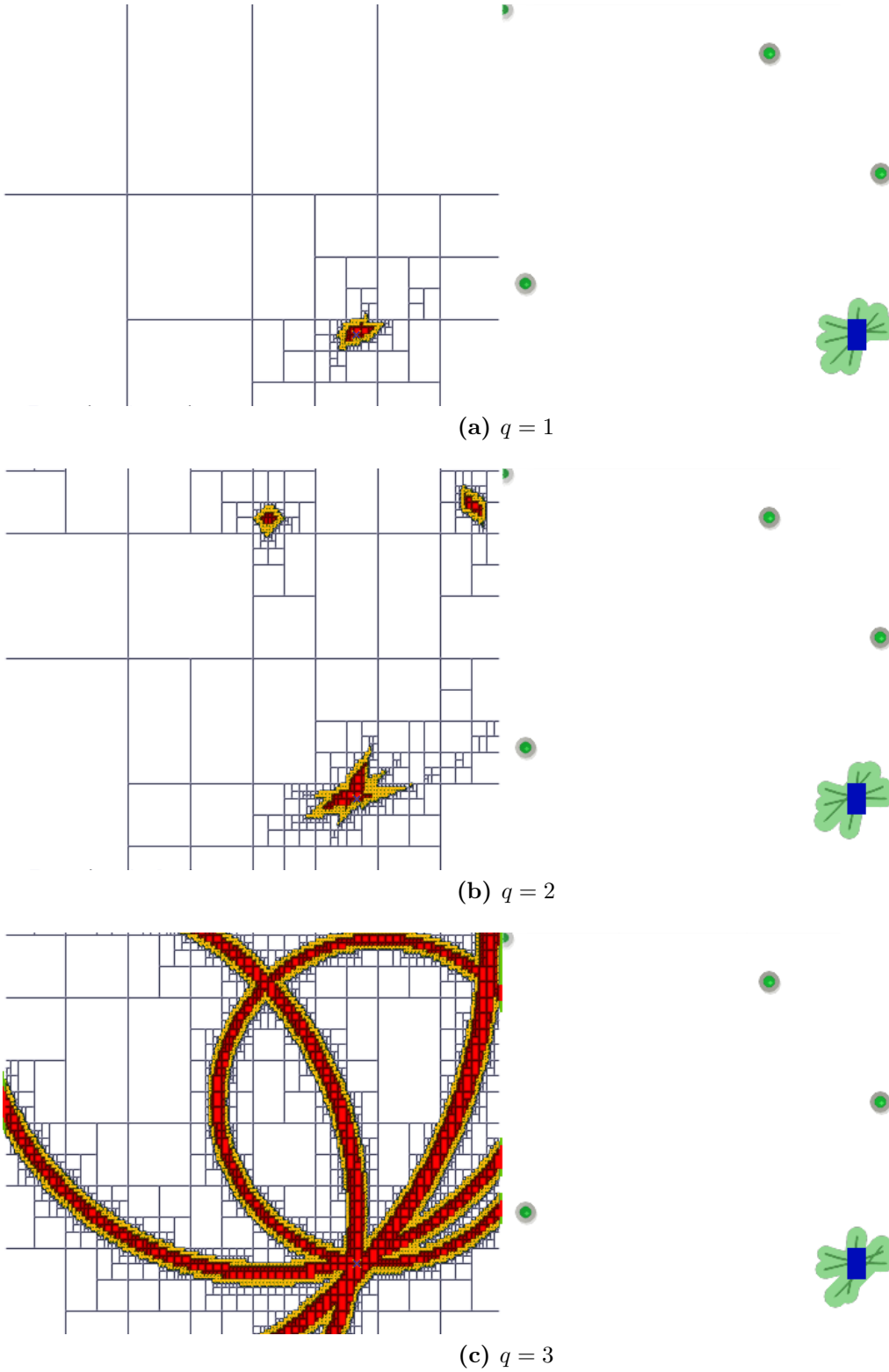


Figure 3.15: Résultat de l'algorithme RSIVIA sur un problème de localisation avec 4 balises (données simulées)

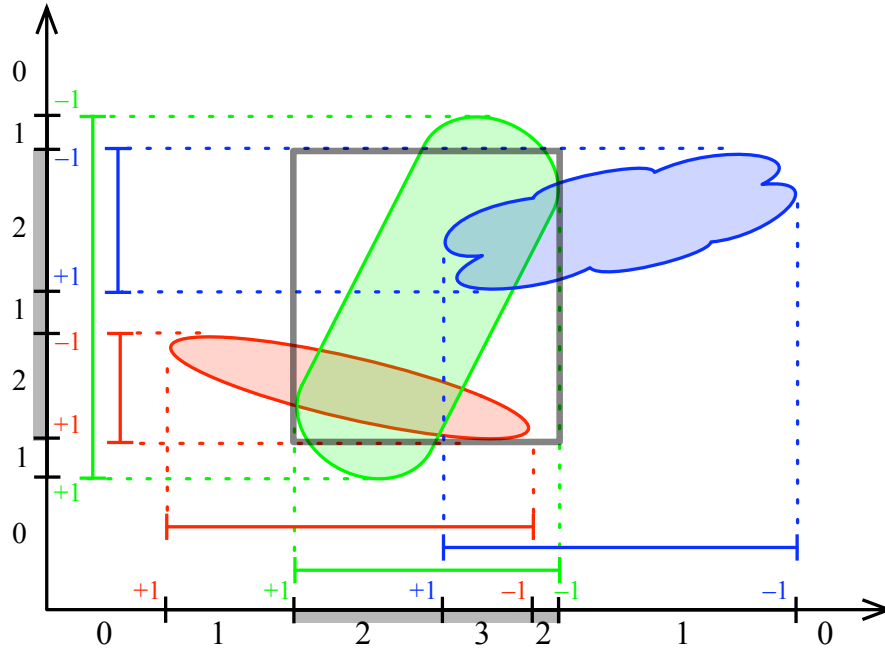


Figure 3.16: Une itération du contracteur rapide pour l'intersection q -relaxée, avec 3 contraintes (en vert, rouge et bleu) et $q = 1$. Le résultat de la contraction est en gris.

déterminer l'intersection q -relaxée. On va alors considérer chaque dimension du problème séparément. Les boîtes contractées sont projetées sur l'axe correspondant à la dimension considérée. On crée alors une liste triée des bornes des m intervalles obtenus, en leur associant une étiquette : les bornes inférieures (crochets ouvrants) sont associées à l'étiquette $+1$, tandis que les bornes supérieures (crochets fermants) sont associées à l'étiquette -1 . On construit une telle liste de bornes étiquetées pour chaque dimension de l'espace des paramètres à estimer. On obtient ainsi n listes triées de bornes ouvrantes et fermantes, respectivement étiquetées $+1$ et -1 .

Pour chaque dimension, un compteur de bornes est initialisé à 0, et on commence à parcourir la liste des bornes dans l'ordre croissant. Chaque fois qu'une borne inférieure est rencontrée, le compteur est incrémenté, et chaque fois qu'une borne supérieure est rencontrée, le compteur est décrémenté. La borne pour laquelle le compteur atteint pour la première fois la valeur $m - q$ est gardée comme borne inférieure de l'intersection q -relaxée pour l'axe considéré. De même, la dernière borne de la liste qui fait tomber le compteur sous la valeur $m - q$ est définie comme borne supérieure de l'intersection q -relaxée sur l'axe considéré. La boîte contractée est obtenue en prenant le produit cartésien des intervalles contractés obtenus sur chaque dimension. Le contracteur ainsi obtenu peut être appliqué plusieurs fois, jusqu'à obtention d'un point fixe.

Ce contracteur n'est pas optimal : il ne permet pas d'atteindre le pavé enveloppe de l'intersection q -relaxée des boîtes dans le cas général. Cependant, sa complexité est

faible comparée à la résolution exacte. La complexité du contracteur q -relaxé rapide par projection est de $O(n \cdot m \log(m))$, en supposant que l'on dispose d'un algorithme de tri de complexité linéarithmique¹.

3.5.2 GOMNE

L'inversion ensembliste robuste, basée sur l'intersection q -relaxée des contraintes permet le calcul d'une solution avec la garantie que la vraie valeur y sera incluse, tant que le nombre de mesures aberrantes est inférieur ou égal à q .

En pratique q peut être déterminé de manière empirique ou par expertise du problème considéré. Cependant, lorsque q est trop supérieur au nombre de mesures aberrantes, la solution est très pessimiste. Une autre stratégie consiste à estimer q en même temps que les paramètres. Plus précisément, on va estimer le nombre minimal de valeurs aberrantes parmi les observations pour que le modèle puisse expliquer les autres mesures : c'est le principe de GOMNE, *Guaranteed Outlier Minimal Number Estimator* [Jaulin et al., 1996].

Pour ce faire, on va successivement calculer une solution avec q croissant, en commençant par $q = 0$. Dès qu'une solution non-vide est obtenue, alors la valeur q_{min} de q est retenue comme le nombre minimal de mesures aberrantes, et l'ensemble solution est retourné. Le principal avantage de cette stratégie est d'éviter de renvoyer un ensemble vide en présence de mesures aberrante, tout en n'ayant pas besoin de surestimer la valeur de q utilisée pour le calcul. La stratégie de GOMNE est résumée dans l'algorithme 3.8.

Algorithme 3.8 GOMNE(in: $[\mathbf{x}]$, $\mathbf{f}$, $[\mathbf{y}]$; out: q , $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$)

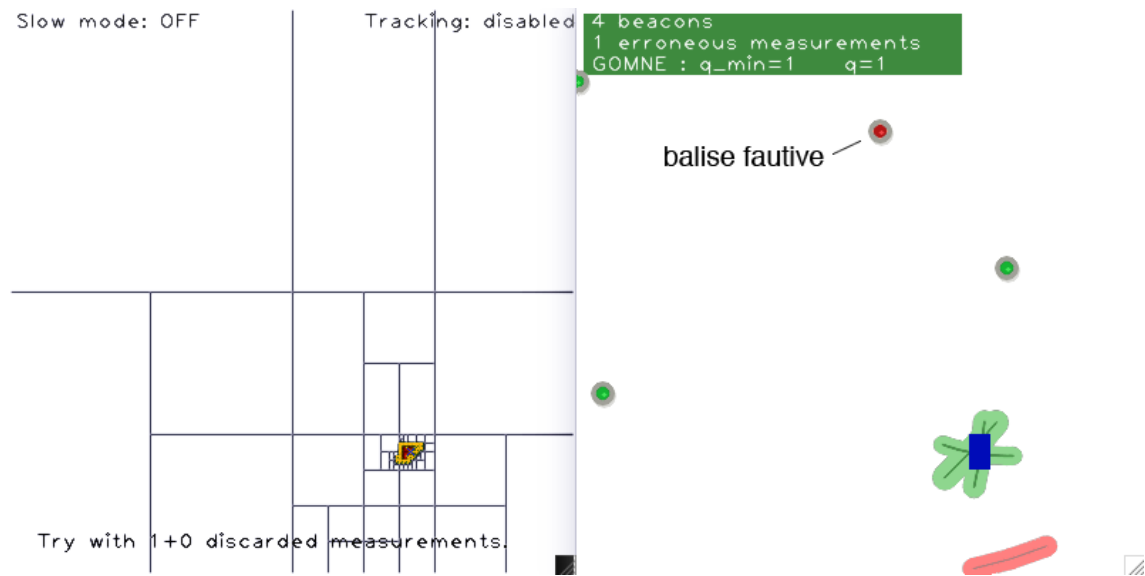
```

1:  $q := -1$ 
2: repeat
3:    $\underline{\mathbb{X}} := \emptyset$ ,  $\overline{\mathbb{X}} := \emptyset$ 
4:    $q := q + 1$ 
5:   RSIVIA( $[\mathbf{x}]$ ,  $\mathbf{f}$ ,  $[\mathbf{y}]$ ,  $q$ ,  $\underline{\mathbb{X}}$ ,  $\overline{\mathbb{X}}$ )
6: until  $\underline{\mathbb{X}} \neq \emptyset$ 

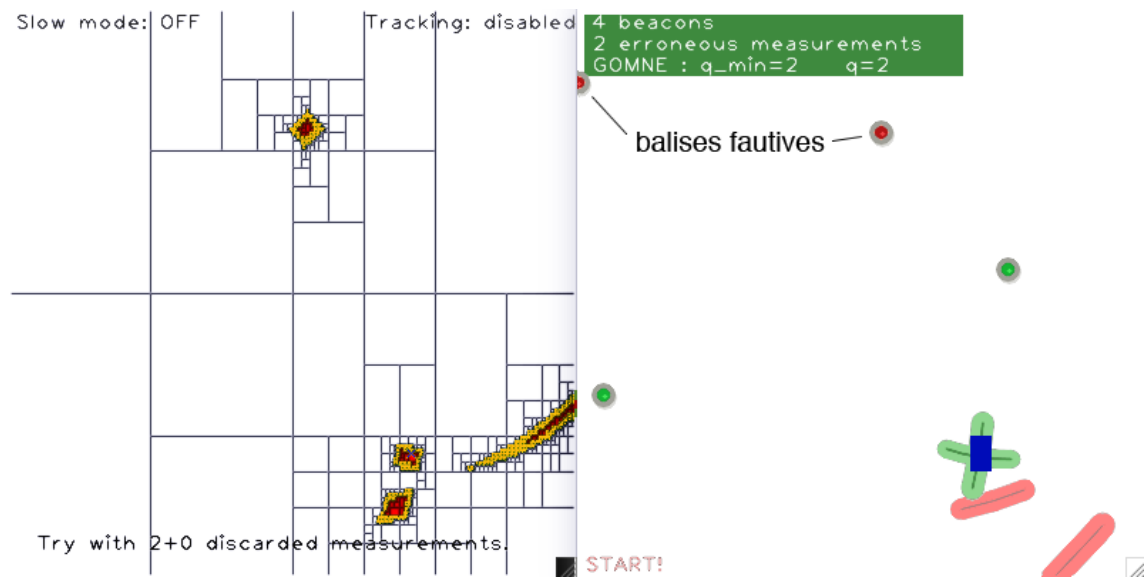
```

La figure 3.17 montre l'utilisation de GOMNE dans le problème de localisation à quatre balises, où les balises en rouge sont fautives et génèrent des observations aberrantes. Les arcs de cercle sur les vues de droite représentent les distances mesurées par les balises, afin de mettre en évidence les observations erronées. On observe ici que, grâce à la redondance des mesures, le nombre de mesures aberrantes est correctement estimé par GOMNE et que l'ensemble solution calculé est aussi consistant avec la vraie position.

1. $O(n \log n)$



(a) Une mesure aberrante. GOMNE retourne $q = 1$



(b) Deux mesures aberrantes. GOMNE retourne $q = 2$

Figure 3.17: Résultat de l'algorithme GOMNE (sans contraction) sur un problème de localisation à 4 balises, en présence de données aberrantes (données simulées)

Le critère d'arrêt de GOMNE est l'existence d'une solution non-vide. Cependant, les exemples (c) et (d) de la figure 3.12 montrent bien que l'existence d'une solution non-vide n'est pas la preuve que cette solution est valide. En effet, certaines mesures aberrantes peuvent ne pas être inconsistantes avec les autres mesures, et la redondance peut également être insuffisante pour la détection des fautes. Aussi, il faut bien remarquer que GOMNE estime le nombre minimal de mesures aberrantes, mais rien ne garantit qu'il n'y en a pas plus en réalité. Sur la figure 3.18a, une mesure aberrante est présente, mais elle n'est pas détectable à cause de sa trop faible amplitude. GOMNE estime ainsi le nombre minimal de mesures aberrantes à $q_{min} = 0$, et la solution calculée ne contient pas la vraie position. Ce problème de sous-estimation du nombre de mesures aberrantes arrive aussi lorsque le nombre de mesures aberrantes est élevé : plusieurs mesures aberrantes combinées avec une ou plusieurs mesures saine ont parfois une solution en commun.

Afin d'augmenter la robustesse de l'estimation dans le cas d'erreurs non-détectées, on peut améliorer la stratégie de GOMNE. Ainsi, après avoir déterminé le nombre minimal de mesures aberrantes q_{min} par l'exécution classique de GOMNE, on va calculer l'ensemble solution $(q_{min} + r)$ -relaxé, où r est une marge de sécurité prenant en compte d'éventuelles mesures aberrantes non-détectées [Jaulin et al., 2002]. Dans ce cas, la solution est garantie tant que l'hypothèse « il reste au plus r mesures aberrantes non détectées par GOMNE » est vraie. La figure 3.18b montre le résultat GOMNE sur l'exemple précédent, où la stratégie de base avait échoué, robustifié avec $r = 1$. On calcule ainsi une solution en tenant compte de la présence de $q_{min} + 1$ valeurs aberrantes. Cette fois-ci, la vraie position appartient bien au sous-pavage calculé.

3.5.3 Comparaison entre les méthodes d'estimation robustes et l'approche DIA

Les méthodes d'estimation robustes ainsi que l'approche DIA présentée dans la section 2.5 du chapitre précédent cherchent à estimer un vecteur de paramètres à partir de mesures, en présence de défauts. Leurs philosophies sont toutefois différentes, et on peut tenter d'y rattacher les deux méthodes d'inversion ensembliste robustes que nous venons de présenter.

Les méthodes d'estimation robuste cherchent la meilleure estimation d'un paramètre en favorisant dans le calcul les mesures qui sont compatibles avec le modèle. Les mesures aberrantes sont pondérées ou rejetées, en fonction de l'importance de l'erreur et de la méthode utilisée. On les utilise typiquement pour des applications où l'on dispose de beaucoup de données redondantes, parmi lesquelles se trouvent des données aberrantes. Les problèmes de vision, où les erreurs d'appariement sont fréquentes mais le nombre de points d'intérêt généralement important, sont souvent résolus à l'aide de méthodes d'estimation robuste. Lors de l'utilisation de ces méthodes, on fait une hypothèse sur le nombre attendu de mesures aberrantes parmi

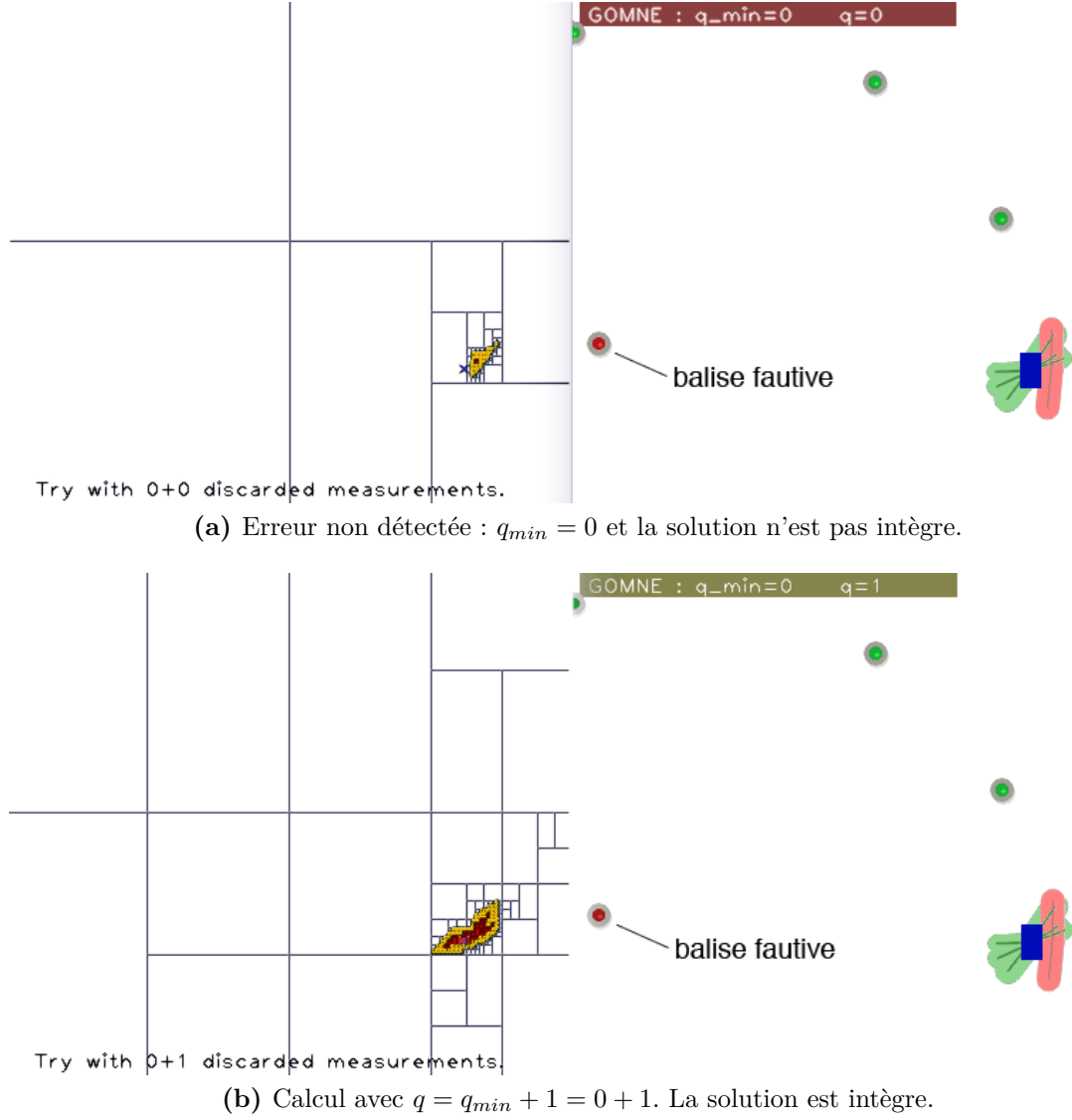


Figure 3.18: Résultat de l'algorithme GOMNE sur un problème de localisation à 4 balises, en présence d'une mesure aberrante non détectable

l'ensemble des valeurs mesurées. L'inversion ensembliste q -relaxée se rapproche de cette logique : on effectue une estimation ensembliste après avoir fait une hypothèse sur le nombre global de mesures aberrantes.

L'approche DIA se concentre quant à elle sur les défauts. C'est une approche de culture automatique, où l'on peut disposer d'un modèle dynamique, et où la quantité de données observées est souvent réduite. Les défauts possibles sont explicités, et on teste un nombre limité d'hypothèses de défaut dans la phase d'identification. La phase d'adaptation consiste à adapter le modèle pour coller aux données. Dans le cas le plus simple, l'adaptation peut se résumer à ignorer les mesures identifiées comme aberrantes, mais elle peut également consister à rajouter des paramètres à estimer dans le modèle. Le GOMNE se rapproche plus de cette logique, dans le sens où plusieurs hypothèses de nombre de défauts présents sont testées successivement, et le modèle est adapté en conséquence en incrémentant le nombre de mesures aberrantes tolérées dans l'inversion q -relaxée.

Qu'il s'agisse de l'inversion q -relaxée ou de GOMNE, les méthodes d'inversion ensembliste robuste permettent la détection et l'identification des défauts, comme nous allons le voir dans la section suivante.

3.6 Détection et identification de mesures aberrantes

La détection de mesures aberrantes consiste à déterminer s'il existe, parmi les mesures, des mesures incompatibles avec le modèle d'observation à erreurs bornées.

Dans le cas d'une inversion ensembliste non robuste ($q = 0$), la présence d'une mesure aberrante peut être détectée quand le sous-pavage solution est vide. On peut alors affirmer de façon garantie qu'au moins une mesure n'est pas consistante avec le modèle d'observation à erreurs bornées choisi.

Démonstration. Soit $\mathbf{x}$ la vraie valeur du paramètre vectoriel à estimer. Une mesure, dont l'incertitude est représentée par l'intervalle $[y_i]$, est consistante avec le modèle d'observation s'il existe un $y_i \in [y_i]$ tel que $y_i = f_i(\mathbf{x})$. Ainsi, si toutes les mesures respectent le modèle d'observation choisi, on a

$$\forall i \in \{1 \dots m\}, \exists y_i \in [y_i], f_i(\mathbf{x}) = y_i,$$

ou encore, en considérant le vecteur de mesures $[\mathbf{y}]$

$$\exists \mathbf{y} \in [\mathbf{y}], \mathbf{f}(\mathbf{x}) = \mathbf{y}.$$

L'ensemble solution $\mathbb{S}_{\mathbf{x}}$ est défini tel que $\mathbb{S}_{\mathbf{x}} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{f}(\mathbf{x}) \in [\mathbf{y}]\}$. On prouve ainsi que lorsque toutes les mesures respectent le modèle d'observation à erreurs bornées, l'ensemble solution est non-vidé. De plus, dans ce cas, l'ensemble solution contient la vraie valeur du paramètre $\mathbf{x}$.

Par contraposée, si l'ensemble solution est vide, alors au moins une mesure ne respecte pas le modèle d'observation à erreurs bornées. $\square$

De même, dans le cadre de l'inversion ensembliste robuste par intervalles, la présence de mesure aberrante est détectée lorsque le sous-pavage solution ne contient aucune boîte compatible avec toutes les mesures. On a alors la preuve qu'au moins une mesure n'est pas consistante avec le modèle d'observation à erreurs bornées.

L'inversion ensembliste robuste permet également de procéder à l'identification des mesures aberrantes, sachant qu'une identification est plus difficile à réaliser qu'une détection. En effet, une mesure aberrante est une mesure qui n'est pas compatible avec l'ensemble solution. Si aucune boîte du sous-pavage solution n'est compatible avec une mesure, alors cette mesure peut être identifiée comme mesure aberrante.

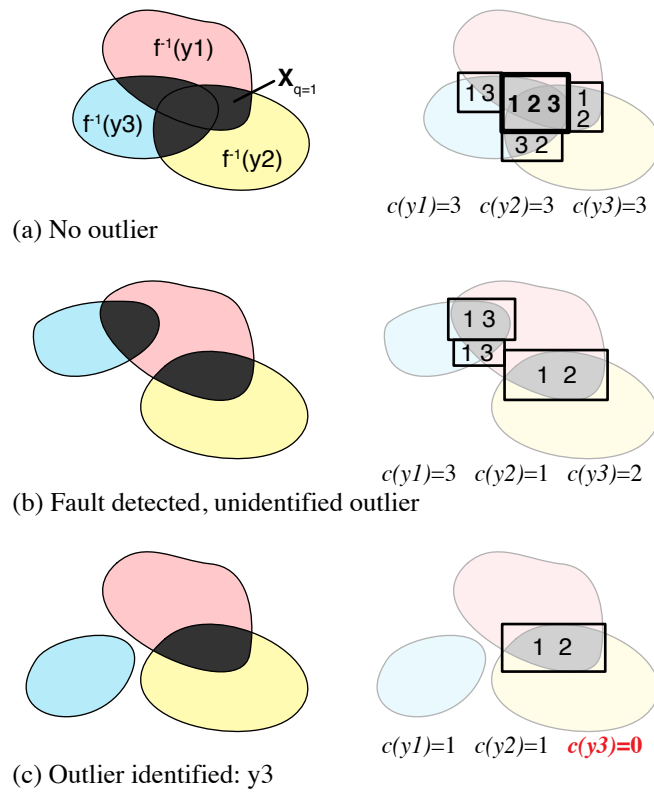


Figure 3.19: Détection et identification de mesure aberrante lors du calcul d'une solution 1-relaxée avec trois mesures. À gauche, les images réciproques des mesures dans l'espace d'état. À droite, les sous-pavages solutions. Sur chaque boîte est indiquée la liste des mesures compatibles. Les compteurs indiquent le nombre de boîtes compatibles avec chaque mesure.

Une méthode pour identifier rapidement les mesures aberrantes est d'associer un compteur à chaque observation (Figure 3.19) et de gérer ce compteur au cours du

traitement. Ce compteur va garder une trace du nombre de boîtes du sous-pavage solution compatibles avec l'observation concernée. La mise à jour des compteurs est effectuée au cours de l'inversion ensembliste, pour que le résultat d'identification ne nécessite pas une phase de post-traitement. Au début de l'inversion ensembliste, les compteurs sont initialisés à 0, puis l'inversion ensembliste robuste est effectuée. À chaque fois qu'une boîte est contractée ou testée, sa compatibilité avec chacune des observations est vérifiée. Si la boîte est compatible, le compteur associé à la mesure est incrémenté, sinon, le compte reste inchangé. Lorsque le calcul du sous-pavage solution est terminé, il suffit d'observer les compteurs dont la valeur est restée à zéro pour identifier d'éventuelles mesures aberrantes. L'algorithme (3.9) présente une implémentation possible du mécanisme de comptage en ligne du nombre de boîtes compatibles avec chaque mesure.

3.7 Conclusion

Dans ce chapitre, nous avons introduit des méthodes permettant d'estimer un paramètre à partir d'observations à erreurs bornées. Une méthode ensembliste robuste basée sur les intervalles, permettant de caractériser l'ensemble solution sous forme d'un sous-pavage a été présentée. Les parties de l'espace de recherche incompatibles avec les mesures sont exclues, pour ne conserver que les parties dont l'appartenance à la solution est prouvée ou indéterminée. On obtient ainsi un sous-pavage solution garanti d'inclure l'ensemble solution du problème. La méthode est robuste à la présence d'un nombre q donné de mesures aberrantes, par l'utilisation de l'intersection q -relaxée, mais on peut également mettre en place une stratégie d'estimation en ligne de ce nombre (sans toutefois pouvoir garantir que le résultat sera consistant avec la vérité). Un exemple simple de localisation, mettant en jeu un modèle non-linéaire, a servi d'illustration. Le problème de l'identification des mesures aberrantes a également été traité et nous avons présenté une méthode pour le faire en parallèle de l'estimation de l'ensemble solution.

Nous n'avons pas discuté de l'observation d'état d'un système dynamique (c'est-à-dire des problème d'estimation dans lesquels on dispose d'un modèle d'évolution comme par exemple une équation différentielle ordinaire). Nous reviendrons sur ce problème dans le chapitre 6 qui traitera du problème de l'estimation de la pose à partir d'historiques de données.

Dans le chapitre suivant, nous allons appliquer la méthode d'estimation robuste par intervalles que nous venons d'aborder au problème concret de la localisation par GPS, afin de calculer un domaine de position à risque quantifié.

Algorithm 3.9 RSIVIAExtCount(in : $[\mathbf{x}]$, $\{\mathcal{C}_i, i = 1 \dots m\}$, $\mathbf{f}$, $[\mathbf{y}]$, ε ; inout : $\bar{\mathbb{S}}_{\mathbf{x}}$)

```

1: push( $[\mathbf{x}]$ ,  $\mathfrak{L}$ )
2: {La boîte initiale n'est pas supposée incompatible avec les mesures}
3: for  $i = 1$  to  $m$  do
4:    $c_i = 1$ 
5: end for
6: while  $\mathfrak{L} \neq \emptyset$  do
7:    $[\mathbf{x}] = \text{pull}(\mathfrak{L})$ 
8:   {On retire une boîte de la liste}
9:   for  $i = 1$  to  $m$  do
10:     $c_i = c_i - 1$ 
11:   end for
12:   for  $i = 1$  to  $m$  do
13:     $[\mathbf{x}^i] = \mathcal{C}_i([\mathbf{x}])$ 
14:   end for
15:    $[\mathbf{x}] = \square \left( \bigcap_{i \in \{1, \dots, m\}}^{\{q\}} [\mathbf{x}^i] \right)$ 
16:   if  $[\mathbf{x}] \neq \emptyset$  then
17:     if  $w([\mathbf{x}]) < \varepsilon$  then
18:        $\bar{\mathbb{S}}_{\mathbf{x}} := \bar{\mathbb{S}}_{\mathbf{x}} \cup [\mathbf{x}]$ 
19:       {Mise à jour des compteurs de compatibilité}
20:       for  $i = 1$  to  $m$  do
21:         if  $[f_i]([\mathbf{x}]) \cap [y_i] \neq \emptyset$  then  $c_i = c_i + 1$ 
22:       end for
23:     else
24:        $([\mathbf{x}_1], [\mathbf{x}_2]) = \text{bisect}([\mathbf{x}])$ 
25:       push( $[\mathbf{x}_1]$ ,  $\mathfrak{L}$ ); push( $[\mathbf{x}_2]$ ,  $\mathfrak{L}$ )
26:       {Mise à jour des compteurs de compatibilité (2 sous-boîtes)}
27:       for  $i = 1$  to  $m$  do
28:         if  $[f_i]([\mathbf{x}]) \cap [y_i] \neq \emptyset$  then  $c_i = c_i + 2$ 
29:       end for
30:     end if
31:   end if
32: end while

```

Chapitre 4

Estimation à risque quantifié d'un domaine de localisation

4.1 Introduction

Les services de localisation deviennent incontournables dans de nombreux domaines, dont les applications telles que les aides à la conduite qui peuvent mettre en jeu la sécurité des personnes. Un service de localisation fournit une estimation de la position de l'utilisateur, mais elle n'est utilisable que si elle est associée à un critère quantifiant l'incertitude et la confiance que l'on peut accorder à cette information. Une mesure de l'incertitude de positionnement est utile pour la plupart des applications. En effet, pour un simple système de géomarkage de photos, il n'est pas pertinent de marquer une photo comme prise au *Palais Impérial de Compiègne* si l'incertitude de localisation est supérieure à la centaine de mètres. Il serait plus pertinent d'indiquer simplement que la photo a été prise à Compiègne (ce n'est malheureusement pas ce que font la plupart de ces systèmes). Pour un véhicule autonome, connaître la précision de la position est essentiel afin de savoir si une manœuvre est réalisable dans un espace donné sans risquer d'entrer en collision avec un obstacle. Comme la sécurité des personnes peut être mise en jeu, cette information devra de plus être accompagnée d'un moyen de garantir son intégrité.

Nous allons, dans ce chapitre, nous attacher à caractériser le domaine d'incertitude associé à la solution de positionnement. On peut ainsi voir la localisation comme un problème ensembliste consistant à déterminer, à partir des observations, la zone dans laquelle l'utilisateur peut se trouver avec un niveau de confiance associé. Les méthodes ensemblistes par intervalles présentées au chapitre précédent ont déjà été employées avec succès pour résoudre des problèmes de localisation dans le domaine de la robotique [Jaulin et al., 2002, Meizel et al., 2002, Kieffer et al., 2000, Lévêque, 1998]. Ces méthodes ont des propriétés intéressantes, dont la capacité à représenter une solution ambiguë par un ensemble non connexe. De plus, quand les erreurs de mesures sont bornées, les méthodes ensemblistes permettent le calcul de solutions garanties. En pratique, il est difficile de garantir les bornes d'erreurs des mesures sans être trop pessimiste. De plus, le bruit est souvent modélisé par une loi

normale, dont le support est infini. Le choix de bornes d'erreurs représente donc un risque pour l'intégrité de la solution, que nous allons quantifier.

Les systèmes de positionnement par satellites comme le GPS offrent une solution globale pour le calcul d'une position. Nous allons donc utiliser le GPS et calculer une zone de localisation compatible avec les mesures de pseudodistance reçues [Drevelle and Bonnifait, 2009b]. Contrairement aux méthodes par intervalles proposées par [Gning and Bonnifait, 2006] qui utilisaient les solutions de navigation calculées par le récepteur, nous allons traiter directement les mesures brutes. Cependant, l'utilisation de mesures réelles pose le problème des mesures aberrantes. Dans le cas du GPS, on peut se prémunir des défaillances du système en utilisant un système d'augmentation tel qu'EGNOS¹. Néanmoins, le milieu urbain est source d'erreurs *locales* qui ne peuvent pas être compensées par EGNOS ou une méthode différentielle : c'est en particulier le cas des multitrajets, qui peuvent grandement affecter la mesure. Il est donc nécessaire de tenir compte de ces mesures aberrantes, en mettant en place un mécanisme d'intégrité interne au récepteur (comme le RAIM vu au chapitre 2.5). On peut, à l'aide des méthodes ensemblistes robustes telles que RSIVIA (voir la section 3.5.1 page 74), détecter la présence d'un défaut, et caractériser une zone de localisation tenant compte de l'hypothèse d'un défaut non-détecté. Les signaux pouvant être bloqués par les constructions, il est parfois difficile d'avoir suffisamment de redondance pour appliquer une méthode de détection de défaut en milieu urbain. Afin de diminuer le nombre de pseudodistances nécessaires, l'ajout d'une contrainte d'altitude par le biais d'un modèle numérique de terrain [Li et al., 2005] permet de contraindre la solution sur le relief terrestre [Drevelle and Bonnifait, 2009a, Drevelle and Bonnifait, 2011a].

Dans ce chapitre, nous allons tout d'abord présenter les contraintes liées aux mesures de pseudodistance sur la position, puis nous étudierons comment contraindre l'altitude par un modèle numérique de terrain. Nous calculerons ensuite le risque associé à l'ensemble satisfaisant ces contraintes, en fonction du choix des intervalles d'erreurs, et nous proposerons une méthode pour choisir les bornes d'erreur en fonction d'un risque d'intégrité donné. Les contracteurs de pseudodistance et d'altitude seront mis en œuvre dans un algorithme d'inversion ensembliste robuste et testé sur des données réelles. Enfin, l'implémentation temps-réel de la méthode sera discutée.

4.2 Contrainte liée aux mesures de pseudodistances

4.2.1 Modèle d'observation

Comme nous l'avons vu dans le chapitre 2.3, le positionnement GPS à l'aide de pseudodistances est un problème à quatre inconnues. On cherche à déterminer la

1. Bien que sa mise en œuvre en ville nécessite l'utilisation d'un moyen de communication terrestre, les satellites géostationnaires étant fréquemment masqués par les bâtiments.

position (x, y, z) du récepteur dans un repère cartésien, ainsi que le décalage d'horloge du récepteur dt_u .

En considérant les pseudodistances ρ_i , le modèle d'observation des mesures de code est, sous sa forme la plus simple, le suivant :

$$\begin{pmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_m \end{pmatrix} = \begin{pmatrix} \sqrt{(x - x_1^s)^2 + (y - y_1^s)^2 + (z - z_1^s)^2} + cdt_u \\ \sqrt{(x - x_2^s)^2 + (y - y_2^s)^2 + (z - z_2^s)^2} + cdt_u \\ \vdots \\ \sqrt{(x - x_m^s)^2 + (y - y_m^s)^2 + (z - z_m^s)^2} + cdt_u \end{pmatrix}.$$

Ce modèle d'observation est applicable dans tout repère cartésien, notamment le repère ECEF lié au référentiel WGS84, mais également dans un repère ENU (*East, North, Up*) localement tangent à l'ellipsoïde de référence. Les pseudodistances mesurées ρ_i^m doivent être corrigées afin de compenser les effets relativistes ainsi que les retards de propagation causés par la traversée de l'ionosphère et de la troposphère. Ces corrections sont obtenues soit en utilisant un modèle standard d'atmosphère, soit en utilisant les informations transmises par EGNOS. Ainsi, à l'erreur de mesure commise par le récepteur s'ajoute l'imprécision des corrections appliquées sur les pseudodistances brutes. L'erreur commise sur les pseudodistances corrigées est caractérisée par l'URA² transmis dans les messages de navigation, dans le cas du GPS naturel, soit en calculant l'UDRE³ dans le cas où un système d'augmentation comme EGNOS est utilisé. Afin de tenir compte de cette imprécision, les mesures ρ_i vont être représentées par des intervalles.

De la même façon, les positions des satellites à l'instant d'émission du signal sont déterminées à partir des éphémérides diffusées dans les messages de navigation. On peut tenir compte de l'incertitude sur les positions des satellites en les représentant par des boîtes.

Le calcul de la position du récepteur consiste ainsi à caractériser l'ensemble $\mathbb{X}$ des positions et décalages d'horloges compatibles avec les m mesures et les positions des satellites, soit

$$\mathbb{X} = \left\{ (x, y, z, cdt_u) \in \mathbb{R}^4, \forall i = 1 \dots m, \exists \rho_i \in [\rho_i], \exists (x_i^s, y_i^s, z_i^s) \in [\mathbf{x}_i^s], \right. \\ \left. \rho_i = \sqrt{(x - x_i^s)^2 + (y - y_i^s)^2 + (z - z_i^s)^2} + cdt_u \right\}.$$

2. User Range Accuracy : indication de l'erreur de pseudodistance due à l'imprécision des paramètres d'horloge et d'éphémérides, transmise dans les messages de navigation GPS.

3. User Differential Range Error : indication de l'erreur résiduelle sur les pseudodistances, après correction par un SBAS. Elle est calculée à partir des informations transmises et des modèles de correction utilisés.

4.2.2 Contracteur par propagation de contraintes

Un contracteur pour la contrainte associée à une mesure de pseudodistance peut être obtenu en mettant en œuvre la propagation de contraintes. La contrainte de pseudodistance est décomposée en un arbre de contraintes élémentaires (Fig. 4.1), afin d'appliquer l'algorithme de *propagation-rétropropagation* (Alg. 4.1). Les contraintes sont propagées des feuilles vers la racine, puis rétropropagées vers les feuilles. Comme le graphe de contraintes d'une pseudodistance ne contient pas de cycle, la contraction est optimale [Benhamou et al., 1999].

Algorithme 4.1 pseudorange_contract($[\mathbf{x}], [\rho], [\mathbf{x}^s]$)

Contracteur d'une contrainte de pseudodistance par propagation-rétropropagation

<pre>// Propagation [i9] := [x] - [x^s] [i8] := square [i9] [i7] := [y] - [y^s] [i6] := square [i7] [i5] := [z] - [z^s] [i4] := [i6] + [i8] [i3] := square [i5] [i2] := [i3] + [i4] [i1] := $\sqrt{[i2]}$ [ρ] := [ρ] ∩ ([i1] + [d])</pre>	<pre>// Rétropropagation [d] := [d] ∩ ([ρ] - [i1]) [i1] := [i1] ∩ ([ρ] - [d]) [i2] := [i2] ∩ square [i1] [i3] := [i3] ∩ ([i2] - [i4]) [i4] := [i4] ∩ ([i2] - [i3]) [i5] := [i5] ∩ square⁻¹ [i3] [i6] := [i6] ∩ ([i4] - [i8]) [i7] := [i7] ∩ square⁻¹ [i6] [i8] := [i8] ∩ ([i4] - [i6]) [i9] := [i9] ∩ square⁻¹ [i8] [x] := [x] ∩ ([i5] + [x^s]) [y] := [y] ∩ ([i7] + [y^s]) [z] := [z] ∩ ([i9] + [z^s])</pre>
--	--

return [x]

Le positionnement par satellites consiste à satisfaire l'ensemble des contraintes de pseudodistance. Avec m satellites en vue, un contracteur basé sur la propagation de contraintes est donné dans l'algorithme 4.2. Les contracteurs associés à chaque pseudodistance sont successivement appliqués, jusqu'à obtention d'une boîte fixe. La contraction n'est en effet pas optimale dès le premier passage, en raison de cycles dans le graphe de contraintes, de même que pour l'exemple présenté dans le chapitre (3.3.5.2).

Algorithme 4.2

gps_contract(in : $[\mathbf{x}], [\rho]_1, \dots, [\rho]_m, [\mathbf{x}^s]_1, \dots, [\mathbf{x}^s]_m$)

```
1: for  $i := 1$  to  $m$  do
2:    $[\mathbf{x}] := [\mathbf{x}] \cap \text{pseudorange\_contract}([\mathbf{x}], [\rho]_i, [\mathbf{x}^s]_i)$ 
3: end for
4: return [x]
```

Compte tenu de la distance des satellites, se trouvant à au moins 20 000 km du

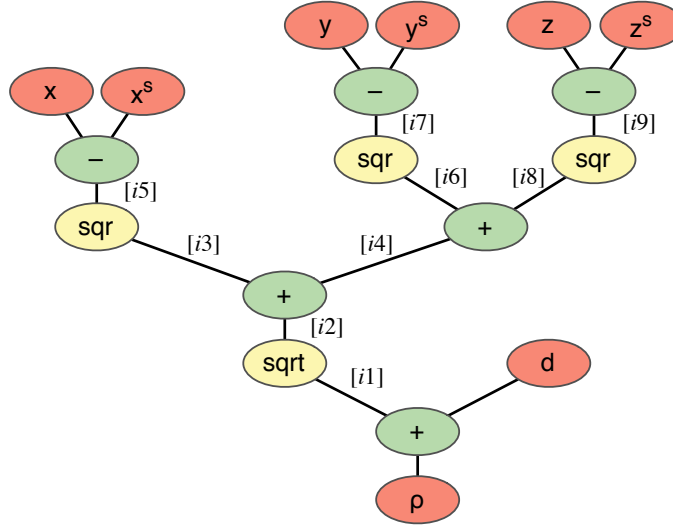


Figure 4.1: Décomposition en contraintes primitives de la contrainte liée à une pseudodistance

récepteur, le problème de géolocalisation par satellites est très bien approché par un système linéaire dans un rayon de plusieurs dizaines de kilomètres autour de la position du récepteur. Aussi l'utilisation du contracteur par linéarisation parallèle présenté au chapitre précédent semble tout à fait appropriée. Elle est détaillée dans l'annexe B. Le contracteur par linéarisation parallèle a l'avantage de tenir compte simultanément de toutes les contraintes. Cependant, nous allons par la suite considérer la présence éventuelle de mesures aberrantes parmi les pseudodistances mesurées. Aussi, comme la méthode d'inversion ensembliste robuste est basée sur l'exploitation séparée de chaque contrainte, l'utilisation du contracteur par linéarisation parallèle est beaucoup moins avantageuse et ce pour deux raisons : la propagation-rétropropagation de contraintes appliquée à une pseudodistance isolée donne une contraction optimale et les calculs sont relativement simples.

4.2.3 Robustesse

En milieu urbain, le risque de mesurer une pseudodistance erronée n'est pas négligeable. En effet, outre la défaillance d'un satellite, dont le risque est spécifié à 10^{-4} occurrences par heure, la mesure de pseudodistance peut être altérée par un trajet multiple du signal (mélange du signal direct et du signal réfléchi sur les obstacles). On peut également mesurer des signaux réfléchis sur les bâtiments, sans ligne de vue directe vers le satellite (Figure 4.2). On parle alors de NLoS (*Non Line of Sight*).

Dans ces conditions, le modèle d'observation supposant une ligne de vue dégagée ne s'applique plus à l'ensemble des mesures. Le problème de satisfaction de contraintes associé à la localisation risque de n'admettre aucune solution, ou de ne pas contenir la vraie position dans l'ensemble des solutions.

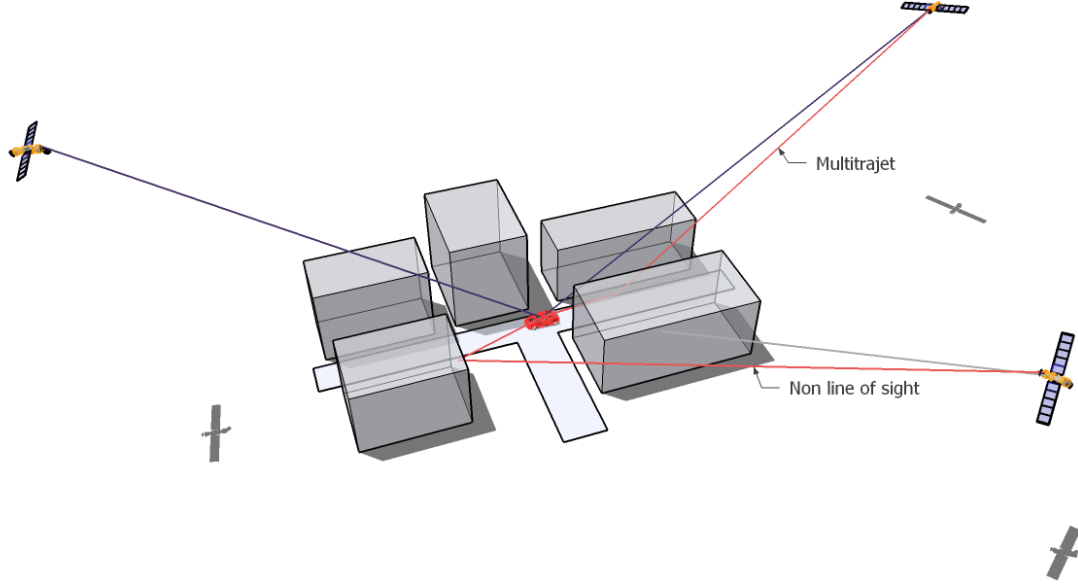


Figure 4.2: Multitrajets et propagation sans ligne de vue directe (NLoS) en environnement urbain

Afin d'introduire de la robustesse dans l'estimation de position, nous allons faire l'hypothèse de la présence d'un nombre maximum q de mesures aberrantes parmi les m pseudodistances reçues. Nous pouvons alors utiliser l'intersection q -relaxée présentée dans la section 3.5.1 page 74, afin de déterminer l'ensemble des positions compatibles avec au moins $m - q$ contraintes de pseudodistance. En construisant un contracteur q -relaxé avec le contracteur de pseudodistance par propagation de contraintes (Algorithme 4.2), on obtient le contracteur robuste présenté dans l'algorithme 4.3.

Algorithme 4.3 $\text{gps_contract_qrel}([\mathbf{x}], [\rho]_1, \dots, [\rho]_m, [\mathbf{x}^s]_1, \dots, [\mathbf{x}^s]_m, q)$

```

1: repeat
2:   for  $i := 1$  to  $m$  do
3:      $[\mathbf{x}_c](i) := \text{pseudorange\_contract}([\mathbf{x}], [\rho]_i, [\mathbf{x}^s]_i)$ 
4:   end for
5:    $[\mathbf{x}] := \square \left( \bigcap_{i \in \{1, \dots, m\}} \{q\} [\mathbf{x}_c](i) \right)$    enveloppe intervalle de l'intersection q-
                                                relaxée de m boîtes
6: until no more contraction can be done on  $[\mathbf{x}]$ 
7: return  $[\mathbf{x}]$ 
    
```

Le domaine de position est contracté indépendamment avec chaque pseudodistance. On obtient ainsi m boîtes contractées $[\mathbf{x}_c](i)$. On calcule alors l'intersection q -relaxée des m boîtes, afin de contracter le domaine de la position $[\mathbf{x}]$. La contraction est répétée jusqu'à ce qu'on aboutisse un point fixe du contracteur.

4.3 Contrainte d'altitude

Le calcul instantané d'une position GPS à l'aide de pseudodistances impose de recevoir les signaux d'au moins quatre satellites. Nous avons vu dans la section 2.5 page 30 que cinq satellites sont nécessaires pour détecter une faute, et six satellites sont requis pour tenter une identification de la mesure fautive. L'ajout d'une contrainte supplémentaire sur l'altitude permet de réduire d'un le nombre satellites requis pour les trois opérations précédentes. On peut ainsi se localiser à partir de 3 satellites en vue. Les récepteurs GPS aéronautiques peuvent ainsi être couplés au baromètre, afin d'améliorer la capacité de détection et d'identification de fautes. En effet, bien qu'un avion reçoive toujours assez de signaux pour se localiser et utiliser les mécanismes de FDE, la géométrie rend parfois certains satellites fautifs difficilement identifiables. L'information d'altitude apportée par le baromètre peut alors permettre d'améliorer les performances de la FDE, et en particulier de réduire le HPL.

Dans un contexte urbain, le nombre de satellites visibles peut se trouver très réduit en raison du masquage par les bâtiments. La contrainte d'altitude, en réduisant le nombre de satellites requis, va ainsi permettre d'augmenter la disponibilité du service de positionnement. Ainsi, la plupart des récepteurs GPS basculent automatiquement vers un mode de positionnement dit « 2D » quand le nombre de satellites est insuffisant pour le positionnement 3D, en faisant l'hypothèse que l'altitude reste constante.

4.3.1 Modèle numérique de terrain

Pour un véhicule terrestre, l'information d'altitude peut être facilement obtenue en exploitant un modèle numérique de terrain (MNT, ou *DEM*⁴ en anglais). Le MNT est une représentation numérique de l'altimétrie. La représentation la plus simple est sous forme de maillage carré ou rectangulaire, mais on trouve également des maillages hexagonaux, triangulaire réguliers et triangulaires quelconques.

Différents moyens peuvent être mis en œuvre pour créer un MNT. L'interférométrie radar, la photogrammétrie à partir de couples d'images aériennes ou satellitaires, ou encore la numérisation des courbes de niveau d'une carte existante permettent d'obtenir un maillage régulier. Le relevé topologique précis de points par GPS, triangulation ou télémètre laser permet d'obtenir un modèle maillé triangulaire quelconque.

La NASA fournit un MNT couvrant l'essentiel du globe, réalisé à partir de mesures par interférométrie radar acquises par la navette spatiale Endeavour en février 2000 lors de la « Shuttle Radar Topography Mission » (*SRTM*⁵). L'Institut Géographique

4. *Digital Elevation Model*

5. <http://www2.jpl.nasa.gov/srtm/>

National (IGN) fournit un MNT sur le territoire français à différents niveaux de précision, au sein de la BD ALTI et de la BD TOPO (Figure 4.3). Nous utiliserons ces deux modèles de terrain dans la suite du chapitre, afin d'étudier l'impact de la précision de l'information d'altitude.

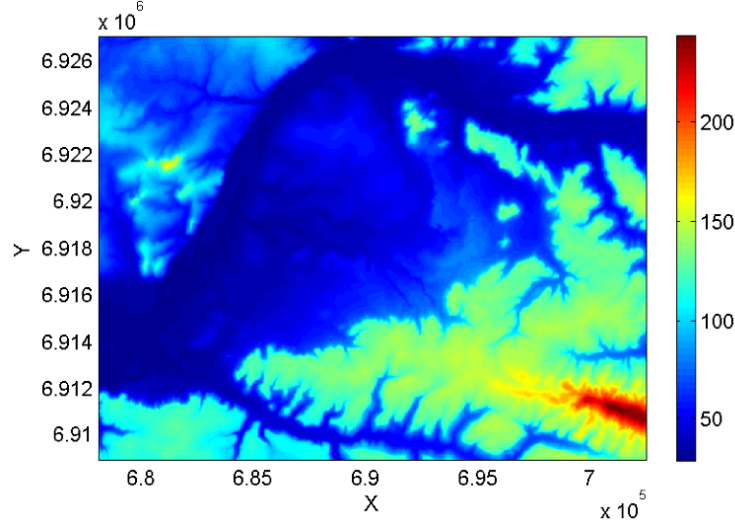


Figure 4.3: Modèle numérique de terrain des environs de Compiègne, IGN.

4.3.2 Contracteur d'altitude

Pour un modèle de terrain de précision planimétrique $\pm\Delta p$ et dont la précision en altimétrie est de $\pm\Delta h$, la contrainte d'altitude peut être appliquée à une boîte $[\mathbf{x}]$ par le contracteur $\mathcal{C}_{\text{MNT}}$ décrit dans l'algorithme 4.4. Pour appliquer ce contracteur,

Algorithme 4.4 $\mathcal{C}_{\text{MNT}}$ (inout : $[\mathbf{x}]$, in : $DEM, \Delta p, \Delta h$)

- 1: $[\mathbf{x}_{\text{DEM}}] = \text{ToDEMCoord}([\mathbf{x}])$
 - 2: $[\mathbf{wnd}] = \begin{pmatrix} [x_{\text{DEM}}] \\ [y_{\text{DEM}}] \end{pmatrix} + \begin{pmatrix} [-\Delta p, +\Delta p] \\ [-\Delta p, +\Delta p] \end{pmatrix}$
 - 3: $h_{\min} = \min_{[\mathbf{wnd}]}(DEM) - \Delta h$
 - 4: $h_{\max} = \max_{[\mathbf{wnd}]}(DEM) + \Delta h$
 - 5: $[\mathbf{x}_c] := \text{FromDEMCoord} \begin{pmatrix} [x_{\text{DEM}}] \\ [y_{\text{DEM}}] \\ [h_{\min}, h_{\max}] \end{pmatrix}$
 - 6: $[\mathbf{x}] := [\mathbf{x}] \cap [\mathbf{x}_c]$
-

la boîte en entrée est d'abord transformée dans le système de coordonnées du MNT. En effet, le repère de travail est un repère cartésien (repère global ECEF ou repère ENU local), mais le MNT est la plupart du temps défini dans un système de coordonnées géographiques si sa couverture est globale, ou dans une projection valide sur

la zone couverte. Les MNT de l'IGN sont ainsi fournis dans le système de projection conique Lambert 93. La boîte est alors contractée après une recherche de l'altitude maximale et minimale sur son emprise. Les incertitudes en altimétrie et planimétrie du MNT sont prises en compte lors de cette recherche : la zone de recherche est dilatée par l'incertitude horizontale, tandis que l'intervalle d'altitude est gonflé par l'incertitude altimétrique du MNT. La dernière étape consiste à transformer la boîte contractée dans le système de coordonnées du MNT pour revenir au repère de travail. La figure 4.4 présente les étapes de la contraction d'un domaine de position par un modèle numérique de terrain. L'effet d'enveloppe introduit du pessimisme, en particulier lors des changements de systèmes de coordonnées. On obtient dans l'exemple de la figure 4.4 une contraction nulle, alors qu'une légère contraction était possible.

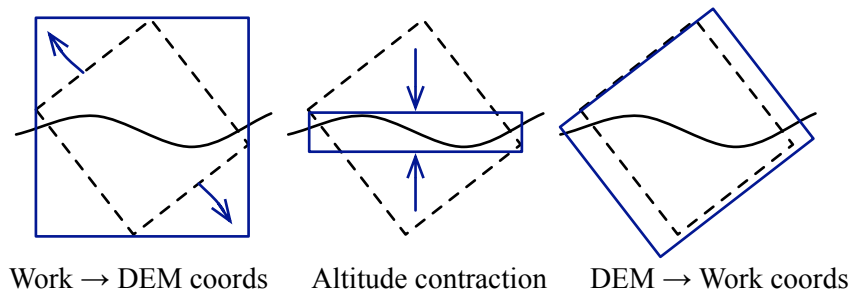


Figure 4.4: Étapes de la contraction d'une boîte de position par un MNT, et influence de l'effet d'enveloppe. *Vue en coupe.*

Pour que le contracteur d'altitude soit efficace, le repère de travail doit être choisi avec soin. En effet, si un des axes du repère de travail est orthogonal au plan du sol, la contraction par le modèle de terrain sera plus importante que si le repère de travail est orienté arbitrairement. La figure 4.5 présente la différence d'efficacité de la contrainte imposée par le MNT en fonction du système de coordonnées choisi. On remarque que, par nos latitudes, la contraction optimale de coordonnées ECEF est très peu efficace. On a ainsi avantage à se placer dans le repère associé à un plan localement tangent à la topologie du terrain ou au géoïde, l'axe z étant ainsi fortement contracté par le modèle d'altitude. En pratique, on utilisera plutôt un repère ENU local tangent à ellipsoïde de référence, afin de simplifier le calcul des changements de repères.

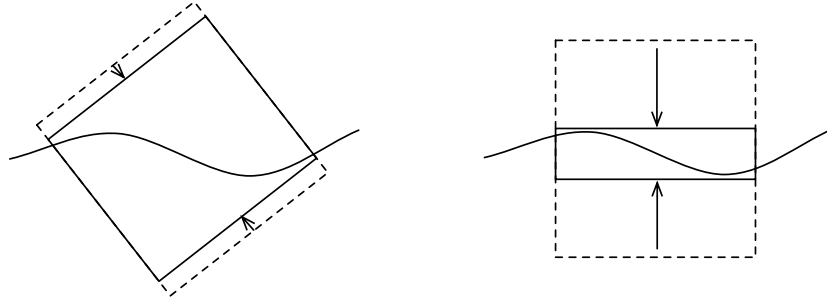


Figure 4.5: Contraction optimale par un modèle de terrain en fonction du système de coordonnées choisi. À gauche, coordonnées ECEF; à droite, coordonnées ENU dans un plan localement tangent à l'ellipsoïde.

4.4 Calcul du risque associé à la zone d'incertitude et choix des bornes

4.4.1 Principe

Les méthodes classiques d'estimation (moindres carrés pondérés, filtrage de Kalman) permettent de déterminer une estimée de la solution ponctuelle de localisation, et de propager la variance des erreurs de mesure pour obtenir une matrice de variance-covariance de l'erreur d'estimation. Le choix d'une borne sur l'erreur de position commise se fait alors à partir de la distribution d'erreur estimée. On en déduit, dans le cas gaussien, des ellipses ou ellipsoïdes d'incertitude.

Dans le cas des méthodes à erreurs bornées, quand l'erreur de mesure n'est pas à support borné, le choix des bornes associées à chaque mesure s'accompagne d'un risque. Les bornes d'erreur vont être choisies en étant associées au risque de ne pas inclure l'erreur vraie. Une qualité majeure des méthodes par intervalles utilisées dans la résolution du problème de localisation ensembliste est qu'elles n'ajoutent aucun risque. En effet, on a la garantie de ne pas perdre de solutions au cours des calculs. Ainsi, le risque que la solution obtenue ne contienne pas la vraie position ne dépend que du risque que les intervalles choisis pour les mesures ne contiennent pas la vraie valeur.

Nous allons, dans cette section déterminer le risque associé *a priori* à la solution de localisation calculée, et proposer une méthode de choix des bornes d'erreur de mesure pour effectuer une estimation à risque donné.

4.4.2 Calcul du risque associé au domaine de localisation

Dans un cadre probabiliste, chaque mesure à erreur bornée peut être considérée comme une expérience de Bernoulli : « Est-ce que la vraie valeur est dans l'intervalle de mesure ? »

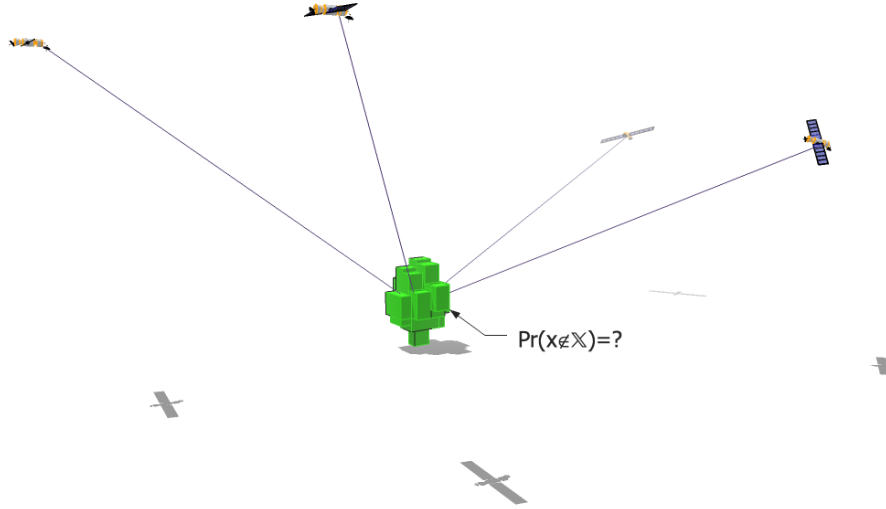


Figure 4.6: Risque associé au sous-pavage solution

Soit $y_{meas} = y + e_y$ une mesure entachée d'une erreur e_y . En choisissant les bornes d'erreur $[\underline{e}_y, \overline{e}_y]$, nous pouvons représenter la mesure par l'intervalle $[y_{meas}] = [y_{meas} + \underline{e}_y, y_{meas} + \overline{e}_y]$. Étant donnée la densité de probabilité f_{e_y} de l'erreur de mesure, la probabilité $p = \Pr(y \in [y_{meas}])$ que la vraie valeur soit dans l'intervalle de mesure est donnée par

$$p = \Pr(y \in [y_{meas}]) = \int_{\underline{e}_y}^{\overline{e}_y} f_{e_y}(\alpha) d\alpha. \quad (4.1)$$

Considérons désormais m mesures. On suppose que les erreurs de mesure sont indépendantes, et que la probabilité de contenir la vraie valeur est la même pour chaque intervalle de mesure et vaut p . Soit n_{ok} le nombre de mesures consistantes avec le modèle à erreurs bornées (on a donc $m - n_{ok}$ valeurs aberrantes ou outliers). La probabilité que tous les intervalles de mesure soient consistants avec les vraies valeurs est

$$\Pr(n_{ok} = m) = p^m. \quad (4.2)$$

Le calcul ensembliste robuste permet de tolérer la présence de mesures aberrantes en conservant une solution consistante avec la vraie position [Jaulin et al., 2002]. Si le calcul est robuste à q mesures fausses, le calcul est garanti si au moins $m - q$ intervalles de mesure sont consistants avec la vraie valeur. Pour calculer la probabilité que cette hypothèse soit vérifiée, déterminons tout d'abord la probabilité $\Pr(n_{ok} = k)$ d'avoir exactement k bonnes mesures. Le nombre de mesures consistantes avec le

modèle d'erreurs bornées suit une loi binomiale : $n_{ok} \sim B(m, p)$. On a donc

$$\Pr(n_{ok} = k) = \binom{m}{k} p^k (1-p)^{m-k}, \quad (4.3)$$

avec

$$\binom{m}{k} = \frac{m!}{k! (m-k)!}. \quad (4.4)$$

En sommant l'équation (4.3) sur les valeurs successives de k , on déduit la probabilité d'avoir au moins $m - q$ bons intervalles de mesure :

$$\Pr(n_{ok} \geq m - q) = \sum_{k=m-q}^m \binom{m}{k} p^k (1-p)^{m-k}. \quad (4.5)$$

L'algorithme d'inversion ensembliste robuste (Algorithme 3.7 page 77) calcule une approximation garantie $\bar{\mathbb{X}}$ de l'ensemble solution $\mathbb{X}$, telle que $\mathbb{X} \subset \bar{\mathbb{X}}$. De plus, si les hypothèses faites sur le nombre de mesures aberrantes sont respectées, l'ensemble solution calculé est garanti de contenir la vraie valeur. Ainsi, on a

$$n_{ok} \geq m - q \implies x \in \mathbb{X} \implies x \in \bar{\mathbb{X}}, \quad (4.6)$$

et on peut poser une borne inférieure sur la probabilité que l'ensemble solution calculé soit consistant avec la vérité

$$\Pr(x \in \bar{\mathbb{X}}) \geq \Pr(x \in \mathbb{X}) \geq \Pr(n_{ok} \geq m - q). \quad (4.7)$$

De manière équivalente, le risque pris en formulant les hypothèses sur les intervalles de mesures est une borne supérieure du risque r que l'ensemble solution ne contienne pas la vraie valeur de la position :

$$r = \Pr(x \notin \bar{\mathbb{X}}) \leq \Pr(x \notin \mathbb{X}) \leq 1 - \Pr(n_{ok} \geq m - q). \quad (4.8)$$

En réintroduisant l'équation (4.5), on obtient donc

$$r \leq 1 - \sum_{k=m-q}^m \binom{m}{k} p^k (1-p)^{m-k}. \quad (4.9)$$

4.4.3 Choix des bornes pour un risque donné

Dans la pratique, les exigences concernant le risque de solution non intègre sont souvent spécifiés, via les RNP par exemple dans l'aéronautique. Nous allons donc suivre la démarche inverse, consistant à déterminer les bornes des intervalles d'erreurs assignés aux mesures afin de satisfaire une spécification du risque maximal

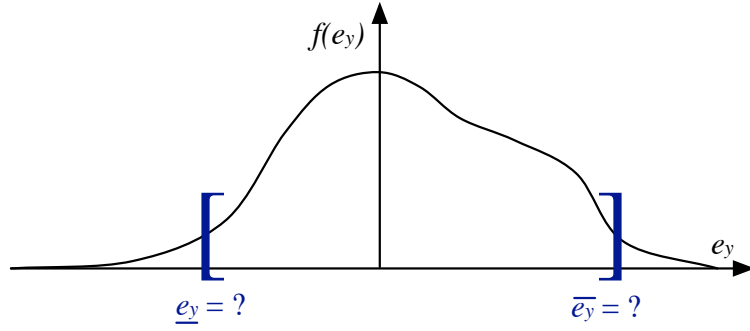


Figure 4.7: Problème du choix des bornes d'erreur

(Figure 4.7). Nous avons vu précédemment que le risque de ne pas respecter les hypothèses d'erreurs bornées et de nombre maximal de mesures aberrantes est une borne supérieure du risque encouru sur la solution. D'après les équations (4.5) et (4.8), le risque maximal est

$$\bar{r} = 1 - \sum_{k=m-q}^m \binom{m}{k} p^k (1-p)^{m-k}. \quad (4.10)$$

Pour un risque spécifié, on peut inverser cette équation afin de déterminer la confiance p requise en chaque mesure en fonction du nombre total de mesures m et du nombre de mesures aberrantes tolérées q . Un calcul par dichotomie permet d'obtenir rapidement des valeurs de p pour différents couples (m, q) , en considérant par exemple un risque maximal $\bar{r} = 10^{-7}$ (Figure 4.8). Un détail de quelques valeurs numériques est présenté dans le tableau 4.1.

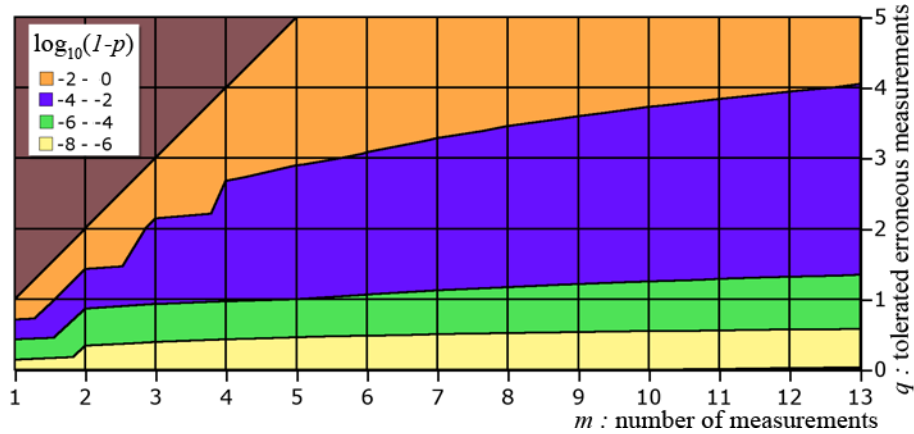


Figure 4.8: Logarithme décimal du risque $(1-p)$ lié à chaque mesure pour que le risque sur la solution soit de 10^{-7} .

Les résultats du tableau 4.1 mettent en avant deux faits survenant lorsqu'on souhaite garantir un risque constant sur la solution. Plus le nombre de mesures considérées est

Table 4.1: Calcul de $(1 - p)$ sur chaque intervalle de mesure pour obtenir un risque global de 10^{-7} avec m mesures et q mesures aberrantes tolérées.

	$m = 4$	$m = 5$	$m = 6$	$m = 7$
$q = 0$	$2.5 \cdot 10^{-8}$	$2 \cdot 10^{-8}$	$1.66 \cdot 10^{-8}$	$1.42 \cdot 10^{-8}$
$q = 1$	* $1.29 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$	$8.16 \cdot 10^{-5}$	$6.90 \cdot 10^{-5}$
$q = 2$	* $2.93 \cdot 10^{-3}$	* $2.16 \cdot 10^{-3}$	$2.71 \cdot 10^{-3}$	$1.42 \cdot 10^{-3}$

important, plus la confiance associée à chaque intervalle de mesure doit être grande. En effet, en augmentant le nombre de mesures à erreurs bornées, on augmente le risque qu'un des intervalles ne contienne pas la vraie valeur. La deuxième observation est que l'on peut grandement réduire la confiance requise dans chaque mesure, en tolérant une mesure aberrante supplémentaire dans le calcul. On voit ainsi que le risque liée à chaque mesure doit être inférieur au risque attendu sur la solution dans le cas non-robuste, mais qu'en tolérant une mesure aberrante, le risque sur chaque mesure peut être de trois ordres de grandeur supérieur au risque spécifié sur la solution (cas $m = 5$).

Les résultats précédés d'un astérisque dans le tableau 4.1 correspondent à des cas sous-déterminés pour la localisation à l'aide des pseudodistances GPS. Cependant, les méthodes ensemblistes permettent de calculer l'ensemble des solutions compatibles avec les mesures. Aussi, même si les solutions au problème de localisation sous-déterminé sont infinies, l'algorithme d'inversion ensembliste permet de caractériser cet ensemble (à l'intérieur de la boîte initiale) en particulier en identifiant toutes les zones de l'espace solution qui sont incompatibles avec les contraintes. On a donc bien apport d'information sur la position du récepteur, associée au risque spécifié. Cette information peut éventuellement être fusionnée avec d'autres contraintes, comme un modèle numérique de terrain.

Une fois déterminée la probabilité p qu'un intervalle de mesure contienne la vraie grandeur, les bornes d'erreurs peuvent être choisies en tenant compte de la fonction de répartition de l'erreur de mesure. Ce choix n'est pas unique, aussi, on choisira de préférence le plus petit intervalle supportant la probabilité p . Dans le cas d'une erreur de mesure Gaussienne centrée d'écart type σ_y , telle que $e_y \sim \mathcal{N}(0, \sigma_y^2)$, et si l'on note Φ la fonction de répartition de la loi normale centrée réduite, alors l'intervalle de mesure $[y_{meas}]$ peut être déterminé par

$$[y_{meas}] = [y_{meas} - \alpha\sigma_y, y_{meas} + \alpha\sigma_y], \quad (4.11)$$

avec

$$\alpha = -\Phi^{-1}\left(\frac{1-p}{2}\right). \quad (4.12)$$

La probabilité que l'erreur sorte de l'intervalle ainsi déterminé est ainsi également distribuée sur les deux queues de la gaussienne (Figure 4.9). Le tableau 4.2 liste les

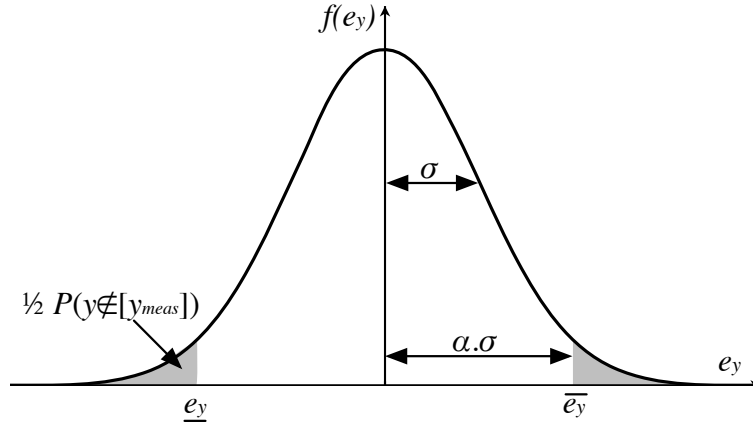


Figure 4.9: Répartition du risque lors du choix des bornes à $\pm\alpha\sigma$ pour une erreur de mesure gaussienne.

Table 4.2: Valeurs de α pour obtenir un risque de 10^{-7} avec m mesures et q mesures aberrantes tolérées. On considère des mesures gaussiennes indépendantes et des bornes d'erreur à $\pm\alpha\sigma$.

	$m = 4$	$m = 5$	$m = 6$	$m = 7$
$q = 0$	5.57	5.61	5.64	5.67
$q = 1$	3.83	3.89	3.94	3.98
$q = 2$	2.98	3.07	3.14	3.19

valeurs du coefficient α calculé pour différentes combinaisons (m, q) afin obtenir un risque de 10^{-7} sur l'ensemble solution, dans le cas d'erreurs de mesures gaussiennes et indépendantes. De même que nous l'avons vu avec le tableau 4.1, il est nécessaire d'agrandir les intervalles d'erreurs quand le nombre de mesures augmente. Cependant, pour des niveaux de risque faibles, l'agrandissement des bornes est modéré par la décroissance exponentielle des queues de la gaussienne. La tolérance de plus de valeurs aberrantes permet de diminuer significativement la largeur des intervalles d'erreur.

Dans le cadre d'une application réelle, ce type de tableau peut être pré-calculé hors-ligne pour le niveau de risque requis par l'application. Le choix des bornes se réduit alors à une requête dans le tableau en fonction du nombre de mesures présentes et une multiplication par l'écart type de l'erreur des mesures reçues.

4.5 Mise en œuvre et comparaison des résultats avec une approche standard

4.5.1 Estimation non contrainte

La méthode ensembliste de localisation présentée dans ce chapitre a été testée en temps réel sur un jeu de données GPS brutes acquises à proximité du laboratoire, avec un récepteur *Septentrio PolaRx* sur le véhicule STRADA. Une présentation détaillée de ces données est fournie en Annexe C.1.

Le système d'augmentation EGNOS est utilisé afin de corriger les pseudodistances mesurées et de déterminer la variance d'une Gaussienne englobante de l'erreur résiduelle des pseudodistances corrigées. Sur la figure 4.10 sont tracées les boîtes englobantes des ensembles solutions calculés à chaque instant de mesure avec la méthode non robuste. L'algorithme SIVIA a été utilisé, avec le contracteur de pseudodistances par propagation de contraintes.

En utilisant une méthode d'inversion ensembliste sans relaxation de contraintes, une inconsistance entre l'ensemble de mesures et le modèle d'observation peut être détectée quand l'ensemble solution est vide. Cependant, l'erreur de mesure peut être trop faible pour être détectée, tout en étant hors de l'intervalle d'erreur choisi. Dans ce cas, la zone de localisation calculée peut ne pas contenir la vraie position. Pour mettre en évidence ce phénomène, nous avons artificiellement appliqué une erreur de type « rampe » sur une pseudodistance du jeu de données. L'erreur ajoutée sur cette pseudodistance croît ainsi linéairement avec le temps, simulant la défaillance du satellite numéro 1 par dérive de l'horloge. Les bornes de la solution calculée sont représentées en fonction du biais simulé par les courbes bleues (en gras) de la figure 4.11. Nous pouvons mettre trois zones de fonctionnement en évidence. Tout d'abord (a), la solution non-robuste reste consistante avec la vraie position, jusqu'à un biais de 18 mètres. Quand l'erreur dépasse 23 mètres, l'ensemble solution est

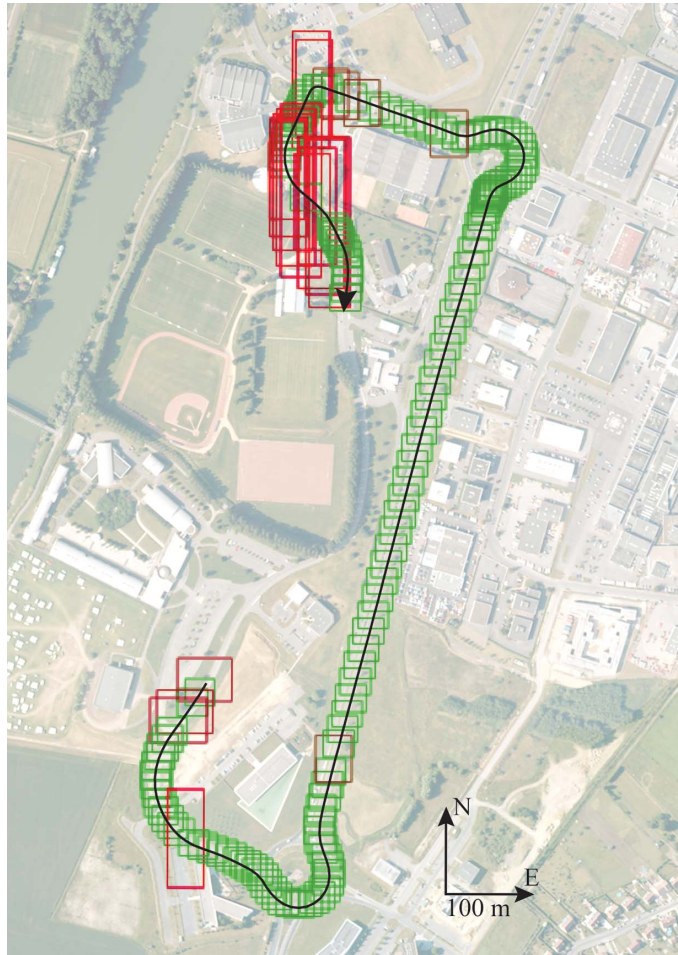


Figure 4.10: Vue d'ensemble des résultats. *Les rectangles représentent les boîtes englobantes des ensembles solutions calculés (solveur non robuste). Ils sont colorés en fonction du volume de la solution. La trajectoire de référence est en noir.*

vide (c). Dans ce cas, la présence d'au moins une mesure aberrante est prouvée. Cependant, pour un biais compris entre 18 et 23 mètres (b), bien que non-vide, l'ensemble solution ne contient plus la vraie position. Le système fournit alors une position erronée.

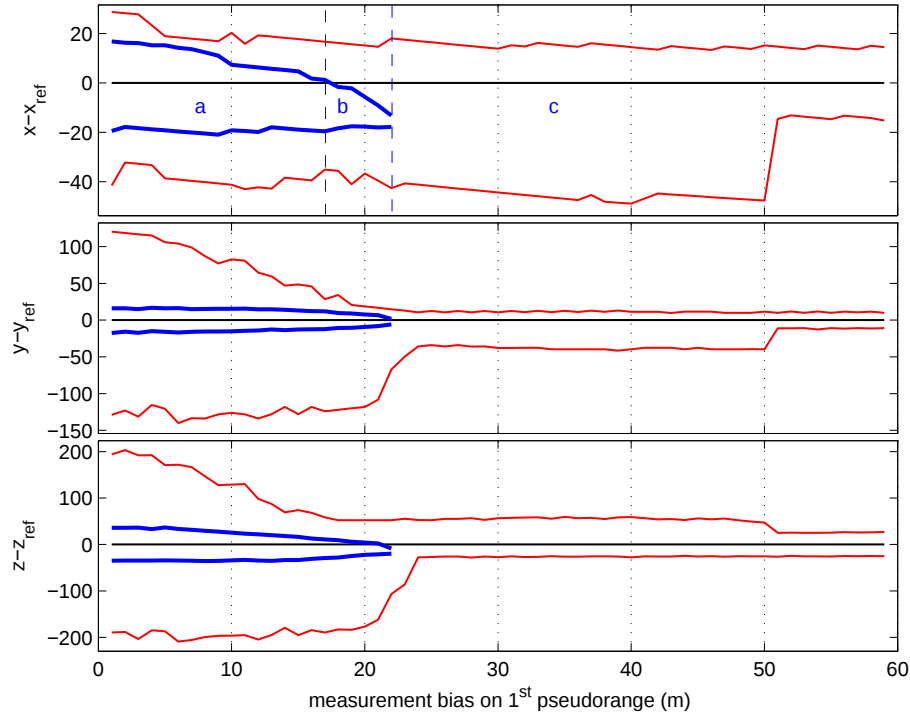


Figure 4.11: Bornes du domaine de localisation, relativement à la vraie position. Une erreur de type rampe est simulée sur le satellite n°1. L'ordonnée zéro correspond à la vraie position. Les résultats de l'inversion ensembliste non robuste sont représentés par les courbes en traits bleus épais. Les courbes rouges correspondent au solveur 1-relaxé.

En utilisant un solveur robuste calculant l'intersection 1-relaxée des contraintes de pseudodistances, la présence d'une mesure aberrante ne compromet pas l'intégrité de l'ensemble solution. En effet, tant qu'une seule mesure ne respecte pas le modèle d'erreurs bornées, l'ensemble solution est garanti de contenir la vraie position. Les courbes rouges (fines) de la figure 4.11 représentent l'évolution des bornes de l'ensemble solution 1-relaxé en fonction du biais rajouté à une pseudodistance du satellite numéro 1. On peut décomposer l'évolution de l'ensemble solution en trois phases illustrées par la figure 4.12. Tant que l'erreur sur la pseudodistance fautive est inférieure à 19 mètres, l'ensemble solution est composé de l'union des solutions à cinq satellites (Figure 4.12a). Ces « sous-solutions » sont toutes non-vides, et le sous-pavage solution contient des pavés présumés compatibles avec les six pseudodistances. L'ensemble solution se réduit à mesure que l'inconsistance entre les mesures s'accroît. À partir de 19 mètres de biais ajouté à la première pseudodistance, aucune

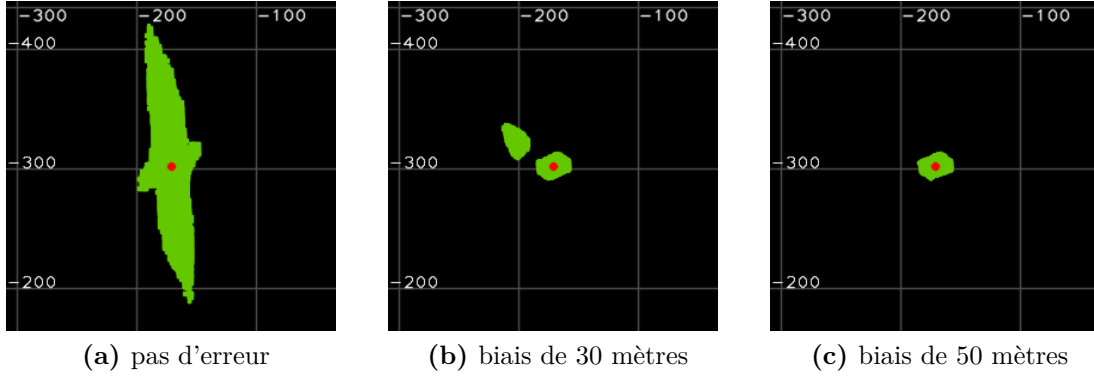


Figure 4.12: Projection sur le plan horizontal de l'ensemble solution 1-relaxé calculé en présence d'un biais sur le satellite n°1. *Le point rouge correspond à la vraie position.*

boîte du sous-pavage solution n'est compatible avec l'ensemble des six contraintes. La présence d'une mesure aberrante est donc détectée. L'ensemble solution est ainsi l'union de « sous-solutions » disjointes à 5 satellites. On peut qualifier le positionnement d'ambigu, car il n'est pas possible de déterminer dans quelle sous-solution se trouve la vraie position. Dans le cas de la figure 4.12b, deux sous-ensembles de 5 satellites conduisent à une solution non vide. On ne peut donc pas déterminer quelle est la mesure aberrante. Enfin, quand l'erreur de pseudodistance dépasse les 50 mètres, une seule solution à 5 satellites est non vide. La solution 1-relaxée correspond ainsi à la solution où la mesure aberrante a été rejetée (Figure 4.12c). On peut dès lors identifier la mesure fautive, comme la mesure correspondant à la contrainte qui n'est satisfaite par aucune boîte du sous-pavage solution.

La figure 4.13 page suivante montre les résultats de détection et d'identification de faute après avoir successivement simulé une erreur de type rampe sur chacun des 6 satellites à l'époque $t = 10$ s. Elle représente, pour chaque satellite, la plus petite erreur ayant pu être détectée, et la plus petite erreur ayant permis l'identification de la mesure corrompue. On remarque une grande disparité dans les erreurs nécessaires à la détection et à l'exclusion de fautes en fonction des satellites considérés. Celle-ci est due à la géométrie du problème. Nous avons comparé les erreurs minimales détectées aux MDBs des méthodes classiques, définis au chapitre 2.5.2. Pour permettre une comparaison de l'inversion ensembliste robuste avec les MDB calculés selon la formule classique de l'équation (2.13), le choix des bornes d'erreurs pour ce calcul a été réalisé de façon à obtenir une probabilité de fausse alarme de $5 \cdot 10^{-5}$ et une probabilité de non-détection de 10^{-3} .

En raison de la géométrie du problème, le MDB varie très significativement selon les satellites. Alors qu'on peut détecter des erreurs d'une quinzaine mètres sur les pseudodistances de trois satellites de la constellation, on remarque que pour le satellite 23, qui est particulièrement mal conditionné pour la détection d'erreur, seule une erreur de mesure de plus de cent mètres peut être détectée. Les performances

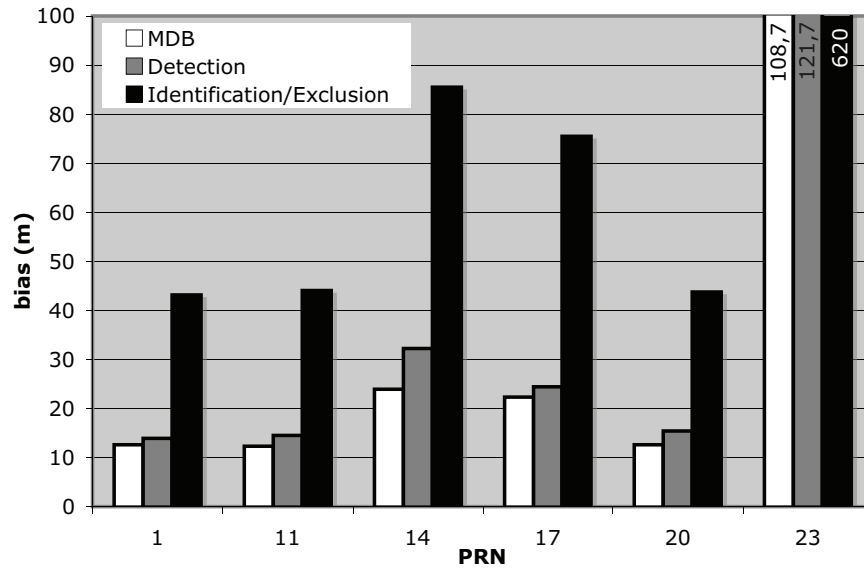


Figure 4.13: Seuils de détection et d'exclusion de faute sur chaque satellite avec l'inversion ensembliste 1-relaxée à $t = 10$ s. Le MDB est donné pour référence. Les valeurs pour le satellite 23 ont été tronquées.

observées de détection de fautes de la méthode d'inversion ensembliste par intervalles sur cet exemple sont parfaitement cohérentes avec les valeurs calculées pour le MDB.

L'intersection 1-relaxée des contraintes liées aux pseudodistances permet ainsi d'introduire de la robustesse aux mesures aberrantes, tout en rendant possible leur détection et leur identification. Néanmoins, la contrepartie de cette robustesse est l'augmentation du volume de l'ensemble solution, en particulier lorsque la redondance des mesures est faible. La figure 4.14 montre ainsi les ensembles solutions calculés pour les mêmes 6 pseudodistances et avec le même risque (sous hypothèse d'erreurs de mesure Gaussiennes et indépendantes), l'un avec le solveur non robuste, l'autre avec le solveur 1-relaxé. Le volume de la solution robuste est beaucoup plus grand. En effet, la solution 1-relaxée correspond à l'union des solutions correspondant aux six sous-ensembles de 5 satellites. Or, une de ces combinaisons de cinq satellites se trouve dans une configuration géométrique défavorable, qui entraîne une grande incertitude sur la localisation.

4.5.2 Ajout d'une contrainte d'altitude

Pour compenser la relaxation d'une contrainte de pseudodistance, nous introduisons une contrainte sur l'altitude. L'ajout d'une telle contrainte, provenant par exemple d'un baromètre dans le cas aéronautique, permet, outre la réduction du nombre de pseudodistances nécessaires à la localisation, d'améliorer le positionnement robuste en diminuant les MDB [Gutmann et al., 2009]. Comme nous considérons des

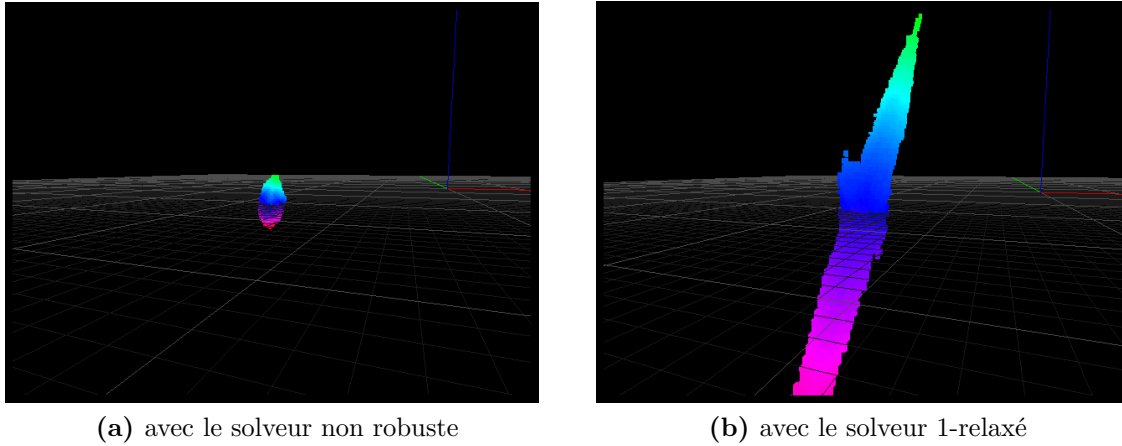


Figure 4.14: Vue 3D^{1/2} des sous-pavages solutions du problème de localisation avec 6 pseudodistances. Les couleurs représentent le décalage d’horloge.

véhicules terrestres, nous allons utiliser un modèle numérique de terrain afin de contraindre l’altitude [Li et al., 2005]. Le MNT utilisé dans cette section est issu de la BD TOPO de l’IGN, couvrant Compiègne et ses environs par un maillage carré de 25 m avec une précision métrique en altitude.

Afin d’optimiser le fonctionnement du contracteur d’altitude, présenté à la section 4.3.2, la position est calculée dans un repère local tangent à l’ellipsoïde de référence WGS84 en un point proche du lieu d’expérimentation. Le contracteur d’altitude est utilisé en coopération avec le contracteur robuste de pseudodistances. On ne discutera pas de la composante verticale de la solution, car elle est contrainte par le MNT, avec un intervalle d’incertitude de ± 1 m.

L’utilisation de cette contrainte d’altitude permet de réduire la largeur de la zone de localisation par un facteur 5 sur le plan horizontal (Figure 4.15). Quand moins de six satellites sont utilisés pour la localisation, la contrainte d’altitude est encore plus efficace, et permet de conserver une zone de localisation dont le rayon est inférieur à 40 mètres avec 5 satellites.

La réduction de l’incertitude de localisation sur le plan horizontal par la contrainte d’altitude est particulièrement visible quand peu de satellites sont visibles, ou quand la robustesse à un grand nombre de mesures aberrantes est requise. L’information d’altitude fournie par le MNT permet ainsi la localisation robuste avec un minimum de quatre mesures de pseudodistance. Sans cette contrainte, cinq satellites sont requis pour assurer la redondance nécessaire à une localisation robuste.

La figure 4.16 décrit l’influence d’une mesure biaisée sur les bornes de la zone d’incertitude calculée, lorsque les pseudodistances GPS sont utilisées en combinaison avec la contrainte d’altitude du modèle de terrain. Les données GPS et le biais simulé sur le satellite 1 sont identiques à ceux utilisés pour la figure 4.11, afin de permettre d’étudier l’apport de l’utilisation du MNT. L’approche non robuste n’est

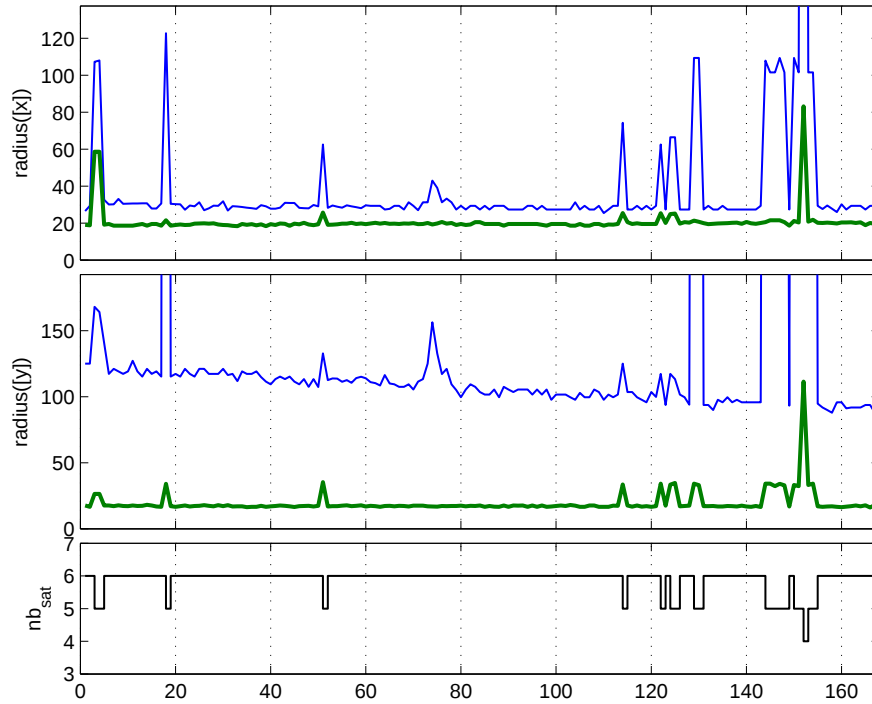
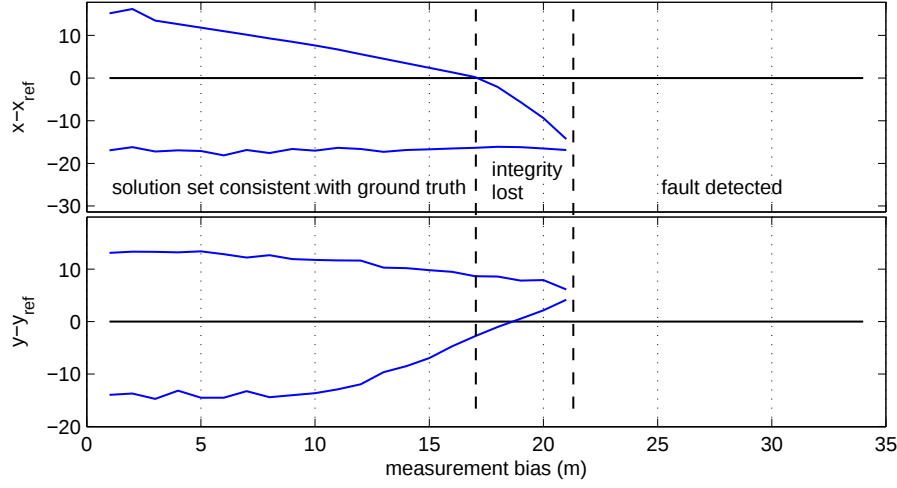


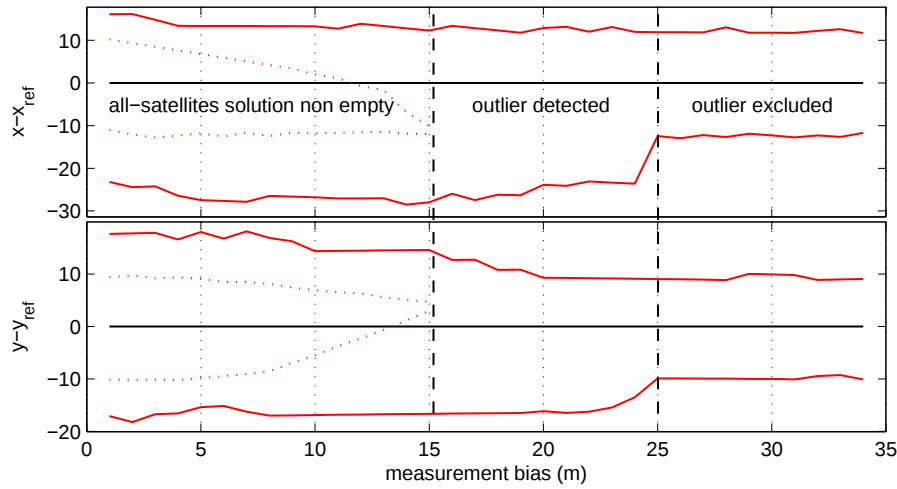
Figure 4.15: Rayon de la solution 1-relaxée avec (*lignes vertes épaisses*) et sans (*lignes bleues*) la contrainte d'altitude du MNT.

pas grandement améliorée par la contrainte d'altitude : la zone d'incertitude est légèrement réduite, et la faute est détectée quand le biais atteint 22 m, soit deux mètres de mieux que sans MNT. Cela s'explique par le fait qu'avec 6 satellites en vue, le problème de localisation est déjà bien contraint. Lorsque l'on utilise l'inversion 1-relaxée, l'apport du MNT est beaucoup plus significatif. Outre la réduction du volume de la zone d'incertitude, la contrainte d'altitude améliore grandement les performances de détection et exclusion de faute. La présence d'une mesure aberrante est détectée à partir d'un biais de 15 mètres (19 m étaient nécessaires sans MNT), et surtout, la mesure fautive est exclue de la solution dès que le biais atteint 25 mètres. Cette exclusion ne se faisait qu'à partir de 50 m sans MNT.

Afin de mettre en évidence l'influence de la précision du MNT sur les performances de détection et exclusion de fautes, nous avons successivement appliqué une erreur de type rampe sur chaque satellite et relevé les seuils de détection et d'exclusion de mesures aberrantes avec deux MNT de précisions différentes : le MNT de l'IGN présentant une précision d'un mètre en altitude, et le MNT issue des données SRTM de la NASA, avec une précision de l'ordre de 14 m en altitude. Les résultats, présentés sur la figure 4.17, montrent l'impact significatif de la précision du MNT sur la FDE. Ainsi, un modèle d'altitude plus précis permet non seulement la détection de biais plus petits, mais aussi la réduction de l'erreur de mesure nécessaire pour identifier la mesure aberrante.



(a) Inversion ensembliste non robuste



(b) Inversion ensembliste robuste (1-relaxée). Les courbes pointillées représentent les bornes de l'ensemble compatible avec toutes les pseudodistance.

Figure 4.16: Bornes du domaine d'incertitude de localisation, relativement à la vraie position. Un biais croissant est appliqué sur la pseudodistance du satellite n°1. La contrainte d'altitude du MNT est appliquée.

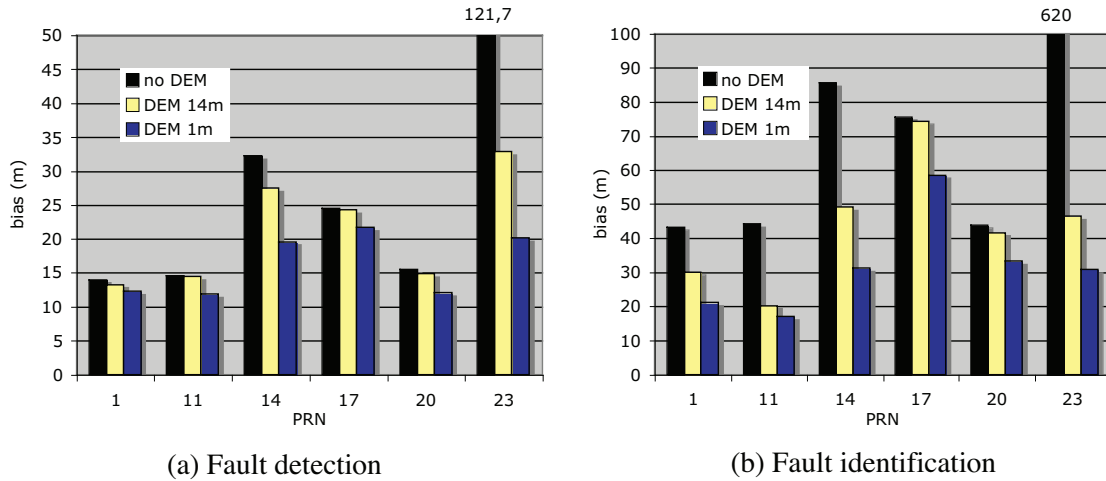


Figure 4.17: Influence de la contrainte d'altitude sur la détection et l'exclusion de faute avec l'algorithme SIVIA 1-relaxé

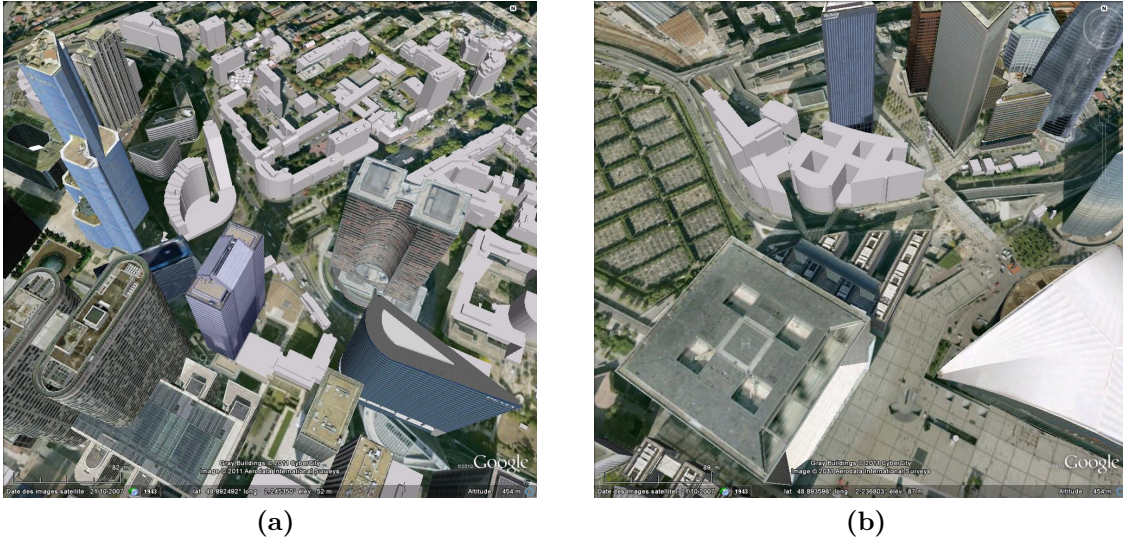
4.5.3 Essai avec un récepteur haute sensibilité

L'algorithme robuste de calcul d'une zone de localisation contrainte par l'altitude a également été testé sur un jeu de données mettant en œuvre un récepteur GPS bas coût *uBlox LEA-4T* associé à une antenne patch. Le récepteur est particulièrement sensible, et son antenne non-polarisée ne rejette pas les signaux réfléchis. On peut donc observer de nombreuses mesures aberrantes en milieu urbain, dues aux réflexions sur les bâtiments. Il n'est pas rare d'observer plusieurs mesures aberrantes simultanées [Le Marchand et al., 2009].

Nous présentons ici des résultats de calculs effectués sur un rejeu en temps réel de données acquises à La Défense. Comme le montre la figure 4.18, le cadre est propice à l'observation de multitrajets et de trajets sans ligne de vue directe (NLoS), en raison de la présence de hauts immeubles et gratte-ciels.

La figure 4.19 présente le résultat du calcul de zones de localisation contraintes en altitude par le MNT SRTM de la NASA à La Défense, aux lieux correspondant à la figure 4.18. L'utilisation d'un récepteur avec une antenne patch ne permet pas de rejeter les signaux réfléchis à la réception, on est donc en présence de plusieurs mesures aberrantes dues aux trajets NLoS des signaux GPS. Le calcul utilise l'intersection 2-relaxée des contraintes de pseudodistance. Le résultat est donc garanti tant que le nombre de mesures aberrantes ne dépasse pas deux.

Sur la figure 4.19a, une mesure est identifiée comme aberrante, grâce au mécanisme de compteurs présenté dans la section 3.6 page 85. La ligne de vue du satellite correspondant est représentée en rouge. Cependant, on constate que l'ensemble solution n'est pas connexe, ce qui laisse deux hypothèses de localisation. Ceci est dû à la présence d'une seconde mesure aberrante, détectée mais non identifiée. La détection

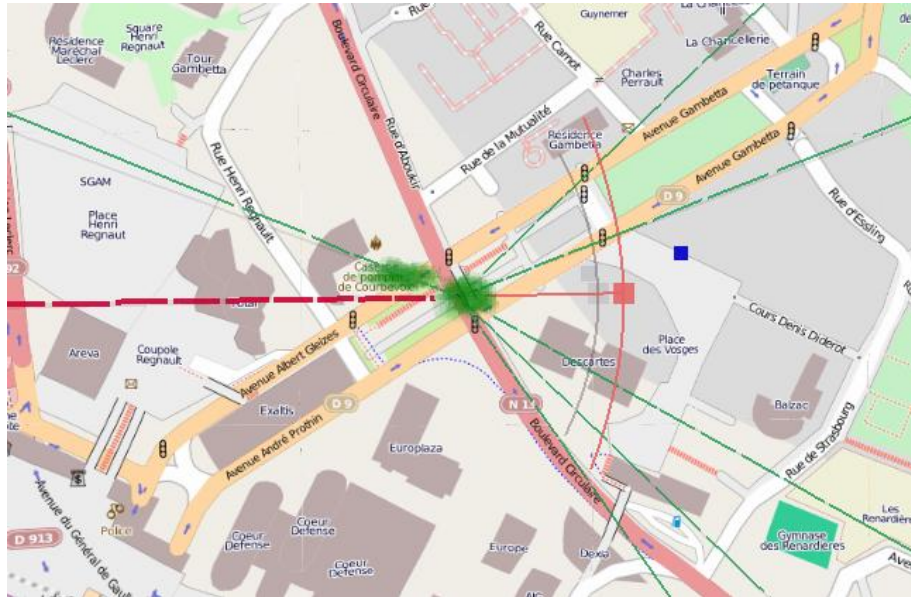
Figure 4.18: Vues *Google Earth* à La Défense

de la présence de deux mesures aberrantes est obtenue en vérifiant qu’aucune des boîtes du sous-pavage solution n’est compatible avec plus de 5 contraintes de pseudodistance sur les 7. Sur la figure 4.19b, en revanche, deux mesures aberrantes ont bien été détectées et identifiées.

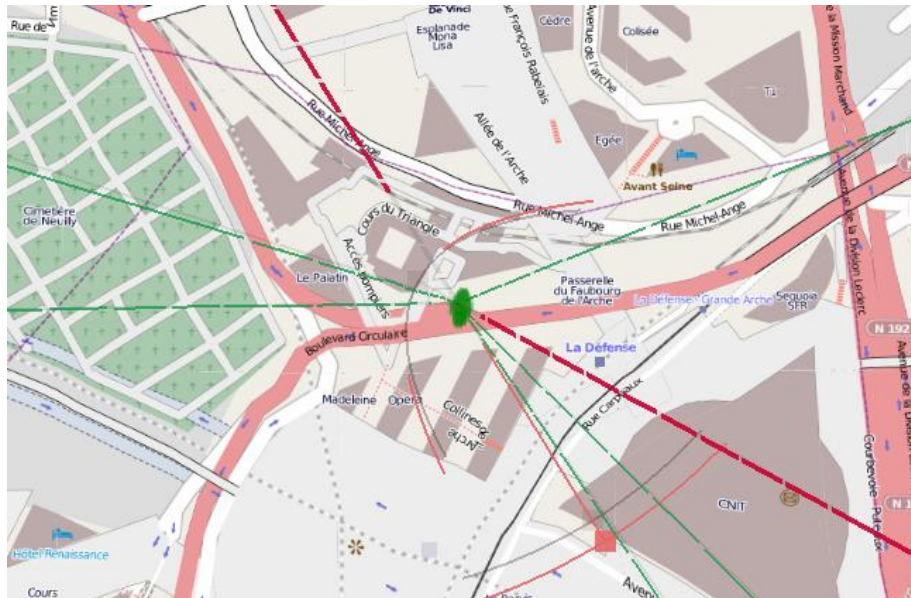
4.6 Implémentation en temps-réel

Le calcul en temps réel d’un domaine de localisation a été intégré à CARMEN, un véhicule expérimental récent du laboratoire. Il est interfacé avec le récepteur *Septentrio PolaRx2*, et peut également fonctionner avec un récepteur bas coût *uBlox* fournissant des données brutes (modèles LEA-4T et LEA-6T). Le décodage des éphémérides, le calcul des positions des satellites et la correction des pseudodistances sont réalisés à l’aide de la bibliothèque open source GPSTk [Tolman et al., 2004].

La méthode robuste de calcul d’une zone de localisation contrainte par l’altitude a été présentée en temps réel sur le véhicule CARMEN lors de la journée démonstrations de la conférence IV2011 *Intelligent Vehicle Symposium* à Baden-Baden (Allemagne). Cette section présente quelques détails de l’implémentation en temps réel de l’algorithme.



(a) Deux mesures aberrantes détectées, dont une identifiée



(b) Deux mesures aberrantes identifiées

Figure 4.19: Détection et identification de mesures aberrantes (en rouge) à La Défense, avec une solution 2-relaxée.



Figure 4.20: Véhicule CARMEN sur la piste d'essais lors de la journée démonstrations de la conférence IV2011 à Baden-Baden (Allemagne)

4.6.1 Calcul du sous-pavage solution

4.6.1.1 Calcul interruptible

Pour être embarqué dans la voiture, l'algorithme de localisation doit pouvoir répondre dans un temps borné, afin de satisfaire les contraintes de temps-réel sur la latence du système de positionnement. Pour satisfaire cette contrainte, il peut ainsi être nécessaire de stopper l'évaluation d'une solution en cours de calcul, et de se contenter du résultat obtenu jusque là. Il faut donc s'assurer que le résultat dégradé sera quand même exploitable.

L'algorithme d'inversion ensembliste SIVIA procède par bisections et contractions d'un domaine initial, afin d'obtenir un sous-pavage. Il est garanti de ne perdre aucune solution, aussi l'arrêt en cours de l'algorithme ne compromet pas l'intégrité du domaine de localisation fourni, mais peut néanmoins dégrader la précision. Parmi les différentes stratégies d'exploration des solutions pouvant être mises en pratique, il convient d'en choisir une qui limite l'impact de l'arrêt prématuré de SIVIA sur la précision de la solution. Une recherche en profondeur d'abord utilisant une pile pour stocker les boîtes bissectées ou, de façon équivalente, une implémentation réursive, n'est pas adaptée si le calcul doit être interrompu en cours pour satisfaire des contraintes temporelles. En effet, seule une partie du domaine initial aura été bissectée jusqu'à la taille de pavé minimale, tandis que le reste du domaine sera resté inexploré. Une meilleure solution consiste à utiliser une file pour stocker les boîtes bissectées. Ainsi, on obtient un parcours en largeur d'abord. Cependant, comme la contraction n'est pas aussi efficace sur toutes les boîtes, le sous pavage obtenu lors d'une interruption de calcul peut être contenir des boîtes de taille très irrégulière. Afin d'obtenir le sous-pavage le plus régulier possible lors d'une interruption du calcul avant la terminaison de l'algorithme SIVIA, les boîtes sont stockées dans une file

triée. À chaque itération de SIVIA, c'est la plus grande boîte de la file qui est ainsi traitée.

L'algorithme de calcul de position embarqué dans le véhicule est ainsi semblable à l'algorithme 3.9 page 88, dans lequel la contrainte d'altitude aurait été rajoutée, et où la liste des boîtes à traiter est triée.

4.6.1.2 Parallélisation

L'opération de contraction et bisection d'une boîte du sous-pavage est indépendante du résultat du traitement d'une autre boîte. Or, SIVIA nécessite de nombreuses contractions et bisections, le sous-pavage solution étant d'autant plus précis qu'il contient de boîtes.

S'agissant d'effectuer la même opération de façon indépendante sur un grand nombre d'objets, SIVIA est un algorithme qui se prête tout particulièrement à la parallélisation. Notre implémentation en C++ de SIVIA répartit les calculs entre plusieurs fils d'exécution (*threads*), afin de profiter de l'intégralité de la puissance disponible dans les machines multicœur actuelles. La liste des boîtes à traiter est partagée entre les fils d'exécution, qui piochent une boîte en tête de liste, la traitent et réinjectent les boîtes bisectées dans la liste une fois le traitement terminé. Cependant, quand le nombre de threads augmente, l'accès concurrentiel à la liste devient un goulot d'étranglement du problème. Le partage de la liste engendre des attentes lors des opérations de synchronisation sur son mutex⁶. On peut augmenter l'efficacité du calcul en mettant en place un cache de boîtes local à chaque thread, afin de traiter plusieurs boîtes séquentiellement et de limiter les temps d'attente.

4.6.2 Visualisation

Afin de présenter les résultats en temps réel dans le véhicule, une interface de visualisation a été réalisée en OpenGL (Figure 4.21). Elle présente le sous-pavage solution en 3D, sur un fond de carte OpenStreetMap⁷ associé au relief du modèle numérique de terrain SRTM. Les lignes de vue des satellites sont également représentées, afin de facilement se rendre compte de la géométrie du problème.

Quand une faute est détectée sur une pseudodistance, la ligne de vue vers le satellite correspondant s'affiche en rouge. En milieu urbain, une cause fréquente de mesure aberrante est la réception d'un signal réfléchi. Ainsi, le lieu des possibles points de réflexion du signal est affiché dans l'espace. Il est calculé sous hypothèse d'une erreur causée par un trajet sans vue directe avec une seule réflexion sur une surface

6. Les mutex (*mutual exclusion*) sont des primitives de synchronisation utilisées pour éviter que des ressources partagées d'un système ne soient utilisées en même temps.

7. OpenStreetMap est une carte du monde entier, librement modifiable. Elle est accessible sous forme raster et sous forme vectorielle XML à l'adresse <http://www.openstreetmap.org>



Figure 4.21: Visualisation 3D des zones de localisation calculées sur les Quais de Seine à Paris et à La Défense. À droite, une mesure aberrante a été identifiée et représentée en rouge.

verticale. Ce lieu correspond à une ellipse. L'association de la ligne de vue et du lieu de réflexion permet de repérer sur la carte les constructions responsables de la mesure fautive.

4.7 Discussion par rapport à l'approche classique

Les méthodes ensemblistes robustes que nous avons appliquées dans ce chapitre permettent de calculer un domaine de localisation, en propageant les intervalles d'erreurs bornées des mesures. Ce domaine, calculé à partir des mesures courantes, peut être considéré comme un domaine d'incertitude, au même titre que les niveaux d'incertitude (HUL et VUL) définis dans les métriques d'intégrité externe. On peut ainsi déterminer facilement un niveau d'incertitude scalaire à partir du résultat du calcul ensembliste, en prenant la plus grande distance entre l'estimée ponctuelle de la position et la frontière du sous-pavage solution.

De la même façon, on pourrait calculer un « domaine de protection », sans connaissance des mesures réelles. Le calcul consisterait alors à déterminer les mesures théoriques pour la position donnée, à leur associer les intervalles d'erreur correspondant au bruit nominal attendu, puis à réaliser l'inversion ensembliste. On pourrait alors déduire le niveau de protection correspondant à partir de la plus grande distance séparant la frontière de la solution de la position pour laquelle le calcul a été réalisé.

Pour les méthodes d'intégrité externes classiques présentées dans la section 2.5, le niveau d'incertitude augmente à mesure que les résidus augmentent. Autrement dit, plus l'inconsistance entre les mesures est grande, plus le niveau d'incertitude augmente. En effet, on effectue le calcul sans prendre en compte la présence éventuelle

d'un défaut, on suppose que l'augmentation des résidus est liée à la présence d'un défaut, et on quantifie l'impact de ce défaut sur la solution.

La méthode ensembliste a un comportement complètement différent. Plus l'inconsistance entre les mesures augmente, plus le volume de l'ensemble solution se réduit, jusqu'à devenir vide en cas d'inconsistance totale. En effet, la méthode ensembliste calcule l'ensemble des positions compatibles avec les mesures, dont l'erreur est supposée bornée, sous hypothèse qu'au moins $m - q$ mesures sont bonnes. Cet ensemble est donc d'autant plus réduit que les mesures ne sont pas concordantes, et l'absence de solution est la preuve qu'au moins q mesures sont aberrantes.

La robustesse du calcul, basé sur l'inversion ensembliste q -relaxée, provient de la spécification du nombre maximal de mesures aberrantes q dans les hypothèses. Il est cependant parfois difficile de pouvoir évaluer ce nombre, en particulier en milieu urbain où l'on peut rencontrer plusieurs mesures aberrantes simultanément. Nous avons étudié le positionnement dans le cadre d'une application temps réel, où le calcul peut être interrompu et doit renvoyer un résultat. Si cette contrainte était levée, on pourrait utiliser la stratégie de GOMNE présentée au chapitre précédent, consistant à déterminer le nombre minimal q_{min} de mesures aberrantes présentes. Une fois cette valeur déterminée, on pourrait calculer la solution $(q_{min} + 1)$ -relaxée afin d'être robuste à une mesure aberrante non-détectée. Cette approche s'apparenterait alors aux méthodes de FDE présentées dans la section 2.5, où l'étape de détection et exclusion de faute est répétée jusqu'à ce que plus aucune faute ne soit détectée, puis les niveaux d'intégrité externe sont calculés sous hypothèse d'un seul défaut non détecté.

4.8 Conclusion

Nous avons, dans ce chapitre, étudié une méthode ensembliste de localisation robuste à partir de données GPS brutes et d'un modèle numérique de terrain. La méthode calcule un domaine de localisation, associé à un critère de confiance, à savoir une borne du risque que la vraie position ne soit pas incluse dans le domaine calculé. La méthode permet également la détection et l'identification de mesures aberrantes.

Notre approche ensembliste a été comparée expérimentalement aux approches classiques de FDE et plusieurs essais expérimentaux ont été rapportés et analysés avec différents types de récepteurs GPS dans des environnements d'évolution très différents.

Nous nous sommes pour l'instant limités à l'utilisation d'une contrainte d'altitude, en plus des mesures de pseudodistances. Cependant, des informations cartographiques bien plus précises sont disponibles pour contraindre le domaine de localisation d'un véhicule routier. Dans le chapitre suivant, nous allons nous appuyer sur une cartographie précise de la chaussée pour la localisation de véhicules en milieu urbain.

Chapitre 5

Positionnement intègre contraint en milieu urbain

5.1 Introduction

La méthode de calcul d'un domaine de localisation présentée dans le chapitre précédent présuppose la disponibilité d'au moins quatre mesures de pseudodistances à chaque époque. L'ajout d'une contrainte d'altitude permet de réduire ce nombre à trois satellites. Cependant, la robustesse aux mesures aberrantes et l'application d'une détection et exclusion de faute est conditionnée par la redondance des données. Ainsi, il faut au moins une mesure supplémentaire pour espérer détecter une faute, et deux mesures supplémentaires pour tenter d'identifier et exclure une mesure aberrante.

En milieu urbain, la disponibilité des mesures peut se trouver très réduite et, de plus, des données aberrantes dues aux multitrajets et trajets sans ligne de vue directe peuvent contaminer les mesures reçues à une époque. L'utilisation d'un récepteur haute sensibilité permet de recevoir un grand nombre de signaux en milieu urbain, mais le risque d'inclure des pseudodistances sans ligne de vue vers le satellite ou contaminées par un multitrajet est d'autant plus élevé. Au contraire, un récepteur haut de gamme, couplé à une antenne polarisée, va rejeter l'essentiel des mesures erronées. Le récepteur est en effet équipé d'algorithmes de réjection ou d'estimation des multitrajets, et l'antenne polarisée atténue fortement les signaux réfléchis. Le risque est que dans des conditions difficiles, le nombre de mesures exploitables restantes peut être insuffisant pour calculer une solution de localisation.

Les contraintes cartographiques peuvent être mises à profit pour le positionnement de véhicules terrestres. De nombreuses recherches ont été menées sur ce sujet avec l'utilisation de cartes numériques classiques [Lahrech et al., 2005, Syed and Cannon, 2005], mais on peut désormais disposer de modèles numériques de surface précis [Paparoditis et al., 2000] et de cartes détaillant les voies de circulation [Bétaille et al., 2008]. La fusion de mesures GPS avec des informations géographiques pour améliorer la localisation a été un domaine actif de recherche : on peut par exemple calculer des corrections sur les mesures GPS à l'aide de la carte, comme dans le *Road Reduction Filter*

[Taylor et al., 2006] ou encore ajouter la contrainte de route après la mise à jour d'un filtre de Kalman comme la méthode de projection en deux étapes proposée par [Julier and LaViola, 2007] et appliquée au positionnement entre deux bords de voies circulaires par [Dunik et al., 2010]. Le filtrage particulaire a également été appliqué à la localisation sur route. Une méthode de bootstrap a été étudiée avec une représentation de l'espace roulant par facettes dans [Chausse et al., 2005], et également sur des surfaces clothoïdales avec une modélisation des connexions entre les voies de circulation parallèles [Selloum et al., 2010].

Afin de réduire le nombre de mesures nécessaires à la localisation d'un véhicule en milieu urbain, nous allons dans ce chapitre introduire deux contraintes supplémentaires sur la solution de localisation. La première contrainte concerne la position : nous allons tenir compte du fait que le véhicule n'évolue que sur les routes afin de contraindre son altitude et sa position sur une représentation 3D surfacique de « l'espace roulant » [Drevelle and Bonnifait, 2011b]. La seconde contrainte concerne le décalage d'horloge du récepteur : les récepteurs GPS embarquent une horloge relativement stable sur des temps courts, il est donc possible de prédire le décalage d'horloge à partir des valeurs précédentes. Ces deux contraintes peuvent être associées à la contrainte q -relaxée de pseudodistances introduite au chapitre précédent afin de caractériser un domaine de localisation. Nous présentons dans ce chapitre des résultats de localisation à partir de données réelles acquises à Paris.

5.2 Positionnement contraint par la route

5.2.1 Représentation 3D surfacique de la route

Contrairement aux approches classiques exploitant une représentation linéaire du réseau routier sous forme de polygones, une représentation précise de la surface de la route va ici être utilisée comme contrainte sur la position. C'est ainsi l'ensemble de la surface sur laquelle le véhicule peut évoluer qui est modélisée en trois dimensions. L'espace roulant définit la surface sur laquelle le véhicule peut physiquement évoluer. Pour une voiture, il peut être défini comme la surface de la route limitée par les bordures de trottoirs, et dont les obstacles tels que poteaux, panneaux, feux, terre pleins et séparateurs de voies ont été soustraits. L'annexe D présente une méthode pour la création d'une carte 3D de l'espace roulant.

La représentation retenue est un maillage triangulaire de la surface de l'espace roulant. On dispose ainsi d'un ensemble de points 3D, connectés afin de former des facettes triangulaires (Figure 5.2). Cette représentation permet également de retrouver la topologie du graphe de facettes, à partir de l'analyse des sommets en commun. L'information d'adjacence des facettes ainsi obtenue sera utilisée dans ce chapitre pour contraindre l'évolution du véhicule sur l'espace roulant.

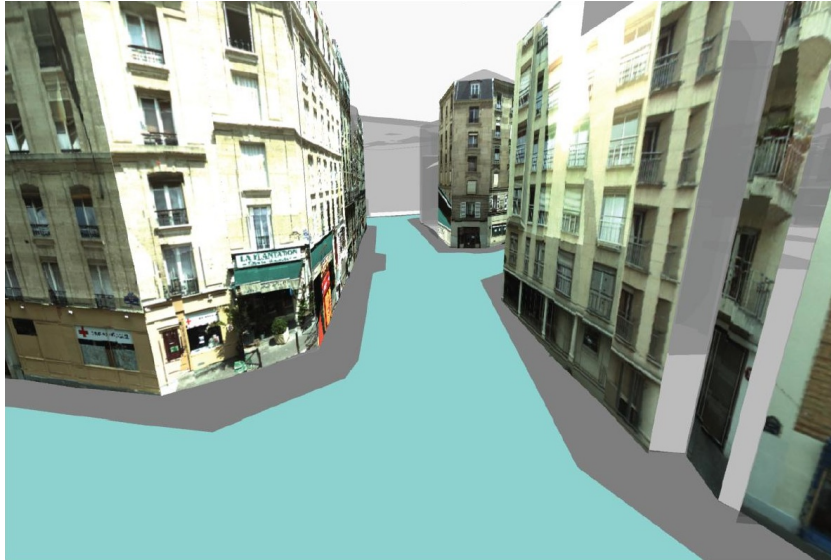


Figure 5.1: Modélisation 3D d'un quartier du XII^e arrondissement de Paris pour le projet CityVIP, incluant une représentation surfacique de la route en bleu clair. (IGN)

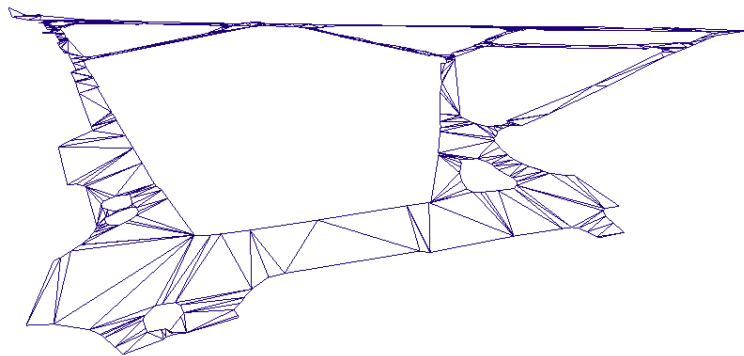


Figure 5.2: Vue « fil de fer » en perspective de l'espace roulant représenté sous forme de maillage triangulaire.

5.2.2 Contrainte de route

5.2.2.1 Contrainte liée à une facette

Le fait d'être situé sur une facette triangulaire peut être exprimé par quatre contraintes simples : une contrainte représentant l'appartenance au plan défini par les trois sommets de la facette, et trois contraintes définissant les limites imposées par ses arêtes. Considérons la facette ABC de la figure 5.3. On peut définir une première contrainte

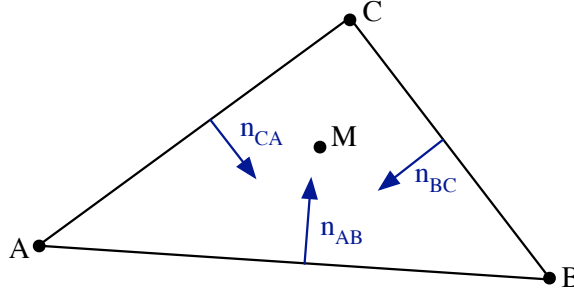


Figure 5.3: Facette ABC et normales associées à chaque arête.

à l'aide de l'équation du plan contenant les trois sommets de la facette. Le point M appartient au plan défini par les points A , B et C , ainsi on a

$$(\overrightarrow{AB} \wedge \overrightarrow{AC}) \cdot \overrightarrow{AM} = 0.$$

Les contraintes d'appartenance du point M à la facette imposées par les arêtes peuvent être exprimées par le signe du produit scalaire entre le vecteur liant un point de l'arête considérée à M et le vecteur normal à l'arête. Ainsi, on a

$$\overrightarrow{AM} \cdot n_{\overrightarrow{AB}} \geq 0$$

$$\overrightarrow{BM} \cdot n_{\overrightarrow{BC}} \geq 0$$

$$\overrightarrow{CM} \cdot n_{\overrightarrow{CA}} \geq 0.$$

Une autre façon de représenter l'appartenance du point M à la facette ABC est de considérer M comme le barycentre du système pondéré $\{(A, a); (B, b); (C, c)\}$, où a , b et c sont compris entre 0 et 1. On a ainsi

$$\{a\overrightarrow{MA} + b\overrightarrow{MB} + c\overrightarrow{MC} = \vec{0}, (a, b, c) \in [0, 1]^3\}.$$

5.2.2.2 Contraction par clipping de Sutherland-Hodgman

La contraction optimale d'une boîte avec la contrainte liée à une facette peut être obtenue en utilisant l'algorithme de découpage de polygones (*clipping*) de Sutherland-Hodgman [Sutherland and Hodgman, 1974]. Il s'agit d'une méthode principalement

utilisée pour la découpe des polygones affichées dans le domaine de la synthèse d'images. Cet algorithme, illustré dans un cas 2D par la figure 5.4, consiste à successivement découper la facette avec chaque (hyper)plan frontière de la boîte représentant le domaine initial $[x]$.

L'algorithme de Sutherland-Hodgeman consiste, pour chaque côté de la boîte initiale x , à vérifier que les points du polygone à découper vérifient la contrainte imposée par le côté considéré de x , et, le cas échéant, calculer le point d'intersection du segment du polygone avec le côté de x considéré.

La boîte englobante du polygone résultant du clipping de la facette par le domaine initial $[x]$ est définie comme le nouveau domaine contracté $[x_c]$.

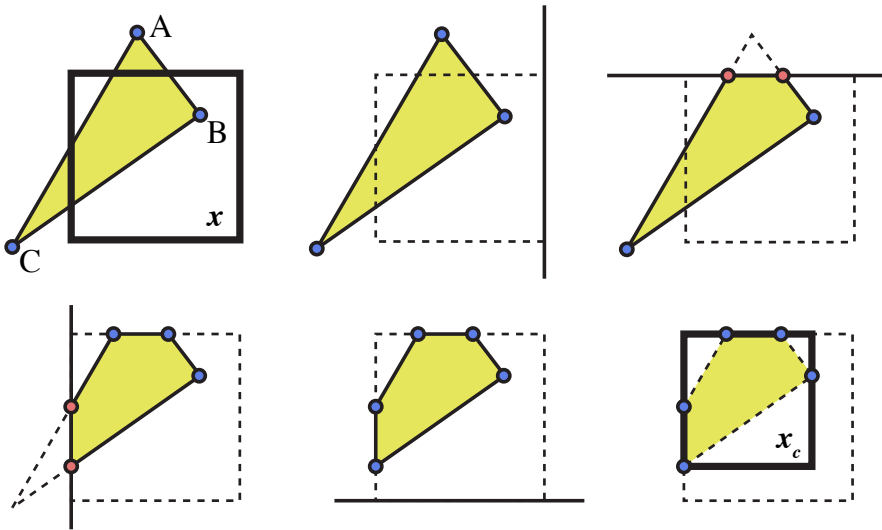


Figure 5.4: Contraction du domaine de position $[x]$ avec la facette triangulaire ABC, en utilisant l'algorithme de Sutherland-Hodgman. $[x_c]$ est le domaine contracté.

5.2.2.3 Contrainte liée à l'espace roulant

L'espace roulant est un maillage triangulaire. Il peut donc être considéré comme l'union des facettes qui le composent. Ainsi, l'ensemble des solutions satisfaisant l'appartenance à la route est l'union des solutions satisfaisant l'appartenance à chacune des facettes. Un contracteur pour la contrainte d'appartenance à l'espace roulant peut donc être construit en retournant l'union des domaines contractés par chaque facette. L'algorithme 5.1 présente un tel contracteur, qui effectue une première étape de sélection des facettes candidates pour limiter le nombre de contracteurs à appeler. Il est en effet préférable, pour limiter le temps de calcul, d'éviter la contraction avec les facettes dont on sait qu'elle sont *a priori* incompatibles avec le domaine de position considéré. Les facettes sont ainsi stockées avec leurs boîtes englobantes, et un

premier test de compatibilit  est effectu  pour rejeter rapidement les facettes incompatibles. Dans le cas o  la base de donn es cartographique couvre une vaste zone, les facettes peuvent  tre stock es dans un syst me d'information g ographique (SIG), permettant d'extraire efficacement au moyen d'une requ te l'ensemble des facettes pertinentes pour la position courante. La fonction *extract_facets* de l'algorithme 5.1 repr sente cette  tape de pr s lection.

Algorithme 5.1 road_contract(in : $[\mathbf{x}], \mathcal{M}$)

```

1:  $[\mathbf{x}_c] := \emptyset$ 
2:  $\mathcal{F} := \text{extract\_facets}(\mathcal{M}, [\mathbf{x}])$ 
3: for each  $f$  in  $\mathcal{F}$  do
4:    $[\mathbf{x}_c] := [\mathbf{x}_c] \sqcup \text{facet\_contract}([\mathbf{x}], f)$ 
5: end for
6: return  $[\mathbf{x}_c]$ 
    
```

La figure 5.5 illustre le fonctionnement de l'algorithme de contraction par l'espace roulant. La contraction d'une bo te $[\mathbf{x}_0]$ par la contrainte d'appartenance   l'espace roulant $\mathcal{M}$ consiste ainsi   tout d'abord s lectionner les facettes de $\mathcal{M}$ compatibles avec la bo te initiale. On obtient ainsi un sous ensemble $\mathcal{F}$ de facettes candidates (Figure 5.5b). La bo te initiale est ensuite contract e ind pendamment avec chaque facette candidate f_i , en utilisant le contracteur de facette pr sent  pr c demment (Figure 5.5c). La bo te contract e par la route $[\mathbf{x}_c]$ est obtenue en prenant l'union intervalle des bo tes contract es par les facette candidates (Figure 5.5d).

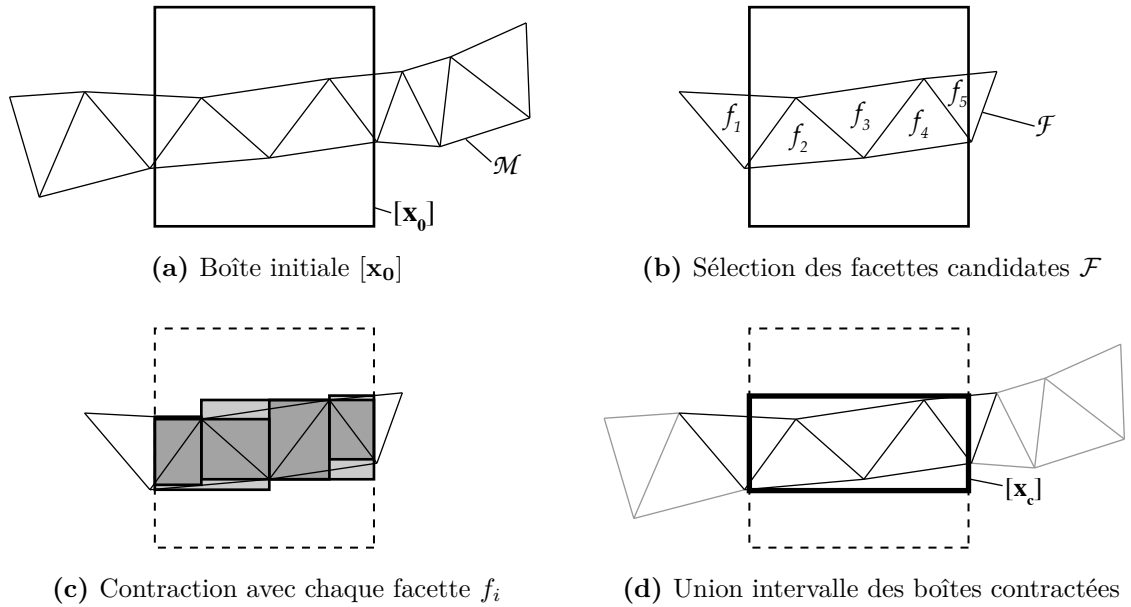


Figure 5.5: Contraction d'une bo te $[\mathbf{x}_0]$ avec l'espace roulant $\mathcal{M}$

5.2.2.4 Sélection des facettes candidates

La sélection des facettes candidates avant contraction permet de limiter le nombre de calculs en évitant de traiter les facettes incompatibles avec le domaine de position à contracter. Nous venons de voir que l'on peut extraire un « cache de facettes » à partir d'un SIG afin de limiter la zone de recherche au voisinage de la position courante, et qu'une première étape de tri à partir de la boîte englobante permet de limiter le nombre de facettes à traiter. La recherche des facettes candidates peut être accélérée en utilisant une structure indexée spatialement, telle qu'un R-arbre (*R-tree*)¹ [Guttman, 1984].

Une des particularités de l'inversion ensembliste est de pouvoir calculer des ensembles solutions non connexes. Cela induit de l'ambiguïté dans la localisation, chaque composante connexe pouvant être vue comme une hypothèse. Dans le chapitre précédent, c'est l'ajout de robustesse, en relaxant une ou plusieurs contraintes, qui pouvait conduire à une localisation ambiguë. Cependant, avec l'ajout de la contrainte de route, même dans le cas d'une inversion ensembliste simple, sans relaxation de contraintes, l'ensemble solution risque d'être non connexe. En effet, dans le cas où peu de satellites sont visibles, les contraintes liées au GPS peuvent mener à une solution large de plusieurs dizaines de mètres. Les contraintes de deux routes proches peuvent ainsi être satisfaites, et mener à une solution ambiguë.

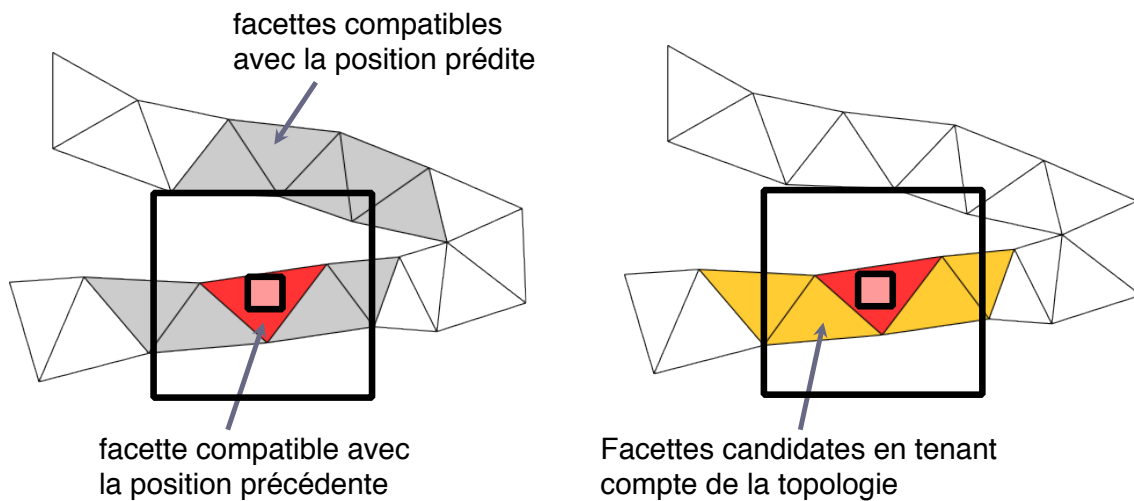


Figure 5.6: Utilisation de la topologie de la route pour limiter l'ambiguïté lors de la sélection des facettes candidates

Afin de limiter les cas d'ambiguïté, la sélection des facettes candidates peut être utilisée pour introduire une contrainte liée à la topologie du graphe de facettes

1. Les R-arbres sont des structures de données similaires aux B-arbres, utilisées pour l'indexation spatiale des données. La structure sépare l'espace avec des boîtes englobantes imbriquées et pouvant se chevaucher.

(Figure 5.6). En effet, il n'est possible de naviguer qu'en passant successivement de facettes en facettes voisines. Connaissant une position pr c dente $[x_{k|k}]$ au temps t_k et la distance s parcourue depuis t_k , on cherche   contracter le domaine de position $[x_{k+1|k}]$ au temps t_{k+1} . Le domaine peut avoir  t  pr dit avec un mod le d' volution du v hicule, tandis que la distance s peut provenir d'un odom tre.

Nous allons tout d'abord caract riser la zone $\mathbb{Z}$ dans laquelle le v hicule a pu  voluer entre les instants t_k et t_{k+1} . La zone d'int r t $\mathbb{Z}$ est l'ensemble des points M de coordonn es $\mathbf{m}$ tels que le chemin de $\mathbf{x}_k$   $\mathbf{x}_{k+1}$ faisant escale par le point M ait une longueur inf rieure ou  gale   la distance parcourue s , soit

$$\mathbb{Z} = \left\{ \mathbf{m} \in \mathbb{R}^3, \|\mathbf{m} - \mathbf{x}_k\| + \|\mathbf{x}_{k+1} - \mathbf{m}\| \leq s \right\}.$$

Une approximation externe grossi re de $\mathbb{Z}$ par une bo te $[\mathbf{z}]$ telle que $\mathbb{Z} \subset [\mathbf{z}]$ peut  tre obtenue en appliquant la contrainte de distance parcourue ind pendamment sur chaque dimension

$$[\mathbf{z}] = \frac{[\mathbf{x}_{k|k}] + [\mathbf{x}_{k+1|k}] + [-s, s]}{2}.$$

Les facettes dont l'intersection avec $\mathbb{Z}$ est non vide sont extraites de la carte, pour obtenir la sous-carte   utile   $\mathcal{M}_{\mathbb{Z}}$. En effet, seules les facettes contenues dans $\mathcal{M}_{\mathbb{Z}}$ peuvent avoir  t  parcourues entre t_k et t_{k+1} . Pour simplifier la s lection, on peut choisir de conserver les facettes intersectant $[\mathbf{z}]$, on utilisera ainsi la sous-carte $\mathcal{M}_{[\mathbf{z}]}$ en lieu et place de $\mathcal{M}_{\mathbb{Z}}$. Notons que puisque $\mathbb{Z} \subset [\mathbf{z}]$, alors $\mathcal{M}_{\mathbb{Z}} \subset \mathcal{M}_{[\mathbf{z}]}$.

Une fois la partie utile $\mathcal{M}_{\mathbb{Z}}$ de la carte extraite, nous allons utiliser la topologie du graphe de facettes pour d terminer l'ensemble des facettes candidates   l'instant t_{k+1} . Tout d'abord, les facettes de $\mathcal{M}_{\mathbb{Z}}$ correspondant   la position $x_{k|k}$   l'instant t_k sont marqu es. Ensuite, on examine chaque facette marqu e : pour chaque facette marqu e et non pr c demment visit e, ses facettes adjacentes sont marqu es. Quand il ne reste plus de facettes   visiter, l'ensemble des facettes marqu es forme une ou plusieurs composantes connexes, correspondant aux routes ayant pu  tre emprunt es entre les instants t_k et t_{k+1} . On d duit l'ensemble des facettes candidates   t_{k+1} en conservant les facettes marqu es dont l'intersection avec le domaine $[x_{k+1|k}]$ est non vide.

5.2.3 Synth se

La figure suivante pr sente de fa on graphique le fonctionnement de la m thode.

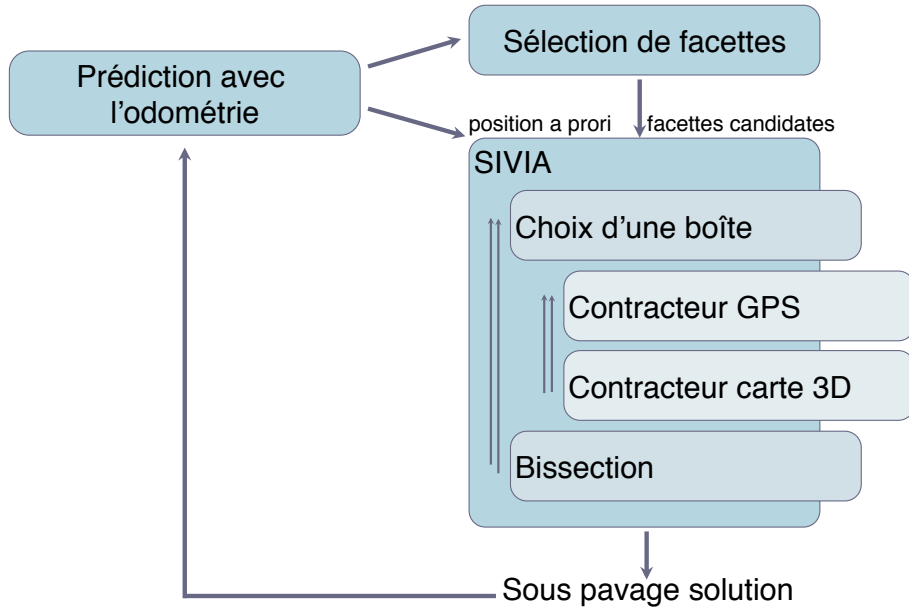


Figure 5.7: Algorithme de positionnement contraint par l'espace roulant.

La méthode contraction de la position par la contrainte d'espace roulant est associée au contracteur robuste de pseudodistances (Algorithme 4.3 page 94) dans un algorithme d'inversion ensembliste par intervalles.

5.3 Contrainte liée à la dérive d'horloge du récepteur

Une partie du problème de localisation GPS consiste à déterminer le décalage dt^u entre l'horloge du récepteur et le temps GPS. Les récepteurs GPS embarquent une base de temps utilisant un quartz, qui dérive en fréquence en fonction de la température. Il est fait usage d'un oscillateur à quartz compensé en température (TCXO) pour améliorer les performances d'acquisition, tandis que les récepteurs haut de gamme dédiés à la mesure du temps embarquent un oscillateur à quartz à température régulée (OCXO). La figure 5.8 montre l'évolution de la dérive d'horloge d'un récepteur bas coût *ublox LEA-4T* sur 18 heures de mesure.

5.3.1 Prédiction du décalage d'horloge

Sur des temps courts, de l'ordre de la minute avec un quartz, on peut considérer que l'oscillateur ne dérive pas en fréquence, et que l'horloge du récepteur est « parfaite ». Pour des observations plus longues (plusieurs minutes), on peut considérer que la dérive de l'oscillateur est linéaire. Il est alors possible d'estimer la dérive dt^u de l'horloge du récepteur par rapport au temps GPS, afin de prédire le décalage d'horloge dt^u .

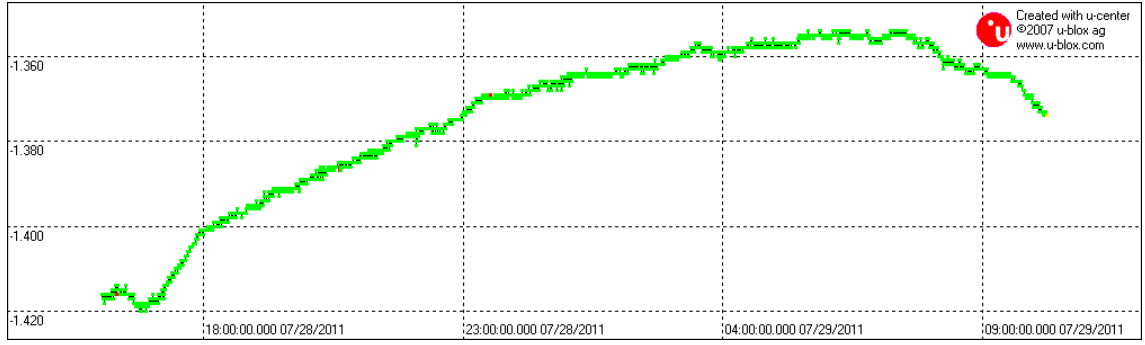


Figure 5.8: Dérive d’horloge (en $\mu\text{s/s}$) sur 18 h de fonctionnement d’un récepteur *ublox LEA-4T*.

Pour prédire le décalage d’horloge $dt^u(t)$ au temps t , nous effectuons une régression linéaire sur les 15 derniers intervalles estimés de dt^u . Comme les paramètres de la droite vont servir à prédire le décalage d’horloge, on va effectuer l’estimation au temps le plus récent afin de réduire l’influence de l’incertitude des paramètres sur la prédiction. On cherche ainsi à estimer les paramètres α et β tels que $dt^u(t) = \alpha + \beta(t - t_N)$, en supposant que t_N corresponde à la date du décalage d’horloge le plus récent de l’historique. Les domaines intervalles $[\alpha]$ et $[\beta]$ pour α et β sont obtenus en contractant le système de contraintes suivant :

$$\begin{cases} dt^u(t_k) = \alpha + \beta(t_k - t_N) & k = 1 \dots N \\ dt^u(t_k) \in [dt^u](t_k) \\ \alpha \in [\alpha] \\ \beta \in [\beta] \end{cases}.$$

Les intervalles $[\alpha]$ et $[\beta]$ sont obtenus en utilisant la propagation de contraintes et des bisections. Le décalage d’horloge au temps t peut alors être prédit sous forme d’intervalle à partir des domaines contractés des paramètres de la droite. On a

$$[\hat{dt}^u](t) = [\alpha] + [\beta](t - t_N).$$

La figure 5.9 résume le principe de la prédiction du décalage d’horloge contraint par l’historique des valeurs précédentes.

5.3.2 Gestion des sauts d’horloge

Hormis certains récepteurs haut de gamme qui asservissent leur base de temps sur le temps GPS, la plupart des récepteurs laissent dériver leur horloge et estiment le décalage. Dans ce cas, le décalage d’horloge du récepteur n’est toutefois pas continu. En effet, afin d’éviter les débordements, dt^u est maintenu dans un intervalle prédéfini. Cela cause donc des sauts à chaque fois que le décalage d’horloge dépasse une

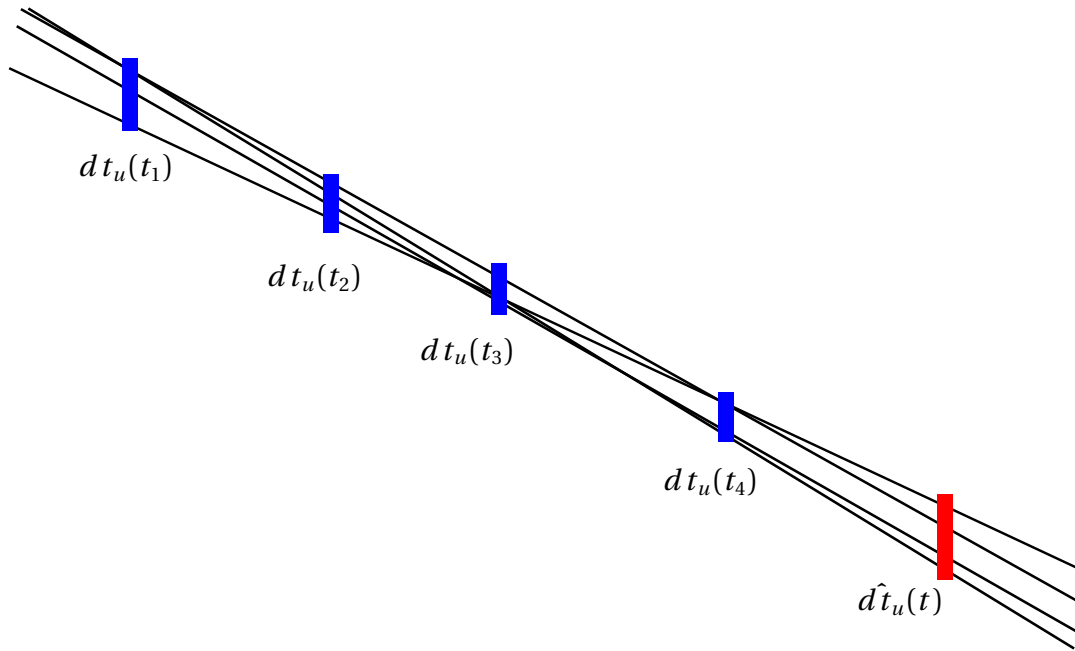


Figure 5.9: Estimation de la dérive d'horloge et prédiction

des bornes fixées. Pour le récepteur *ublox* de la figure 5.10, les sauts d'horloge ont une amplitude d'une dizaine de millisecondes. De tels sauts peuvent facilement être

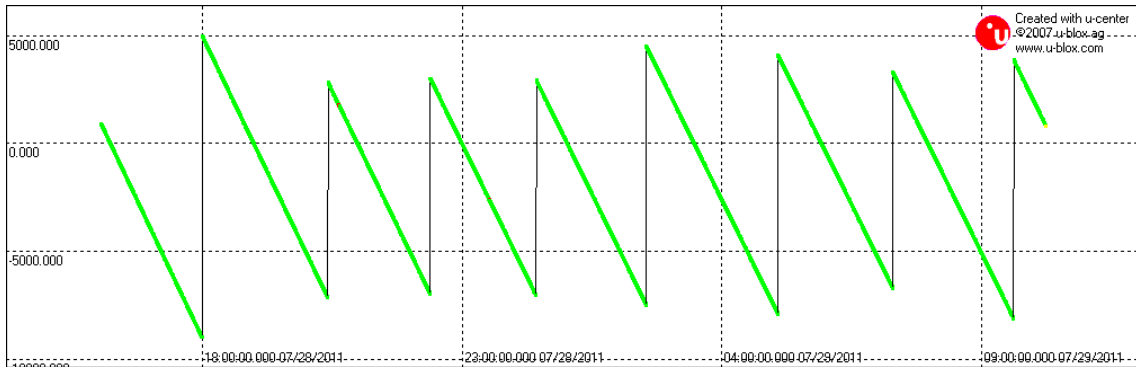


Figure 5.10: Évolution du décalage d'horloge (en μs) d'un récepteur *ublox LEA-4T* sur 18 h de mesures. On observe très nettement les sauts d'horloge.

déTECTÉS : un saut d'horloge typique a une amplitude d'au moins une milliseconde (soit environ 300 km parcourus à la vitesse de la lumière), tandis que le décalage d'horloge est généralement prédit avec une incertitude de quelques dizaines de nanosecondes. Lors d'un saut d'horloge, l'ensemble des pseudodistances mesurées présentent ainsi un saut facilement détectable d'au moins 300 km. La figure 5.11 représente l'évolution des pseudodistance mesurés pour le satellite 24 sur 5 heures d'observation. Les sauts d'horloge du récepteur utilisé sont d'au moins 10 ms, ce qui crée des sauts de 3000 km dans la grandeur mesurée. Quand un saut d'horloge est

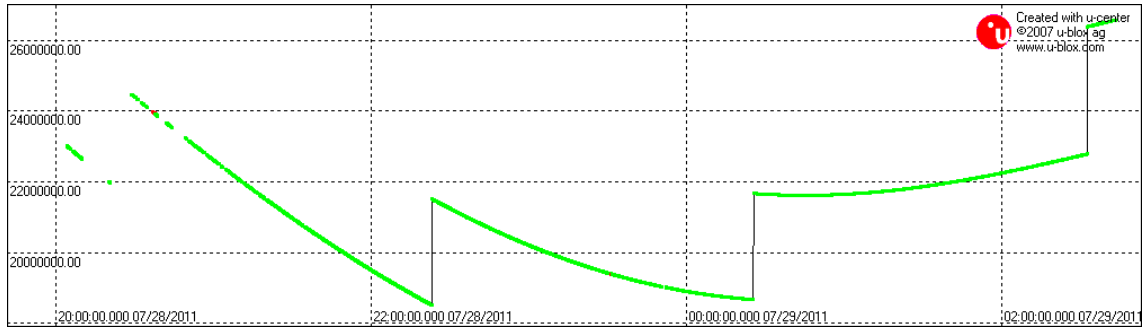


Figure 5.11: Influence des sauts d'horloge sur les pseudodistances mesurées (en m) du satellite 24.

détecté, le prédictor ensembliste de décalage d'horloge est réinitialisé.

L'ajout de la contrainte d'évolution du décalage d'horloge récepteur permet le positionnement avec deux mesures de pseudodistance, quand il n'y a pas d'ambiguïté sur le choix du tronçon de route courant [Fouque and Bonnifait, 2010]. Il est donc nécessaire d'avoir pu observer auparavant plusieurs satellites, afin de stocker les valeurs de dt^u nécessaires à la régression linéaire et de positionner le véhicule sur une unique route. Une fois la prédiction d'horloge initialisée (c'est-à-dire une fois qu'on a une estimation fiable et précise des intervalles $[\alpha]$ et $[\beta]$), et s'il n'y a pas d'ambiguïté sur le choix du segment de route courant, la prédiction du décalage d'horloge permet la localisation sans ambiguïté de l'utilisateur avec une seule pseudodistance jusqu'à la prochaine intersection.

5.4 Résultats

La mise en œuvre du contracteur d'espace roulant présuppose la disponibilité d'une représentation surfacique de la route sur l'aire d'essai. Au cours du projet CityVIP et parallèlement à une campagne d'acquisition de données GPS et vidéo, l'IGN a réalisé une telle modélisation de la route autour de la mairie du XII^e arrondissement de Paris. Cette carte a été créée par photogrammétrie, les opérateurs saisissant manuellement en trois dimensions les bords de trottoirs à partir de couples d'images aériennes [Paparoditis et al., 2000]. La précision des sommets de la carte est de 5 cm en planimétrie et de 25 cm en altitude. La carte est fournie dans la projection Lambert93. Elle est convertie dans le repère de travail (repère tangent en la base IGN de Saint-Mandé) afin d'être facilement utilisée avec le contracteur présenté dans ce chapitre.

Nous utilisons les mesures GPS échantillonnées à 2 Hz du récepteur *Septentrio PolaRx2* embarqué sur le véhicule Stéréopolis de l'IGN. Les données odométriques brutes n'étant pas accessibles, elles sont reconstruites à partir de la sortie de la centrale *Applanix*. L'algorithme de localisation est testé lors d'un jeu en temps

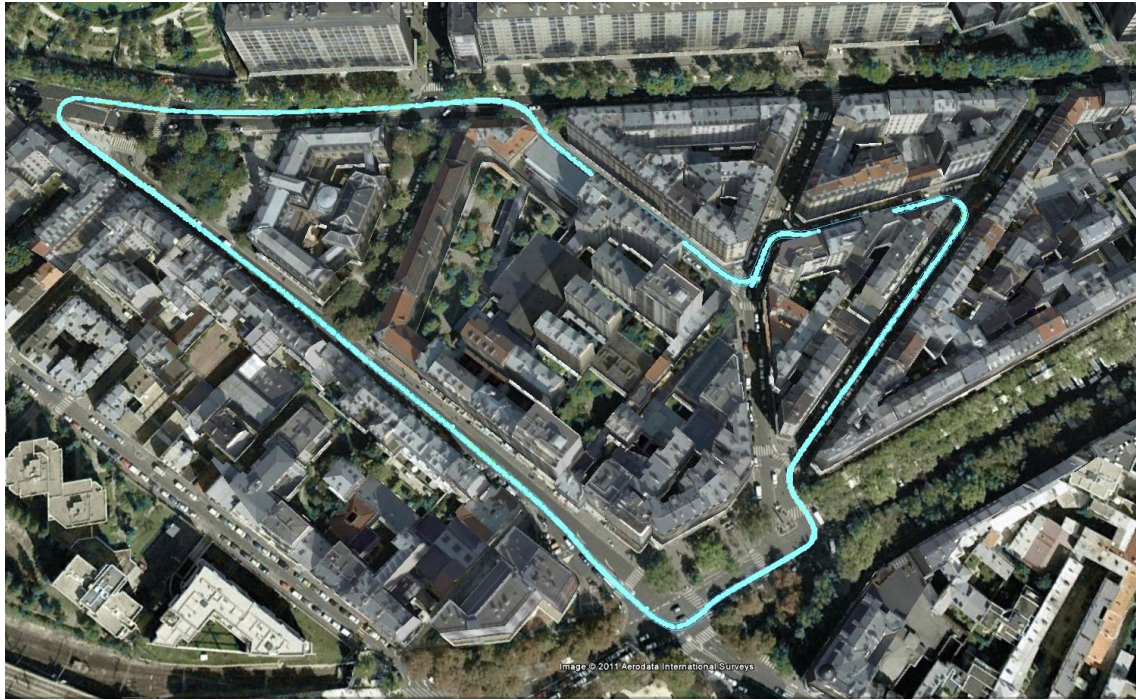


Figure 5.12: Trajectoire de l’essai autour de la mairie du XII^e arrondissement de Paris. *Vue Google Earth.*

réel des données acquises sur une boucle d’un kilomètre autour de la mairie du XII^e arrondissement, représentée sur la figure 5.12. Le jeu de données est présenté dans l’annexe C.3.

Les données acquises autour de la mairie du XII^e arrondissement sont particulièrement difficiles pour le positionnement par satellites. Le véhicule parcourt des rues étroites où la constellation GPS visible est réduite. En effet, on peut voir sur l’histogramme de la figure 5.13 que moins de quatre satellites sont visibles pendant plus des trois quarts de l’essai, et on a moins de trois satellites en vue 40% du temps.

La méthode de positionnement contraint par la route de la section 5.2 est couplée à la prédiction du décalage d’horloge de la section 5.3 afin de réduire la boîte initiale soumise à l’algorithme d’inversion ensembliste (Figure 5.14 page suivante).

Le nombre q de pseudodistances erronées tolérées par le contracteur robuste de position est déterminé dynamiquement en fonction du nombre m de satellites visibles. Avec moins de quatre mesures de pseudodistances, une inversion ensembliste non robuste est effectuée ($q = 1$). À partir de quatre satellites visibles, l’intersection q -relaxée est employée. Le nombre de mesures GPS aberrantes tolérées est réglé de

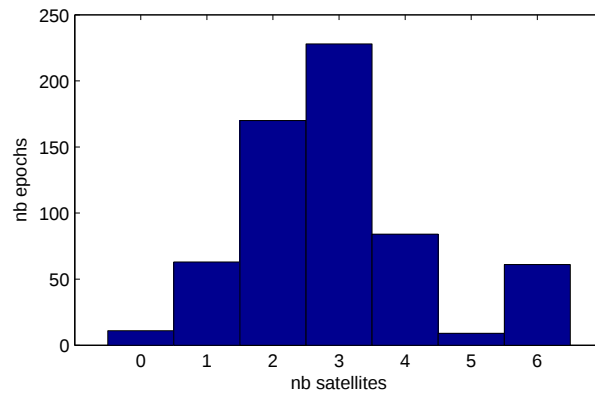


Figure 5.13: Histogramme du nombre de satellites visibles avec le récepteur *Polara* sur une boucle autour de la mairie.

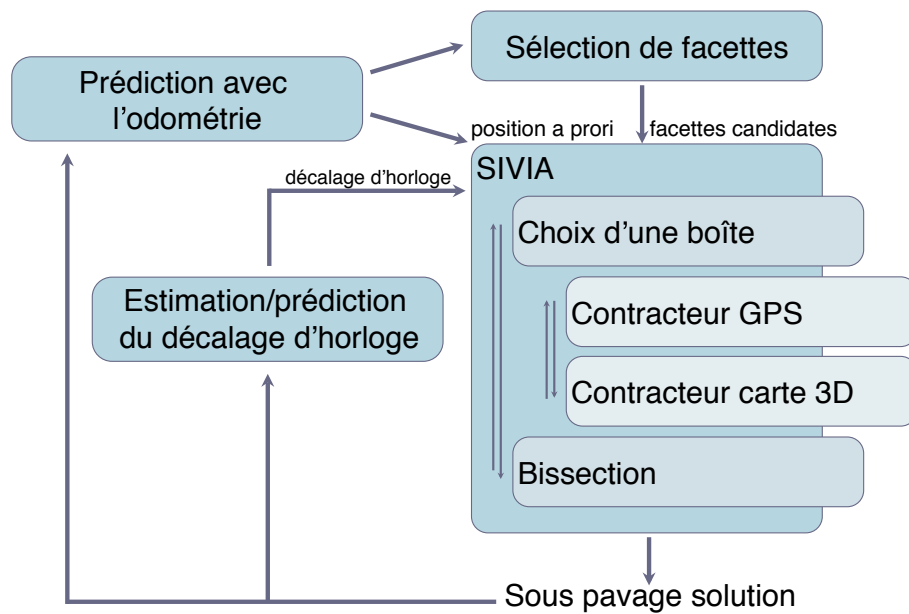


Figure 5.14: Algorithme de positionnement contraint par l'espace roulant et le décalage d'horloge prédit

la façon suivante :

$$q = \begin{cases} 0 & m < 4 \\ 1 & m = 4 \\ 2 & m > 4 \end{cases}.$$

Les intervalles d'erreur associées aux mesures de pseudodistances sont ajustés pour que le risque associé au domaine d'incertitude calculé soit de 10^{-5} , en supposant des erreurs gaussiennes et indépendantes, avec un mètre d'écart type. Ce calcul de risque est effectué à l'aide de la méthode décrite dans la section 4.4, en supposant que le véhicule ne quitte pas la route et que la carte ne comporte pas d'erreurs.

La figure 5.15 présente les domaines d'incertitude sur la position calculés en différents points de la trajectoire du véhicule, ainsi que les facettes candidates à ces instants. Au début de l'essai, les conditions GPS sont bonnes, avec six satellites en vue, ce qui permet un positionnement sans ambiguïté. Puis le véhicule entre dans des rues étroites, formant des canyons urbains où le nombre de pseudodistances utilisées est fortement réduit. Après trente secondes, à l'époque de mesure #69, le domaine d'incertitude de localisation est réduit, permettant une localisation précise malgré l'utilisation de seulement deux pseudodistances (Figure 5.15a). Cela est rendu possible par l'utilisation de la contrainte cartographique, associée au fait que l'ambiguïté du choix de la route est levée grâce à la prise en compte de la connexion des facettes. À l'époque #220, le véhicule se trouve à un carrefour avec seulement deux satellites en vue (Figure 5.15b). La sélection du tronçon de route est ambiguë, mais la solution de positionnement reste connexe. En revanche, à l'époque #553, le véhicule franchit un croisement avec seulement une pseudodistance disponible. Grâce à la prédiction d'horloge, on peut continuer à calculer une solution non ambiguë avec une seule pseudodistance, tant que le segment de route courant est identifié et que l'incertitude sur le décalage d'horloge prédit n'est pas trop grande. Ce n'est plus le cas ici, car le croisement apporte deux hypothèses de choix de route. On obtient un domaine d'incertitude de position non connexe, présentant les deux hypothèses de localisation (Figure 5.15c).

La figure 5.16 montre l'erreur ensembliste de la localisation, à savoir l'écart entre les bornes des domaines de localisation calculés à chaque instant et la trajectoire de référence. On peut noter que les domaines calculés sont consistants avec la vraie position pendant tout l'essai.

Le centre de gravité du sous-pavage solution (représenté par des points sur la figure 5.16) fournit une bonne estimation de la position du véhicule. Avec au moins deux satellites en vue, l'erreur de position est inférieure à 6.5 m pendant 95% du temps de l'essai (Figure 5.17). Dans ces mêmes conditions, le rayon du domaine d'incertitude de localisation reste inférieur à 16 m 95% du temps.

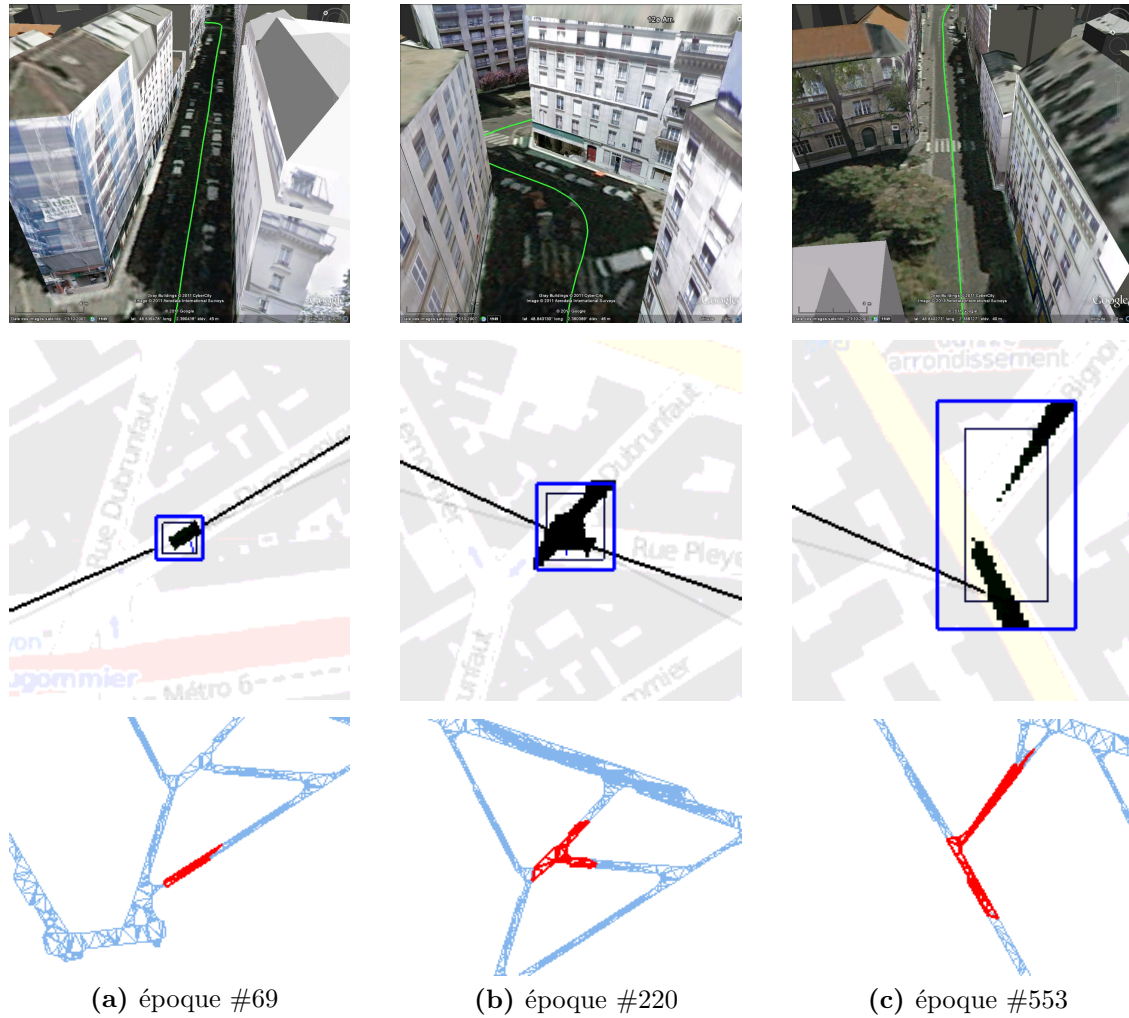


Figure 5.15: En haut : vue 3D de l'environnement (*Google Earth*). Au milieu : sous-pavage solution (en noir) de la localisation contrainte par l'espace roulant. Le rectangle bleu est la position prédite, les lignes noires sont les lignes de vue vers les satellites GPS. En bas : espace roulant. Les facettes candidates sont représentées en rouge.

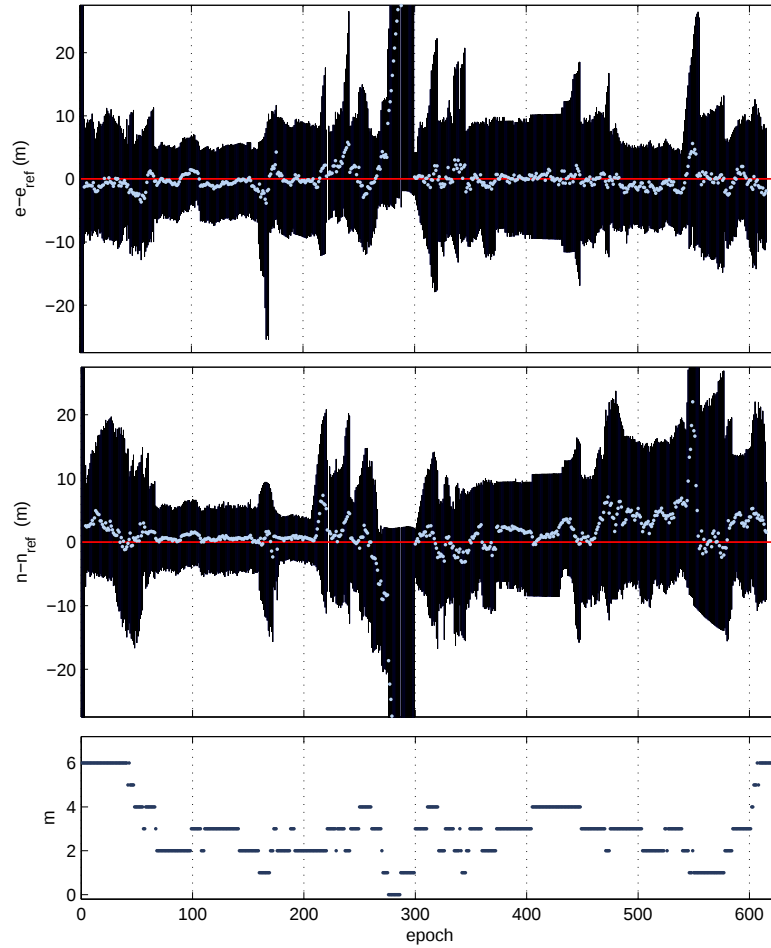


Figure 5.16: Bornes du sous-pavage solution. *Les coordonnées e et n sont exprimées dans un plan local tangent à la terre. L'ordonnée zéro correspond à la vraie position. Les points indiquent le centre de gravité du sous-pavage.*

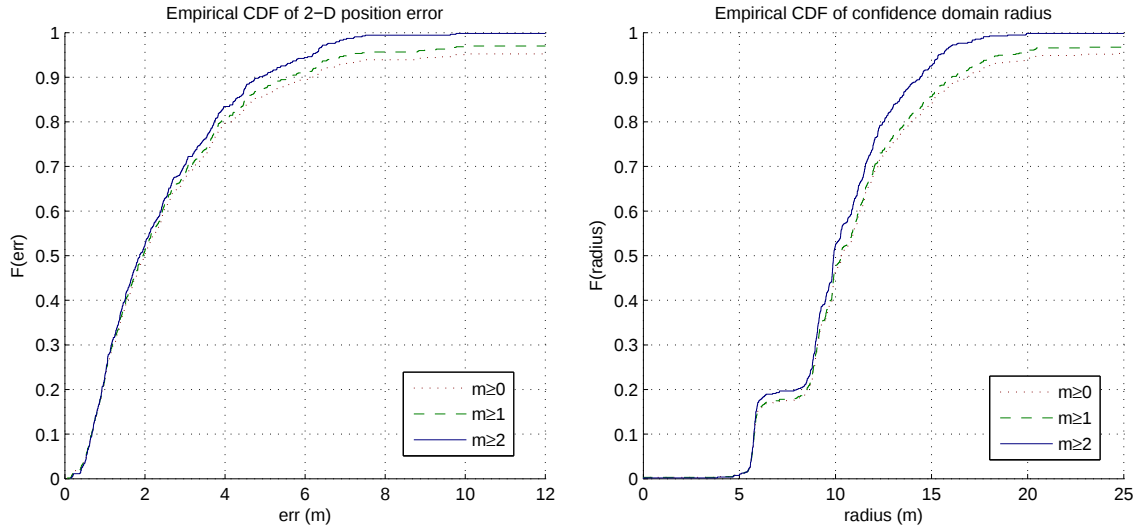


Figure 5.17: À gauche : fonction de répartition empirique de l'erreur de position 2D du centre de gravité du sous-pavage solution. À droite : fonction de répartition empirique du rayon du domaine de confiance de position. Les fonctions de répartitions sont calculées avec toutes les solutions ($m \geq 0$), et en ne conservant que les époques avec au moins un ($m \geq 1$) ou deux ($m \geq 2$) satellites utilisés.

5.5 Conclusion

Ce chapitre a été dédié à l'introduction de la contrainte de route dans le calcul de localisation. Les cartes que nous avons utilisées représentent la surface roulable en 3 dimensions sous la forme de facettes triangulaires. Nous avons présenté les contracteurs ensemblistes que nous avons développés. Du fait de l'utilisation de cette approche dans des environnements urbains très contraignants vis-à-vis de l'utilisation du GPS, un prédicteur de dérive d'horloge a été développé. Des résultats expérimentaux ont été présentés sur un essai réalisé à Paris avec de nombreux canyons urbains. Un calcul époque par époque a été réalisé en rejeu temps-réel des données.

En l'absence totale de signaux GPS ou quand un seul satellite est visible, les domaines de localisation calculés avec cette méthode peuvent devenir arbitrairement grands. Nous allons donc nous intéresser maintenant à une amélioration de cette approche en formalisant le problème sous l'angle de l'observation d'état sur un horizon fini de données.

Chapitre 6

Estimation robuste de la pose sur un horizon de données

6.1 Introduction

Dans les chapitres précédents, nous avons étudié une méthode de localisation ensembliste robuste statique d'un véhicule en milieu urbain. Le chapitre 5 a permis l'ajout de contraintes fortes sur la solution GPS, en utilisant une carte délimitant la surface sur laquelle le véhicule peut se situer, et en modélisant l'évolution du décalage d'horloge du récepteur. Toutefois, l'évolution du véhicule n'a pas été prise en compte.

L'essentiel des véhicules modernes sont équipés, de série, de systèmes d'aide à la conduite qui nécessitent la présence de capteurs proprioceptifs. Le système de freinage anti-blocage (ABS), obligatoire depuis 2004 sur les véhicules neufs commercialisés dans l'Union Européenne, fonctionne grâce à des codeurs mesurant la vitesse de rotation de chaque roue. Le contrôle électronique de stabilité (ESP) nécessite quant à lui, outre la vitesse des roues, la présence de capteurs de l'angle au volant, de vitesse de lacet et d'accélération.

Les informations fournies par ces différents capteurs peuvent donc être utilisées pour contraindre la position du véhicule. Cherchant à contraindre la trajectoire du véhicule avec les mesures fournies par les capteurs proprioceptifs embarqués, le problème revient à estimer, en plus de la position, l'attitude du véhicule. En effet, l'orientation du véhicule dans l'espace entre en compte dans le modèle d'évolution. Nous devons donc estimer la pose du véhicule. En considérant que le véhicule se déplace sur des pentes faibles, le problème d'estimation de pose peut être simplifié en se plaçant dans un repère plan localement tangent à l'ellipsoïde de référence. On passe ainsi de 6 degrés de libertés (trois pour la position et trois pour les angles d'attitude) à une pose à trois degrés de liberté : deux coordonnées pour la position, et un angle de cap.

Une approche simple consiste à utiliser un filtre ensembliste de type estimateur-prédicteur [Kieffer, 1999]. Ainsi, partant du domaine d'incertitude de la pose du véhicule à l'instant précédent, on prédit la pose du véhicule à l'instant courant grâce

aux mesures proprioceptives et à un modèle d'évolution, puis on contracte le domaine prédit avec les contraintes de pseudodistances et de route. Le principal problème de cette approche est que la solution calculée à un instant dépend de l'ensemble des mesures précédentes. Aussi, comme un risque a été pris sur chacune des mesures afin de choisir les bornes d'un intervalle d'erreur, la confiance que l'on peut accorder à l'ensemble solution décroît au fil des estimations [Jaulin, 2011b].

Afin de fournir une zone d'incertitude assortie d'un paramètre de confiance constant, une solution est de limiter le nombre de données utilisées pour le calcul de la solution. Nous allons donc utiliser un nombre borné de mesures, sous forme d'un horizon glissant de données. Cet historique de données est source de redondance d'information dont nous allons pouvoir tirer profit pour assurer l'intégrité du positionnement [Drevelle and Bonnifait, 2010]. Suite aux chapitres précédents, nous disposons d'une méthode de calcul d'une zone d'incertitude sur la position, utilisant les pseudodistances GPS en couplage serrée avec une carte 3D de l'espace roulant. Nous pouvons ainsi réaliser l'estimation de pose en utilisant un horizon des dernières positions calculées, couplé à l'historique des dernières mesures proprioceptives et à un modèle d'évolution du véhicule.

Dans ce chapitre, nous présentons tout d'abord le modèle de véhicule utilisé, puis nous abordons la gestion de l'historique des positions et des mesures proprioceptives. Nous décrivons ensuite un contracteur de la pose par propagation de contraintes à partir d'une position de l'historique, des données proprioceptives et de la carte de l'espace roulant. L'implémentation en temps-réel est ensuite détaillée, avant de présenter des résultats de localisation autour de la mairie du XII^e arrondissement de Paris.

6.2 Modèle cinématique 2D

Le véhicule est modélisé par son essieu arrière, à l'aide d'un modèle de type « char » (Figure 6.1), et on considère la pose $\mathbf{x} = (e, n, \psi)^T$ du milieu de l'essieu.

On suppose que le véhicule se déplace sans glissement sur un plan horizontal. Cette hypothèse est raisonnable car on se place dans un repère tangent, et on suppose que les variations d'altitude sont faibles. L'estimation de pose consiste donc à déterminer les coordonnées (e, n) ¹ de la position du véhicule sur le plan, ainsi que son cap ψ par rapport à l'est. Le modèle d'évolution est ainsi donné par

$$\begin{cases} \dot{e}(t) = v(t) \cdot \cos \psi(t) \\ \dot{n}(t) = v(t) \cdot \sin \psi(t) \\ \dot{\psi}(t) = \omega(t) \end{cases}, \quad (6.1)$$

1. Nous utilisons ici (e, n) à la place de (x, y) utilisé dans les chapitres précédents pour indiquer clairement que le calcul se fait dans un plan tangent à la terre.

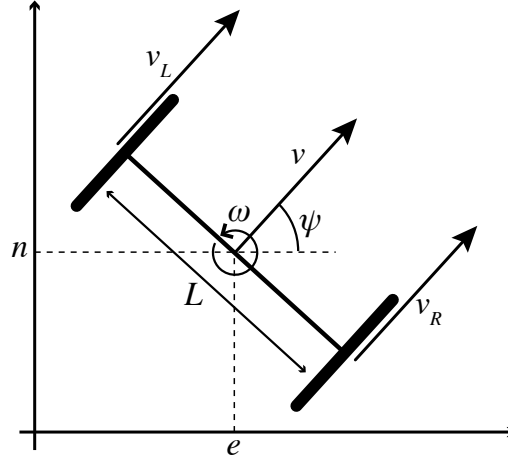


Figure 6.1: Modèle de véhicule type « char »

où la vitesse $v(t)$ peut être obtenue à partir des codeurs ABS mesurant la vitesse de rotation des roues, et où la vitesse angulaire de lacet $\omega(t)$ peut également être directement mesurée si le véhicule est équipé d'un gyromètre pour l'ESP.

Après discrétisation de (6.1), en utilisant par exemple la méthode d'Euler, on obtient le modèle d'évolution du véhicule entre deux instants t_k et t_{k+1} . En notant $T_s = t_{k+1} - t_k$, on a

$$\begin{cases} e(t_{k+1}) = e(t_k) + T_s \cdot v(t_k) \cdot \cos \psi(t_k) \\ n(t_{k+1}) = n(t_k) + T_s \cdot v(t_k) \cdot \sin \psi(t_k) \\ \psi(t_{k+1}) = \psi(t_k) + T_s \cdot \omega(t_k) \end{cases} \quad (6.2)$$

6.3 Gestion de l'historique de données

L'estimation de pose se base sur un nombre fini de données antérieures. Deux historiques de données sont utilisés afin conserver les informations nécessaires à l'estimation de pose. L'historique des positions L_p contient o pavés $[\mathbf{p}]$ représentant les positions $\mathbf{p} = (e, n)^T \in \mathbb{R}^2$ par lesquelles le véhicule est passé aux instants t_k^p , $k = 1 \dots o$. On a ainsi

$$L_p(t) = \{[\mathbf{p}](t_1^p), \dots, [\mathbf{p}](t_o^p)\}.$$

L'historique $L_u(t)$ des mesures proprioceptives contient n boîtes $[\mathbf{u}](t_i^u) = ([v](t_i^u), [\omega](t_i^u))^T$ acquises aux instants t_i^u , $i = 1 \dots n$, où v et ω représentent respectivement la vitesse linéaire et la vitesse angulaire de lacet du véhicule au milieu de l'essieu arrière. Soit

$$L_u(t) = \{[\mathbf{u}](t_1^u), \dots, [\mathbf{u}](t_n^u)\}.$$

Chaque donnée est conservée dans l'historique avec sa date d'acquisition. Il est ainsi possible de traiter des données dont la fréquence d'acquisition n'est pas constante, mais également des données dont l'ordre d'arrivée ne respecte pas la chronologie de l'acquisition (*out of sequence*) [Ranganathan et al., 2007]. Cela permet notamment de prendre en compte la latence liée au calcul d'une position GPS. En effet, pendant le temps nécessaire au calcul, entre la mesure et la communication de la position, des données proprioceptives ont le temps d'être acquises.

L'estimation de pose étant fondée sur un historique fini des positions précédentes, un soin particulier doit être accordé à la sélection des positions devant être conservées. De même, la quantité de données proprioceptives stockées doit être limitée. Les historiques de données sont donc gérés de façon à favoriser l'estimation de pose, tout en limitant la quantité de données à traiter.

6.3.1 Gestion de l'historique des positions

Comme la liste de positions conservées est de taille fixe en régime permanent, l'ajout d'une nouvelle position implique l'oubli d'une position précédente. L'ajout systématique de toutes les nouvelles positions dans l'historique n'est donc pas toujours une bonne stratégie. En effet, quand la distance couverte par l'historique est du même ordre de grandeur que l'incertitude sur les positions, il devient impossible d'estimer un cap à partir des positions conservées. Ce problème se pose en particulier à chaque arrêt du véhicule : le système commence alors à accumuler un historique de positions redondantes, tout en oubliant l'information de mouvement du passé qui contraignait le cap.

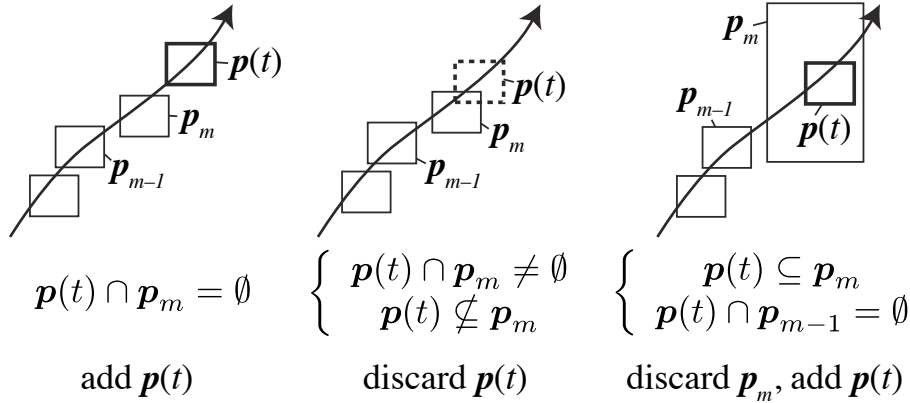


Figure 6.2: Règles d'ajout de positions dans l'historique L_p .

Pour contrer ce problème et assurer l'observabilité du cap à partir de l'historique de positions, le remplissage de l'historique de positions est régi par des critères spatiaux illustrés par la figure (6.2) :

1. Si la position candidate à l'ajout intersecte la position la plus récente de l'historique, alors elle n'est pas ajoutée. Si l'intersection est vide, alors la position est ajoutée. Cette règle peut cependant empêcher l'ajout d'une position précise, si la dernière position stockée présentait un grand domaine d'incertitude. Une seconde règle est donc nécessaire pour prendre en compte ce cas.
2. Si la position à stocker est incluse dans le domaine d'incertitude de la dernière position de l'historique, et si elle n'intersecte pas l'avant dernière position de l'historique, alors la nouvelle position remplace la dernière position de l'historique.

Afin de rendre modulable l'échantillonnage spatial des positions mémorisées, un facteur de *densité spatiale* est introduit (Figure 6.3). Ce facteur permet de régler le taux de recouvrement ou l'espacement des positions dans l'horizon. Avant de vérifier les règles d'ajout des positions, les boîtes sont « gonflées » d'un facteur inversement proportionnel à la densité spatiale. Ainsi, si la densité spatiale est réglée à 2, les largeurs des domaines de position sont réduites de moitié avant de vérifier les intersections. Cela permet d'introduire du chevauchement entre les boîtes de l'historique. Inversement, une densité spatiale de $1/2$ va introduire l'espacement d'une boîte entre deux boîtes de l'historique.

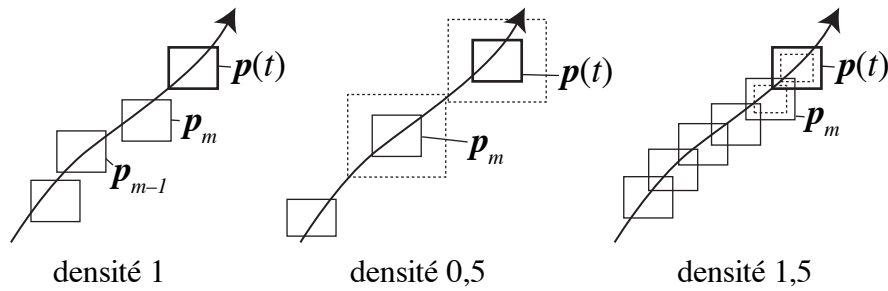


Figure 6.3: Facteur de densité spatiale dans l'historique L_p .

La diminution de la densité permet ainsi, en éloignant les boîtes, de diminuer l'incertitude sur l'estimation de cap tout en conservant un historique de taille réduite. Cette amélioration est cependant limitée par les incertitudes sur les données proprioceptives.

6.3.2 Gestion de l'historique des mesures proprioceptives

Afin de permettre l'estimation de pose, les mesures de vitesse et de vitesse de lacet doivent être disponibles entre la position la plus ancienne de l'historique et l'instant courant. Les mesures proprioceptives antérieures à la position la plus ancienne n'ont pas d'intérêt pour le calcul, aussi, dès que l'on ajoute une position dans l'historique, toutes les données proprioceptives obsolètes sont oubliées.

La longueur o de l'historique de positions est limitée. Ainsi, en considérant T_e^u la période constante d'acquisition des mesures proprioceptives et T_e^p la période constante des informations de position, le nombre n de mesures proprioceptives à conserver peut être borné par

$$n \leq \frac{o \cdot T_e^u}{T_e^p}. \quad (6.3)$$

Néanmoins, en introduisant des contraintes spatiales pour la mise à jour de l'historique, l'échantillonnage des positions ne se fait plus à période constante, et il n'est plus possible de borner le nombre de mesures proprioceptives conservées par l'équation (6.3). Ainsi, lorsque le véhicule est à l'arrêt ou lorsque que l'incertitude sur la position empêche la mise à jour de l'historique L_p , le nombre de données proprioceptives stockées croît rapidement. Afin de limiter l'occupation mémoire et l'augmentation du temps de calcul, deux mécanismes ont été mis en place pour limiter la taille de l'historique des données proprioceptives L_u .

6.3.2.1 Suppression des données redondantes à l'arrêt

Lorsque les mesures proprioceptives successives prennent des valeurs identiques, il n'est pas nécessaire de toutes les conserver. Le premier mécanisme de réduction de la quantité de mesures stockées consiste donc à ne pas mettre à jour l'historique des mesures proprioceptives si la nouvelle mesure est égale à la précédente.

Cette méthode est censée être particulièrement efficace lorsque le véhicule est à l'arrêt, les mesures proprioceptives étant théoriquement nulles jusqu'au redémarrage. Cependant, en raison du bruit sur la mesure de vitesse de lacet, les mesures successives ne sont pas toujours identiques à l'arrêt. Pour tenir compte de la présence de l'essieu avant, empêchant toute rotation à vitesse nulle, nous allons corriger les mesures du gyromètre. Lorsque la vitesse mesurée v est nulle, la vitesse de lacet ω est forcée à 0. Ainsi, dès que le véhicule s'arrête, les mesures successives sont identiques et ne sont donc plus dupliquées dans l'historique L_u .

6.3.2.2 Ré-échantillonnage des données sur l'historique

La simplification des mesures à l'arrêt et la suppression des doublons, bien qu'elles permettent d'efficacement réduire le nombre de mesures conservées entre deux instants, ne suffisent pas à garantir une borne sur la taille de l'historique L_u . Nous allons donc imposer une taille maximale $n_{\max}$ à l'historique L_u , et ré-échantillonner l'ensemble des mesures stockées si jamais cette taille vient à être dépassée.

Quand une mesure proprioceptive doit être ajoutée et $n = n_{\max}$, l'historique proprioceptif est plein. On effectue donc une décimation des données en ne conservant qu'une valeur sur deux. L'historique est donc désormais à moitié vide et l'on peut ajouter la nouvelle valeur. Le pas d'échantillonnage des données stockées dans L_u

n'est pas constant : lors des arrêts du véhicule, seule une mesure nulle est conservée. La décimation des données doit tenir compte de ce fait, sous peine de supprimer les arrêts du véhicule et fausser les calculs. Ainsi, lors de la décimation, si une mesure nulle est rencontrée, elle n'est pas supprimée, de même que la mesure suivante.

On peut remarquer que dans le cas où plusieurs ré-échantillonnages sont appliqués à différents instants, les données les plus récentes de l'historique sont moins altérées que les mesures les plus anciennes. De cette façon, les mesures les plus dégradées sont les premières à être oubliées.

6.4 Contracteur robuste de pose sur l'horizon

6.4.1 Contraintes liées aux positions de l'historique

Chaque position dans l'historique, associée aux données proprioceptives et au modèle d'évolution, contraint la pose du véhicule à tout instant t dans un ensemble en forme d'hélice épaisse dans l'espace des poses 2D (Figure 6.4). La forme d'hélice vient du fait que le cap n'est pas mesuré.

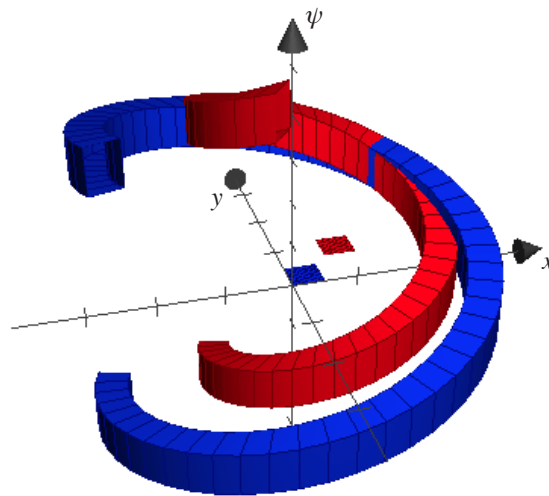


Figure 6.4: Prédiction de la pose au temps t à partir de deux positions. L'ensemble des poses compatibles avec une position a une forme d'hélice.

La figure 6.5 montre l'estimation de la pose du véhicule à l'instant t à partir de deux positions dans l'historique aux instants t_1 et t_2 , dans le cas où le véhicule roule tout droit. Chaque position contraint la pose du véhicule dans un ensemble de l'espace des poses, dont la projection dans le plan est un ensemble en forme d'anneau. La pose au temps t appartient ainsi à l'intersection des deux contraintes imposées par

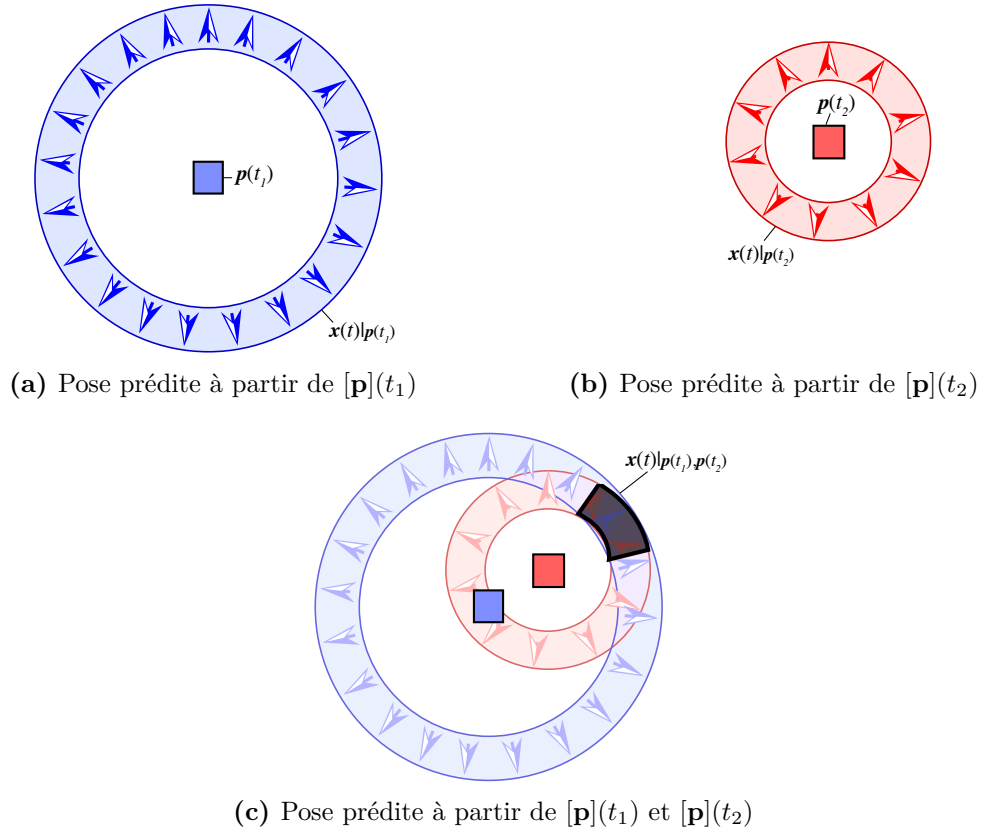


Figure 6.5: Estimation de la pose courante (au temps t) à partir de deux positions et de données proprioceptives. Dans cet exemple, le déplacement est supposé linéaire. *Vue de dessus, les flèches représentent le cap.*

chaque information de position :

$$\mathbb{S}_x(t) = \bigcap_{k=1\dots o} \mathbb{S}_x^+(t) \mid [\mathbf{p}](t_k^p), [\mathbf{u}](t_1^u), \dots, [\mathbf{u}](t_n^u) \quad (6.4)$$

où $\mathbb{S}_x^+$ désigne la prédiction ensembliste de la pose connaissant une position de l'historique et les données des capteurs proprioceptifs.

Afin de prendre en compte la présence éventuelle de positions erronées dans l'historique, nous allons considérer l'intersection q -relaxée des ensembles de poses prédites. Ainsi, la solution ensembliste de pose calculée sera garantie robuste à la présence de q boîtes ne contenant pas la vraie position dans l'historique. L'estimation robuste de pose consiste ainsi à calculer la position et le cap du véhicule à l'instant t , connaissant un nombre fini de positions précédentes et l'historique des valeurs relevées par les capteurs proprioceptifs, le tout sous l'hypothèse que jamais plus de q positions mémorisées sont fausses.

$$\mathbb{S}_x^{\{q\}}(t) = \bigcap_{k=1\dots o}^{\{q\}} \mathbb{S}_x^+(t) \mid [\mathbf{p}](t_k^p), [\mathbf{u}](t_1^u), \dots, [\mathbf{u}](t_n^u) \quad (6.5)$$

6.4.2 Contraintes associant historique de position et route

Lorsque l'on dispose d'une carte de l'espace roulant, l'ensemble des poses à l'instant t compatibles avec une position de l'historique se trouve bien plus contraint. Il ne s'agit pas simplement de l'intersection de l'ensemble hélicoïdal obtenu sans contrainte avec le produit cartésien de la carte et de l'espace des caps. En effet, les données proprioceptives associées au modèle d'évolution contraignent la trajectoire du véhicule entre les deux instants, et il est nécessaire que cette trajectoire soit compatible avec la carte. On le remarque en particulier dans la figure 6.6b, où la partie gauche de l'ensemble des poses prédites est réduit afin de satisfaire la contrainte de trajectoire linéaire entre t_2 et t .

L'algorithme 6.1 présente un contracteur pour l'ensemble $\mathbb{S}_x^+(t) \mid [\mathbf{p}](t_k^p), [\mathbf{u}](t_1^u), \dots, [\mathbf{u}](t_n^u)$. Il utilise la propagation de contraintes avec la fonction d'évolution f de l'équation (6.2), la carte de l'espace roulant $\mathcal{M}$ et les mesures proprioceptives. Pour la rétropropagation des contraintes, le modèle inverse d'évolution f^{-1} est utilisé. La figure (6.7) résume l'ensemble des contraintes s'appliquant sur la pose à l'instant t , connaissant une position à l'instant t_k .

Les données proprioceptives sont généralement acquises à une fréquence élevée, si bien qu'entre deux pas de temps, la position du véhicule évolue très peu. Or, la contrainte d'appartenance à l'espace roulant est censée être appliquée à chaque point de la trajectoire du véhicule. Cela entraîne un nombre élevé d'appels du contracteur de route. On peut remarquer qu'il est inutile d'appeler le contracteur de route à chaque pas de temps si les déplacements sont minimes. Pour réduire le temps de calcul, sans toutefois réduire significativement l'efficacité de la contrainte

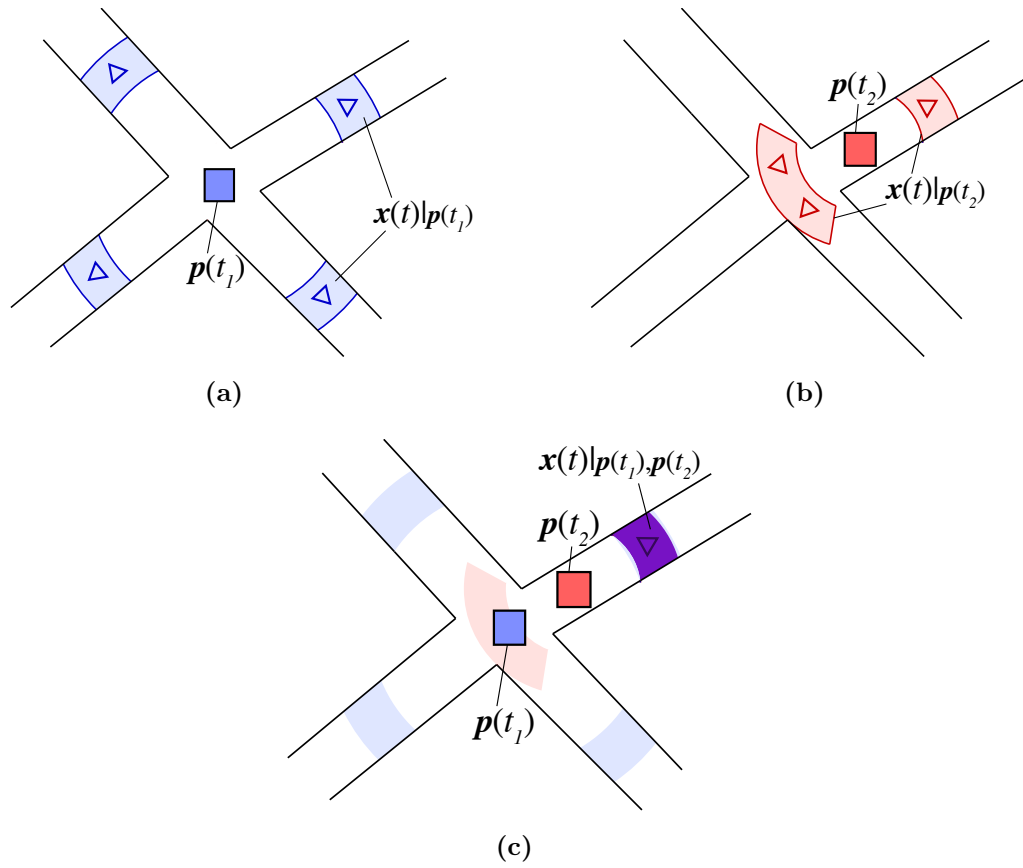


Figure 6.6: Estimation de la pose courante à partir de deux positions, de mesures du déplacement et d'une carte. Le déplacement est ici supposé linéaire.

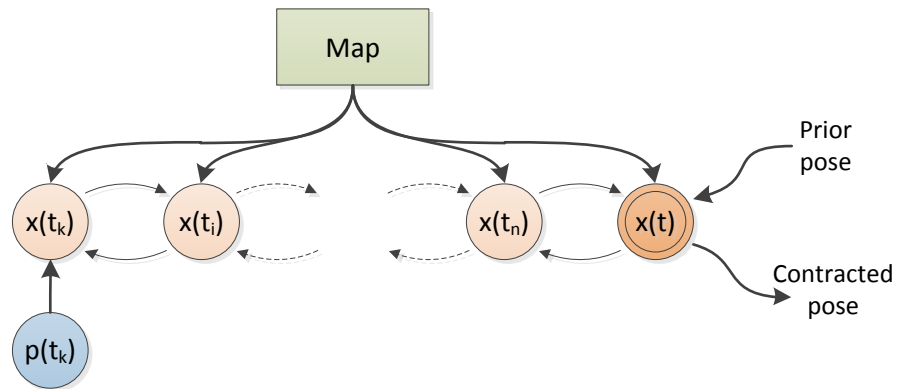


Figure 6.7: Propagation de contraintes sur la pose $[x](t)$ avec une position $[p](t_k)$ et les mesures proprioceptives. La contrainte cartographique est appliquée à chaque pas de temps.

Algorithme 6.1 $\mathcal{C}_{p_k}$ (in : $[\mathbf{x}](t_n^u), [\mathbf{p}](t_k^p), L_u, \mathcal{M}$)

Contracteur de pose par propagation de contraintes avec une position $[\mathbf{p}](t_k^p)$, les mesures proprioceptives et la contrainte de route

```

1: find  $j$  that verifies  $t_{j-1}^u < t_k^p \leq t_j^u$ 
2: for  $i = n - 1$  down to  $j$  do
3:    $[\mathbf{x}](t_i^u) := f^{-1}([\mathbf{x}](t_{i+1}^u), [\mathbf{u}](t_i^u))$ 
4:    $[\mathbf{x}](t_i^u) := \text{road\_contract}([\mathbf{x}](t_i^u), \mathcal{M})$ 
5: end for
6:  $[\mathbf{x}](t_k^p) := f^{-1}([\mathbf{x}](t_j^u), [\mathbf{u}](t_{j-1}^u))$ 
7:  $[\mathbf{x}]_{1..2}(t_k^p) := [\mathbf{x}]_{1..2}(t_k^p) \cap [\mathbf{p}](t_k^p)$ 
8:  $[\mathbf{x}](t_j^u) := [\mathbf{x}](t_j^u) \cap f([\mathbf{x}](t_k^p), [\mathbf{u}](t_{j-1}^u))$ 
9: for  $i = j$  to  $n - 1$  do
10:   $[\mathbf{x}](t_{i+1}^u) := [\mathbf{x}](t_{i+1}^u) \cap f([\mathbf{x}](t_i^u), [\mathbf{u}](t_i^u))$ 
11:   $[\mathbf{x}](t_{i+1}^u) := \text{road\_contract}([\mathbf{x}](t_{i+1}^u), \mathcal{M})$ 
12: end for
13: return  $[\mathbf{x}](t_n^u)$ 

```

cartographique, la contrainte de route va donc être appliquée selon un critère de distance parcourue. L'entrée odométrique est utilisée pour comptabiliser la distance parcourue depuis la dernière contraction avec l'espace roulant. Quand cette distance dépasse un seuil (4 mètres dans notre cas), le contracteur de route est appelé et le compteur réinitialisé.

Une approximation extérieure de $\mathbb{S}_x^{\{q\}}(t)$ est calculée par l'algorithme SIVIA avec le contracteur q -relaxé $\mathcal{C}_{q\text{rel}}([\mathbf{x}_0], q, \mathcal{C}_{p_1}, \dots, \mathcal{C}_{p_o})$. Les contraintes apportées par chaque position de l'historique sont mises en œuvre par le contracteur $\mathcal{C}_{p_k}$ défini par l'algorithme 6.1. Ainsi, q contraintes parmi les o contraintes liées aux positions stockées dans l'historique sont relaxées.

6.5 Calcul du risque

Le choix d'utiliser un historique fini de données permet, outre la robustesse à la présence de positions aberrantes, de permettre une estimation à risque constant. En effet, il est montré dans [Jaulin, 2011b] que la succession de prédictions et estimations à partir des résultats précédents fait tendre vers zéro la confiance que l'on peut accorder dans l'estimation. En n'effectuant l'estimation de pose qu'à partir d'un nombre restreint de positions, et sans tenir compte des résultats d'estimation de pose précédents, le risque d'intégrité de la solution n'est lié qu'au risque associé à chaque donnée de l'historique.

Nous considérons ici que les données proprioceptives sont garanties et que le modèle d'évolution en 2D est valide, aussi le risque ne porte que sur les positions conservées dans l'historique. Le calcul de pose étant basé sur l'inversion ensembliste q -relaxée,

nous pouvons nous appuyer sur les calculs réalisés dans la section 4.4 page 98 pour quantifier le risque, en supposant que les erreurs sur les informations de position sont indépendantes. Une borne supérieure du risque R de la solution de pose calculée par inversion ensembliste robuste est ainsi obtenue à partir de l'équation (4.9) :

$$R \leq 1 - \sum_{k=o-q}^o \binom{o}{k} p_{pos}^k (1 - p_{pos})^{o-k}, \quad (6.6)$$

où p_{pos} est la probabilité qu'une boîte de l'historique L_p contienne bien la vraie position.

En inversant l'inéquation (6.6), on peut déterminer une borne inférieure de la confiance requise dans chaque information de position pour garantir que le risque d'intégrité de l'estimation de pose sera inférieur à R .

Si on utilise la méthode d'inversion ensembliste robuste pour calculer les positions enregistrées dans l'historique, alors on a

$$R \leq 1 - \sum_{k=o-q}^o \binom{o}{k} (1 - r)^k r^{o-k}, \quad (6.7)$$

où r est le risque associé à la zone de localisation calculée à partir des pseudodistances, déterminé à partir de l'équation (4.9). On peut alors utiliser la méthode présentée dans la section 4.4.3 pour déterminer les bornes des intervalles d'erreurs associés aux pseudodistances, afin de satisfaire les exigences sur le risque d'intégrité de la pose.

6.6 Implémentation en temps-réel

L'algorithme d'estimation robuste de pose prend en entrée un historique de positions et de mesures proprioceptives, ainsi qu'une représentation 3D surfacique de la route.

Afin de fournir les entrées nécessaires à l'algorithme à partir des données capteur, on peut construire un système de localisation autour de trois tâches principales représentées sur la figure 6.8 :

- Une première tâche détermine un domaine de localisation à partir des observables GPS mesurés. Une solution consiste à utiliser l'algorithme de calcul interne au récepteur, qui fournit une solution PVT (Position, Vitesse et Temps) à chaque époque de mesures. On déduit alors le domaine d'incertitude à partir des ellipses de confiance associées à la solution de position. Cependant, on peut vouloir profiter de l'information cartographique disponible, puisque cette dernière est utilisée dans l'estimation de pose. Aussi, on peut mettre en place un calcul en couplage serré utilisant les mesures GPS brutes et la contrainte de l'espace roulant. L'algorithme ensembliste présenté au chapitre 5 peut ainsi être implémenté dans cette tâche de fusion serrée GPS-carte.

- La seconde tâche est chargée de gérer les historiques de données. Elle récupère les positions calculées par la première tâche, ainsi que les mesures provenant des capteurs proprioceptifs du véhicule, et gère les positions et mesures stockées en fonction des règles définies dans la section 6.3.
- Enfin, la dernière tâche consiste en l’algorithme d’estimation robuste d’un domaine de confiance de la pose à partir des historiques de données et de la carte de l’espace roulant.

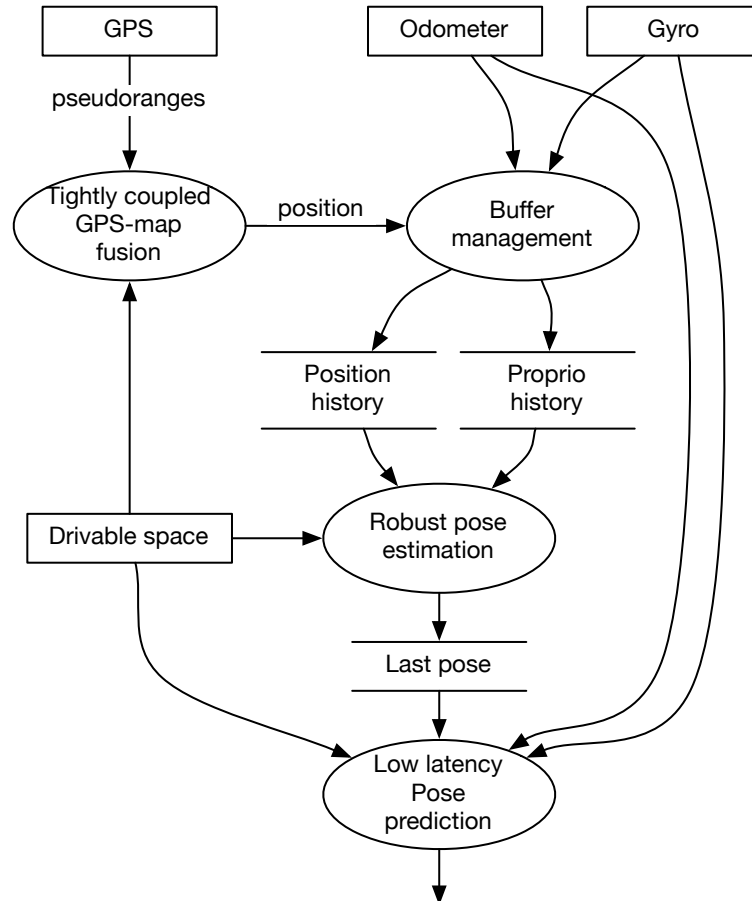


Figure 6.8: Implémentation en temps-réel du système d’estimation robuste de pose.

Le calcul en couplage serré GPS-carte d’une zone d’incertitude de localisation est assez rapide. Au contraire, l’estimation robuste de la pose sur l’historique de données est un processus coûteux en temps de calcul. Le système doit donc être conçu de façon à permettre la fourniture d’une pose en temps réel, malgré la grande latence de l’estimateur ensembliste robuste. L’estimation de pose est ainsi scindée en deux tâches complémentaires, travaillant en parallèle :

- Une tâche d’estimation robuste de la pose à partir de la carte 3D, des positions précédentes et des mesures proprioceptives. Elle est réalisée à chaque fois que de nouvelles positions sont mémorisées dans l’historique, et que la tâche d’estimation

n'est pas déjà active. En raison des temps de calculs élevés, les poses calculées par cette tâche sont retournées assez tard et à une fréquence en général trop faible pour une utilisation temps réel.

- Une tâche de prédiction rapide de la pose, qui calcule la pose prédite à chaque fois qu'une mesure proprioceptive est reçue. Ce calcul se base sur la dernière pose calculée, la carte et le modèle d'évolution associé aux mesures proprioceptives. À partir du sous-pavage de pose, une étape de prédiction du *box particle filter* [Abdallah et al., 2008] est effectuée. Le calcul est rapide, et se base sur estimation de pose robuste aux positions aberrantes. On obtient ainsi une faible latence, tout en conservant la robustesse.

6.7 Résultats

Tout comme l'algorithme de positionnement intègre contraint par la route décrit dans le chapitre précédent, l'algorithme ensembliste d'estimation de pose présenté dans ce chapitre a été testé sur les données acquises pour le projet CityVIP autour de la mairie du XII^e arrondissement de Paris, afin de profiter de la cartographie 3D de l'espace roulant. Les détails concernant ce jeu de données sont fournis dans l'annexe C.3. Les résultats présentés ont été obtenus par rejou en temps-réel des données.

Les données GPS utilisés sont les pseudodistances acquises à 2 Hz par un récepteur *Septentrio PolaRx2*, équipé d'une antenne polarisée de qualité géodésique. Les pseudodistances sur la fréquence L1 dont le rapport signal à bruit est inférieur à 35 dBHz sont éliminées, afin de réduire les mesures aberrantes. Comme le montre l'histogramme de la figure 6.9, ces données sont particulièrement difficiles pour le calcul de solution GPS. En effet, 88% des époques de mesure ont moins de 4 satellites utilisables, et il n'y a pas plus de deux pseudodistances disponibles pendant 56% de l'essai.

Les positions sont calculées par inversion ensembliste, en couplage serré à partir des pseudodistances GPS et de la carte de l'espace roulant, avec un risque $r = 10^{-4}$. Les mesures de pseudodistances sont supposées avoir une erreur gaussienne de variance égale à 1 m. En suivant la méthode décrite dans la section 4.4 (page 98), on a ainsi des bornes d'erreur de ± 4.15 m sur chaque pseudodistance quand trois satellites sont utilisés. Ces bornes passent à ± 3.89 m quand une seule pseudodistance est utilisée. Les boîtes englobantes des positions ainsi calculées sont mémorisées dans l'historique.

La figure 6.10 indique, sur la trajectoire de référence, le centre de gravité de chaque domaine de localisation calculé à partir des pseudodistances et de la carte. Seules les positions calculées avec au moins deux satellites sont représentées, car les solutions calculées avec un seul satellite sont souvent ambiguës. Cette figure met en évidence la mauvaise visibilité satellitaire dans le canyon urbain que constituent les rues étroites

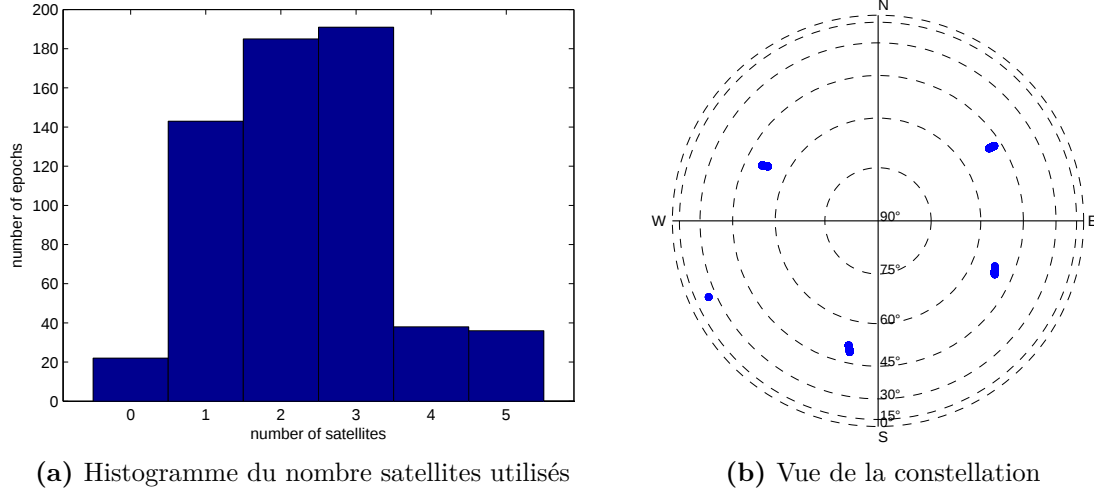


Figure 6.9: Visibilité satellitaire après seuillage sur le rapport signal bruit $C/N_0 \geq 35\text{dBHz}$

de la zone d'essai. On remarque, outre l'absence de continuité de la localisation par manque de satellites, la présence de sauts lorsque le nombre de pseudodistances est faible et que l'on approche d'une intersection. Le peu de mesures ne permet pas en effet de contracter le domaine d'incertitude sur un unique tronçon de route. De par la difficulté du positionnement époque par époque, l'estimation robuste de pose basée sur l'historique des positions prend, dans de telles conditions difficiles, tout son intérêt.

La longueur de l'historique est fixée à 15 positions ($o = 15$), et le facteur de densité est réglé à 2. On autorise ainsi le chevauchement des boîtes dans l'historique. L'estimation de pose est faite sous l'hypothèse d'au maximum une position erronée dans l'horizon ($q = 1$). Les positions étant calculées avec un risque $r = 10^{-4}$, on peut calculer le risque maximal lié à l'estimation de pose en s'appuyant sur l'équation (6.7) :

$$R \leq 1 - \sum_{k=o-q}^o \binom{o}{k} (1-r)^k r^{o-k}$$

$$R \leq 1 - 15 \times 0.9999^{14} \times 0.0001 - 0.9999^{15}$$

On obtient ainsi un risque $R \leq 1.05 \cdot 10^{-6}$ pour chaque estimation de position.

L'estimation robuste de pose fournit des ensembles solutions sous forme de sous-pavages garantis de contenir l'ensemble des poses compatibles avec l'historique des positions. Cette représentation sous forme de domaine d'incertitude offre une bonne information sur l'incertitude de localisation et permet de gérer les hypothèses multiples. Cependant, une estimée ponctuelle est souvent requise par le système utilisant la pose en entrée, tel qu'un algorithme de contrôle. L'estimée de la pose est calculée

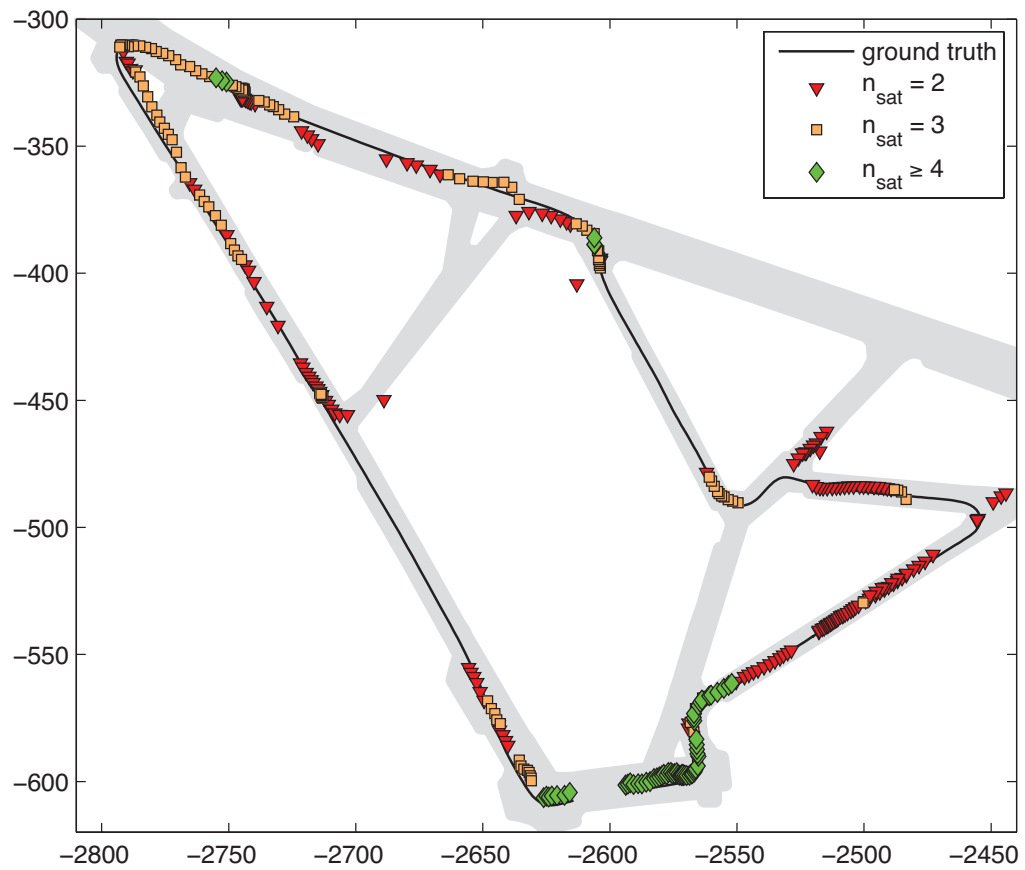


Figure 6.10: Centre de gravité des positions calculées époque par époque par fusion des pseudodistances GPS avec l'espace roulant. *Les symboles indiquent le nombre de satellites utilisés pour le calcul.*

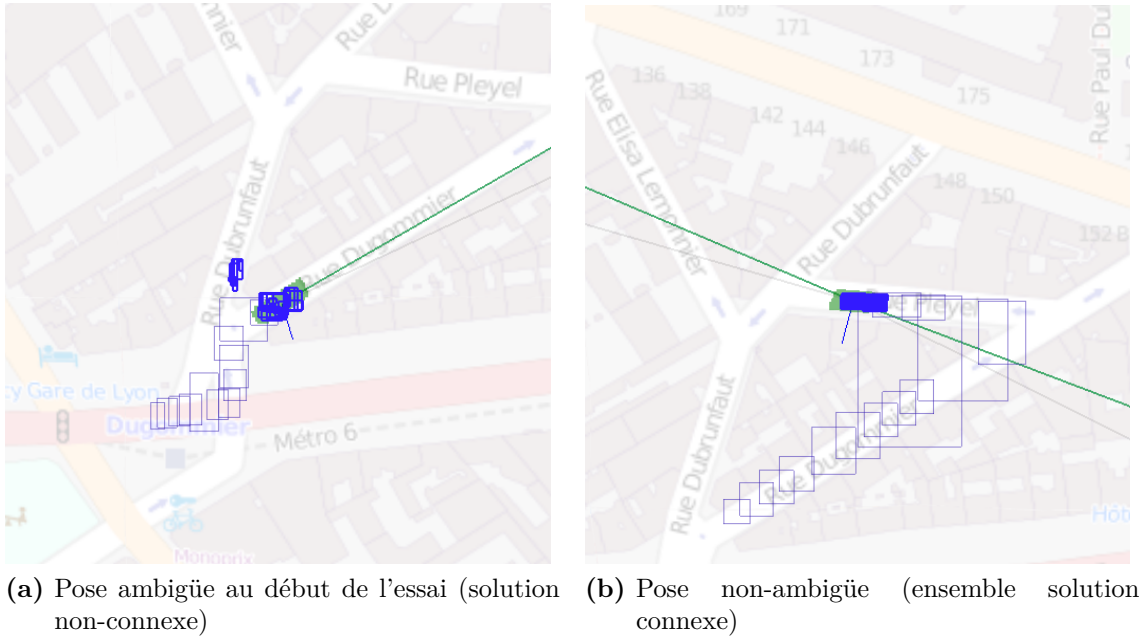


Figure 6.11: Sous-pavage solution, représenté par des rectangles épais à deux époques sur la carte OpenStreetMap du quartier. *Les rectangles fins figurent l'historique de positions. Les lignes de vues vers les satellites sont en vert.*

comme le centre du gravité du sous-pavage. Au début de l'essai, l'estimée ponctuelle de la pose ne suit que grossièrement la trajectoire de référence (Figure 6.12). En effet, l'algorithme d'estimation démarre sans connaissance *a priori* de la pose du véhicule : le domaine initial pour le cap est $[\psi] = [-\pi, \pi]$. Quand l'historique de positions n'est pas encore rempli, on a des solutions non connexes représentant plusieurs hypothèses de pose (Figure 6.11a). Le centre de gravité est influencé par les hypothèses multiples, et se retrouve donc éloigné de la trajectoire de référence. Quand assez de données ont été stockées dans l'historique, l'ambiguïté de pose est résolue (Figure 6.11b). Les boîtes correspondant aux mauvaises hypothèses « meurent » en raison de leur inconsistance avec les données et la carte, et une estimée plus précise de la pose est obtenue.

Le sous-pavage solution de pose est consistant avec la trajectoire de référence sur l'ensemble de la boucle. La figure 6.13 montre l'erreur ensembliste (bornes inférieures et supérieures du domaine) sur les trois composantes du domaine de pose estimé, ainsi que l'erreur commise en prenant le centre de gravité du sous-pavage comme estimée de la pose. Pendant les 30 premières secondes de l'essai, le cap n'est pas bien initialisé. L'historique ne contient en effet pas assez de positions pour contraindre suffisamment le domaine de $[\psi]$. Quand assez de positions ont été mémorisées, le domaine du cap est fortement contracté et le centre de gravité fournit une estimée précise. L'estimation du cap permet une meilleure réjection des mauvaises hypo-

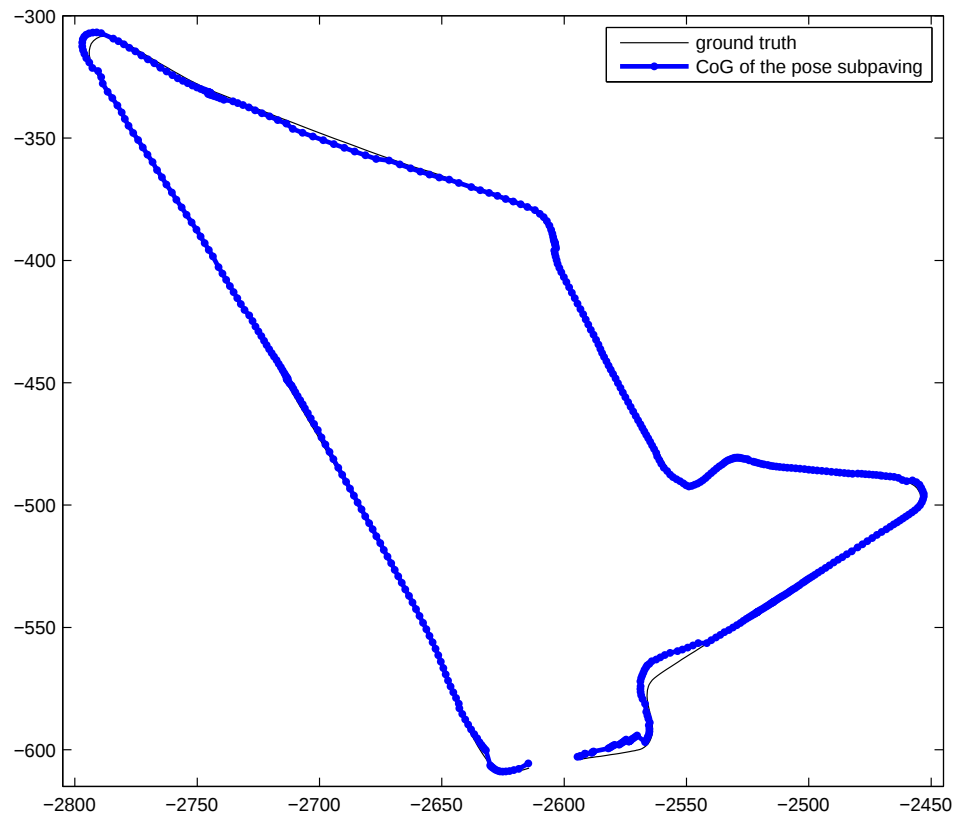


Figure 6.12: Trajectoire 2D du centre de gravité du sous-pavage solution (courbe bleue épaisse). *La vraie position est représentée par la courbe fine et noire.*

thèses de localisation aux croisements.

Durant le parcours, l'erreur longitudinale est globalement plus élevée que l'erreur latérale. Ceci est principalement dû à l'importance de la contrainte cartographique. L'itinéraire suit des routes étroites, ainsi, une fois l'ambiguïté sur le segment de route courante levée, la contrainte cartographique limite l'extension latérale du domaine d'incertitude, tandis que les pseudodistances GPS contraignent principalement la position dans le sens de la route. Les satellites situés dans l'axe de la route sont généralement visibles, tandis que les satellites situés sur les cotés du véhicule sont le plus souvent masqués par les constructions. La carte étant supposée bonne, la source d'erreur principale vient du GPS, et, en raison du faible nombre de mesures disponibles, les biais dus à l'imprécision des corrections et aux multitrajets influent directement sur le positionnement le long de la route. Cela est particulièrement sensible quand seulement un ou deux satellites sont utilisés.

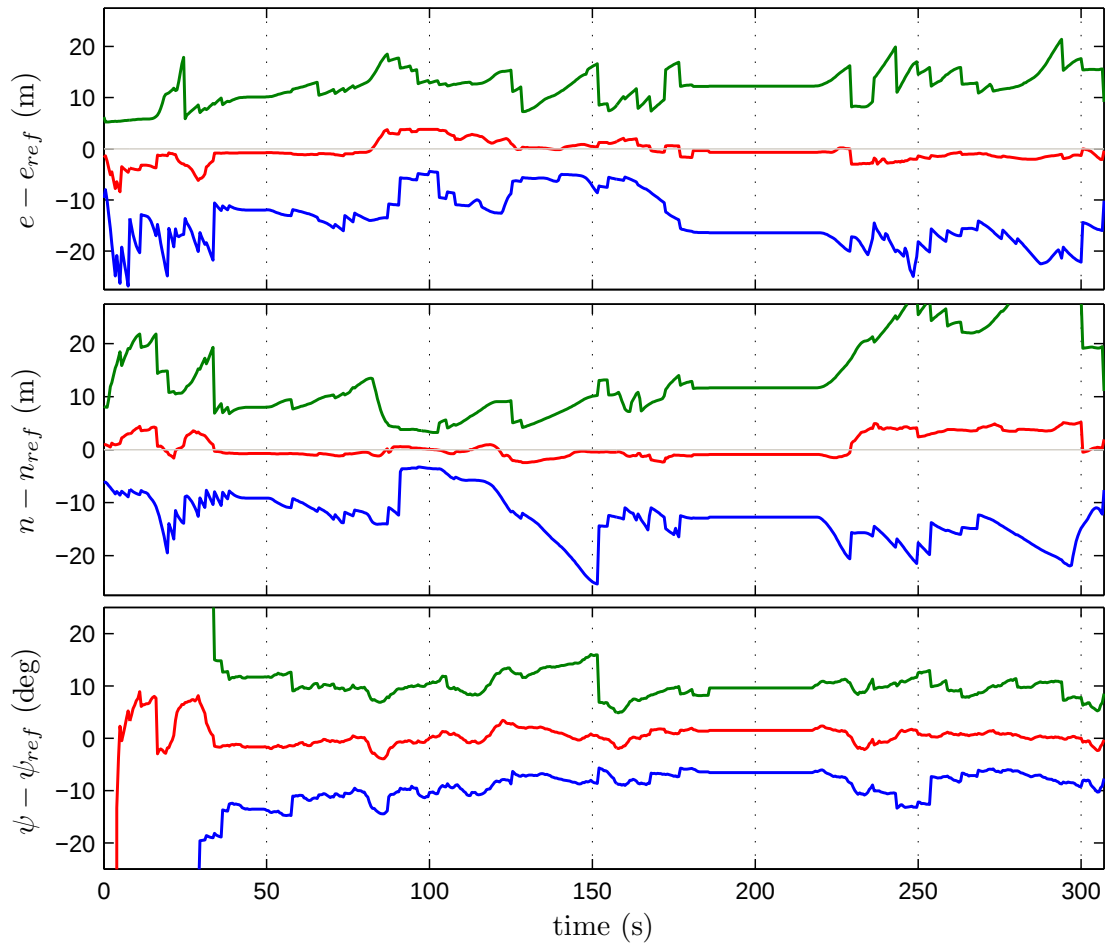


Figure 6.13: Erreurs d'estimation de position et de cap par le centre de gravité du sous-pavage (en rouge) et bornes de la zone de confiance (courbes vertes et bleues).

Sur la figure 6.14 est tracée la fonction de répartition empirique de l'erreur horizontale de position, commise en utilisant le centre de gravité du sous-pavage comme estimateur de la pose. L'erreur horizontale de position est inférieure à 5.25 m pendant 95% du temps sur l'essai. La moitié des estimations affichent une erreur position inférieure à 2 m.

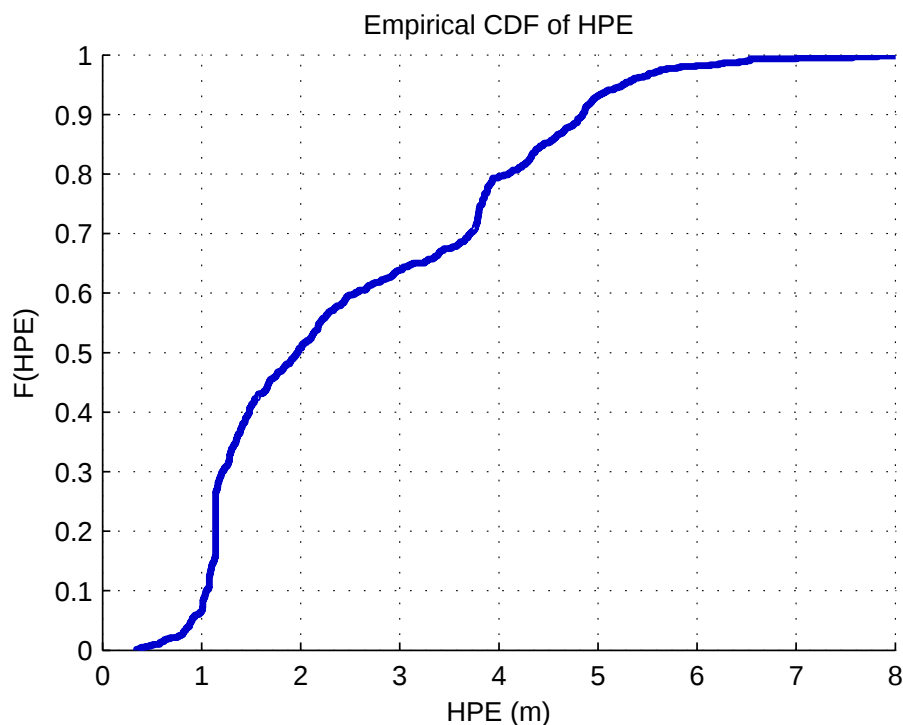


Figure 6.14: Fonction de répartition empirique de l'erreur de position horizontale (HPE).

6.8 Conclusion

L'utilisation d'un buffer de données est une solution pour apporter suffisamment d'informations redondantes à un module de positionnement et ainsi fournir une solution intègre de bonne qualité. En effet, si l'on compare les résultats de cette approche avec ceux du chapitre précédent, on peut observer une nette amélioration. Ainsi l'erreur horizontale de position est inférieure à 5.25 m 95% du temps, alors que cette valeur était de 9.5 m sans l'utilisation de l'horizon de données. Les domaines d'incertitude sont également réduits. Ainsi, pour un niveau d'exigence spécifié, la continuité de service du positionnement s'en trouve nettement améliorée.

Les positions utilisées en entrée de l'estimation de pose étaient, dans notre exemple, calculées avec une méthode ensembliste non-robuste aux mesures aberrantes. L'intégrité de positionnement vient de l'utilisation d'un historique fini de positions, parmi

lesquelles on tolère un nombre de données erronées sans risque pour la solution de pose. Les positions aberrantes dans l'historique peuvent être détectées (en utilisant la méthode décrite dans la section 3.6 page 85), et les données erronées non-détectées sont oubliées au fur et à mesure du renouvellement de l'historique.

La contrainte cartographique associée aux mesures proprioceptives permet de fortement contraindre la solution. Le réglage des paramètres (taille de l'horizon, densité des positions et nombre de positions aberrantes tolérées) se fait en s'assurant que l'historique permettra de suffisamment contraindre le cap du véhicule, et que l'on peut déterminer une borne du nombre de mesures aberrantes dans l'horizon.

Chapitre 7

Conclusion et perspectives

7.1 Conclusion

Ces travaux de thèse ont été consacrés à l'étude de méthodes de localisation intégrées pour la navigation de véhicules en milieu urbain. Nous avons donc présenté les systèmes de localisation par satellites, qui offrent l'avantage d'une couverture globale et d'un coût réduit pour le récepteur. Ces systèmes ne sont néanmoins pas infaillibles, aussi un moyen de surveillance de l'intégrité de la position calculée est nécessaire dès que l'application peut mettre en danger la vie des personnes. C'est le rôle des mécanismes classiques d'intégrité que nous avons décrits, utilisées notamment dans le domaine de l'aéronautique.

L'intégrité consiste à fournir une borne de l'erreur commise en position, associée à un risque, nommé *risque d'intégrité*, de fournir une position dont l'erreur dépasse l'erreur maximale spécifiée. Ce problème peut être vu comme le calcul d'un domaine d'incertitude de position, contenant l'ensemble des positions possibles de l'utilisateur pour un risque donné. Cela rend particulièrement pertinente l'étude des méthodes ensemblistes par intervalles. Nous avons donc présenté dans ce document les principes du calcul par intervalles et son extension aux boîtes, avant d'introduire des méthodes d'inversion ensembliste et leur robustification aux mesures aberrantes.

L'application du calcul ensembliste au positionnement par satellites à partir des pseudodistances mesurées par le récepteur permet l'estimation robuste d'un domaine de localisation. Les propriétés de l'inversion ensembliste à partir de mesures à erreurs bornées rendent possible la détection d'anomalies de mesure. L'utilisation de l'inversion ensembliste q -relaxée permet l'identification et l'exclusion des mesures aberrantes. On peut de plus calculer un domaine d'incertitude de position sous hypothèse de mesures aberrantes non-détectées. L'ajout d'une contrainte d'altitude provenant d'un modèle numérique de terrain améliore, outre la précision de la localisation, les capacités de détection et d'exclusion de défaut de la méthode.

La formulation du problème de localisation sous forme de contraintes liées aux mesures de pseudodistance facilite la fusion avec un *a priori* cartographique, afin de calculer une solution en *couplage serré*. En effet, le schéma classique dans lequel l'étape de *map matching*, permettant d'identifier le tronçon de route sur lequel le

véhicule se trouve avant de l'utiliser pour contraindre la position, n'a plus lieu d'être : on peut utiliser la carte dans sa globalité comme une contrainte sur la position, sans supposer que le véhicule évolue sur une route en particulier. Nous avons ainsi pu utiliser une carte précise de l'espace roulant, sous forme d'un maillage triangulaire en trois dimensions. La nature ensembliste de la solution permet de représenter précisément les cas d'ambiguïté qui apparaissent lorsque la précision du GPS ne suffit pas à identifier un unique segment de route. De plus, la méthode est capable de gérer les problèmes non-linéaires, que l'on rencontre en particulier aux intersections.

En milieu urbain, la visibilité satellitaire est fortement réduite par les bâtiments, qui sont également cause de mesures aberrantes par réflexion. Cela rend le positionnement par satellites particulièrement ardu, même si une carte est disponible. Une première conséquence est le risque de calculer un ensemble solution erronée si le nombre de mesures aberrantes à considérer n'est pas correctement estimé ; on a alors une perte d'intégrité. Une autre conséquence est la diminution de la continuité du service de positionnement, soit par la détection d'une mesure aberrante se solvant par un ensemble solution vide, soit par manque de mesures, conduisant à un domaine de localisation trop étendu pour être utilisé. Pour pallier ce manque de continuité, tout en assurant l'intégrité de la localisation, nous avons utilisé un historique des derniers domaines de localisation calculés, associé aux mesures de vitesse linéaire et de vitesse de lacet fournies par les capteurs intégrés au véhicule. Une méthode d'inversion ensembliste robuste est alors appliquée pour déterminer la pose courante du véhicule, en tenant compte de l'éventuelle présence d'une position aberrante dans l'historique.

Les méthodes de localisation présentées ont été implémentées en temps-réel et testées sur des données exploitées en mode « rejeu » ou bien directement à bord du véhicule expérimental CARMEN du laboratoire.

Un intérêt majeur des méthodes ensemblistes mis en avant dans ces travaux est la capacité de caractériser un ensemble solution quand on a peu de données, en particulier dans le cas sous-déterminé où un ensemble solution peut toujours être calculé. Cet ensemble solution pourra éventuellement être affiné quand d'autres informations seront disponibles. La plupart des méthodes classiques reposent sur la linéarisation du problème. Cependant, quand on dispose de trop peu de données, il n'est pas toujours possible de définir un point de linéarisation. Une mauvaise linéarisation peut également amener à tomber dans un minimum local.

La plupart des systèmes classiques ont en entrée et sortie des valeurs associées à leurs variances ou à une matrice de variances-covariances, ce qui n'est pas directement compatible avec la représentation ensembliste. Nous avons traité le problème de ces entrées probabilistes en les transformant en intervalles associés à un risque dépendant de la fonction de répartition de l'erreur. À l'inverse, la sortie d'une méthode ensembliste doit pouvoir être convertie en une représentation par les deux premiers moments statistiques avec des bornes de domaine d'intégrité. Ce problème présente des difficultés particulières dans le cas où une solution ambiguë a été calculée.

7.2 Contributions

Ces travaux de thèse sont la première application de méthodes ensemblistes robustes par intervalles à la localisation par satellites avec des données brutes, dans l'objectif de caractériser des domaines de confiance. Un calcul du risque *a priori* lié à l'estimation ensembliste robuste a également été introduit, accompagné d'une méthode de choix des bornes des bornes d'erreur en fonction du risque d'intégrité toléré.

Un contracteur pour la contrainte d'appartenance à un espace roulant sous forme d'un maillage triangulaire en trois dimensions a également été introduit.

La détection et l'identification des mesures aberrantes ont été implémentées au fil de l'exécution de l'algorithme robuste d'inversion ensembliste. Lorsqu'une mesure GPS aberrante est détectée, le lieu possible de réflexion peut être représenté sur la carte, sous hypothèse d'un trajet sans ligne de vue directe.

Une méthode ensembliste de fusion robuste de positions avec une carte 3D de l'espace roulant et des mesures proprioceptives a été proposée. Elle est basée sur un historique fini de données, associé à une stratégie d'échantillonnage spatial limitant le nombre de données à traiter.

Enfin, les différentes propositions ont donné lieu à des études expérimentales sur données réelles, avec différents types de capteurs dans des conditions parfois très contraignantes pour la réception GPS. La plupart des algorithmes présentés ont été implémentés en temps-réel et testés sur les véhicules. Leur capacité à être interrompus et fournir un résultat en temps borné a été testée sur différentes machines au gré des expérimentations.

7.3 Perspectives

Estimation avec la médiane géométrique en cas de solutions multiples

Sur l'ensemble de ces travaux, lorsqu'une estimée ponctuelle de la position était requise, nous avons utilisé le centre de gravité du sous-pavage solution. Cette méthode possède l'avantage d'être rapide à calculer, le calcul peut même se faire par accumulation au cours de l'inversion ensembliste, ce qui permet de connaître à tout moment de l'exécution de l'algorithme la position du centre de gravité. On peut ainsi interrompre l'algorithme à tout instant sans latence supplémentaire pour le calcul d'une estimée ponctuelle. Le centre de gravité est de plus une très bonne estimée ponctuelle de la solution, dans le cas où l'ensemble calculé est convexe.

Une des propriétés intéressantes des méthodes ensemblistes est de pouvoir caractériser des ensembles solutions quelconques, c'est à dire non-convexes (par exemple

en « fourche » à une intersection) voire non-connexes dans le cas d'hypothèses multiples. Le problème est que le centre de gravité de tels ensembles peut se trouver en dehors des ensembles. Or, en utilisant les méthodes d'inversion ensembliste par intervalles, on prouve que la position ne peut pas se trouver en dehors du sous-pavage solution. Il est ainsi particulièrement contradictoire de livrer une estimée ponctuelle dont on a la garantie qu'elle ne peut pas être la solution du problème.

Une solution pour fournir une estimée plus cohérente avec l'ensemble solution calculé serait d'utiliser une généralisation multidimensionnelle de la médiane, telle que la médiane géométrique pondérée par la volume des boîtes. La médiane géométrique est le point qui minimise la somme des distances à l'ensemble des données (là où le barycentre minimise la somme des carrée des distances). Elle est ainsi beaucoup moins sensible que le centre de gravité aux valeurs extrêmes, ce qui permet de garder l'estimée ponctuelle proche du sous-ensemble connexe de plus grand volume dans le cas d'une estimation multi-hypothèse. Un défaut de cette approche est que le calcul de la médiane géométrique est itératif.

Adaptation automatique au nombre de mesures aberrantes

Nous avons utilisé l'inversion ensembliste q -relaxée afin d'assurer la robustesse en présence de mesures aberrantes. L'ensemble solution est alors garanti de contenir la vraie position si le nombre de mesures aberrantes est inférieur ou égal à q . Si on disposait de nombreuses mesures simultanées, on pourrait déterminer une valeur de q à partir des statistiques d'occurrence des défauts. Dans le cas du GPS, le nombre de satellites visibles est réduit. Il est possible de faire l'hypothèse d'une seule erreur simultanée en champ libre, l'erreur venant du système GPS. Cette hypothèse ne tient plus en milieu urbain, les sources d'erreur principales étant les multitrajets et NLoS.

Dans ces travaux, le nombre de mesures aberrantes tolérées a été fixé empiriquement, afin d'assurer le fonctionnement en temps réel du calcul et son interruptibilité. Cette stratégie est aidée par le seuillage du rapport signal à bruit des mesures, qui permet d'éliminer une partie des mesures aberrantes. Elle peut cependant réduire la précision des domaines calculés quand le nombre q est surestimé par rapport au nombre réel de mesures aberrantes.

L'utilisation d'une stratégie d'estimation du nombre de mesures aberrantes telle que GOMNE permettrait de limiter le nombre de contraintes à relâcher. La solution pourrait alors être calculée en relâchant $q_{min} + 1$ contraintes, où q_{min} est le nombre minimal d'outliers trouvé par GOMNE. Il faudrait néanmoins tenir compte des contraintes temps réel.

Développement d'un contracteur robuste par linéarisation parallèle

Le contracteur par linéarisation parallèle est très efficace sur le problème de localisation par satellites, de par la grande linéarité du modèle d'observation au niveau du récepteur (voir l'annexe B). Il serait intéressant d'étudier la mise au point d'un contracteur robuste basé sur la linéarisation, et de comparer son efficacité avec la propagation de contraintes mise en œuvre dans cette thèse.

Utilisation conjointe des méthodes ensemblistes avec des méthodes probabilistes

Les méthodes ensemblistes par intervalles sont bien adaptées pour le calcul des solutions globales, en présence de non-linéarités. Elles permettent de tenir compte de la présence de mesures aberrantes et de caractériser un domaine d'incertitude de localisation. Elles sont donc particulièrement indiquées pour le calcul d'une position intègre. Dans le cas linéaire, ou quand la non-linéarité est modérée, le filtre de Kalman ou sa version étendue fournit une bonne estimation, et permet de tenir compte de la représentation des mesures par les deux premiers moments statistiques. Une utilisation conjointe de ces deux méthodes pourrait permettre, tout en garantissant un domaine d'intégrité et en vérifiant que la non-linéarité est assez modérée pour faire fonctionner le filtre de Kalman, de fournir une position plus précise que le centre de gravité du sous pavage.

Prédiction du HPL en ville avec une base de donnée du bâti 3D

Les paramètres orbitaux des satellites étant connus, nous avons vu au chapitre 4 qu'il était possible de calculer le domaine d'incertitude nominal en une position et à un instant donnés, afin de calculer un niveau de protection tel que le HPL. Ce calcul suppose néanmoins que tous les satellites sont visibles. Pour généraliser cette méthode en environnement urbain, il est nécessaire de pouvoir déterminer les masquages. En utilisant une modélisation 3D des bâtiments (telle que le Bâti 3D fourni par l'IGN au sein du projet ANR CityVIP), on peut déterminer si les satellites sont visibles d'une position donnée ou s'ils sont masqués par une construction. On peut alors déterminer les domaines de localisation correspondant à la constellation visible et à la contrainte de route pour une trajectoire planifiée à l'avance.

Ce calcul est cependant très dépendant de la position, contrairement au calcul en champ libre, car les masquages des satellites dépendent de l'environnement d'évolution du véhicule. Un simple échantillonnage ponctuel le long de la trajectoire prévue peut ne pas donner de bons résultats. En effet, cette méthode peut prévoir une constellation visible différente de celle qui sera réellement vue par le véhicule, s'il dévie légèrement de la trajectoire prévue. On pourrait donc également envisager

l'utilisation de méthodes ensemblistes pour tenir compte de l'erreur de suivi de trajectoire lors du calcul de la visibilité satellitaire.

Utilisation d'une représentation plus fine dans l'horizon de positions

Le calcul robuste de pose du chapitre 6 est basé sur l'utilisation d'un horizon de positions et de données proprioceptives. Les positions sont représentées par une boîte englobante. Cependant, cette représentation introduit beaucoup de pessimisme. On pourrait envisager l'utilisation de sous-pavages, afin de pouvoir représenter plus finement les positions. Cette représentation serait particulièrement utile dans le cas de positions ambiguës, qui sont pour l'instant représentées de manière très approchée par la boîte englobante de l'ensemble des solutions.

On pourrait également travailler à une meilleure stratégie de décimation des données proprioceptives, tenant compte de la dynamique.

Implémentation sur une architecture massivement parallèle

Les méthodes d'inversion ensembliste par intervalles reposent sur l'exploration de l'espace des solutions par évaluation, contraction et bisection de boîtes. Le traitement de chaque boîte est complètement indépendant du traitement des autres boîtes, aussi l'essentiel des opérations peuvent être parallélisées. L'implémentation temps réel réalisée au cours de cette thèse partage le calcul entre plusieurs fils d'exécution, ce qui permet d'accélérer le traitement sur les machines multiprocesseurs ou multicœurs. Cependant, l'algorithme pourrait grandement bénéficier d'une implémentation sur une architecture massivement parallèle telle que les GPU¹ ou sur FPGA².

1. Graphics Processing Unit : processeur graphique

2. Field-Programmable Gate Array : circuit logique programmable

Annexes

Annexe A

Méthodes classiques d'estimation robuste

Dans cette annexe, nous présentons brièvement trois méthodes d'estimation robuste classiquement utilisées en robotique et en vision : les M-estimateurs, le Least Median of Squares et RANSAC.

A.1 M-estimateurs

Les M-estimateurs sont une classe d'estimateurs qui généralisent les estimateurs du maximum de vraisemblance. Pour n observations x_i , l'estimateur du maximum de vraisemblance maximise $\prod_{i=1}^n f(x_i, \theta)$ ou, de façon équivalente, minimise $\sum_{i=1}^n -\log f(x_i, \theta)$.

En 1964, Peter Huber a proposé une généralisation de l'estimation du maximum de vraisemblance, par la minimisation de

$$\sum_{i=1}^n \rho(x_i, \theta),$$

où ρ est une fonction mesurable réelle symétrique d'une mesure et du paramètre à estimer. Les solutions

$$\hat{\theta} = \arg \min_{\theta} \left(\sum_{i=1}^n \rho(x_i, \theta) \right)$$

sont appelées M-estimateurs [Huber, 1964]. Les estimateurs du maximum de vraisemblance sont ainsi un cas particulier de M-estimateur où $\rho = -\log f$.

La minimisation de $\sum_{i=1}^n \rho(x_i, \theta)$ peut être faite directement ou, si ρ est différentiable, en résolvant $\sum_{i=1}^n \psi(x_i, \theta) = 0$, où ψ est la dérivée de ρ par rapport à θ .

On distingue deux types de M-estimateurs, selon leur formulation à l'aide de la fonction ρ ou ψ . La figure A.1 présente quatre fonctions ρ .

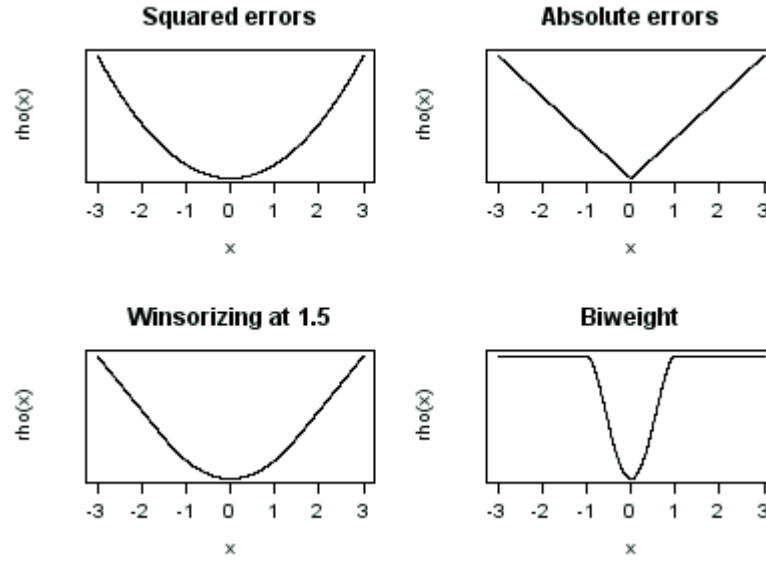


Figure A.1: Exemples de fonctions ρ

La première fonction correspond à un estimateur aux moindres carrés, où $\rho(x)$ augmente d'autant plus vite que l'erreur est grande. En utilisant l'erreur absolue, ρ croît à une vitesse constante, ce qui limite l'influence des valeurs aberrantes. La fonction *Winsorizing* combine ces deux effets : en dessous d'un seuil (1,5 sur la figure), ρ croît avec le carré de l'erreur, et, une fois le seuil passé, la croissance de ρ devient constante. On appelle parfois cette fonction ρ la fonction de Huber. La fonction *Biweight* de Tuckey se comporte comme l'erreur au carré pour de petites valeurs avant de s'aplatir et devenir constante.

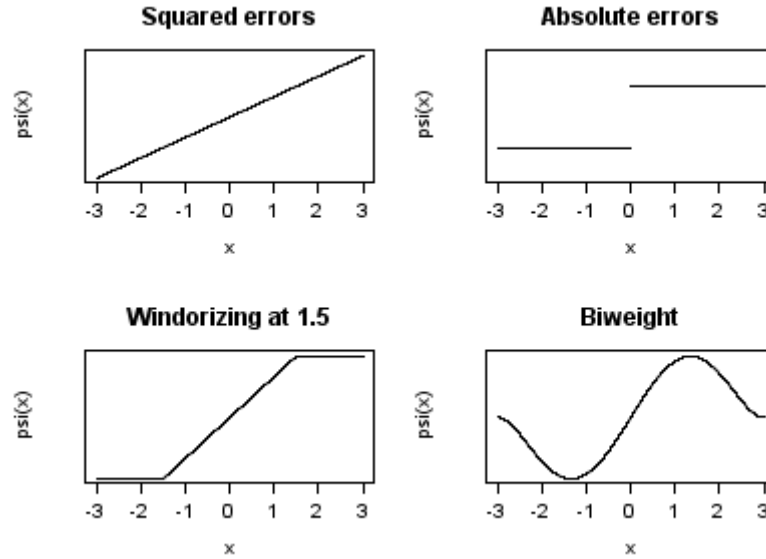
Dans la plupart des cas pratiques, on utilise des M-estimateurs qui peuvent être définis par la fonction ψ . La solution $\hat{\theta}$ est alors donnée par

$$\sum_{i=1}^n \psi(x_i, \hat{\theta}) = 0.$$

La figure A.2 représente la fonction ψ pour les quatre M-estimateurs présentés précédemment.

Si la fonction ψ tend vers zéro quand x tend vers plus ou moins l'infini, on obtient un estimateur redescendant. On a alors un rejet complet des valeurs aberrantes. Dans ce cas, la solution M-estimée n'est pas nécessairement unique. C'est par exemple le cas avec la fonction de Tuckey.

Pour la plupart des choix de fonctions ρ ou ψ , il n'existe pas de solution analytique au problème de M-estimation. On utilise donc une approche itérative. Il

Figure A.2: Exemples de fonctions ψ

est possible d'utiliser les techniques standard d'optimisation comme la méthode de Newton-Raphson. En règle générale, on utilise un algorithme de moindres carrés itérativement repondérés (IRLS¹).

A.2 Least Median of Squares

Un estimateur robuste couramment utilisé est le *Least Median of Squares* (LMedS) [Rousseeuw, 1984]. Cette méthode estime le paramètre $\hat{\theta}$ en résolvant le problème de minimisation non-linéaire suivant :

$$\hat{\theta} = \arg \min_{\theta} \text{med}_i r_i^2.$$

L'estimateur est tel qu'il minimise la médiane des résidus aux carrés sur l'ensemble des données. Cet estimateur à un point de rupture très élevé, il est en effet résistant à la contamination de 50% des données.

Contrairement aux M-estimateurs, qui peuvent être réduits à un problème aux moindres carrés pondérés et résolus par l'algorithme IRLS, l'estimateur LMedS se base sur une recherche dans un espace d'estimées possibles générées à partir des données.

1. Iteratively Re-weighted Least Squares

Une fois l'estimateur LMedS calculé, on réalise généralement une estimation aux moindres carrés sur les données saines (déterminées à partir de la solution LMedS).

A.3 RANSAC

RANSAC, abréviation de « RANdom SAMple Consensus », est une méthode itérative robuste d'estimation non déterministe, publié par Fischler et Bolles en 1981 [Fischler and Bolles, 1981]. L'algorithme fournit un résultat correct avec une certaine probabilité, croissante avec le nombre d'itérations effectuées.

L'hypothèse est que le jeu de données est constitué d'*inliers*, données pouvant être expliquées par le modèle, et d'*outliers*, données aberrantes au regard du modèle choisi. Les données peuvent être contaminées par du bruit. Ainsi, les outliers peuvent venir soit de valeurs extrêmes du bruit, de mesures erronées, ou d'hypothèses fausses lors du choix de modèle. On suppose également qu'il existe une procédure permettant d'estimer les paramètres du modèle à partir d'un ensemble réduit d'inliers.

L'algorithme RANSAC prend en entrée un ensemble de données observées, un modèle paramétré pouvant expliquer les observations et des paramètres d'intervalle de confiance.

Le principe consiste à sélectionner itérativement un sous-ensemble aléatoire des données observées, et de faire l'hypothèse qu'il s'agit d'inliers. Cette hypothèse est ensuite testée par la procédure suivante :

1. Les paramètres du modèle sont estimés de façon à ajuster le modèle aux inliers hypothétiques choisis.
2. Toutes les autres données observées sont ensuite testées sur le modèle estimé. Si un point correspond bien au modèle, il est considéré comme un inlier candidat.
3. En fonction du nombre d'inliers candidats, le modèle estimé est considéré comme correct ou non.
4. Les paramètres du modèle sont ré-estimés à partir de l'ensemble des inliers candidats.
5. L'adéquation du modèle aux données est évaluée par l'estimation de l'erreur des inliers par rapport au modèle.

La procédure est répétée un nombre fixe de fois. À chaque itération, soit le modèle est rejeté car il explique un nombre trop faible de données, soit on calcule un modèle réajusté et une mesure d'erreur associée. On conserve alors les paramètres du modèle réajusté si la mesure d'erreur est plus faible que celle du meilleur modèle obtenu lors des itérations précédentes.

Annexe B

Contracteur par linéarisation parallèle du calcul GPS

Nous avons, dans la section 4.2 (page 90) proposé un contracteur par propagation et rétropropagation de contraintes, permettant de contracter un domaine de position par des mesures de pseudodistances. Chaque pseudodistance est cependant appliquée successivement comme une contrainte indépendante des autres mesures sur la position. Or comme toutes ces contraintes dépendent des mêmes variables, la contraction n'est pas optimale. On a donc recours aux bisections pour améliorer la précision de l'ensemble solution.

Dans cette annexe, nous présentons un contracteur permettant de tenir compte simultanément de toutes les contraintes de pseudodistance.

Les satellites GPS évoluant à plus de 20000 km d'altitude, les distances $R_1, \dots, R_m$ des satellites au récepteur sont du même ordre de grandeur. Le problème de localisation par satellites se prête ainsi très bien à la linéarisation, la jacobienne M de la fonction d'observation restant valide pour des positions éloignées de plusieurs dizaines de kilomètres du point de linéarisation.

$$\mathbf{M} = \mathbf{g}'(\mathbf{x}) = \begin{bmatrix} \frac{\partial g_1(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial g_1(\mathbf{x})}{\partial x_4} \\ \dots & \dots & \dots \\ \frac{\partial g_m(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial g_m(\mathbf{x})}{\partial x_4} \end{bmatrix} = \begin{bmatrix} \frac{x-x_1}{R_1} & \frac{y-y_1}{R_1} & \frac{z-z_1}{R_1} & 1 \\ \dots & \dots & \dots & \dots \\ \frac{x-x_m}{R_m} & \dots & \frac{z-z_m}{R_m} & 1 \end{bmatrix} \quad (\text{B.1})$$

Afin de prendre en compte simultanément toutes les contraintes de pseudodistances, un contracteur par linéarisation parallèle $\mathcal{C}_{\parallel}$ peut être construit. Nous l'avons présenté dans la section 3.3.5.4, à la page 66. Le problème de localisation par satellites étant linéaire sur un large voisinage de la solution, le contracteur $\mathcal{C}_{\parallel}$ est efficace sur des domaines initiaux de plusieurs dizaines de kilomètres de large.

Le tableau B.1 présente l'implémentation sous Matlab du contracteur GPS par linéarisation externe.

Afin de comparer le contracteur par propagation-rétropropagation de contraintes et le contracteur par linéarisation parallèle, nous avons contracté une boîte de position par les deux méthodes avec 5 mesures de pseudodistance. La figure B.1 représente

Table B.1: Contracteur par linéarisation parallèle sous Matlab. La fonction J_obs_code calcule la jacobienne de la fonction d'observation des pseudodistances.

```

function [ X, PR ] = obs_code_linext( X, PR, Xsat )
% X      : boîte [x, y, z, dtu]
% PR     : intervalles de pseudodistances
% Xsat   : positions des satellites

x0 = mid(X);
A = mid(J_obs_code(x0, Xsat));

b = obs_code(x0, Xsat) - A*x0' ...
    + (J_obs_code(X, Xsat) - A)*(X - x0)';

lpA = [-A; A];
lpb = [-inf(PR) + sup(b); sup(PR) - inf(b)];

options = optimset('LargeScale', 'off', ...
    'Simplex', 'on', 'Display', 'off');

lb = inf(X) - rad(X) .* 1e-7;
ub = sup(X) + rad(X) .* 1e-7;

x1l = linprog([ 1;0;0;0],lpA,lpb,[],[],lb,ub,x0,options);
x1u = linprog([-1;0;0;0],lpA,lpb,[],[],lb,ub,x0,options);
x2l = linprog([0; 1;0;0],lpA,lpb,[],[],lb,ub,x0,options);
x2u = linprog([0;-1;0;0],lpA,lpb,[],[],lb,ub,x0,options);
x3l = linprog([0;0; 1;0],lpA,lpb,[],[],lb,ub,x0,options);
x3u = linprog([0;0;-1;0],lpA,lpb,[],[],lb,ub,x0,options);
x4l = linprog([0;0;0; 1],lpA,lpb,[],[],lb,ub,x0,options);
x4u = linprog([0;0;0;-1],lpA,lpb,[],[],lb,ub,x0,options);

Xlin = infsup([x1l(1), x2l(2), x3l(3), x4l(4)], ...
    [x1u(1), x2u(2), x3u(3), x4u(4)]);
X = intersect(X, Xlin);

```

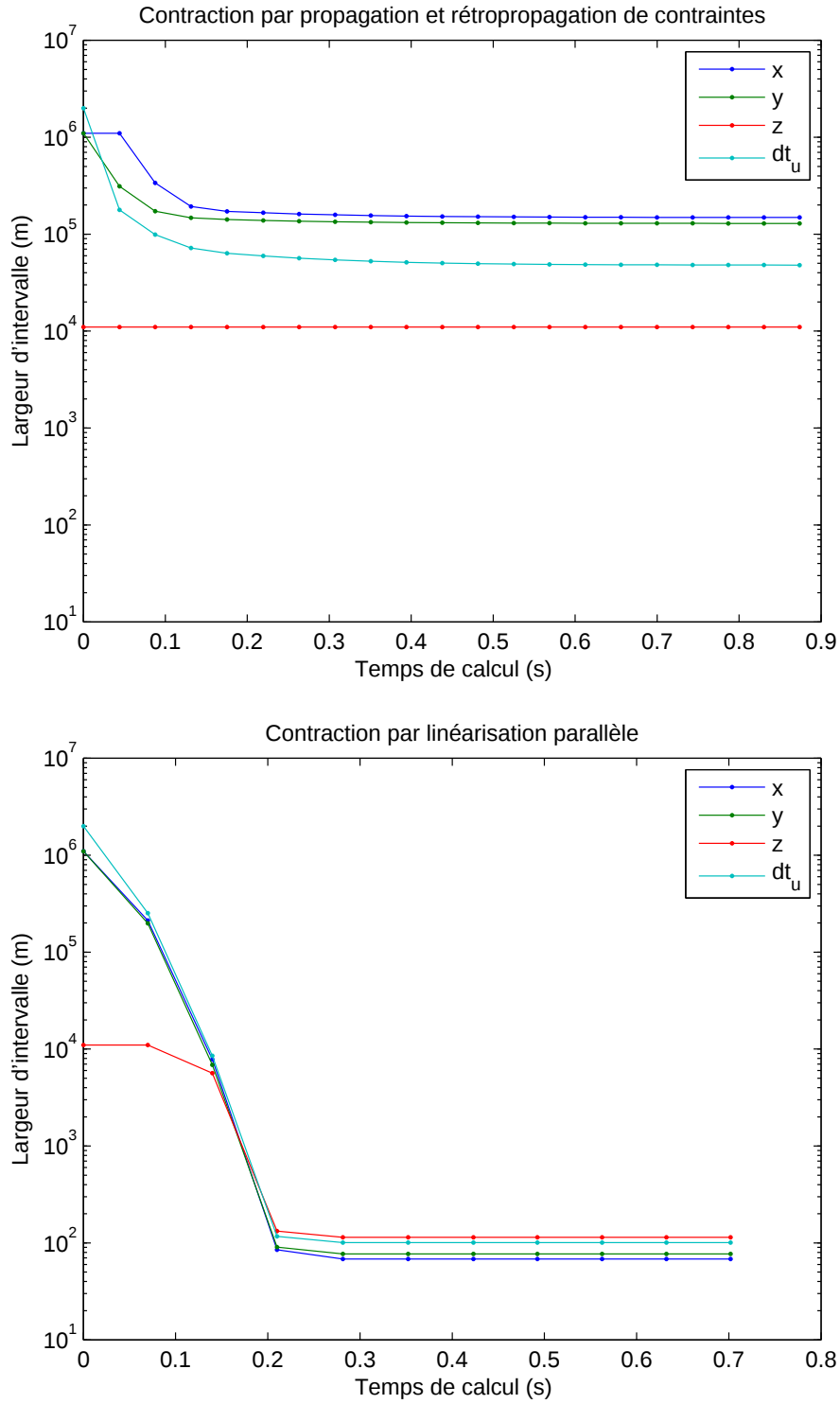


Figure B.1: Itérations de contraction de la boîte $[-1000000, 100000] \times [-100000, 1000000] \times [-1000, 10000] \times [-1000000, 1000000]$ avec les pseudodistances de 5 satellites, sous Matlab. Chaque point représente une itération du contracteur.

la largeur de chaque composante ($[x], [y], [z], [dt_u]$) de la boîte contractée en fonction du temps de calcul sous *Matlab* (avec la toolbox de calcul intervalle *Intlab*). La boîte initiale choisie fait 1100 km de largeur sur le plan horizontal et 11 km de hauteur, et elle n'est pas centrée sur la solution.

On observe sur le premier graphique de la figure B.1 que la propagation de contraintes appliquée indépendamment pour chaque pseudodistance ne parvient pas efficacement à contracter l'ensemble solution. Des bisections seront donc nécessaires pour aboutir à une solution précise. La largeur de l'intervalle de décalage d'horloge est néanmoins réduite de plus d'un ordre de grandeur, de même que la largeur des composantes horizontales est divisée par 8. La contraction permet ainsi de réduire le nombre de bisections nécessaires.

L'utilisation du contracteur par linéarisation externe permet de considérer l'ensemble des pseudodistance simultanément, et parvient à un point fixe après quatre contractions successives (second graphique de la figure B.1). On obtient une boîte contractée large de 70 m sur le plan horizontal et haute de 115 m. On remarque que chaque itération du contracteur par linéarisation parallèle est environ deux fois plus coûteuse en temps que la propagation de contraintes, mais la contraction par linéarisation parallèle est beaucoup plus efficace.

Le contracteur par linéarisation parallèle atteint néanmoins ses limites lorsqu'il n'est plus possible de linéariser la fonction d'observation sur le domaine de position initial. Nous avons réalisé sur la figure B.2 la même expérience de contraction d'une boîte à partir de 5 pseudodistances, mais à partir d'un domaine initial dont la largeur de chaque composante a été multipliée par 10. La boîte initiale choisie fait ainsi 11000 km de largeur sur le plan horizontal, 110 km de hauteur et le décalage d'horloge est connu à ± 10000 km. On observe sur le second graphique de la figure 2 que la linéarisation parallèle ne permet aucune contraction. La propagation de contraintes, en revanche, permet toujours de gagner un ordre de grandeur sur la largeur des composantes $[x]$, $[y]$ et $[dt_u]$. En utilisant les deux contracteurs en alternance, on peut obtenir une boîte contractée identique à celle obtenue lors de l'expérience précédente (troisième graphique de la figure B.2). On bénéficie en effet de la propriété de contraction de la propagation de contrainte quand la linéarisation n'est pas efficace, puis, quand le domaine se réduit la linéarisation permet une forte contraction du domaine.

L'utilisation du contracteur par linéarisation parallèle est particulièrement appropriée pour le calcul de solutions GPS en l'absence de mesures aberrantes. Cependant, nous avons dans cette thèse étudié l'utilisation de méthodes robustes basées sur l'intersection q -relaxée de contraintes. Ces méthodes nécessitent la contraction indépendante par rapport à chaque mesure. Nous avons donc utilisé la propagation de contraintes.

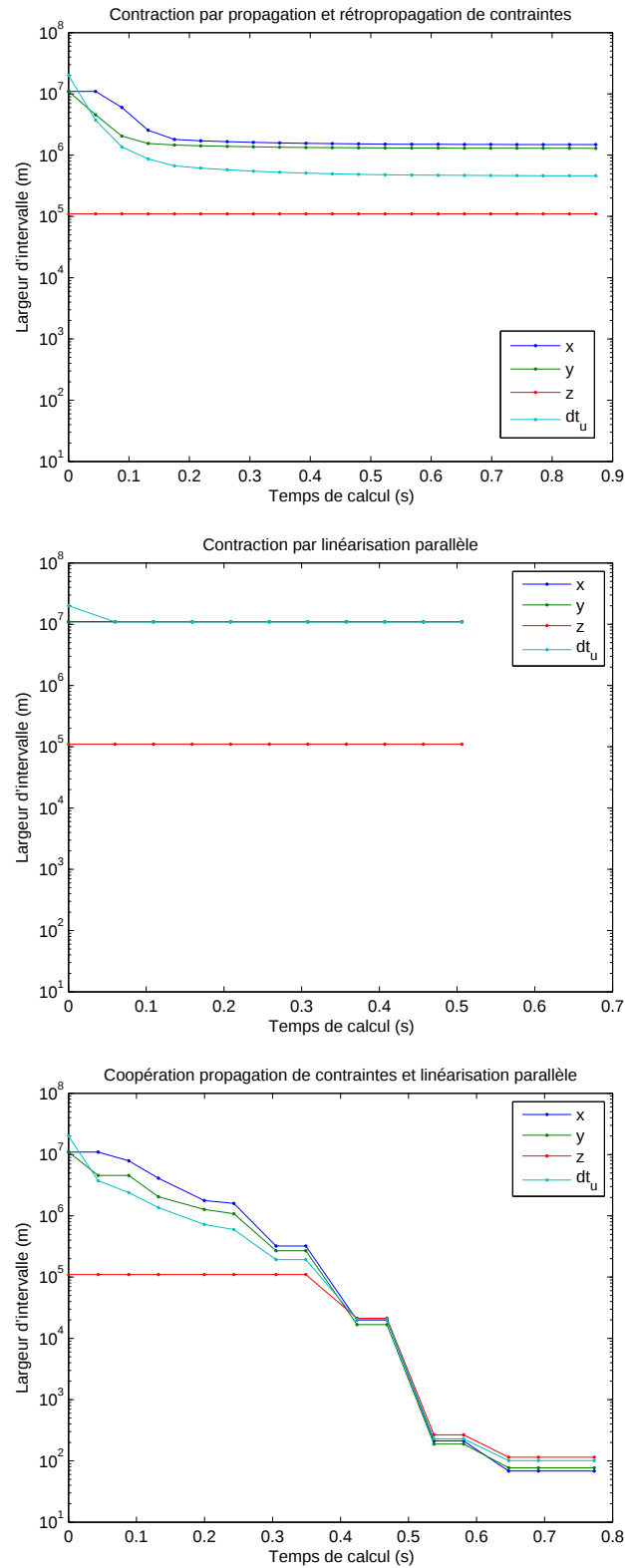


Figure B.2: Itérations de contraction de la boîte $[-10000000, 1000000] \times [-1000000, 1000000] \times [-10000, 10000] \times [-10000000, 10000000]$ avec les pseudodistances de 5 satellites, sous Matlab. Chaque point représente une itération du contracteur.

Annexe C

Jeux de données utilisés

Cette annexe présente les trois jeux de données utilisés pour les résultats expérimentaux présentés dans cette thèse. Il s'agit d'un essai à Compiègne, d'un essai à La Défense, et d'un essai autour de la mairie du XII^e arrondissement de Paris.

C.1 Essai à Compiègne

Ce jeu de données correspond à la partie à visibilité dégagée d'une boucle enregistrée à Compiègne avec le véhicule expérimental Strada du laboratoire Heudiasyc. Il a été intensivement testé par Clément Fouque durant sa thèse [Fouque, 2010] ce qui en fait un jeu de données de choix pour des premiers essais en conditions favorables. Le nombre de satellites varie au cours de l'essai, tout en restant suffisant pour un calcul de position époque par époque, ce qui permet d'étudier l'influence du nombre de pseudodistances sur la précision.

C.1.1 Parcours

La séquence utilisée dans les travaux présentés correspond à 170 secondes d'enregistrement, sur une distance de 1800 mètres, au sud de Compiègne (Figure C.1). Une partie de la séquence originale, où la localisation est rendue difficile par la présence d'arbres perturbant la réception des signaux GPS, n'a pas été utilisée. Les conditions difficiles ont été simulées ultérieurement par l'ajout de biais sur les valeurs mesurées.

C.1.2 Vérité terrain

La vérité terrain pour l'essai à Compiègne provient d'un récepteur GPS géodésique *Trimble 5700* en mode différentiel, post-traité à l'aide du logiciel *Trimble Total Control*.



Figure C.1: Parcours de l'essai à Compiègne

C.1.3 Données

On dispose des données GPS brutes enregistrées par un récepteur *Septentrio PolaRx2*, partageant son antenne avec le récepteur de référence Trimble.

Le jeu de données contient également des mesures proprioceptives provenant de l'odométrie du véhicule, ainsi que d'un gyromètre de lacet à fibre optique *KVH*. Nous n'avons néanmoins pas utilisé ces données proprioceptives, s'agissant uniquement de calcul de position époque par époque.

C.2 Essai à La Défense

Cet essai fait partie d'un jeu de données acquis au cours de la thèse d'Olivier Le Marchand [Le Marchand, 2010], afin d'observer les erreurs d'observation GPS dans les milieux de réception difficiles et de tester la robustesse d'algorithmes de localisation.

C.2.1 Parcours

Le parcours se déroule à La Défense, avec la proximité de hauts bâtiments, la traversée d'un tunnel et des conditions de canyon urbain, encaissé entre les gratte-ciel (Figure C.2). Deux parcours identiques ont été réalisés afin d'obtenir une variation dans la constellation GPS visible, et donc des masquages et multitrajets observés.

C.2.2 Vérité terrain

La vérité terrain de la pose sur l'essai à La Défense provient d'une centrale de navigation inertielle *LandINS* post-traitée en mode différentiel avec une base locale à l'aide du logiciel *Inertial Explorer*. La précision est sub-décimétrique et se dégrade tout en restant sub-métrique quand les conditions de réception se dégradent (tunnel, canyon urbain).

C.2.3 Données

Les données des capteurs embarqués sur le véhicule ont été enregistrées à l'aide du logiciel *RTmaps*.

On dispose des données de trois récepteurs GPS bas coût uBlox, reliés à des antennes patch : deux LEA-4T fournissant des données brutes à 1 Hz et à 10 Hz, et un LEA-5H fournissant une solution de localisation à 1 Hz. Les données GPS brutes d'un récepteur Septentrio PolaRx2 relié à une antenne polarisée ont également été enregistrées à 10 Hz.

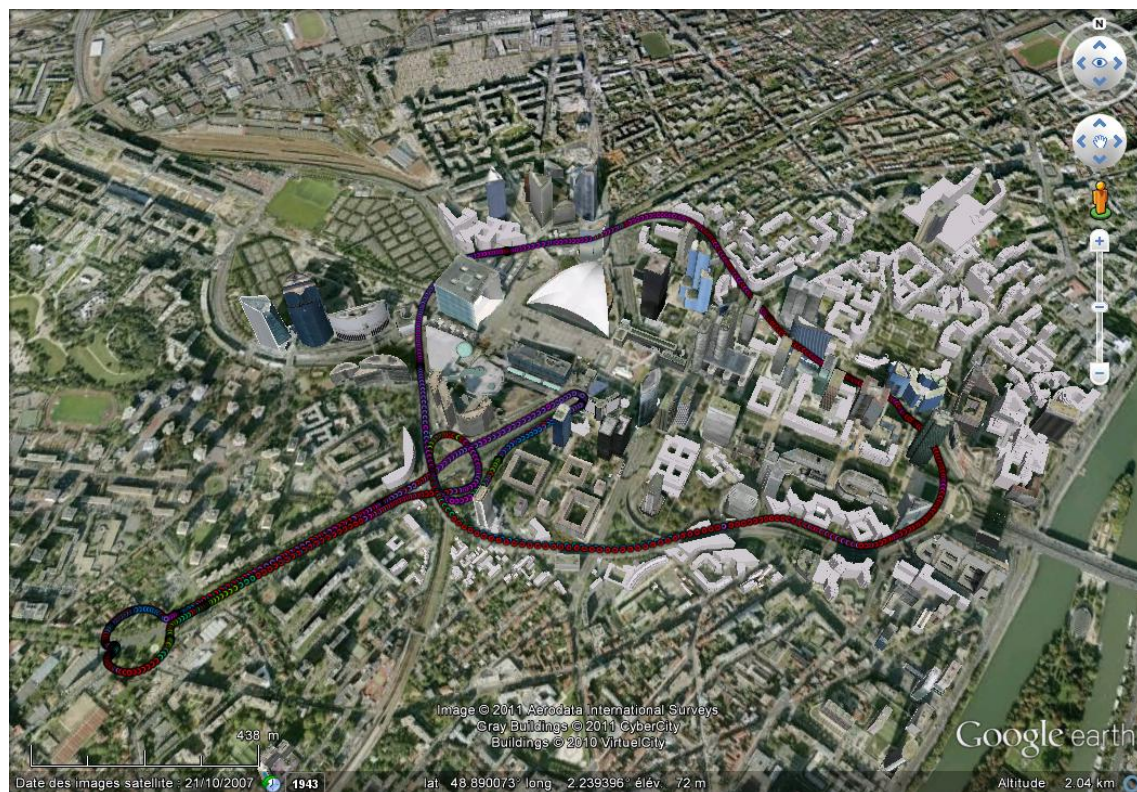


Figure C.2: Parcours de l'essai à La Défense

Le véhicule était également équipé d'une caméra filmant vers l'avant et d'un télémètre laser *Alasca*.

Enfin, le jeu de données contient les informations proprioceptives du véhicule, telles que les vitesses des roues, la vitesse de lacet, l'angle au volant.

C.3 Essai à Paris, Mairie du XII^e arrondissement

Cet essai a été réalisé pour le projet ANR CityVIP, pour la navigation autonome de véhicules individuels publics en milieu urbain.

C.3.1 Parcours

Trois boucles, chacune d'une longueur d'un kilomètre et d'une durée d'environ 5 minutes ont été réalisées autour de la mairie du XII^e arrondissement de Paris (Figure C.3). La principale caractéristique du parcours est la difficulté de réception GPS, les routes suivies étant soit étroites au milieu d'immeubles de cinq étages, soit bordées d'arbres.

C.3.2 Vérité terrain

La vérité terrain pour la pose sur l'essai à Paris autour de la mairie du XII^e arrondissement provient d'un système de navigation *Applanix* après post-traitement par l'IGN. Le véhicule de l'IGN embarquait également deux systèmes de navigation *LandINS* pouvant aussi être utilisés comme vérité terrain.

C.3.3 Données

Les données d'un récepteur uBlox LEA-4T à 0,2 Hz, d'un récepteur Septentrio PolaRx2 à 10 Hz ainsi que d'un récepteur Trilble AG132 ont été enregistrées. On dispose également des données de deux centrales inertielles MEMS (une MicroStrain 3DMG2 et une XSens MTI) ainsi que d'un gyromètre vertical à fibre optique Crossbow VG700.

Le véhicule était également équipé de nombreuses caméras dont une paire stéréoscopique et une tête panoramique, ainsi que de télémètres laser dont un *Alasca* 4 nappes à l'avant, des RIEGL couvrant les façades et un Velodyne couvrant 360° avec 64 nappes.



Figure C.3: Parcours de l'essai à Paris (mairie du XII^e arrondissement)

Annexe D

Génération d'une carte 3D de l'espace roulant

Cette annexe décrit comment l'information d'espace roulant 3D est représentée, ainsi qu'une méthode pour générer ce type d'espace roulant à partir de relevés GPS ou de données cartographiques.

D.1 Représentation de l'information

Au sein du projet ANR CityVIP, l'IGN a fourni une carte 3D de l'espace roulant couvrant les rues environnantes de la mairie du XII^e arrondissement de Paris. La surface de l'espace roulant est représentée par un ensemble de facettes triangulaires.

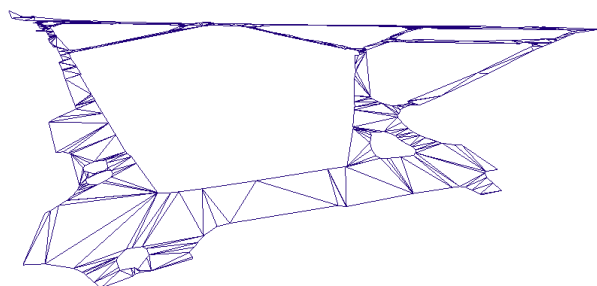


Figure D.1: Vue 3D de l'espace roulant autour de la mairie du XII^e arrondissement de Paris

Cette carte est fournie en coordonnées Lambert 93, notamment dans un format d'échange portable basé sur la représentation textuelle des coordonnées des points et des indices des sommets des facettes. Les points sont représentés en trois dimensions par leurs coordonnées X,Y dans la projection Lambert 93, ainsi que leur altitude par rapport au géoïde.

Le format de fichier, très simple, est le suivant :

- Les commentaires de début de fichier commencent par le caractère '#' ;

- Tous les sommets de l'espace roulant sont listés : chaque ligne contient les coordonnées X et Y en Lambert 93 ainsi que l'altitude d'un sommet, séparés par des tabulations ;
- Un commentaire marque la fin de la liste des sommets ;
- L'ensemble des facettes est énuméré : chaque facette est représentée sur une ligne par les indices (comptés à partir de zéro) de ses trois sommets dans la liste précédente, séparés par des tabulations.

Le tableau D.1 montre un exemple d'espace roulant représenté au format texte.

D.2 Création d'une carte 3D

Nous décrivons dans cette section une méthode permettant de réaliser notre propre carte 3D de l'espace roulant, à l'aide du SIG libre *Quantum GIS*¹ (QGIS). Elle consiste en deux étapes, la première étant la saisie des coordonnées 3D des bords de voie, et la seconde la génération des facettes triangulaires.

D.2.1 Saisie des bords de voie

La carte 3D réalisée par l'IGN repose sur la saisie des coordonnées des bords de voie en 3D, à partir de paires d'images aériennes à haute résolution. La saisie s'effectue en manipulant un curseur 3D sur un écran stéréoscopique, et permet une précision de l'ordre de 5 cm en planimétrie et de 20 cm en altitude.

Nous présentons ici trois autres méthodes plus faciles d'accès afin de saisir les coordonnées des points de bords de voie.

Par relevé de points GPS

Un première méthode pour la saisie des bords de voie consiste à utiliser un récepteur GPS. Idéalement, il faut disposer d'une station de base et d'un moyen de communication radio afin d'utiliser le récepteur en mode RTK. On peut alors obtenir une précision relative centimétrique.

Lorsque le relevé se fait près de bâtiments, le masquage des satellites diminue la précision de localisation, et des signaux réfléchis peuvent être reçus, ce qui peut être source de points erronés. Le relevé de points peut ainsi être impossible dans les environnements où le bâti est haut et dense. Il est de plus difficile de saisir les points lorsqu'un bâtiment constitue la limite d'espace roulant.

Nous avons réalisé cette expérience à l'aide du récepteur Trimble 5700 et de son PDA qui embarque un logiciel de relevé de points. Le positionnement a été réalisé

1. <http://www.qgis.org/>

Table D.1: Exemple d'espace roulant au format texte

```

# Petite boucle en face de la sortie du garage Heudiasyc
# Lambert93
# 13 vertices
685468.001000    6922116.640000    36
685428.872000    6922141.450000    35
685404.736000    6922105.750000    35
685399.765000    6922109.110000    35
685463.980000    6922204.080000    36
685470.176000    6922203.180000    36
685512.047000    6922174.490000    36
685512.832000    6922171.730000    36
685471.935000    6922117.370000    36
685469.082000    6922121.880000    36
685507.055000    6922171.850000    36
685467.674000    6922198.830000    36
685431.674000    6922145.600000    35
# 13 triangles
9      8      0
12     0      1
12     9      0
3      1      2
10     7      8
10     6      7
9      10     8
4      5      11
12     4      11
5      10     11
5      6      10
3      12     1
3      4      12

```

en mode naturel, faute de liaison de données pour la transmission des corrections différentielles. On observe une erreur d'un mètre entre le même point relevé au début et à la fin de la campagne de mesures.

À partir d'une orthoimage et d'un MNT

La méthode par relevé de points GPS nécessite des moyens logistiques (récepteur, station de base, liaison radio) et beaucoup de temps si la surface à couvrir est grande. Si l'on dispose d'orthoimages² géoréférencées suffisamment précises de la zone à cartographier, on peut procéder à la saisie des bords de voie directement dans un logiciel SIG (tel que le logiciel libre QGIS).

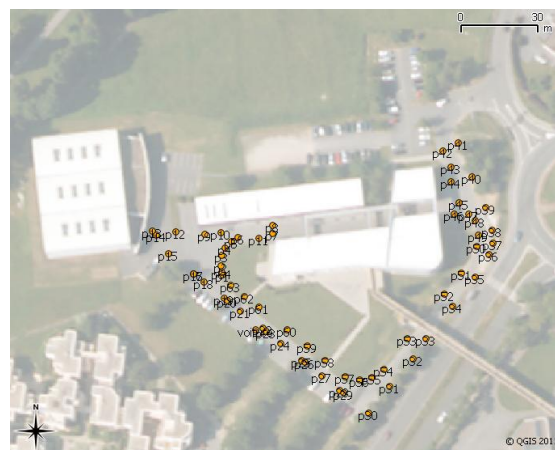
Pour ce faire, on importe l'orthoimage en tant que couche raster, et on procède à la saisie des bords de voie sur une couche vectorielle. Pour simplifier la saisie, on peut utiliser une couche de type ligne pour la saisie des bords de voie, et extraire les points de contour dans une couche d'objets ponctuels une fois la saisie terminée.

Les points obtenus sont en 2D. Afin d'obtenir l'information d'altitude, on importe un modèle numérique de terrain de la zone, et on l'utilise pour définir un attribut d'altitude sur tous les points de bord de voie.

À partir d'une carte 2D et d'un MNT

Si on ne cherche pas à décrire précisément les détails des bords de voie, on peut se baser sur une carte 2D classique, représentant les segments de route par des

2. Une orthophotographie ou orthoimage est une photographie aérienne dont la géométrie a été redressée de sorte que chaque point soit superposable à une carte plane qui lui correspond.



(a) Vue des points sous QGIS

Figure D.2: Saisie des bords de voie par relevé de points GPS

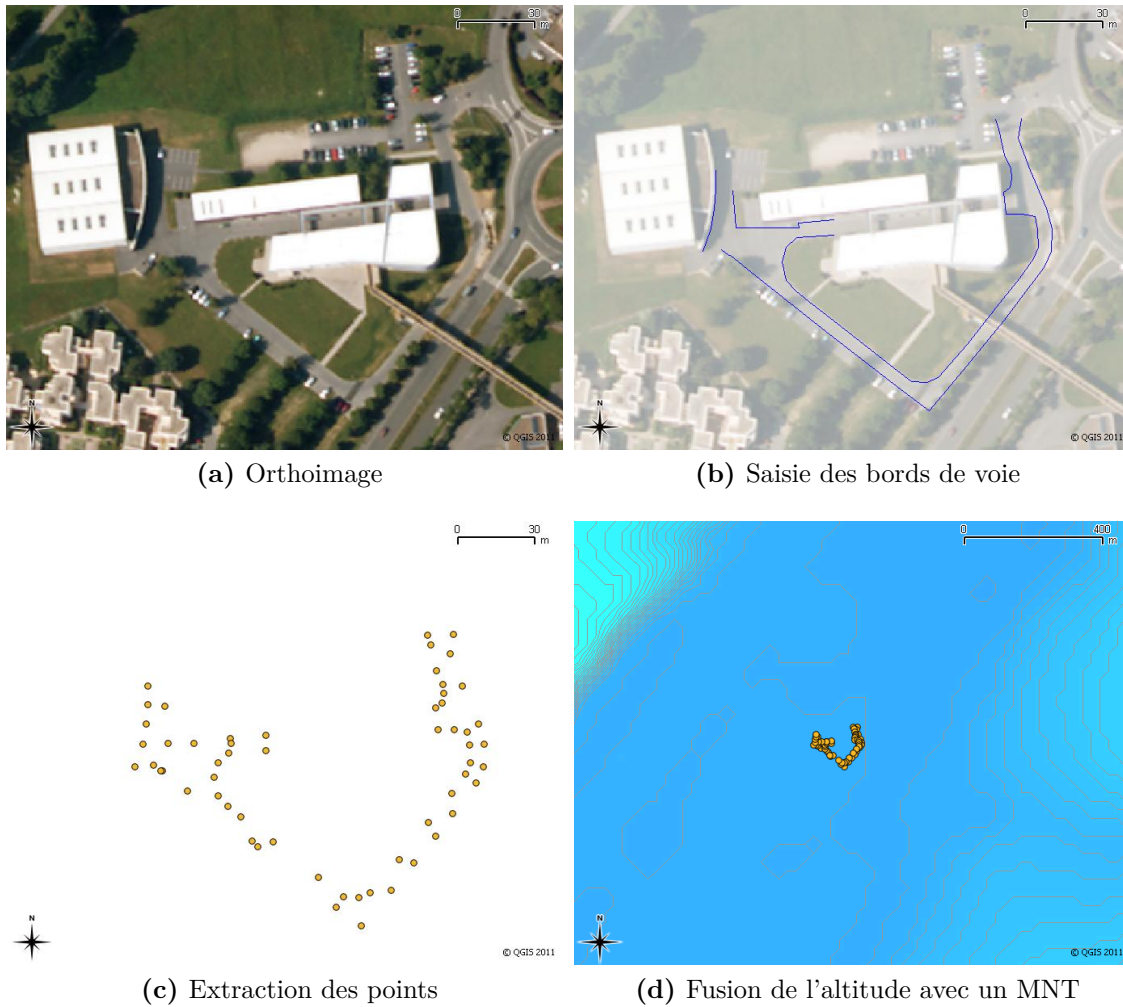


Figure D.3: Saisie des bords de voie à partir d'une orthoimage et d'un MNT sous QGIS

polylignes. On peut ainsi utiliser les routes de la BDTopo de IGN, ou encore la cartographie libre d'OpenStreetMap.

Une fois les routes considérées importées dans une couche du SIG et converties en Lambert 93, la première étape consiste à recréer une surface à partir des segments. On a donc besoin d'information de largeur de voie. Soit cette information est déjà présente dans les données importées, soit il faudra la définir pour chaque tronçon de route. On calcule alors un champ « demi-largeur de voie » à partir du champ « largeur de voie », et on calcule la surface des points éloignés d'au plus une demi-largeur de voie de la route (à l'aide, par exemple, de l'outil *buffer*). Après une éventuelle étape de fusion et simplification des polygones obtenus, on peut extraire les points de contour de la surface obtenue. On peut alors charger un MNT couvrant la zone saisie afin d'attribuer une altitude aux points de bord de voie générés.



Figure D.4: Génération des bords de voie à partir d'une carte 2D sous QGIS

D.2.2 Génération des facettes

La génération des facettes de l'espace roulant peut être effectuée en réalisant une triangulation de Delaunay sur les points de bords de voie obtenus précédemment. Cette triangulation va aussi bien couvrir l'espace roulant que son complémentaire dans l'enveloppe convexe des points de bord de voie saisis. Aussi, il va falloir ne conserver que les triangles appartenant à l'espace roulant. Une méthode simple consiste à sélectionner les triangles qui entrent en intersection avec la ligne de centre de voie (que l'on peut tracer grossièrement si l'on ne dispose pas de carte numérique de la zone). Il peut être nécessaire de retoucher manuellement la sélection sur certains

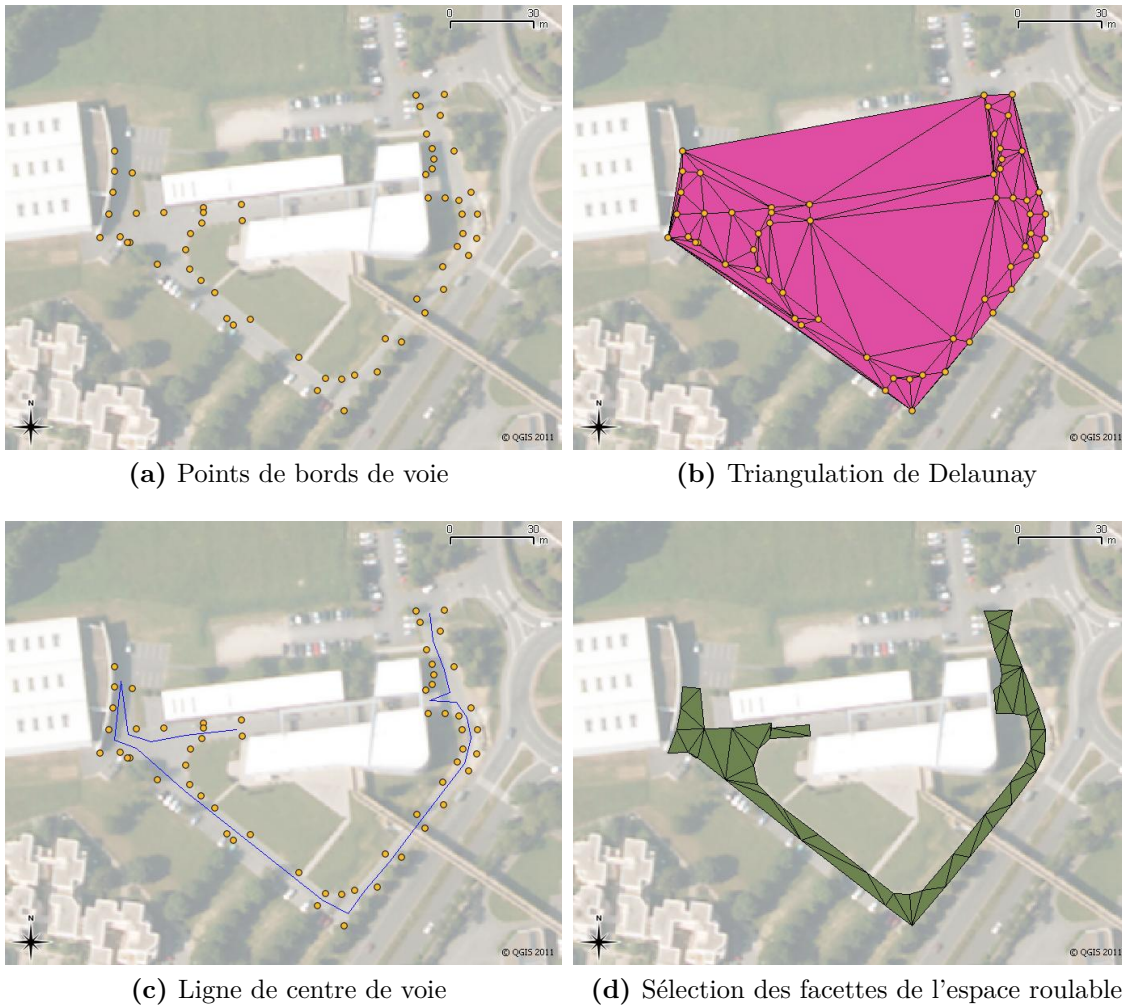


Figure D.5: Création de l'espace roulant sous QGIS à partir des points de bord de voie

points où la ligne de centre de voie n'intersecte pas tous les triangles de l'espace roulant.

On exporte alors au format texte les coordonnées des points 3D saisis (en créant les attributs X et Y à partir de la géométrie), et les triplets d'identifiants de points formant chaque facette (ces attributs sont générés automatiquement par l'outil de triangulation de QGIS). On obtient ainsi une description de l'espace roulant en 3D au format texte, telle que présentée dans la section précédente.

Bibliographie

- [Abdallah et al., 2008] Abdallah, F., Gning, A., and Bonnifait, P. (2008). Box particle filtering for non linear state estimation using interval analysis. *Automatica*, 44 :807–815. DOI :10.1016/j.automatica.2007.07.024.
- [Aitken, 1934] Aitken, A. (1934). On least squares and linear combination of observations. *Proc. R. Soc. Edinb*, 55 :42–48.
- [Baarda, 1968] Baarda, W. (1968). A testing procedure for use in geodetic networks. *Ned Geod Comm Publ on Geodesy, New series*, 2(5).
- [Bancroft, 1985] Bancroft, S. (1985). An algebraic solution of the GPS equations. *IEEE T. Aero. Elec. Sys.*, 21(1) :56–59.
- [Basseville and Nikiforov, 1993] Basseville, M. and Nikiforov, I. (1993). *Detection of abrupt changes : theory and application*, volume 15. Prentice-Hall, Inc.
- [Benhamou et al., 1999] Benhamou, F., Goualard, F., Granvilliers, L., and Puget, J.-F. (1999). Revising hull and box consistency. In *Int. Conf. on Logic Programming*, pages 230–244. MIT press.
- [Bétaille et al., 2008] Bétaille, D., Toledo-Moreo, R., and Laneurit, J. (2008). Making an enhanced map for lane location based services. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, pages 711–716. IEEE.
- [Bouron, 2002] Bouron, P. (2002). *Méthodes ensemblistes pour le diagnostic, l'estimation d'état et la fusion de données temporelles*. PhD thesis, Université de Technologie de Compiègne.
- [Brown, 1987] Brown, A. K. (1987). Receiver autonomous integrity monitoring using a 24-satellite GPS constellation. In *Institute of Navigation, Technical Meeting*, pages 256–262.
- [Brown, 1992] Brown, R. G. (1992). A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods. *Navigation*.
- [Chabert and Jaulin, 2009] Chabert, G. and Jaulin, L. (2009). Contractor Programming. *Artificial Intelligence*, 173 :1079–1100. WOS.
- [Chausse et al., 2005] Chausse, F., Laneurit, J., and Chapuis, R. (2005). Vehicle localization on a digital map using particles filtering. In *IEEE Intelligent Vehicles Symposium, 2005*, pages 243–248, Las Vegas, Nevada, U.S.A. IEEE.
- [Dantzig, 1963] Dantzig, G. (1963). *Linear programming and extensions*. Princeton University Press.

- [Drevelle and Bonnifait, 2009a] Drevelle, V. and Bonnifait, P. (2009a). High integrity GNSS location zone characterization using interval analysis. In *Proceedings of ION GNSS 2009*, pages 2178–2187.
- [Drevelle and Bonnifait, 2009b] Drevelle, V. and Bonnifait, P. (2009b). Integrity zone computation using interval analysis. In *Proceedings of the European Navigation Conference (GNSS 2009)*.
- [Drevelle and Bonnifait, 2010] Drevelle, V. and Bonnifait, P. (2010). Robust positioning using relaxed constraint-propagation. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4843–4848, Taipei, Taiwan. IEEE.
- [Drevelle and Bonnifait, 2011a] Drevelle, V. and Bonnifait, P. (2011a). A set-membership approach for high integrity height-aided satellite positioning. *GPS Solutions*, 15(4) :357–368.
- [Drevelle and Bonnifait, 2011b] Drevelle, V. and Bonnifait, P. (2011b). Global positioning in urban areas with 3-D maps. In *IEEE Intelligent Vehicle Symposium, 2011*, pages 764–769, Baden-Baden, Germany.
- [Dunik et al., 2010] Dunik, J., Simandl, M., and Straka, O. (2010). Multiple-model filtering with multiple constraints. In *American Control Conference (ACC), 2010*, pages 6858 –6863.
- [Duquenne et al., 2005] Duquenne, F., Botton, S., Peyret, F., Bétaille, D., and Willis, P. (2005). *GPS : localisation et navigation par satellites*. Hermès - Lavoisier.
- [Durand, 2003] Durand, S. (2003). *Amélioration de la précision de la localisation différentielle temps réel par mesure de phase des systèmes GNSS*. PhD thesis, IGN / LCPC Nantes - Observatoire de Paris.
- [Feng et al., 2006] Feng, S., Ochieng, W., Walsh, D., and Ioannides, R. (2006). A measurement domain receiver autonomous integrity monitoring algorithm. *GPS Solut.*, 10(2) :85–96.
- [Fischler and Bolles, 1981] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24 :381–395.
- [Fouque, 2010] Fouque, C. (2010). *Apport des cartes routières navigables pour le positionnement par satellites des véhicules*. PhD thesis, Université de Technologie de Compiègne.
- [Fouque and Bonnifait, 2010] Fouque, C. and Bonnifait, P. (2010). Multi-hypothesis map-matching on 3d navigable maps using raw gps measurements. In *13th IEEE International Conference on Intelligent Transportation Systems*, pages 1498–1503, Madeira Island Portugal.
- [Gning and Bonnifait, 2006] Gning, A. and Bonnifait, P. (2006). Constraints propagation techniques on intervals for a guaranteed localization using redundant data. *Automatica*, 42(7) :1167–1175.

- [Gutmann et al., 2009] Gutmann, J., Marti, L., and G., L. (2009). Multipath detection and mitigation by means of a MEMS based pressure sensor for low-cost systems. In *Proceedings of the 22nd International Meeting of the Satellite Division of The Institute of Navigation*, pages 2077–2087.
- [Guttman, 1984] Guttman, A. (1984). R-trees : A dynamic index structure for spatial searching. In *Proc. SIGMOD International Conference on Management of Data*, pages 47–57.
- [Huber, 1964] Huber, P. J. (1964). Robust estimation of a location parameter. *Ann Math Stat*, 35(1) :73–101.
- [IS-GPS-200D, 2004] IS-GPS-200D (2004). Navstar GPS space segment/Navigation user interfaces – Interface specification. Technical report, GPS Joint Program Office.
- [Jaulin, 2000] Jaulin, L. (2000). *Le calcul ensembliste par analyse par intervalles et ses applications*. PhD thesis, Habilitation à diriger des recherches.
- [Jaulin, 2009] Jaulin, L. (2009). Robust set-membership state estimation ; application to underwater robotics. *Automatica*, 45(1) :202–206.
- [Jaulin, 2011a] Jaulin, L. (2011a). Range-only SLAM with occupancy maps ; a set-membership approach. *IEEE T-RO*, 27(5) :1004–1010.
- [Jaulin, 2011b] Jaulin, L. (2011b). Set-membership localization with probabilistic errors. *Robotics and Autonomous Systems*, 59(6) :489 – 495.
- [Jaulin et al., 2001] Jaulin, L., Kieffer, M., Didrit, O., and Walter, É. (2001). *Applied Interval Analysis*. Springer-Verlag.
- [Jaulin et al., 2002] Jaulin, L., Kieffer, M., Walter, E., and Meizel, D. (2002). Guaranteed robust nonlinear estimation with application to robot localization. *IEEE T. Syst. Man. Cy. C.*, 32(4) :374–382.
- [Jaulin and Walter, 1993] Jaulin, L. and Walter, E. (1993). Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4) :1053–1064.
- [Jaulin et al., 1996] Jaulin, L., Walter, É., and Didrit, O. (1996). Guaranteed robust nonlinear parameter bounding. In *CESA'96 IMACS Multiconference (Symposium on Modelling, Analysis and Simulation)*, volume 2, pages 1156–1161, Lille.
- [Julier and LaViola, 2007] Julier, S. J. and LaViola, J. J. (2007). On kalman filtering with nonlinear equality constraints. *Signal Processing, IEEE Transactions on*, 55(6) :2774 –2784.
- [Kaplan and Hegarty, 2006] Kaplan, E. and Hegarty, C. (2006). *Understanding GPS : Principles and Applications Second Edition*. Artech House Publishers.
- [Kieffer, 1999] Kieffer, M. (1999). *Estimation ensembliste par analyse par intervalles. Application à la localisation d'un véhicule*. PhD thesis, Université de Paris 11, Orsay, FRANCE.
- [Kieffer et al., 2000] Kieffer, M., Jaulin, L., Walter, É., and Meizel, D. (2000). Robust autonomous robot localization using interval analysis. *Reliable Computing*, 6(3) :337–362.

- [Lahrech et al., 2005] Lahrech, A., Boucher, C., and Noyer, J.-C. (2005). Accurate vehicle positioning in urban areas. *IECON Proceedings (Industrial Electronics Conference)*, 5 :486–490.
- [Le Marchand, 2010] Le Marchand, O. (2010). *Estimation robuste de la position d’un véhicule dans un environnement complexe et mesure de l’intégrité de l’information*. PhD thesis, Université de Technologie de Compiègne.
- [Le Marchand et al., 2009] Le Marchand, O., Bonnifait, P., Ibanez-Guzmán, J., Betaille, D., and Peyret, F. (2009). Characterization of GPS multipath for passenger vehicles across urban environments. *ATTI dell’Istituto Italiano di Navigazione*, (189) :77 – 88.
- [Leick, 2004] Leick, A. (2004). *GPS satellite surveying, Third edition*. Wiley.
- [Lévêque, 1998] Lévêque, O. (1998). *Méthodes Ensemblistes pour la Localisation de Véhicules*. PhD thesis, Université de Technologie de Compiègne.
- [Li et al., 2005] Li, J., Taylor, G., and Kidner, D. B. (2005). Accuracy and reliability of map-matched GPS coordinates : The dependence on terrain model resolution and interpolation algorithm. *Comput. Geosci.*, 31(2) :241–251.
- [Meizel et al., 2002] Meizel, D., Leveque, O., Jaulin, L., and Walter, E. (2002). Initial localization by set inversion. *IEEE T. Robot. Autom.*, 18(6) :966–971.
- [Moore, 1966] Moore, R. (1966). *Interval analysis*. Prentice Hall.
- [Paparoditis et al., 2000] Paparoditis, N., Thom, C., and Jibrini, H. (2000). Surface reconstruction in urban areas from multiple views of aerial digital frame cameras. *International Archives of Photogrammetry and Remote Sensing*, 33(Suppl. B3) :43–50.
- [Ranganathan et al., 2007] Ranganathan, A., Kaess, M., and Dellaert, F. (2007). Fast 3d pose estimation with out-of-sequence measurements. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2486 –2493.
- [Röhrig and Müller, 2009] Röhrig, C. and Müller, M. (2009). Indoor location tracking in non-line-of-sight environments using a IEEE 802.15.4a wireless network. In *Conference Proceedings of IROS09. St. Louis*, pages 552–557.
- [Rousseeuw, 1984] Rousseeuw, P. (1984). Least median of squares regression. *Journal of the American statistical association*, pages 871–880.
- [RTCA SC-159, 2006] RTCA SC-159 (2006). RTCA/DO-229D - Minimum operational performance standards for global positioning system/Wide area augmentation system airborne equipment.
- [Selloum et al., 2010] Selloum, A., Betaille, D., Le Carpentier, E., and Peyret, F. (2010). Robustification of a map aided location process using road direction. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 1504–1510. IEEE.
- [Septentrio, 2005] Septentrio (2005). *PolaRx2/2e User Manual*.

- [Sutherland and Hodgman, 1974] Sutherland, I. E. and Hodgman, G. W. (1974). Reentrant polygon clipping. *Communications of the ACM*, 17(1) :32–42.
- [Syed and Cannon, 2005] Syed, S. and Cannon, M. E. (2005). Map-aided gps navigation. *GPS World*, 16(11) :39–44.
- [Taylor et al., 2006] Taylor, G., Blewitt, G., and Wiley, J. (2006). *Intelligent positioning : GIS-GPS unification*. John Wiley.
- [Teunissen, 1990] Teunissen, P. J. G. (1990). Quality control in integrated navigation systems. *IEEE Aerosp Electron Syst Mag*, 5(7) :35–41.
- [Tolman et al., 2004] Tolman, B., Harris, R. B., Gaussiran, T., Munton, D., Little, J., Mach, R., Nelsen, S., and Renfro, B. (2004). The GPS Toolkit : Open Source GPS Software. In *Proceedings of the 16th International Technical Meeting of the Satellite Division of the Institute of Navigation*, Long Beach, California.
- [Waltz, 1972] Waltz, D. L. (1972). *Generating Semantic Description from Drawings of Scenes with Shadows*. PhD thesis, Massachusetts Institute of Technology.

Liste des symboles

ABS	Antiblockiersystem : système de freinage anti-blocage
BPSK	Binary Phase-Shift Keying
CDMA	Code-Division Multiple Access
CSP	Constraint Satisfaction Problem, Problème de Satisfaction de Contraintes
DEM	Digital Elevation Model
DIA	Detection, Identification and Adaptation
DoD	Department of Defense
ECEF	Earth Centered Earth Fixed
EGNOS	European Geostationary Navigation Overlay Service
ENU	East, North, Up
ESA	European Space Agency
ESP	Electronic Stability Program (aussi appelé ESC pour Electronic Stability Control) : Contrôle Electronique de Stabilité
FAA	Federal Aviation Administration
FDE	Fault Detection and Exclusion
FDMA	Frequency-Division Multiple Access
GBAS	Ground-Based Augmentation System
GLONASS	GLObalnaya NAVigatsionnaya Spoutnikovaya Sistema
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HAL	Horizontal Alert Limit
HPL	Horizontal Protection Level (Niveau de protection horizontal)
HUL	Horizontal Uncertainty Level (Niveau d'incertitude horizontal)
IRLS	Iteratively Re-weighted Least Squares
MDB	Minimal Detectable Bias

Bibliographie

MNT	Modèle Numérique de Terrain
MOPS	Minimal Operational Performance Standards
MSAS	MTSAT-Based Augmentation System
MTSAT	Multifunctional Transport Satellites
NAVSTAR	NAVigation System by Timing And Ranging
NGA	National Geospatial-Intelligence Agency
NLoS	Non Line of Sight
OCXO	Oven Controlled X-tal(Crystal) Oscillator : Oscillateur à Quartz Thermostaté
PL	Protection Level (Niveau de Protection)
PVT	Position, Velocity, and Time
RAIM	Receiver Autonomous Integrity Monitoring
RANSAC	RANdom SAMple Consensus
RNP	Required Navigation Performance
RSIVIA	Robust Set Inverter via Interval Analysis
SA	Selective Availability
SBAS	Satellite-Based Augmentation System
SIG	Système d'Information Géographique
SIVIA	Set Inversion via Interval Analysis
SLAM	Simultaneous Localization And Mapping
TCXO	Temperature Compensated X-tal(Crystal) Oscillator : Oscillateur à quartz compensé en température.
TDoA	Time Difference of Arrival
TGD	Total Group Delay
UDRE	User Differencial Range Error
UL	Uncertainty Level
URA	User Range Accuracy
VAL	Vertical Alert Limit
VPL	Vertical Protection Level (Niveau de protection vertical)
WAAS	Wide Area Augmentation System
WGS 84	World Geodetic System 1984

Résumé

On cherche dans cette thèse à caractériser un domaine de confiance pour la localisation d'un véhicule, en utilisant des méthodes ensemblistes robustes par intervalles.

La localisation est essentielle à la navigation en robotique mobile, en particulier pour les véhicules intelligents. Lorsque la position est utilisée dans un contexte pouvant mettre en jeu la sécurité des personnes, tel que la navigation autonome, on doit avoir recours à un mécanisme d'intégrité vérifiant que l'erreur commise reste inférieure à l'erreur maximale tolérable pour la mission. Dans le domaine aéronautique, on définit ainsi un niveau de protection associé à un risque d'intégrité. Dans un esprit similaire aux niveaux de protections utilisés en aéronautique, ce travail vise à déterminer un domaine de confiance, dans lequel l'utilisateur est garanti de se trouver avec un risque d'intégrité donné. L'utilisation de méthodes ensemblistes robustes permet de calculer un domaine de localisation tenant compte de l'éventuelle présence de mesures aberrantes.

Les mesures provenant de capteurs ainsi que les paramètres des modèles sont entachés d'erreurs, souvent modélisées par leur distribution de probabilité. Dans le cadre ensembliste, on peut représenter ces erreurs par des intervalles. Quand les bornes ne sont pas spécifiées ou trop pessimistes, on peut déterminer des bornes associées à un risque, et propager ce risque au domaine de confiance calculé.

Les systèmes de navigation par satellites permettent un positionnement absolu avec une bonne précision en milieu ouvert. Cependant, les mesures de pseudodistance GPS sont sujettes aux problèmes de trajets multiples ou réfléchis en zone urbaine. Une robustesse aux valeurs aberrantes est donc nécessaire. Pour compenser le manque de satellites en milieu à visibilité satellitaire réduite, des contraintes sur la position sont apportées par la cartographie 3D de l'espace roulant et l'intégration des capteurs proprioceptifs présents sur les véhicules modernes.

Trois méthodes de localisation, basées sur un algorithme robuste d'inversion ensembliste par intervalles associé aux mesures GPS sont présentées dans ce document.

- La première consiste en un calcul époque par époque, fusionnant les mesures de pseudodistance d'un récepteur GPS avec l'information d'un modèle numérique de terrain.
- La seconde s'appuie sur une carte surfacique précise de « l'espace roulant » en trois dimensions, ainsi que l'observation de la dérive de l'horloge du récepteur.
- Enfin, la troisième méthode concerne l'estimation de la pose du véhicule à partir d'un historique fini de positions et de mesures proprioceptives.

Ces trois méthodes ont été implémentées en temps-réel, et testées sur des données réelles acquises dans des environnements difficiles pour la localisation par satellites.

Summary

In this thesis, confidence domains for vehicle localization are characterized by using robust interval methods.

Positioning is of prime importance in mobile robotics and more specifically for intelligent vehicle applications. When position information is used in a safety-critical context, like autonomous vehicle navigation, an integrity method is needed to check that the positioning error stays within the limits specified for the mission. In aeronautical navigation, protection levels are defined as bounds on the position error associated to a given integrity risk. This work aims to compute a confidence domain in which the user is guaranteed to be located with a given integrity risk. The possible presence of outliers is handled by the use of robust set-membership methods.

Sensor measurements and model parameters are prone to errors, which are often modeled by their probability distribution. In the set-membership working frame, errors can be represented by intervals, thus making the assumption of bounded errors. When guaranteed error bounds are unknown or too pessimistic, error bounds associated with a risk can be used. The risk taken on measurements is then propagated to the computed confidence domain.

Global navigation satellite systems enable high precision absolute positioning in open sky environments, but measurements suffer from multipath and non-line-of-sight propagation in urban areas. Robustness to outliers is thus needed. To counter the lack of visible satellites in urban canyons, position is constrained by a 3D map of the drivable space and by using the proprioceptive sensors embedded in recent vehicles.

This document presents three positioning methods based on a robust set inversion via interval analysis with GPS pseudorange measurements :

- Snapshot computation of a position confidence domain, with GPS measurements and altitude constraint from a digital elevation model.
- Use of a precise 3D model of the drivable space as a positioning constraint, and observation of the GPS receiver's clock drift.
- Robust pose estimation from a sliding horizon of positions and proprioceptive measurements, constrained by a 3D map.

These positioning methods have been implemented in real-time and tested with real data in difficult environments for satellite positioning.