

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 DESCRIPTION DU PROJET DE RECHERCHE ET REVUE DE LITTÉRATURE	3
1.1 Mise en contexte et problématique	3
1.2 Objectifs principaux et spécifiques du projet.....	6
1.3 Méthodologie	7
1.4 Retombées prévues	9
1.5 Revue des caractéristiques des simulateurs de constellation GPS/GNSS actuels	10
1.6 Contributions générales	18
1.7 Conclusion	18
CHAPITRE 2 DESCRIPTION DE L'ARCHITECTURE INITIALE DU SIMULATEUR LACIME	21
2.1 Description de l'architecture du simulateur LACIME existante	21
2.1.1 Fonctionnement global et banc de test du simulateur initial.....	21
2.1.2 Description de la partie logicielle associée au simulateur et à son GUI...	22
2.1.3 Description des aspects matériels reliés au simulateur	26
2.1.3.1 Description de l'architecture de la section IF	26
2.1.3.2 Description des interfaces des mémoires du simulateur LACIME	29
2.1.3.3 Description de l'architecture de la section RF	32
2.1.4 Limitations de l'architecture actuelle et problèmes rencontrés	37
2.2 Discussion sur les performances initiales du simulateur LACIME	40
2.3 Conclusion	44
CHAPITRE 3 AMÉLIORATIONS ET VALIDATION DE LA PARTIE LOGICIELLE DU SIMULATEUR DE CONSTELLATION GPS L1	45
3.1 Ajout d'une fonction d'enregistrement des mesures du simulateur.....	45
3.2 Validation des algorithmes du simulateur LACIME	46
3.2.1 Calcul des positions des satellites	46
3.2.2 Calcul des vitesses des satellites	49
3.2.3 Calcul de l'azimuth des satellites.....	51
3.2.4 Calcul de l'élévation des satellites.....	53
3.2.5 Calcul de la distance géométrique des satellites	55
3.2.6 Calcul de la pseudo-distance des satellites	56
3.2.7 Calcul de la fréquence Doppler des satellites	59
3.3 Validation du message binaire GPS L1 du simulateur LACIME	60
3.3.1 Amélioration de l'architecture de la génération du message de navigation.....	61

3.3.2	Calcul de l'indicateur de disponibilité de la correction du message de navigation.....	65
3.3.3	Calcul du Estimated Range Deviation (ERD)	66
3.3.4	Statut de l'antibrouillage et de la configuration des satellites	66
3.3.5	Calcul de l'index User Range Accuracy (URA).....	67
3.3.6	Calcul des paramètres Issue Of Data Clock (IODC) et Issue Of Data Ephemeris (IODE)	68
3.3.7	Calcul du temps de référence de l'horloge et des éphémérides	68
3.3.8	Calcul du numéro courant de la sous-trame.....	69
3.3.9	Calcul du paramètre Time Of Week (TOW)	69
3.3.10	Calcul de la semaine de référence de l'almanach	70
3.3.11	Synchronisation des sous-frames	70
3.3.12	Spécification des bits réservés	70
3.3.13	Génération des messages spéciaux	71
3.3.14	Particularité de la parité des mots #10	72
3.3.15	Problème de perte des signaux de tous les satellites.....	73
3.4	Discussion relative aux résultats finaux de la partie logiciel.....	73
3.5	Conclusion	75
CHAPITRE 4 VALIDATION DE LA PARTIE IF DU SIMULATEUR LACIME.....		77
4.1	Discussion des données brutes initiales décodées par le récepteur u-blox	77
4.2	Problème de délai UTC.....	78
4.3	Délais sur la génération des adresses XMEM.....	80
4.4	Problème lors de la génération de l'horloge principale	81
4.5	Présentation et correction des erreurs sur l'architecture des NCOs.....	83
4.5.1	Erreur de précision du NCO	84
4.5.2	Calcul des phases corrigées du simulateur LACIME	86
4.5.3	Calcul des fréquences porteuses et du code.....	87
4.5.4	Calcul de la phase du message de navigation	90
4.6	Calcul de l'index de la position courante du code du satellite.....	91
4.7	Calcul de l'index de la position courante du bit du message de navigation	94
4.8	Calcul du sinus et du cosinus de la porteuse.....	96
4.9	Discussion des résultats IF finaux.....	97
4.10	Conclusion	98
CHAPITRE 5 PERFORMANCE ET ANALYSE DU SIMULATEUR LACIME.....		101
5.1	Performance en statique du simulateur LACIME.....	101
5.2	Performance en dynamique du simulateur LACIME	106
5.3	Conclusion	112
CONCLUSION.....		114
RECOMMANDATIONS		117
ANNEXE I PROBLÈMES RENCONTRÉS AVEC LE SIMULATEUR LACIME..		121

ANNEXE II	RÉSULTATS DES PARAMÈTRES DE NAVIGATION DU SIMULATEUR LACIME	127
ANNEXE III	ALGORITHME DU CALCUL DE LA POSITION DES SATELLITES	161
ANNEXE IV	ALGORITHME DU CALCUL DE LA VITESSE DES SATELLITES	165
ANNEXE V	DESCRIPTION DES VARIABLES DU MESSAGE DE NAVIGATION	169
ANNEXE VI	MESURES DE L'ERREUR DU MESSAGE DE NAVIGATION GPS L1 DU SIMULATEUR LACIME	173
ANNEXE VII	RÉSULTATS VUS DU RÉCEPTEUR U-BLOX	177
ANNEXE VIII	RÉSULTATS DE L'ERREUR DE LA PHASE DE LA PORTEUSE, DU CODE ET DU MESSAGE DE NAVIGATION	189
ANNEXE IX	RÉSULTATS INITIAUX ET MODIFIÉS DE ERR_CODE ET DE ERR_MESS	199
ANNEXE X	RÉSULTATS FINAUX DES SIMULATIONS EN STATIQUE	207
ANNEXE XI	RÉSULTATS FINAUX DES SIMULATIONS EN DYNAMIQUE	223
BIBLIOGRAPHIE		229

LISTE DES TABLEAUX

	Page
Tableau 1.1	Statut de la constellation pour les différents systèmes de positionnement ..4
Tableau 1.2	Liste des différents simulateurs GNSS disponibles sur le marché et comparaison au simulateur LACIME11
Tableau 1.3	Récapitulation des paramètres et flexibilités des simulateurs.....16
Tableau 1.4	Résumé des fonctionnalités essentielles supportées par les simulateurs GPS/GNSS19
Tableau 2.1	Fréquences théoriques des oscillateurs de la carte RF37
Tableau 2.2	Résumé des problèmes observés du simulateur39
Tableau 2.3	Les moyennes et les écarts-types de l'erreur de position et du DOP (Rx/Spirent versus Rx/LACIME)43
Tableau 3.1	Erreur initiale du message binaire GPS L1 du simulateur LACIME pour le satellite #461
Tableau 3.2	Disponibilité de la table de correction du message de navigation65
Tableau 3.3	Code pour la configuration des satellites67
Tableau 3.4	Index URA selon la précision des données de navigation67
Tableau 3.5	Bits réservés des sous-frames du message binaire de navigation71
Tableau 3.6	Messages spéciaux envoyés71
Tableau 3.7	Nombre de décimales des paramètres de navigation (LACIME versus SimGen)73
Tableau 3.8	Erreur modifiée du message binaire de navigation du satellite #474
Tableau 3.9	Santé des satellites de l'almanach74
Tableau 3.10	Résumé des corrections apportées aux paramètres de navigation75
Tableau 3.11	Résumé des corrections apportées aux variables de navigation75
Tableau 3.12	Résumé des corrections apportées au message binaire de navigation76
Tableau 4.1	Sinus et cosinus de la phase de la porteuse97

XVIII

Tableau 4.2	Quadrant de la phase de la porteuse.....	97
Tableau 4.3	Résumé des corrections apportées à la IF du simulateur LACIME.....	99
Tableau 5.1	Latitudes et longitudes des positions simulées pour l'analyse en statique	102
Tableau 5.2	Moyennes et écarts-types de l'erreur de position en statique à différentes positions (Rx/Spirent versus Rx/LACIME).....	104
Tableau 5.3	Moyennes et écarts-types du DOP en statique à différentes positions (Rx/Spirent versus Rx/LACIME)	105
Tableau 5.4	Moyennes et écarts-types de l'erreur de la position en dynamique à différentes vitesses (Rx/Spirent versus Rx/LACIME).....	110
Tableau 5.5	Moyennes et écarts-types de l'erreur de la vitesse en dynamique à différentes vitesses (Rx/Spirent versus Rx/LACIME).....	111
Tableau 5.6	Moyennes et écarts-types du DOP en dynamique à différentes vitesses (Rx/Spirent versus Rx/LACIME)	111

LISTE DES FIGURES

		Page
Figure 2.1	Montage actuel du banc de test du simulateur LACIME.....	22
Figure 2.2	Morphogramme du processus #1 du simulateur actuel.....	23
Figure 2.3	Morphogramme du processus #2 du simulateur actuel.....	24
Figure 2.4	Architecture logicielle du simulateur.....	25
Figure 2.5	Fonctionnalités de haut niveau du Virtex IV	27
Figure 2.6	Structure d'un canal d'un satellite.....	28
Figure 2.7	Schéma bloc d'un canal GNSS	29
Figure 2.8	Interface XMEM et FSB de la section IF	30
Figure 2.9	Adresse de la mémoire XMEM	31
Figure 2.10	Carte RF du simulateur LACIME.....	32
Figure 2.11	Architecture initiale de la partie RF du simulateur LACIME.....	33
Figure 2.12	Forme d'onde à la sortie de la PLL pour le signal L1	35
Figure 2.13	Forme d'onde obtenue à la sortie de la PLL pour les signaux L5 et E5a ..	36
Figure 2.14	Forme d'onde obtenue à la sortie de la PLL pour le signal E5b	36
Figure 2.15	L'erreur de la position (Rx/Spirent versus Rx/LACIME).....	41
Figure 2.16	Nuage de points (Rx/Spirent versus Rx/LACIME)	41
Figure 2.17	PDOP, HDOP et VDOP (Rx/Spirent versus Rx/LACIME)	42
Figure 2.18	Visibilités des satellites (Rx/Spirent versus Rx/LACIME)	42
Figure 3.1	Diagramme de classes de la fonction d'enregistrement des données.....	45
Figure 3.2	Position en x et son erreur initiale et modifiée du satellite #4	47
Figure 3.3	Position en y et son erreur initiale et modifiée du satellite #4	47
Figure 3.4	Position en z et son erreur initiale et modifiée du satellite #4	48

Figure 3.5	Vitesse en x et son erreur initiale et modifiée du satellite #4	49
Figure 3.6	Vitesse en y et son erreur initiale et modifiée du satellite #4	49
Figure 3.7	Vitesse en z et son erreur initiale et modifiée du satellite #4.....	50
Figure 3.8	Angle d'azimut et d'élévation d'un satellite.....	51
Figure 3.9	L'angle d'azimut et son erreur initiale et modifiée du satellite #4	52
Figure 3.10	L'angle d'élévation et son erreur initiale et modifiée du satellite #4.....	54
Figure 3.11	Γ et Γ_{err} initiale et modifiée du satellite #4	55
Figure 3.12	ρ et ρ_{err} initial et modifié du satellite #4.....	57
Figure 3.13	Δf et Δf_{err} initial et modifié du satellite #4.....	59
Figure 3.14	Format du message de navigation GPS L1	62
Figure 3.15	Diagramme d'état de la génération du message binaire de navigation GPS L1	63
Figure 3.16	Morphogramme modifié de la génération des messages binaires de navigation (GPS L1)	64
Figure 4.1	Morphogramme modifié du processus #2.....	79
Figure 4.2	Représentation du module esclave de la mémoire XMEM	81
Figure 4.3	L'onde initiale et modifiée de l'horloge principale.....	81
Figure 4.4	Montage de la génération de l'horloge.....	82
Figure 4.5	Interaction initiale des NCOs entre le logiciel et le FPGA.....	83
Figure 4.6	$\varphi_{\text{porteuse}}$ et $\varphi_{\text{err_porteuse}}$ d'une porteuse de 30 MHz avec $F_s=100$ MHz.....	85
Figure 4.7	$\varphi_{\text{porteuse}}$, φ_{code} et φ_{mess} du satellite #4 calculé par le logiciel avec $F_s=100$ MHz ($\varphi_{\text{porteuse}}$ et φ_{code}) et $F_s=1000$ Hz (φ_{mess})	86
Figure 4.8	$\varphi_{\text{err_porteuse}}$ initiale et modifiée du satellite #4 avec $F_s=1$ Hz	88
Figure 4.9	$\varphi_{\text{err_code}}$ initial et modifié du satellite #4 avec $F_s=1$ Hz.....	88
Figure 4.10	$\varphi_{\text{err_mess}}$ initiales et modifiée du satellite #4 avec $F_s=1$ Hz	90
Figure 4.11	L'index du code courant calculé par le logiciel pour le satellite #4	92

Figure 4.12	err_code initiale et modifiée du satellite #4 avec $F_s=1\text{Hz}$	93
Figure 4.13	L'index de la position courante du bit du message de navigation du satellite #4	94
Figure 4.14	err_mess initiale et modifiée du satellite #4 avec $F_s=1\text{ Hz}$	96
Figure 5.1	Visibilités des satellites d'une simulation au Colorado (Rx/Spirent versus Rx/LACIME)	102
Figure 5.2	L'erreur de position d'une simulation au Colorado (Rx/Spirent versus Rx/LACIME)	103
Figure 5.3	Nuage de points d'une simulation au Colorado (Rx/Spirent versus Rx/LACIME)	103
Figure 5.4	PDOP, HDOP et VDOP d'une simulation au Colorado (Rx/Spirent versus Rx/LACIME)	104
Figure 5.5	Trajectoire simulée affichée sur Google Earth	107
Figure 5.6	Visibilités des satellites pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)	108
Figure 5.7	L'erreur de la position pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)	108
Figure 5.8	Nuage de points pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)	109
Figure 5.9	Erreur de la vitesse pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)	109
Figure 5.10	PDOP, HDOP et VDOP pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)	110

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

2D	Deux dimensions (latitude, longitude)
3D	Trois dimensions (latitude, longitude, altitude)
AltBOC	Alternate Binary Offset Carrier
BOC	Binary Offset Carrier
BPSK	Binary Phase Shift Keying
C/N ₀	Ratio signal sur interférence
CASM	Coherent Adaptive Subcarrier Modulation
CBOC	Composite Binary Offset Carrier
COMPASS	Système de navigation chinois
CRSNG	Conseil de Recherches en Sciences Naturelles et en Génie
CSV	Comma-Separated Values
DAC	Digital to Analog Converter
DOP	Dilution Of Precision
ECEF	Earth Centered Earth Fixed
EGNOS	European Geostationary Navigation Overlay System
ENU	East North Up
ERD	Estimated Range Deviation
ÉTS	École de Technologie Supérieure
FIFO	First In First Out
FPGA	Field Programmable Gate Array
F _s	Fréquence d'échantillonnage
FSB	Front Side Bus
GAGAN	GPS Aided Geo-Augmented Navigation
GALILÉO	Système de navigation européen
GLONASS	Système de navigation russe
GNSS	Global Navigation Satellite System
GPIB	General Purpose Interface Bus
GPS	Global Positioning System
GUI	Graphical User Interface

HDOP	Horizontal Dilution Of Precision
I	Composante en phase du signal GPS
IF	Intermediate Frequency
IODC	Issue Of Data Clock
IODE	Issue Of Data Ephemeris
LACIME	Laboratoire de Communication d'Intégration de la Micro-Électronique
LAN	Local Area Network
MCLK	Horloge du message de navigation
MHz	MégaHertz
Ms	Milliseconde
MSAS	Multi-functional Satellite Augmentation System
NAV	NAVigation
NCO	Numerical Controlled Oscillator
NMCT	Navigation Message Correction Table
NMEA	National Marine Electronics Association
Ns	Nanosecondes
OCXO	Oven Controlled X-tal (Crystal) Oscillator
PC	Ordinateur personnel
PCLK	Horloge du code
PDOP	Position Dilution Of Precision
PLL	Phase Locked Loop (boucle à verrouillage de phase)
PRN	Pseudorandom noise
Q	Composante en quadrature du signal GPS
QPSK	Quadrature Phase Shift Keying
QZSS	Quasi-Zenith Satellite System
RAM	Random Access Memory
RDC	Recherche et Développement Coopérative
RF	Radio Fréquence
RINEX	Receiver Independent Exchange Format

RTG2	Real Time GNSS Generator
SBAS	Satellite-Based Augmentation System
SEM	System Effectiveness Model
TOW	Time Of Week
TTF	Time To First Fix
URA	User Range Accuracy
URSS	Union des Républiques Socialistes Soviétiques
USB	Universal Serial Bus
UTC	Universal Time Coordinated
VDOP	Vertical Dilution Of Precision
VGA	Voltage Controlled Amplifier/Attenuator
VHDL	VHSIC hardware description language
WAAS	Wide Area Augmentation System
XMEM	eXternal MEMory

LISTE DES SYMBOLES ET UNITÉS DE MESURE

Code FPGA	L'index du code du FPGA
Code logiciel	L'index du code calculé par le logiciel C++.NET
err_code	Erreur du code
err_mess	L'erreur du pointeur de la position du bit du message de navigation
Hz	Hertz
m	Mètre
m/s	Mmètre par seconde
Mess FPGA	L'index du message de navigation du FPGA
Mess logiciel	L'index du message de navigation calculé par le logiciel C++.NET
MHz	Mégahertz
mHz	Millihertz
mm	millimètre
mrad	Milliradian
ms	Milliseconde
nrad	Nanoradian
ns	Nanoseconde
ppm	Partie par million
rad	Radian
Rx/LACIME	Simulateur LACIME vue du récepteur u-blox
Rx/Spirent	Simulateur Spirent vue du récepteur u-blox
sec	Seconde
t_u	Biais d'horloge du récepteur
Γ	Distance géométrique
Γ_{err}	Erreur de la distance géométrique
Δf	Fréquence Doppler
Δf_{err}	Erreur de la fréquence Doppler

μ_{err}	Moyenne de l'erreur de position 2D
μ_{errx}	Moyenne de l'erreur de position selon l'axe des x
μ_{erry}	Moyenne de l'erreur de position selon l'axe des y
μ_{hdop}	Moyenne du HDOP
μ_{pdop}	Moyenne du PDOP
μ_{vdop}	Moyenne du VDOP
$\mu_{vitesse}$	Moyenne de l'erreur de la vitesse
μHz	Microhertz
π	Pi
ρ	Pseudo-distance
ρ_{err}	Erreur de la pseudo-distance
σ_{err}	Écart-type de l'erreur de position 2D
σ_{errx}	Écart-type de l'erreur de position selon l'axe des x
σ_{erry}	Écart-type de l'erreur de position selon l'axe des y
σ_{hdop}	Écart-type du HDOP
σ_{pdop}	Écart-type du PDOP
σ_{vdop}	Écart-type du VDOP
$\sigma_{vitesse}$	Écart-type de l'erreur de la vitesse
φ_{code}	Phase du code
φ_{err_code}	Erreur de la phase du code
φ_{err_mess}	Erreur de la phase du message de navigation
$\varphi_{err_porteuse}$	Erreur de la phase de la porteuse
$\varphi_{logiciel}$	Phase calculée par le logiciel C++.NET
φ_{mess}	Phase du message de navigation
φ_{mess_FPGA}	Phase du message de navigation du FPGA
φ_{NCO}	Phase du NCO
$\varphi_{porteuse}$	Phase de la porteuse

INTRODUCTION

De 2000 à 2010 inclusivement, le système de navigation américain GPS était le seul couvrant la totalité du globe. Le système russe GLONASS a une zone de couverture de 100 % de la surface de la planète depuis le 8 décembre 2011 (RIA Novosti, 2011). Avec le nombre grandissant de satellites de positionnement, l'Agence spatiale fédérale russe a annoncé le lancement du projet « SOCIAL-GLONASS ». Ceci est un service permettant d'améliorer la qualité de vie des personnes souffrant de cécité, des gens âgés et des enfants. Aussi, l'iPhone 4S devient le premier téléphone intelligent compatible GPS/GLONASS (Andrillon, 2011). D'autres idées novatrices pourront se concrétiser avec les futures constellations : Galiléo (européen) et Compass (chinois). Selon les derniers développements, une couverture globale est prévue en 2014 pour Galiléo (Le nouvel Observateur, 2010) et en 2020 pour Compass (Leblal, 2011).

La tendance des prochains récepteurs est de capter tous les satellites de navigation. Ainsi, ceci améliorera les performances des récepteurs GNSS pour ainsi concevoir des applications révolutionnaires. Les récepteurs seront plus précis et une meilleure disponibilité des signaux en continu engendrera moins de perte de positionnement. Avant de les mettre sur le marché, il est nécessaire de tester ces appareils. Comme les systèmes de navigation Galiléo et Compass ne sont pas encore fonctionnels, il n'est pas encore possible de recevoir ces signaux. Donc, la validation des récepteurs GNSS ne peut pas se faire maintenant ce qui retarde la vente du produit. Lorsque tous les signaux GNSS seront disponibles, une solution possible pour tester les récepteurs consiste à poser une antenne sur le toit d'un édifice et de la relier par un câble jusqu'au laboratoire. Cette manière de procéder permet seulement de faire des essais à la même position. La seconde solution nécessite à mettre l'équipement de test dans un véhicule en mouvement et ainsi vérifier la position affichée par le récepteur. Cette technique entraîne des coûts très importants de déplacements et de temps diminuant considérablement l'efficacité de cette méthode.

Le Laboratoire de Communication d'Intégration de la Micro-Électronique (LACIME) rassemble des chercheurs qui travaillent présentement sur le problème de validation des récepteurs GNSS. Un simulateur GNSS est en cours de développement depuis 2005. Celui-ci a pour fonction de reproduire des signaux réels de navigation. Il est composé de trois parties distinctes : logiciel, IF (Intermediate frequencies) et RF (Radio fréquence). La partie logicielle, développée en C++, permet de configurer les paramètres d'une simulation (l'heure, la durée, la trajectoire, etc.) et d'effectuer les calculs pour la génération de la constellation. L'électronique numérique est basée sur un Field Programmable Gate Array (FPGA) Virtex IV de la compagnie Xilinx qui module les signaux de navigation à la fréquence intermédiaire à partir des données calculées du logiciel. Un convertisseur numérique à analogique (DAC) est employé pour convertir les signaux numériques en analogique par une carte RF. Celle-ci transpose les signaux numériques de navigation vers la fréquence réelle des satellites. En procédant de cette manière, tous les récepteurs GNSS seront en mesure de recevoir les signaux et de calculer la position simulée.

Le premier chapitre du mémoire décrira le projet de recherche et fera une revue des caractéristiques des différents simulateurs GNSS présentement sur le marché. Aussi, la problématique de recherche sera énoncée ainsi que les objectifs principaux et spécifiques à atteindre. Une méthodologie devra être suivie pour le bon fonctionnement du projet. Le deuxième chapitre comportera une description initiale de l'architecture logicielle et matérielle du simulateur GPS L1. Une première étude sera effectuée dans le but d'avoir une vue d'ensemble des performances et de trouver les problèmes majeurs du simulateur. Les chapitres III et IV décriront les améliorations et les validations logicielles et matérielles qui ont été apportées au simulateur sur la constellation GPS L1. Pour finaliser le mémoire, à la suite des corrections qui auront été faites, les chapitres V et VI décriront les performances GPS L1 du simulateur en position statique et dynamique à l'aide de scénarios typiques de l'industrie.

CHAPITRE 1

DESCRIPTION DU PROJET DE RECHERCHE ET REVUE DE LITTÉRATURE

Le présent chapitre donne le cadre du projet de recherche, ses objectifs et la méthodologie employée. Après, la contribution générale de l'oeuvre et une revue de littérature des différents simulateurs GNSS vont être également présentées.

1.1 Mise en contexte et problématique

En date du 12 mai 2012, il y a seulement deux systèmes de navigations globales ayant une zone de couverture de 100% : le système américain Global Positionning System (GPS) et le système russe GLObal NAVigation Satellite System (GLONASS). Le système GPS a débuté en 1960 et il était destiné uniquement au domaine militaire. C'est suite à un tragique accident d'avion du transporteur Korean Air Lines en 1983, que Ronald Reagan, le président des États-Unis de l'époque, décida de rendre le système GPS disponible pour les civils (Smarthmore College Computer Society, 2011). Aujourd'hui, il est utilisé dans plusieurs secteurs tels que le transport, l'agriculture et la sécurité.

Pour ne plus dépendre du système de positionnement américain, les Russes ont créé leur propre système de positionnement par satellite (GLONASS) durant la guerre froide. Après la chute de l'Union des Républiques Socialistes Soviétiques (URSS), le budget alloué au projet ne cessa de diminuer ce qui a eu comme conséquence la perte de plusieurs satellites, car certains d'entre eux ont atteint leur durée de vie. Ainsi, ce phénomène dégrade considérablement les performances de GLONASS le rendant inutilisable. À partir des années 2000, les Russes commençaient à remplacer les satellites perdus pour rendre leur système accessible à tous le 8 décembre 2011 (RIA Novosti, 2011).

Prochainement, les Chinois et les Européens auront leur propre système de positionnement qui s'ajoutera à ceux déjà existants. Le Tableau 1.1 montre l'état actuel des satellites de

différents systèmes de positionnement en date du 19 mai 2012 (Federal Space Agency, 2012), (European Space Agency, 2012), (Canton Fair, 2012).

Tableau 1.1 Statut de la constellation pour les différents systèmes de positionnement globaux

Système de navigation	Nombre de satellites			
	En entretien	Opérationnel	En constellation	À lancer
GPS	1	30	31	0
GLONASS	8	23	31	0
GALILÉO	2	0	2	28
COMPASS	0	11	11	24

Le système de navigation Galiléo a subi de nombreux retards durant sa phase de développement. En effet, le projet étant dispendieux, aucune compagnie ne désirait prendre le risque d'investir dans celui-ci. En 2008, la commission européenne prend la décision de financer le nouveau système de positionnement à l'aide de financement intégralement public. (Chicheportiche, 2010). Malgré tout, Galiléo est prévu être fonctionnel en 2014 (Le nouvel Observateur, 2010).

La conception du système de positionnement chinois a débuté en 2000 (Le Quotidien du Peuple en ligne, 2012). Le 25 février 2012, les Chinois ont lancé avec succès leur 11^{ème} satellite et (Le Quotidien du Peuple en ligne, 2012) en 2012, la Chine prévoit en lancer 8 autres. (Le Quotidien du Peuple en ligne, 2011b). Ainsi, le système de positionnement basé sur 35 satellites couvrant la surface complète du globe sera complété d'ici 2020 (Leblal, 2011). À la fin de 2012, la couverture de COMPASS se concentrera sur la région Asie-Pacifique (Le Quotidien du Peuple en ligne, 2011a).

Un système GNSS englobe plusieurs systèmes de positionnement comme GPS, GLONASS, COMPASS et Galiléo. Il y a plusieurs avantages à employer tous ces signaux disponibles. Effectivement, on améliore la disponibilité du service parce qu'il y a davantage de satellites dans le ciel. La précision de la position est fortement accrue tout en assurant une meilleure

intégrité des informations reçues. La multitude des signaux aux différentes fréquences risque d'être problématique quant à leur intégrité, leur redondance, leur sécurité et leur vulnérabilité au brouillage.

Il existe plusieurs manières de vérifier le bon fonctionnement des récepteurs GPS en développement ou en intégration dans une application. La première consiste à poser une antenne sur le toit d'un édifice et de la relier à un câble jusqu'au laboratoire. Cette façon de procéder permet seulement de tester à une seule position fixe tout en perdant de la puissance et de la qualité du signal. La deuxième nécessite de se déplacer dans un véhicule avec le récepteur suivant une trajectoire donnée. Cette méthode engendre des pertes de temps importantes, car les ingénieurs doivent changer de lieu pour faire des tests sur le terrain. Avec les deux techniques décrites précédemment, il n'est pas possible de faire des analyses minutieuses des comportements du récepteur en rejouant les scénarios de navigation.

Un simulateur de constellation GNSS doit reproduire les signaux réels de navigation dans la bande d'origine radio fréquence : les signaux de navigation globaux (GPS, GLONASS, COMPASS, Galiléo) et les signaux d'augmentation comme le Wide Area Augmentation System (WASS) et le Quasi-Zenith Satellite System (QZSS, Japon). Ce dispositif est configurable permettant à l'utilisateur d'exécuter plusieurs scénarios possibles et variés. Ceux-ci sont répétables assurant une grande fiabilité des résultats simulés. Le coût de déploiement des récepteurs GNSS diminue, car la plupart des tests effectués sur le terrain peuvent être réalisés avec un simulateur. Ainsi, il devient plus facile d'interpréter les résultats qui sont envoyés au récepteur et de dériver l'application conçue.

Au moment d'amorcer le projet, le LACIME avait déjà un simulateur qui simulait seulement la constellation GPS L1. Plusieurs collaborateurs ont contribué à la conception logicielle et matérielle de celui-ci durant les dernières années. Par contre, en exécutant des simulations, plusieurs problèmes majeurs ont été observés et ils se répètent peu importe la configuration du scénario. Ceux-ci seront étudiés en détail à la section 2.1.4 et corrigés aux chapitres III et IV.

1.2 Objectifs principaux et spécifiques du projet

L'objectif principal du projet de recherche consiste à générer en temps réel les signaux de navigation GNSS. Dans le but de simplifier la tâche, le simulateur existant au LACIME sera réutilisé. Par contre, il est nécessaire d'effectuer des modifications afin d'obtenir des performances similaires à un simulateur commercial : une moyenne maximale de l'erreur de position de 0,3 mètre pour une simulation de 24 heures dans les conditions idéales. Donc, l'objectif spécifique du projet est de déverminer le simulateur LACIME dans l'intention d'améliorer ses performances et de le commercialiser. Pour parvenir à ce but, certaines cibles secondaires devront être atteintes :

- robustesse de la simulation,
- stabilité des résultats,
- répétabilité des résultats.

Plusieurs actions devront être effectuées pour s'assurer que les points énumérés précédemment seront respectés. Tout d'abord, il faudra vérifier la validité des formules des paramètres de tous les satellites (position, vitesse, pseudo-distance etc). Ainsi, en ayant les bonnes valeurs de pseudo-distances, ceci nous assure que les biais d'horloges des satellites sont estimés correctement ce qui est une étape importante pour la stabilité des horloges. Le message de navigation devra lui aussi être validé. Il contient plusieurs informations importantes dont les paramètres d'éphémérides et d'almanach permettant au récepteur de positionner correctement les satellites dans le ciel.

La stabilité des horloges est influencée par la génération des fréquences du code et du message de navigation. C'est principalement à partir du code qu'un récepteur GNSS détermine son biais d'horloge qui est essentiel pour positionner correctement le véhicule. Aussi, le calcul de la porteuse est très critique parce qu'une erreur sur sa génération empêche la démodulation du signal GNSS et ainsi engendrer une désynchronisation au récepteur.

Donc, le calcul des phases de la porteuse, du code et du message de navigation du simulateur LACIME devront être révisés rigoureusement afin d'émettre un signal GNSS valide.

1.3 Méthodologie

Le projet de recherche comporte particulièrement trois phases. La première consiste à se familiariser avec le système actuel du simulateur GPS L1 dans le but de bien comprendre son fonctionnement et de faciliter sa modification. Pour effectuer cette tâche, il devient nécessaire de lire la documentation du simulateur LACIME et de faire une bonne revue de la littérature. Ainsi, cela permettra de mieux comprendre comment la génération des signaux s'effectue et la façon dont l'architecture matérielle et logicielle a été conçue. Différents scénarios de tests devront être faits en statique et dynamique pour examiner le bon fonctionnement du simulateur et de s'assurer que celui-ci reflète bien la réalité. Le même scénario sera exécuté avec le simulateur LACIME et le simulateur commercial GSS7700 de Spirent. Les performances des deux simulations devront être "similaires", c'est-à-dire la position calculée par le récepteur doit avoir une déviation maximale de 0,3 mètre par rapport à celle simulée par le simulateur.

La deuxième phase consiste à valider et à corriger si nécessaire les algorithmes permettant de calculer les paramètres de navigation à l'aide du simulateur Spirent. En utilisant la fonction d'enregistrement du logiciel SimGen de la compagnie Spirent, il est possible d'avoir accès au message binaire de navigation ainsi que plusieurs autres variables (pseudo-distance, position et vitesse des satellites, etc.). Par contre, cette fonctionnalité n'est pas encore implémentée dans le simulateur LACIME et elle devra d'abord être réalisée. Après, l'erreur du paramètre de navigation qui est la différence entre la valeur calculée de SimGen et du simulateur LACIME sera trouvée. Celle-ci doit être bruitée et inférieure à 6×10^{-5} afin d'assurer la fiabilité des résultats du simulateur LACIME. Chaque bit du message de navigation du simulateur LACIME et Spirent au même temps de la semaine (TOW) sera comparé ensemble et il y a une erreur lorsqu'il est différent entre les deux simulateurs. Il est nécessaire que le

message binaire de navigation produit par ceux-ci soit identique pour ne pas transmettre de mauvaises informations.

La troisième phase comprend la validation du matériel du simulateur LACIME et se fera en deux étapes. La première consiste à exécuter deux simulations identiques : l'une avec le simulateur Spirent et l'autre avec le simulateur LACIME. Un récepteur u-blox EVK-6T recevra les signaux simulés et enregistrera les données brutes reçues. Les biais d'horloge, les pseudo-distances et les Dopplers des satellites recueillis seront comparés ensemble pour les deux simulations. Les données brutes enregistrées par le récepteur au même temps écoulé depuis le début de la simulation devront être semblables pour les deux simulations afin d'obtenir une précision de la position suffisamment élevée à un simulateur commercial. La seconde étape de la validation du matériel permet de corriger les phases ainsi que la valeur du pointeur du bit courant du code et du message de navigation. Les phases sont générées en temps réel par le Field Programmable Gate Array (FPGA) et le logiciel corrige à toutes les secondes l'erreur de celles-ci qui s'accumule au fil du temps. L'index courant du code et du message de navigation calculé par le logiciel initialise seulement les canaux. C'est le FPGA qui l'incrémente selon les valeurs des phases. Pour chaque phase et chaque index courant du code et du message de navigation, il faudra s'assurer que la formule permettant de les calculer est correcte en vérifiant que les fréquences du protocole GPS sont respectées. Ensuite, leur erreur sera estimée et celle-ci représente la différence entre la valeur calculée par le logiciel et le FPGA. Aucune erreur n'est permise, car ceci indique que les fréquences générées ne sont pas les bonnes.

La dernière phase du projet est une caractérisation du simulateur LACIME en statique et en dynamique. Pour l'analyse en position fixe, huit emplacements différents sur le globe terrestre seront simulés avec le simulateur Spirent et LACIME. Quant aux simulations en dynamiques, elles se feront en simulant avec les deux simulateurs la même trajectoire curviligne avec des vitesses initiales constantes de 10 km/h, 50 km/h et 100 km/h. À partir des données enregistrées par le récepteur, la moyenne et l'écart-type de l'erreur de la position et du Dilution Of Precision (DOP) seront calculés pour chacune des simulations de chaque

simulateur. Ainsi, il sera possible d'étudier les performances des deux simulateurs en comparant leur résultat.

1.4 Retombées prévues

Dans les années futures, la technologie du GNSS sera de plus en plus employée pour la localisation. Dans le but de concevoir de meilleurs récepteurs, il est essentiel d'utiliser des outils fiables pour les tester. Un simulateur de constellation GNSS est un outil indispensable pour contrôler les paramètres de la simulation. Cet appareil intéressera les industriels à la condition qu'il soit correctement calibré avec une précision convenable et qu'il respecte les normes de l'industrie. Les entreprises en recherche et développement s'intéresseraient à un tel dispositif, puisqu'il leur permettrait de mieux comprendre le potentiel des signaux et de mettre au point, tester et certifier leurs produits. Ainsi, ceci pourrait être utilisé pour reproduire un scénario problématique à l'heure actuelle dans le but d'améliorer les produits utilisant la technologie du positionnement par satellite.

Un simulateur FPGA a un faible coût, car les portes logiques sont réalisées par programmation. Dans ce cas, il n'est pas nécessaire de se procurer des pièces électroniques comme des diodes pour leur conception. Les outils de synthèse des FPGAs ont la capacité d'optimiser le circuit électronique numérique rapidement ce qui diminue sa complexité. Un FPGA n'a pas besoin d'être calibré et il offre une grande flexibilité. Effectivement, le design se change beaucoup plus facilement et celui-ci est capable de s'adapter à différentes simulations. Par exemple, il est capable de générer une onde sinusoïdale d'une porteuse IF de navigation dont sa fréquence a été calculée par le logiciel. Donc, le logiciel commande la génération des signaux IF effectuée par le FPGA.

En région montagneuse et urbaine, il est plus fréquent de perdre le signal GPS. En effet, les montagnes et les gratte-ciel bloquent les signaux de navigation. En ajoutant d'autres satellites de plusieurs constellations dans la solution de navigation, la disponibilité du service

de positionnement est meilleure. Ainsi, lorsqu'on se promène en voiture, les dispositifs GNSS perdront moins fréquemment la position qu'en employant seulement les signaux GPS.

Le domaine de l'aviation utilise le GPS pour connaître l'emplacement courant des avions. La précision de la position est fondamentale, car les algorithmes de guidage utilisent cette mesure. Une grande erreur de la position calculée pourrait faire atterrir un avion à côté de la piste d'atterrissage. Donc, les signaux GNSS vont permettre aux avions de se positionner avec une très faible erreur.

Tous les téléphones intelligents et les nombreux modèles de portables ont un système GPS pour pouvoir retrouver son propriétaire. Lors de période de détresse pouvant nuire à la vie d'une personne, les autorités utilisent les systèmes de positionnement dans les mobiles afin de trouver plus facilement les gens demandant de l'aide. Avec un système GNSS intégré dans les cellulaires, la position sera encore plus précise et plus sûre ce qui permettra au service d'urgence d'être plus rapide et efficace dans leur intervention. Lorsqu'une vie est en danger, chaque minute compte et peut faire la différence.

1.5 Revue des caractéristiques des simulateurs de constellation GPS/GNSS actuels

Comme la complexité du domaine de la radionavigation augmentera avec l'apparition de nouveaux signaux GNSS, il devient d'autant plus important de valider les récepteurs avec un simulateur. Donc, il faut une analyse des simulateurs GNSS présents sur le marché pour mieux distinguer les fonctions essentielles, les performances et les spécifications qui les composent. Ceci permettra de mieux connaître ce qu'un simulateur GNSS doit posséder. Le Tableau 1.2 énumère différents simulateurs GNSS.

Le GPSG-1000 de la compagnie Aeroflex est un simulateur GNSS portable spécialement conçu pour le monde de l'aviation. Il est facile d'utilisation et configurable tout en ayant la capacité de simuler un maximum de 12 satellites. Il supporte les constellations GPS et Galiléo ainsi que certains systèmes d'augmentation. Celui-ci n'offre pas la possibilité

d'enregistrer les données et d'avoir un contrôle sur le message de navigation et de ses paramètres. Les effets de la troposphère et de l'ionosphère sont simulés chacun avec un modèle. Les C/N_0 s de chaque satellite peuvent être modifiés individuellement pour ajouter du réalisme aux simulations. Le simulateur GPSG-1000 doit être calibré une fois par année pour avoir de meilleures performances et cette tâche se réalise par un technicien d'Aeroflex. La simulation des multitrajets s'effectue selon l'environnement prédéfini : zone rurale, urbaine et aéronautique. Les modèles de Rice et de Rayleigh établissent les atténuations du signal de navigation (Aeroflex, 2010).

Tableau 1.2 Liste des différents simulateurs GNSS disponibles sur le marché et comparaison au simulateur LACIME

Nom de la compagnie	Modèle du simulateur	Signaux GNSS et système d'augmentation simulés	Canaux	Coût (\$ CAN)
Aeroflex	GPSG-1000	GPS L1, L1C, L2C, L5 Galiléo E1, E5, E5a, E5b, E6 WAAS/EGNOS L1, L5	12	17 595
Agilent	EXG N5172B PXB N5106A	GPS L1 GLONASS L1	32	152 634
Cast Navigation	Cast-2000	GPS L1, L2	12	140 000
IFEN	NavX-NCS pro	GPS L1, L2C, L5 Galiléo E1, E5, E6 GLONASS G1, G2 QZSS L1 WAAS, EGNOS, MSAS	108	510 000
LACIME	-	GPS L1	8	-
Racelogic	Labsat 2	GPS L1, GLONASS G1 WAAS/EGNOS L1	-	33 500
Rohde & Schwarz	SMBV100A	GPS L1, L2 Galiléo E1 GLONASS L1, L2	24	74 327
Silicom	RTG2	GPS L1, L2C, L5 GLONASS G1 Galiléo E1, E5, E5a, E5b, E6 COMPASS L1 EGNOS, QZSS et WASS	120	338 850
Spectracom	GSG-62	GPS L1, L2 GLONASS L1, L2 WAAS, EGNOS, GAGAN, MSAS	32	34 280
Spirent	GSS7700	GPS L1 WAAS	16	250 000

Le GSS7700 de Spirent est composé d'un FPGA qui module le signal GPS. Le logiciel SimGen pour Windows permet aux usagers de préciser les paramètres de la simulation reliés au message de navigation, au C/N_0 des satellites et le modèle utilisé pour simuler les délais atmosphériques. En plus, il enregistre les données en format CSV des satellites (position, vitesse, angle d'azimuth et d'élévation etc.) et du véhicule simulé (position, vitesse, etc.). Le GSS7700 est compatible avec d'autres produits de l'entreprise pour tester les systèmes de navigation inertielle. L'extensibilité de l'architecture autorise l'ajout optionnel de modules comme un générateur d'interférence. Pour obtenir de meilleurs résultats, il est nécessaire de calibrer le circuit Radio Fréquence (RF) à tous les ans. Les résultats obtenus du simulateur LACIME seront comparés avec ce produit commercial (Spirent Communications, 2007).

Le Labsat 2 de Racelogic est un simulateur limité et peu dispendieux comparativement à ses concurrents. La simulation se fait à partir de signaux réels, c'est-à-dire que l'appareil reproduit les signaux qu'il enregistre. Il est possible de se procurer le logiciel SatGen qui permet de faire des simulations en dynamique. Ce programme génère un fichier binaire d'une trajectoire qui sera lu par le Labsat 2 lorsqu'on désire recueillir les signaux réels de navigation. Le trajet simulé peut être édité à partir de Google Earth ou simplement d'un fichier National Marine Electronics Association (NMEA). Aussi, les messages de navigation peuvent être envoyés sur les ports RS-232/422/485 pour consulter leur contenu (Racelogic, 2010).

Cast-2000 de l'entreprise Cast navigation est un simulateur GNSS ayant la capacité de simuler 12 satellites sur la bande de fréquence GPS L1 et L2 et d'analyser les performances d'antibrouillages. Les délais causés par la troposphère, l'ionosphère et les multitrajets peuvent être simulés. Le message de navigation est modifiable manuellement avec une interface usager ainsi que chaque C/N_0 des satellites. Une fonctionnalité d'enregistrement des données du simulateur permet de consulter et de produire des graphiques sur les paramètres de la position et des satellites simulés. Pendant la calibration RF qui se fait à l'usine, un technicien génère un fichier de configuration '.rf_loss' dont le simulateur va lire

pour se calibrer. Au fil du temps, le Cast-2000 se décalibre et un utilitaire reproduira un fichier '.rf_loss' pour le recalibrer (Cast Navigation, 2010).

Le NavX-NCS professionnel, un simulateur GNSS, simule 108 satellites en même temps ayant les constellations GPS, Glonass, Galiléo, QZSS et Satellite-Based Augmentation System (SBAS). L'interface graphique du simulateur permet à l'utilisateur de modifier le contenu du message de navigation et chaque C/N_0 des satellites. La fonctionnalité d'enregistrement du dispositif enregistre les données d'éphémérides en format Receiver Independent Exchange Format (RINEX) et le contenu du message de navigation. Il n'est pas nécessaire de calibrer la RF du simulateur, mais il est recommandé de le faire à tous les trois ans. Les délais causés par l'ionosphère sont modélisés avec le modèle Klobuchar ou en important un fichier RINEX. Trois modèles sont employés pour simuler les délais de la troposphère : WAAS Mops, Stanag et Saastamoinen. Deux types de multitrajet peuvent être reproduits : par compensation fixe et par réflexion sur le sol. Le premier consiste à ajouter un retard sur le code et le Doppler du satellite. Le second est simulé selon les conditions géométriques et de la hauteur de l'antenne par rapport au sol. Il est possible de brancher un composant additionnel pour générer du bruit (IFEN, 2010).

Le simulateur LACIME simule seulement les signaux GPS L1. Par contre, la carte RF permet de transposer les signaux IF de navigation de la constellation Galiléo. Le logiciel n'a pas encore implémenté cette fonctionnalité et elle le sera éventuellement dans un développement futur. Le logiciel doit être installé dans un ordinateur et il configure le scénario à simuler. Celui-ci affiche à l'écran en temps réels les paramètres des satellites simulés (position, vitesse, pseudo-distance, etc.) et du véhicule (position, vitesse, etc.). Il n'existe aucune fonction d'enregistrement des mesures, de contrôle du message de navigation et des C/N_0 s des satellites. Les librairies de Google Earth montrent la trajectoire simulée et ils permettent de voir la position du véhicule en temps réel. Le circuit RF doit être calibré manuellement. Pour que le simulateur devienne un simulateur GNSS, il faudrait ajouter une chaîne d'émission RF pour chaque signal de chaque fréquence émise. Aussi, du côté du logiciel, il devient nécessaire d'insérer une classe spécifique à la constellation à simuler et

celle-ci s'occupera de calculer convenablement les paramètres de navigation et de produire le message binaire de navigation.

Le simulateur d'Agilent simule seulement les constellations GPS et GLONASS avec 32 canaux. Il est composé essentiellement de trois parties : le logiciel Signal Studio, le EXB et le PXB. Le premier configure un scénario et s'exécute sur un ordinateur personnel (PC). Le second reçoit les données du scénario par le port Local Area Network (LAN) ou General Purpose Interface Bus (GPIB) pour générer les signaux de navigation (Agilent Technologies, 2012b). Un composant optionnel PXB est utilisé si on désire produire un signal en bande de base. Dans cette configuration, il n'est pas nécessaire d'avoir un PC, puisque la simulation se configure par le EXB. Signal Studio offre la possibilité de lire un fichier kml, trajectoire générée par Google Earth, pour les simulations en dynamique. Aussi, il enregistre les paramètres de navigations et il simule les multitrajets. Il est possible de choisir le modèle de délais atmosphériques ainsi qu'exercer un contrôle sur les C/N_0 s des satellites. À tous les deux ans, il est essentiel de recalibrer les appareils pour de meilleures performances (Agilent Technologies, 2012a).

Le générateur de signaux de navigation SMBV100A de la compagnie Rohde & Schwarz est un produit similaire à celui d'Agilent. Par contre, il supporte la constellation Galiléo ce qui n'est pas encore le cas pour son compétiteur. Le simulateur GNSS SMBV100A modélise les délais atmosphériques et les multitrajets. Les C/N_0 s des satellites peuvent être configurés individuellement de sorte que la puissance des signaux reçue par le récepteur soit différente pour chaque satellite. La fonctionnalité d'enregistrement des mesures du SMBV100A est plus difficile à utiliser comparativement aux autres simulateurs disponibles sur le marché. En effet, il est nécessaire d'écrire un script pour l'activer et spécifier les éléments à enregistrer. Ceci est moins pratique que si cette fonction était intégrée dans une interface usager (GUI). La calibration doit être faite par un professionnel de Rohde & Schwarz à tous les trois ans. Le composant facultatif SMBV-B10 offre la possibilité de recevoir les signaux de navigation en bande de base (Rohde & Schwarz, 2012).

Le simulateur GNSS Spectracom GSG-62 simule seulement les signaux GPS et GLONASS. Aussi, les systèmes d'augmentations simulés sont le WAAS, l'European Geostationary Navigation Overlay System (EGNOS), le GPS Aided Geo-Augmented Navigation (GAGAN) et le Multi-functional Satellite Augmentation System (MSAS). La génération des trajectoires s'effectue avec Google Maps ou à partir d'un fichier NMEA. Il est possible de configurer des multitrajets et ajouter du bruit au C/N_0 de chaque satellite qui est paramétrable dans la simulation. Le logiciel StudioView n'a pas de fonction d'enregistrement des mesures en format texte. Par contre, celui-ci enregistre l'exécution d'un scénario en un fichier NMEA (Spectracom, 2012). La calibration se fait directement avec le logiciel StudioView. Cette opération peut être effectuée à tous les deux ans par l'utilisateur lui-même, mais il est nécessaire de posséder un analyseur de spectre. Les délais atmosphériques sont simulés et un module logiciel du GSG-62 donne la possibilité de contrôler le simulateur à distance avec un PC (Spectracom, 2010).

Le Real Time GNSS Generator (RTG2) de l'entreprise Silicom contient 120 canaux des constellations GPS, GLONASS, Galiléo, COMPASS et quelques systèmes d'augmentations dont : EGNOS, QZSS et WASS. Les délais atmosphériques et les multitrajets sont modélisés. Dans une simulation, il est possible de simuler un brouilleur par une onde continue ou une impulsion. La configuration d'un scénario s'effectue sur un PC avec un fureteur. Les paramètres de la simulation permettent de contrôler individuellement les C/N_0 s des satellites et le message de navigation. Les données des satellites et du véhicule simulées peuvent être enregistrées. Pour l'entretien du RTG2, il est conseillé de calibrer la RF annuellement (Silicom, 2012).

Le Tableau 1.3 fait une récapitulation des paramètres et fonctionnalités d'un simulateur GNSS idéal.

Tableau 1.3 Récapitulation des paramètres et flexibilités des simulateurs

Compagnie / # modèle	Paramètres et flexibilités
Aeroflex / CPSG-1000	<ul style="list-style-type: none"> • Portatif, conçu pour l'aviation • Aucune fonctionnalité d'enregistrement des données et de contrôle du message de navigation • Calibration annuelle requise • Simulation des délais atmosphériques • Contrôle du C/N_0 de chaque satellite • Modélisation des multitrajets et des atténuations des signaux
Agilent / N5172B avec N5106A	<ul style="list-style-type: none"> • Génération des trajectoires avec Google Earth • Simulation des multitrajets • Modélisation de l'ionosphère et de la troposphère • Contrôle des C/N_0s • Calibration RF requise • Fonctionnalité d'enregistrement des données • Peut générer les signaux en bande de base
Cast Navigation / Cast-2000	<ul style="list-style-type: none"> • Génération de multitrajets • Analyse les performances antibrouillage • Simulation de l'effet de la troposphère et de l'ionosphère • Fonctionnalité d'enregistrement des mesures du simulateur • Contrôle du message de navigation et des C/N_0s • Aucune calibration RF manuelle
IFEN / NavX-NCS pro	<ul style="list-style-type: none"> • Générateur de bruit • Simulation des multitrajets • Modélisation de l'ionosphère et de la troposphère • Contrôle du message de navigation et des C/N_0s • Calibration RF requise • Fonctionnalité d'enregistrement des données
LACIME	<ul style="list-style-type: none"> • La RF doit être calibrée manuellement • Compatibilité avec Google Earth pour l'affichage des trajectoires • Affichage en temps réels des paramètres de navigation des satellites et du véhicule • Aucun enregistrement des données, de contrôle du message de navigation et des C/N_0s des satellites

Compagnie / # modèle	Paramètres et flexibilités
Racelogic / Labsat 2	<ul style="list-style-type: none"> • Enregistre et rejoue les signaux réels de navigation • Génération des trajectoires avec Google Earth ou d'un fichier NMEA • Envoie le message de navigation sur les ports RS-232/422/485
Rohde & Schwarz / SMBV100A avec SMBV-B10	<ul style="list-style-type: none"> • Simulation des multitrajets • Modélisation de l'ionosphère et de la troposphère • Contrôle des C/N₀s • Calibration RF requise • Fonctionnalité d'enregistrement des données • Peut générer les signaux en bande de base
Silicom / RTG2	<ul style="list-style-type: none"> • Simulation des délais atmosphériques • Contrôle du C/N₀ de chaque satellite • Contrôle du message de navigation • Calibration RF requise • Simulation des multitrajets • Enregistrement des mesures • Possibilité de brouiller les signaux de navigation
Spectracom / GSG-62	<ul style="list-style-type: none"> • Compatible avec Google Maps pour la génération des trajectoires • Simulation des délais atmosphériques • Contrôle du C/N₀ de chaque satellite • Calibration RF requise • Simulation des multitrajets • Enregistre l'exécution d'un scénario en format NMEA • Générateur de bruit
Spirent / GSS7700	<ul style="list-style-type: none"> • Contrôle C/N₀ de chaque satellite et du message de navigation • Fonctionnalité d'enregistrement des données • Possibilité de connecter d'autres modules au simulateur comme un générateur d'interférence • Compatibilité pour simuler la navigation inertielle • La RF doit être calibrée manuellement • Modélisation de l'ionosphère et de la troposphère

1.6 Contributions générales

La principale contribution de ce mémoire est le déverminage de la constellation GPS L1 du simulateur LACIME. Ainsi, une simulation stable pour une durée de 24 heures sans perte de position du récepteur GPS avec une moyenne de l'erreur de position inférieure à 1 mètre sera obtenue. Ceci a été rendu possible notamment en corrigeant les valeurs des paramètres de navigation des satellites (position, vitesse, élévation, pseudo-distance, etc.) ainsi que leur message binaire de navigation. Le calcul des phases de la porteuse, du code et du message de navigation calculés par le logiciel et générés par le FPGA ont été rectifiés. Le bit courant du code et du message de navigation ont été corrigés pour que les satellites simulés soient correctement synchronisés entre eux. Une solution temporaire a été proposée pour la génération de l'horloge principale du système.

1.7 Conclusion

Dans ce chapitre, il a été question de décrire le contexte du projet de recherche ainsi que ses objectifs et la méthodologie de recherche qui sera adoptées. Les travaux qui seront effectués permettront d'avoir des résultats répétables, stables et robustes dans son ensemble. Pour atteindre ces aspects, il faut se familiariser avec le simulateur LACIME, de valider et de corriger chaque sous-système. Un simulateur commercial Spirent sera utilisé pour confirmer la validité des résultats obtenus. Les retombées économiques d'un simulateur GNSS sont nombreuses. Le secteur des transports en bénéficiera avec des récepteurs plus précis et plus fiables. Les compagnies de conception d'appareil de navigation par satellite reproduiront plus facilement des scénarios dans un environnement contrôlé facilitant grandement la vérification du bon fonctionnement du produit. Une liste de différents simulateurs GPS/GNSS disponibles sur le marché a été établie ainsi que certaines de leurs fonctionnalités principales. Celles-ci ont été résumées dans le Tableau 1.4.

Tableau 1.4 Résumé des fonctionnalités essentielles supportées par les simulateurs
GPS/GNSS

Simulateurs GPS/GNSS	Contrôle du C/N0	Enregistrement des mesures	Contrôle du message de navigation	Modélisation des délais atmosphériques	Simulation des multitrajets	Générateur de bruit
Cast-2000	X	X	X	X	X	
CPSG-1000	X		X	X	X	
GSG-62	X			X	X	X
GSS7700	X	X	X	X		
Labsat 2						
LACIME						
N5172B	X	X		X	X	
NavX-NCS pro	X	X	X	X	X	X
RTG2	X	X	X	X	X	
SMBV100A	X			X	X	

Pour que le simulateur LACIME soit un simulateur GNSS, quelques ajouts logiciels et matériels sont nécessaires. En effet, il faut que le logiciel génère le message binaire de navigation et calcule les paramètres de navigation pour toutes les constellations à simuler. En plus, une nouvelle chaîne d'émission pour chaque signal devra être faite à la carte RF.

Les objectifs de recherche étant connus, le chapitre suivant sera une revue de la littérature du simulateur LACIME. Ces performances initiales seront étudiées plus en détail.

CHAPITRE 2

DESCRIPTION DE L'ARCHITECTURE INITIALE DU SIMULATEUR LACIME

Dans ce chapitre, une description de l'architecture du simulateur GPS L1 existante sera le cœur des discussions, car elle sera utilisée pour la suite des travaux de recherche. Ensuite, les limitations du simulateur seront énumérées ainsi que les problèmes rencontrés. Les performances initiales seront présentées pour démontrer l'état actuel du simulateur.

2.1 Description de l'architecture du simulateur LACIME existante

2.1.1 Fonctionnement global et banc de test du simulateur initial

Le système se compose de trois parties : logiciel, IF et RF. La configuration d'un scénario de simulation et les différents engins de calculs (pseudo-distance, trajectoire, etc.) s'effectuent par logiciel à l'aide d'un ordinateur. Il y a deux composantes électroniques, intégrables dans un ordinateur embarqué, qui sont nécessaires pour le bon fonctionnement du simulateur : une carte de développement FPGA de la compagnie Lyrtech modèle VHS-ADC/DAC et une carte RF développée au LACIME. La première contient un FPGA Virtex IV permettant de coder l'information envoyée par le logiciel et de produire un signal IF modulé. La seconde a été conçue à l'École de Technologie Supérieure (ÉTS) et convertit le signal IF en un signal RF. Comme la puissance du signal à sa sortie est trop élevée comparativement aux vrais signaux GPS, on doit l'atténuer de 50 dB à l'aide d'un atténuateur programmable externe Agilent 11713A.

Le récepteur EVK-6T de la compagnie u-blox permettra de valider le fonctionnement du simulateur. Son port Universal Serial Bus (USB) donne la possibilité d'échanger de l'information à un PC pour afficher les résultats. La Figure 2.1 illustre le banc de test du projet de recherche.

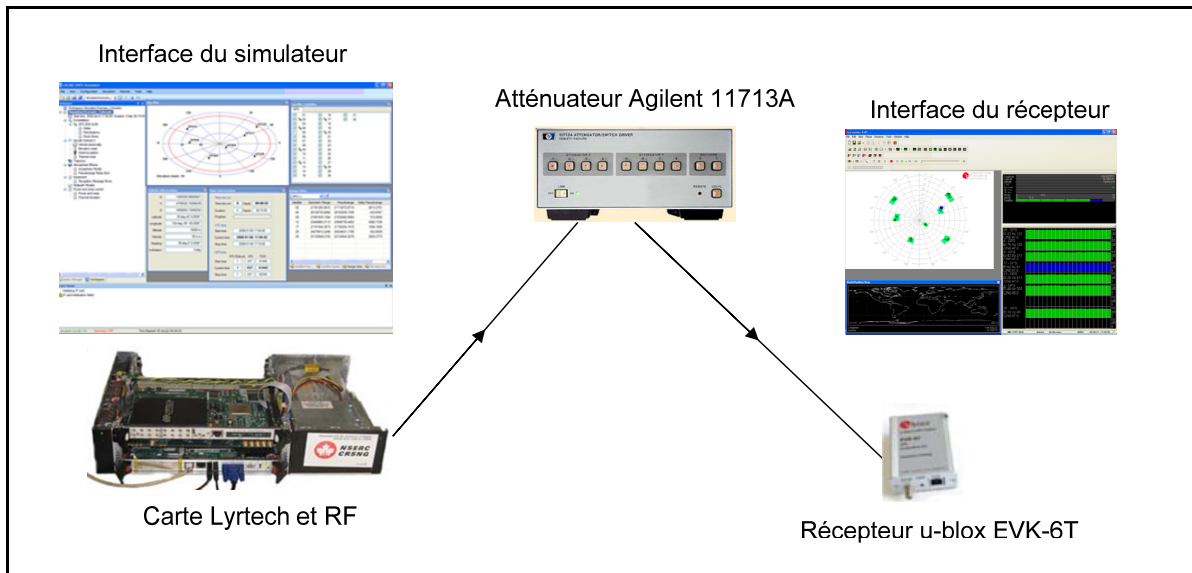


Figure 2.1 Montage actuel du banc de test du simulateur LACIME

2.1.2 Description de la partie logicielle associée au simulateur et à son GUI

L'architecture logicielle est divisée en deux sous-sections: le simulateur proprement dit et le GUI. Ce dernier a été développé par un autre membre de l'équipe et il ne sera pas couvert en profondeur dans ce document. La section simulation sera très importante pour les travaux de recherche et elle consiste à calculer les paramètres de la simulation (azimuth, pseudo-distance, élévation, fréquence Doppler etc). Ceux-ci sont générés à partir de la date et de l'heure de la simulation et d'un fichier almanach pour la position des satellites. Les messages binaires de navigation sont produits à l'aide des paramètres de la simulation et ils sont transmis directement au FPGA à l'aide du bus Front Side Bus (FSB) (voir Figure 2.8). La partie simulation se compose de deux processus principaux qui s'exécutent en parallèle. Le premier calcule à toutes les secondes les paramètres de la simulation. Le deuxième envoie le message de navigation au FPGA à toutes les six secondes et à toutes les secondes, il transmet la fréquence porteuse, la fréquence du code, le gain logiciel et la phase de la porteuse et du code par le biais de registres (voir Figure 2.8). Ceux-ci servent à piloter les circuits du FPGA pour qu'ils génèrent le signal IF approprié. Les morphogrammes des processus #1 et #2 sont représentés aux Figures 2.2 et 2.3.

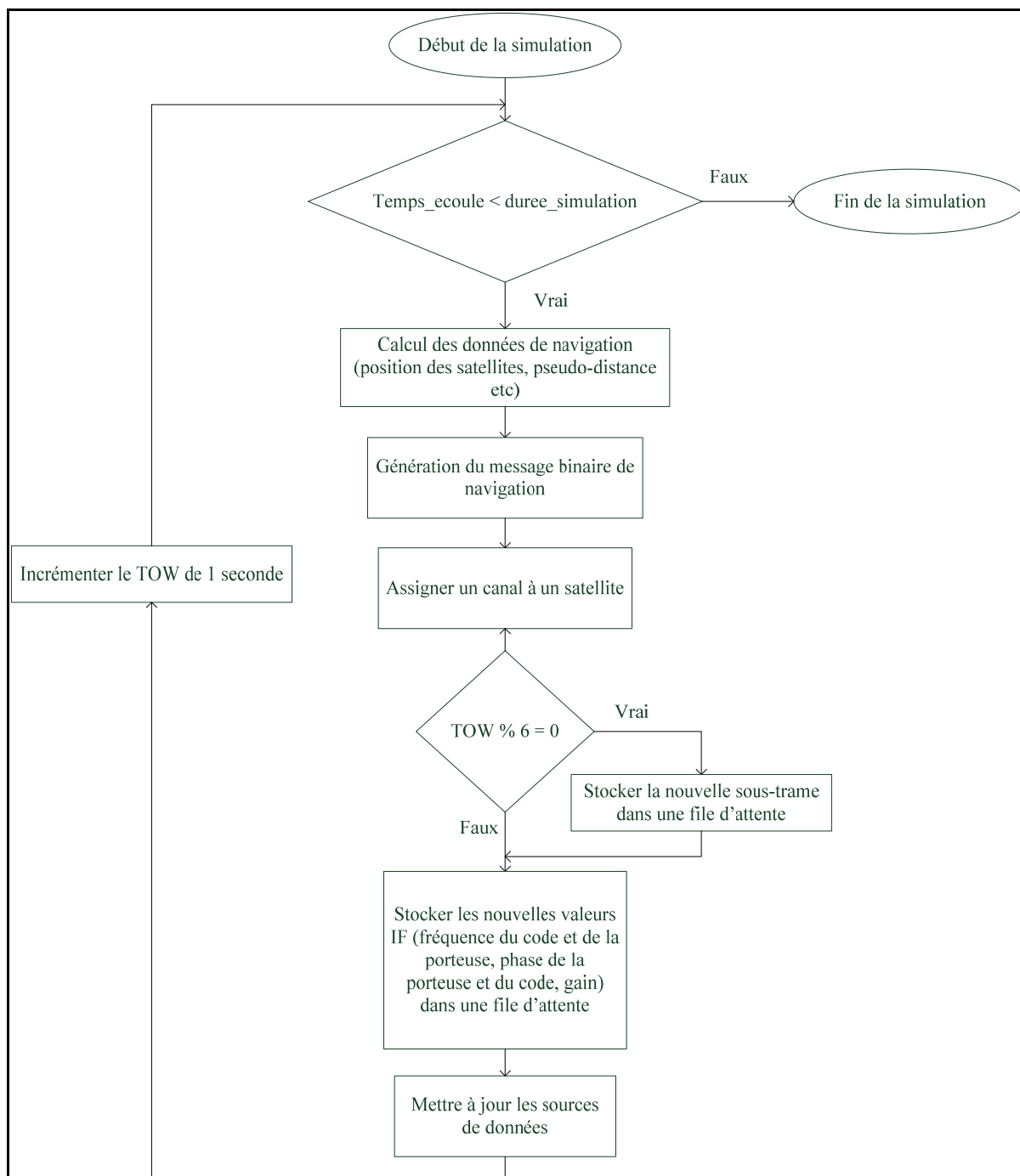


Figure 2.2 Morphogramme du processus #1 du simulateur actuel

La boucle du processus #1 s'exécute une fois à toutes les secondes. Après vérification, il n'y a pas de problème avec sa structure.



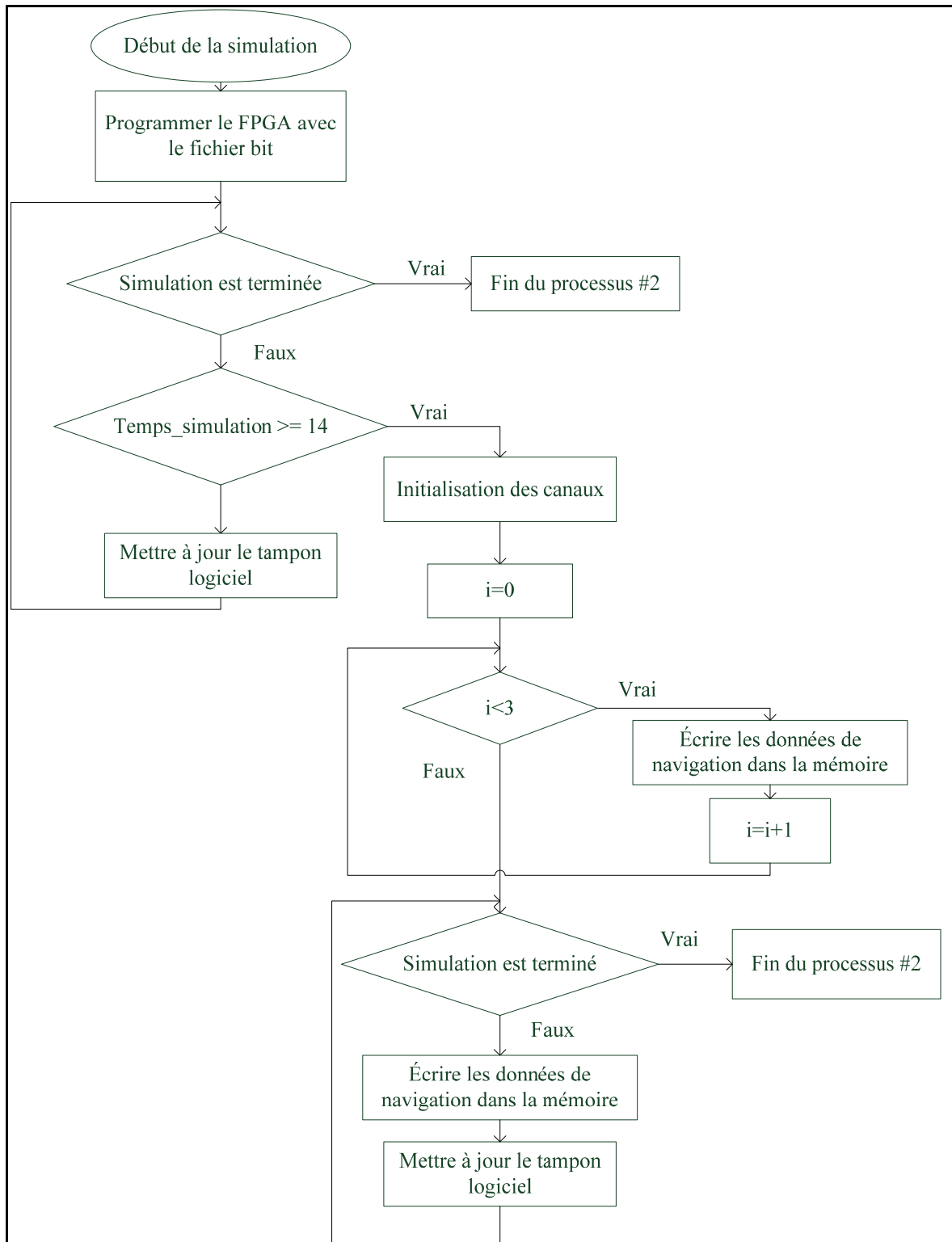


Figure 2.3 Morphogramme du processus #2 du simulateur actuel

Selon la Figure 2.3, il n'y a pas de données qui sont écrites dans les mémoires FSB et eXternal MEMory (XMEM) durant les 13 premières secondes de la simulation. Ce délai est un peu long et des investigations sur cet aspect sont nécessaires. Aussi, après l'initialisation des canaux, il n'y a pas de raison pour laquelle la fonction permettant d'écrire dans les mémoires les données de navigation soit appelée trois fois consécutives. Donc, les problèmes énumérés précédemment sur le processus #2 devront être corrigés.

Pour bénéficier de tous les avantages de la programmation orientée objet dont la réutilisation des classes, chaque module regroupe différentes classes ayant des fonctionnalités semblables. Le processus #1 appartient au module "Simulator Engine" puisqu'il est le processus primaire de la partie simulation du simulateur. Le processus #2 s'occupe de la communication IF et il a été implémenté dans le module "IF/RF interface". L'architecture logicielle du simulateur est représentée à la Figure 2.4.

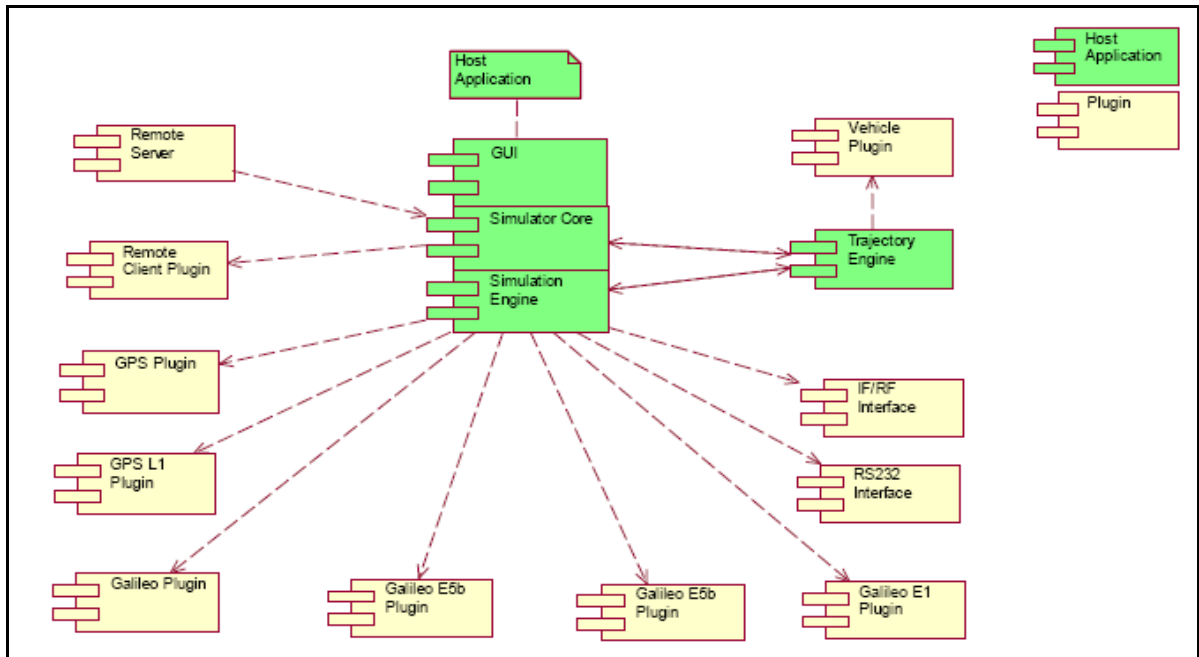


Figure 2.4 Architecture logicielle du simulateur
Tirée de Ben Salah (2010, p. 70)

Les modules en vert représentent des classes de base dont les modules en jaune peuvent dériver. Cela représente une relation d'héritage. Une classe de base a la possibilité de dériver indéfiniment ce qui simplifie la tâche lorsque vient le temps de faire des modifications au programme. Il y a seulement la constellation GPS L1 qui est implémentée. Les autres seront programmés beaucoup plus tard au cours du processus de développement.

2.1.3 Description des aspects matériels reliés au simulateur

Les parties IF et RF ont comme rôle principal de produire les signaux réels de navigation qui seront reçus, démodulés et décodés par un récepteur GPS. Cette section sera divisée en deux parties : la génération des signaux IF et RF.

2.1.3.1 Description de l'architecture de la section IF

La carte Lyrtech est une composante très importante du simulateur LACIME, car elle permet de générer le signal GPS en IF et d'introduire les délais aux codes et aux messages de navigation. Avec les données de navigation, le FPGA produira un signal en fréquence intermédiaire qui sera transposé en RF. Chaque satellite possède son propre module VHDL, appelé canal, permettant de générer les signaux en phases (I) et en quadratures (Q). Pour faciliter les tests, le nombre de canaux a été limité à 8, mais les ressources du FPGA permettraient d'en implémenter près de 16 (Lacasse, 2009).

Un convertisseur numérique/analogique, extérieur à la carte Lyrtech, reçoit une porteuse IF centrée à 30 MHz et le transpose à la fréquence de 70 MHz pour l'envoyer à l'entrée de la carte RF. Un schéma bloc des fonctionnalités de haut niveau du FPGA générant le signal IF du simulateur est illustré à la Figure 2.5.

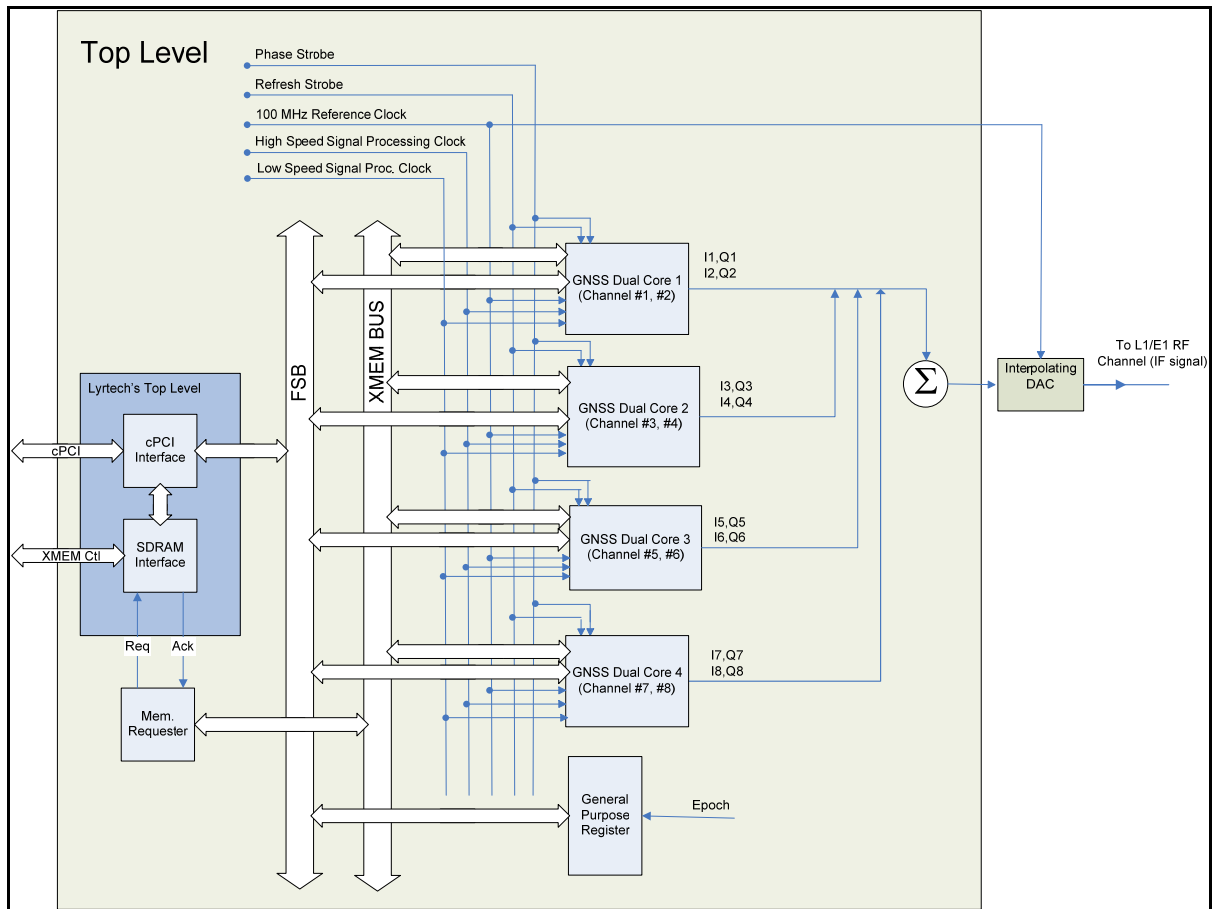


Figure 2.5 Fonctionnalités de haut niveau du Virtex IV

Chaque module « GNSS Dual Core » est composé de deux canaux et une table de vérité de sinus et de cosinus pour calculer I et Q. La Figure 2.6 représente la structure d'un canal d'un satellite.

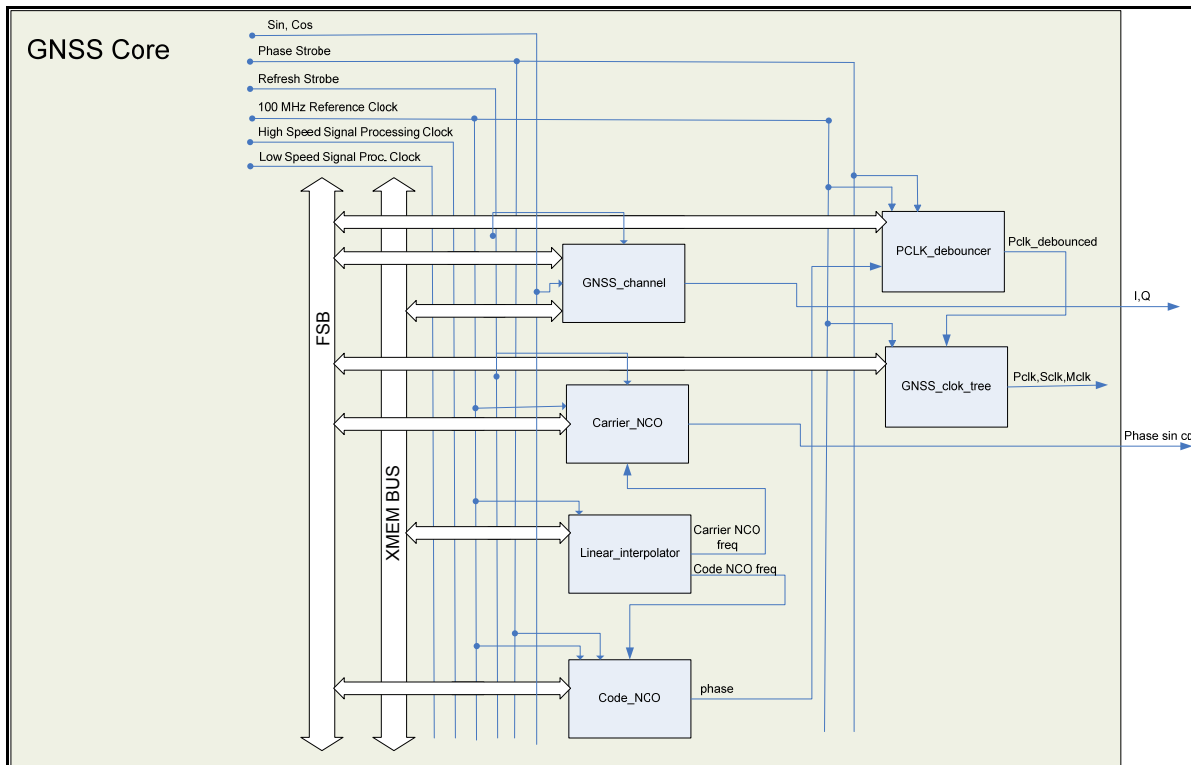


Figure 2.6 Structure d'un canal d'un satellite

Les modules NCO permettent de contrôler numériquement la phase de la porteuse et du code. À chaque 10 ns, la phase du code et de la porteuse (sortie des NCOs) est incrémentée suivant leur fréquence respective. Le module PCLK_debouncer reçoit le bit le plus significatif de la phase du code et l'inverse à la sortie. Ainsi, lorsque GNSS_Clock_tree détecte un front montant sur le signal PCLK_debouncer (débordement de la phase du code), il y a un changement de bit du code Pseudo-Random Noise (PRN) et une incrémentation de la phase du message de navigation. On remplace le bit courant du message de navigation lorsque sa phase est à 20460 (20 ms). L'interpolateur linéaire a été conçu initialement pour interpoler d'un facteur de 10 000 entre chaque fréquence sortie des tampons du code et de la porteuse dans le but de sauver de l'espace mémoire (Lacasse, 2009). En d'autres termes, si la mise à jour s'effectue à toutes les secondes, l'interpolation permettra d'obtenir 10 000 échantillons par seconde. Présentement, ce module ajoute un délai d'horloge pour la fréquence du code et de la porteuse.

La Figure 2.7 représente le schéma bloc du modulateur I et Q d'un canal GNSS et il se situe dans chaque module « GNSS_channel » de la Figure 2.6.

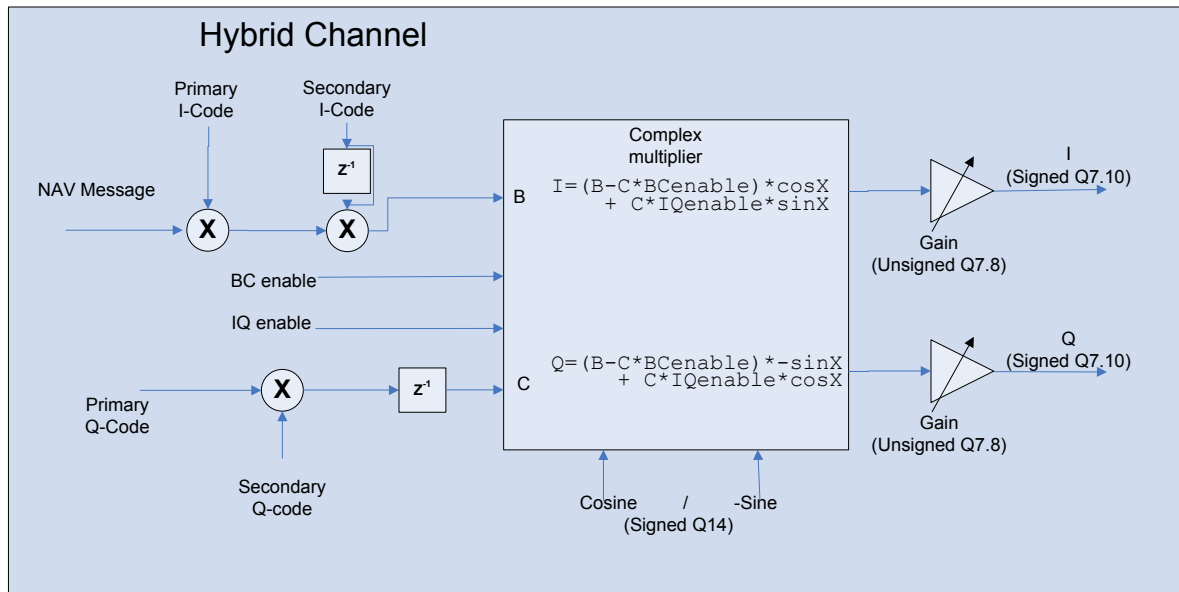


Figure 2.7 Schéma bloc d'un canal GNSS
Tirée de Lacasse (2008, p. 21)

Le canal GNSS peut générer indifféremment les signaux GPS et Galiléo. Par contre, le module ne permet pas encore la génération des signaux Galiléo, seul le signal GPS L1 est produit. Donc, les codes secondaires que nous voyons à la Figure 2.7 ne sont pas utilisés. Les sorties du bloc (I et Q) représentent le signal IF de navigation et ils sont envoyés au convertisseur numérique/analogique afin de produire le signal RF. Le message de navigation arrive à une fréquence de 50 Hz et le code primaire à 1.023 MHz. À partir de la phase du signal, on récupère la valeur du sinus et du cosinus pour calculer I et Q dans une table de vérité partagée par les deux canaux.

2.1.3.2 Description des interfaces des mémoires du simulateur LACIME

La Figure 2.8 montre les interfaces FSB et XMEM du simulateur LACIME.

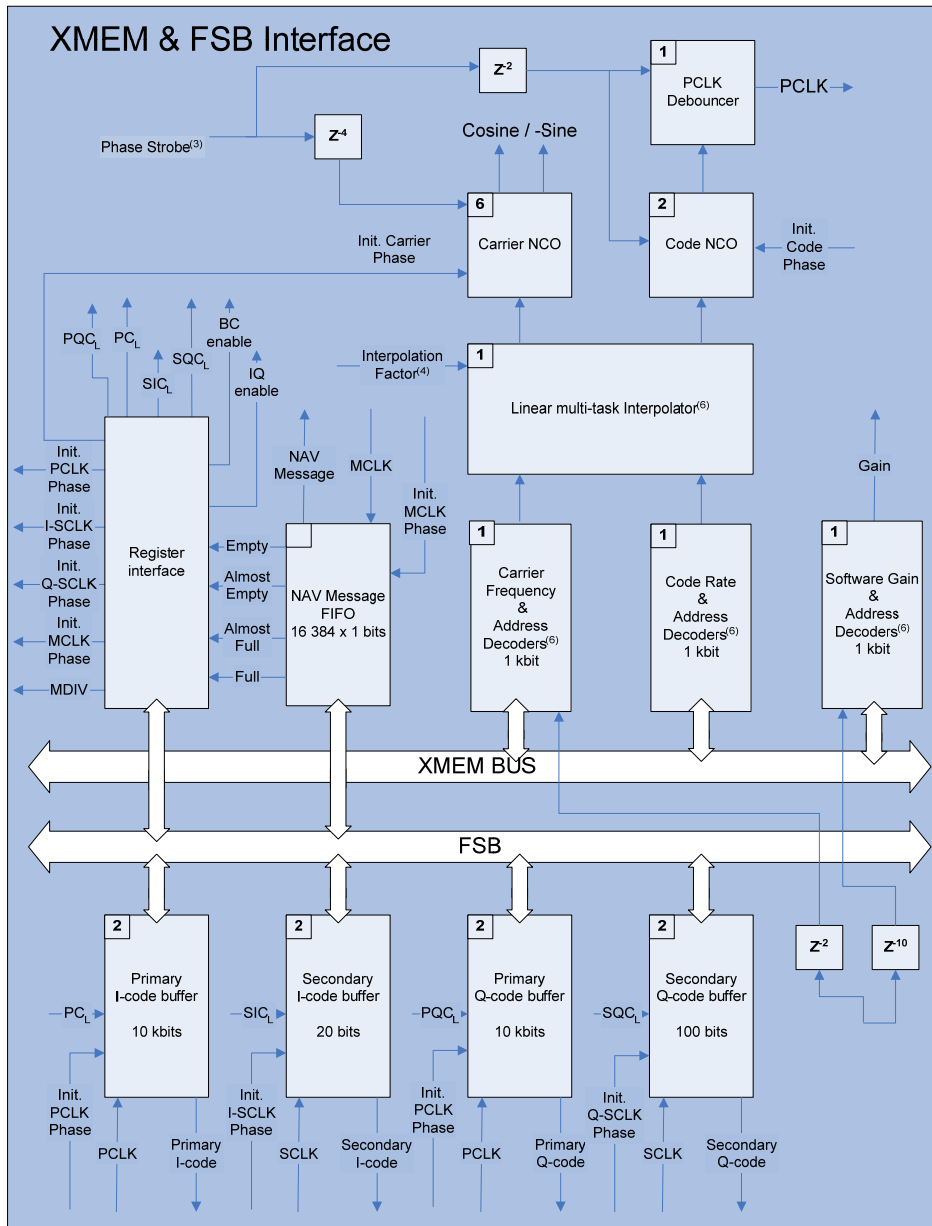


Figure 2.8 Interface XMEM et FSB de la section IF
Adaptée de Lacasse (2008, p.20)

La mémoire FSB contient dans une Random Access Memory (RAM) le message de navigation de tous les satellites. Elle comprend une file First In First Out (FIFO) de 16 384 bits pour chaque canal ainsi que les valeurs des registres pour contrôler et surveiller les différents canaux GNSS.

La mémoire XMEM enregistre la fréquence de la porteuse, la fréquence du code et le gain logiciel. Ce dernier est constant à 5461 pour tous les satellites. En modifiant sa valeur, les C/N₀s des satellites simulés changeraient. Alors, le gain logiciel pourrait être employé pour faire varier la puissance des signaux de navigation de chaque satellite. Le bus d'adressage de la mémoire XMEM est illustré à la Figure 2.9 :

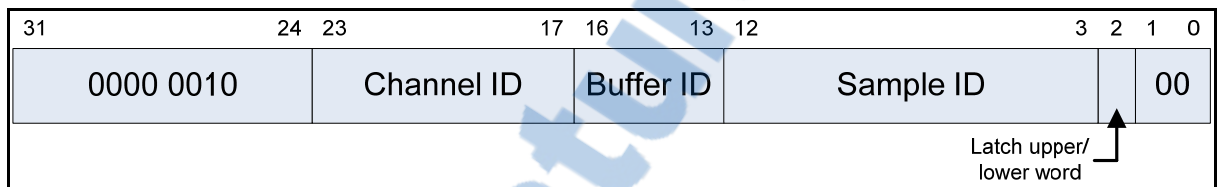


Figure 2.9 Adresse de la mémoire XMEM
Tirée de Sauriol (2007, p. 2)

où :

- Channel ID (Bit 23 :17) est le numéro du canal,
- Buffer ID (Bit 16 :13) est le numéro du tampon (1 pour la fréquence porteuse, 2 pour la fréquence du code et 3 pour le gain logiciel),
- Sample ID (Bit 12 :3) est le numéro d'échantillon dont il est synchronisé avec un compteur de 10 ms,
- Latch upper/lower word (Bit 2) est à 1 si la donnée à mémoriser est représenté sur 64 bits sinon 0 pour 32 bits .

Les 8 bits les plus significatifs de l'adresse sont constants afin de garder une compatibilité avec les autres projets en cours au LACIME (Sauriol, 2007). Il y a au total 4 194 303 adresses disponibles et chacune d'entre elles permet d'enregistrer un mot de 32 bits. Donc, deux cases mémoires sont nécessaires pour une donnée de 64 bits.

2.1.3.3 Description de l'architecture de la section RF

La carte RF récupère les signaux IF à la sortie du DAC pour les transposer vers la fréquence réelle de la constellation GPS L1. Les pseudo-distances sont évaluées par le récepteur selon la différence de temps entre l'horloge du récepteur et celle des satellites. Donc, il devient nécessaire pour le simulateur de posséder une horloge principale de bonne précision. Une horloge stable de 10^{-9} (erreur de $\pm 0,1$ Hz pour une fréquence de 100 MHz) est suffisante pour les besoins du projet. L'horloge principale du simulateur se retrouve dans la carte RF du LACIME et elle est utilisée pour la section IF. Il faut noter que la carte RF n'a pas été calibrée ce qui affecte la valeur du C/N_0 des satellites.

Les Figures 2.10 et 2.11 montrent respectivement la carte RF du simulateur LACIME et son architecture initiale.

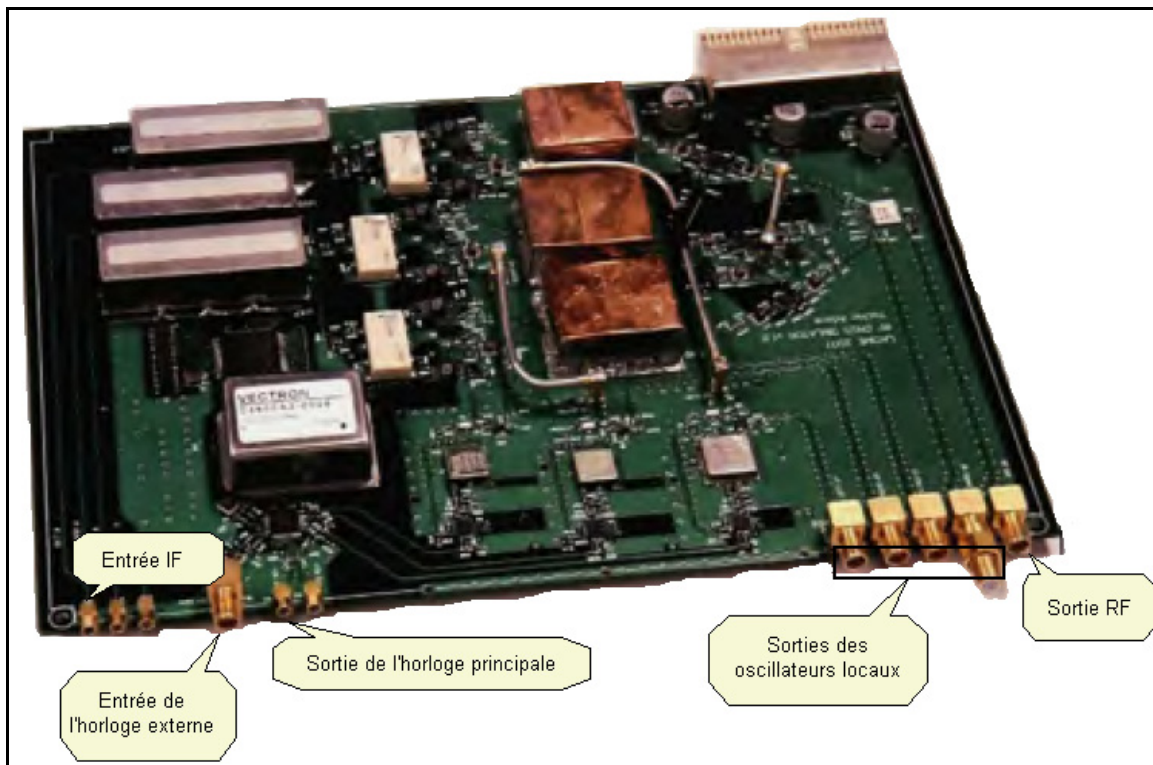


Figure 2.10 Carte RF du simulateur LACIME

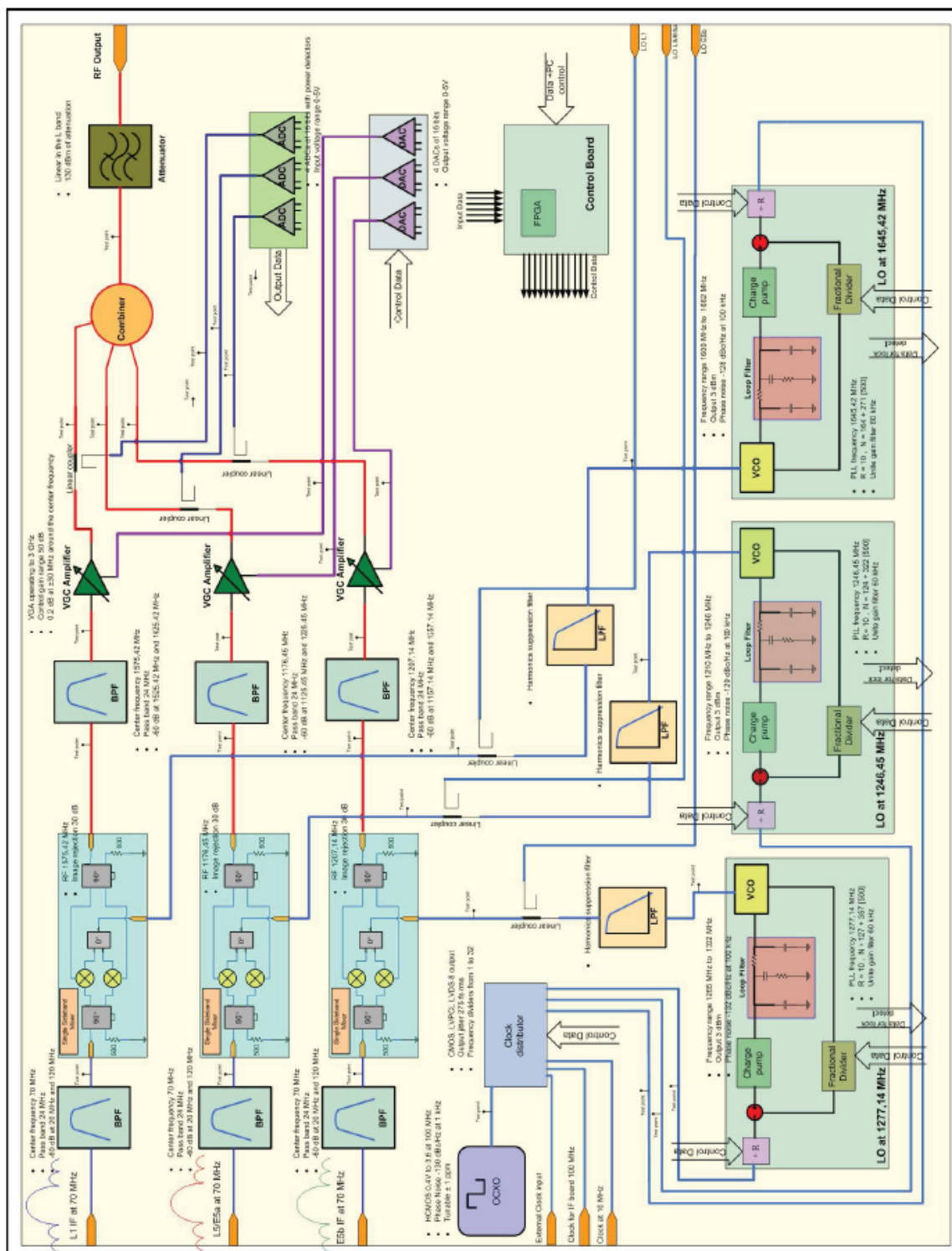


Figure 2.11 Architecture initiale de la partie RF du simulateur LACIME
Adaptée de Adane (2007b, p.2)

La carte RF translate la fréquence des signaux IF centrés à 70 MHz pour les transposer aux fréquences réelles. Elle se compose principalement de trois parties :

1. La génération de l'horloge principale : c'est une horloge Oven Controlled X-tal (Crystal) Oscillator (OCXO) de 100 MHz de haute précision. Le FPGA Virtex IV contient une horloge de 100 MHz, mais celle-ci n'est pas assez de bonne qualité que celle des satellites réels. Donc, le composant OCXO est essentiel pour atteindre une précision similaire aux horloges des satellites qui est de 10^{-12} . Il n'est pas possible avec l'OCXO d'obtenir une onde exacte de 100 MHz (Adane, 2007a). Alors, un Complex Programmable Logic Device (CPLD) configure un potentiomètre MAX5481 pour avoir la bonne tension de raffinement de la précision de la fréquence centrale de l'OCXO. Aussi, le CPLD a pour fonction de distribuer l'horloge interne de la carte RF ou celle fournie par l'utilisateur vers les boucles à verrouillage de phase et la partie IF du simulateur LACIME.
2. Les boucles à verrouillage de phase (PLL) : il y a trois PLLs générant chacune un oscillateur local pour couvrir les fréquences d'émission L1, L5, E5a et E5b. Elles sont composées principalement de trois parties :
 - un synthétiseur de fréquence fractionnelle ADF4153 d'Analog Device qui produit la fréquence désirée de l'oscillateur;
 - un filtre de boucle pour filtrer le bruit de phase;
 - un VCO (Voltage Controlled Oscillator) qui cadence la PLL à partir d'une tension de commande envoyée par le filtre de boucle.
3. Le système de contrôle de puissance : celui-ci contrôle le ratio signal sur bruit (C/N_0) des satellites. Il comprend un Voltage Controlled Amplifier/Attenuator (VGA) ADL5330 de la compagnie Analog Device. Avec une tension de commande configurable par un deuxième CPLD, ce composant fait varier la puissance des signaux à la sortie de la carte RF. Un GUI permet à l'utilisateur de spécifier la valeur du C/N_0 qui sera appliquée pour tous les satellites simulés.

La transposition du signal IF en RF s'effectue à l'aide d'un oscillateur et d'un mélangeur à bande latérale unique. La fréquence d'émission peut être calculée selon l'équation (2.1) (Adane, 2007b) :

$$f_{RF} = (f_{LO} - f_{IF}) \quad (2.1)$$

où :

- f_{LO} est la fréquence du signal à la sortie de la PLL,
- f_{IF} est la fréquence du signal IF centré à 70 MHz.

Sur la carte RF, il y a trois connecteurs (voir Figure 2.10) qui relient la sortie de chacun des oscillateurs locaux. Les Figures 2.12, 2.13 et 2.14 illustrent la forme et l'enveloppe de l'onde obtenue à la sortie des PLLs.

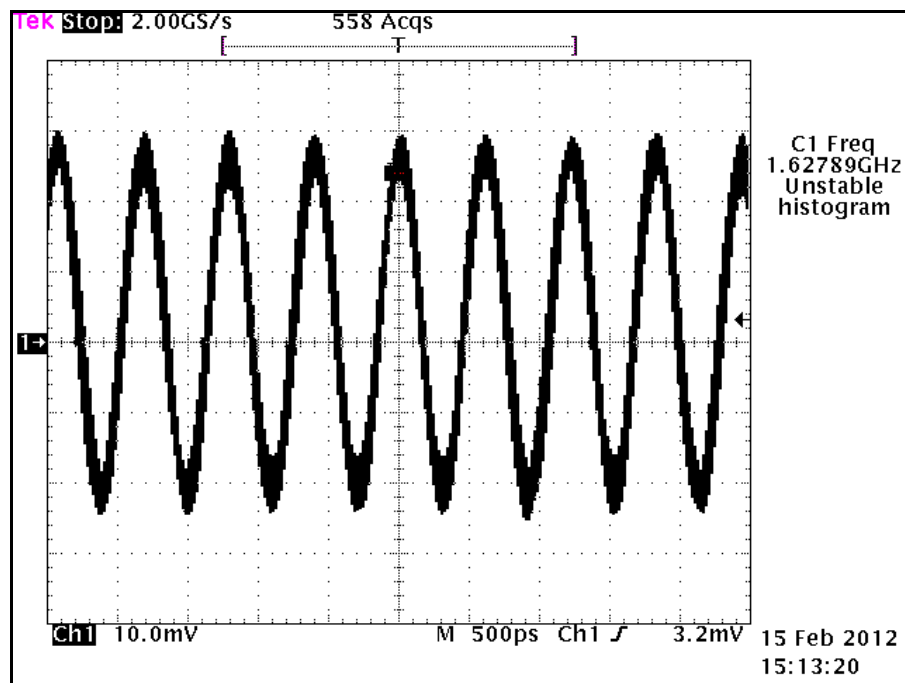


Figure 2.12 Forme d'onde à la sortie de la PLL pour le signal L1

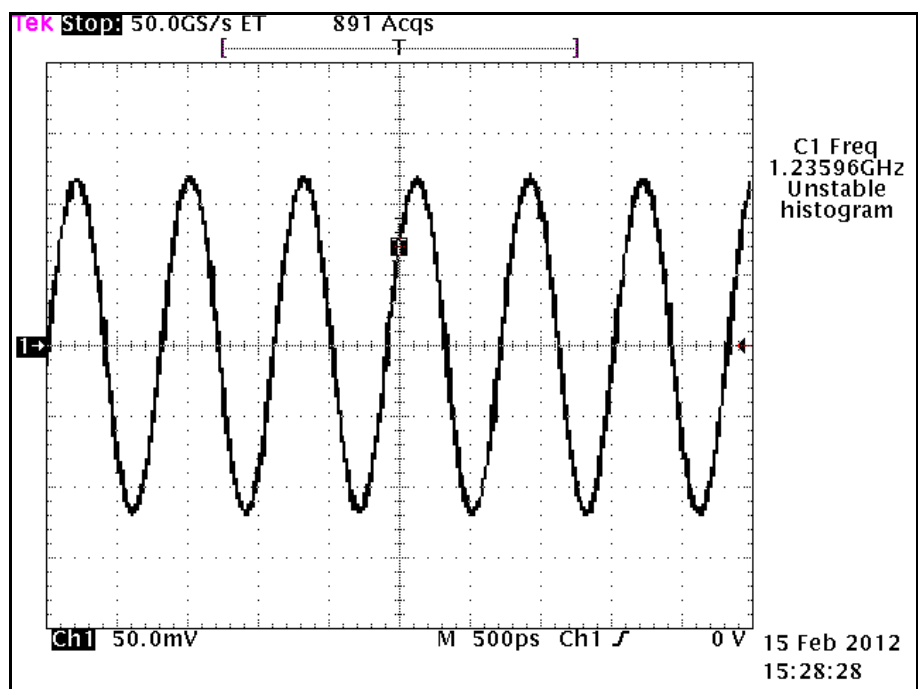


Figure 2.13 Forme d'onde obtenue à la sortie de la PLL pour les signaux L5 et E5a

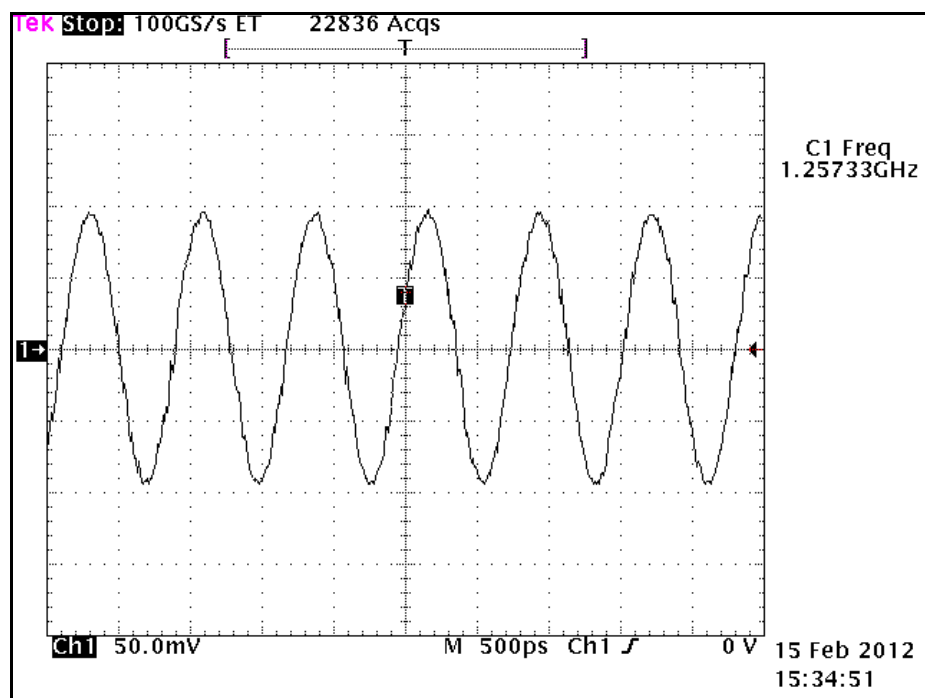


Figure 2.14 Forme d'onde obtenue à la sortie de la PLL pour le signal E5b

Le Tableau 2.1 représente les fréquences théoriques des oscillateurs pour les signaux de navigation implémentés initialement dans la carte RF (Loukil, 2010).

Tableau 2.1 Fréquences théoriques des oscillateurs de la carte RF

Signal	Fréquence d'émission (MHz)	Fréquence théorique de l'oscillateur (MHz)
L1	1575,42	1645,42
L5, E5a	1176,45	1246,45
E5b	1207,14	1277,14

Les sorties des PLLs des signaux L5, E5a et L1 contiennent beaucoup de bruit ce qui rend difficile pour l'oscilloscope de trouver la fréquence exacte de l'onde produite. Il est possible que ce problème survienne parce qu'il y a des connexions sur la carte RF qui n'ont pas été correctement effectuées ou bien les PLLs n'ont pas été correctement réalisées. Une sortie d'une PLL bruitée endommage la qualité des signaux de navigation ce qui complexifie pour le récepteur la recherche d'une position simulée et de trouver la bonne valeur de Doppler des satellites.

Le système de contrôle de puissance ne permet pas de commander correctement la valeur du C/N_0 . Plusieurs tests ont été effectués pour vérifier cette fonctionnalité et les résultats détaillés sont à l'ANNEXE I. Lorsque la configuration est dite fonctionnelle, le C/N_0 des satellites au récepteur ne concorde pas avec ceux affichés par le GUI. En plus, les C/N_0 des satellites affichés au récepteur peuvent avoir une différence entre elles jusqu'à une vingtaine de dBHz. Ce problème se répète peu importe la position simulée. Donc, le système de contrôle de puissance n'est pas fonctionnel.

2.1.4 Limitations de l'architecture actuelle et problèmes rencontrés

Il y a quelques limitations technologiques associées à l'architecture actuelle :

1. Le simulateur ne permet pas de simuler plus que 8 satellites en même temps dus à la saturation des ressources du FPGA. Une optimisation du circuit logique est essentielle sachant que d'autres canaux devront être simulés dans le futur;
2. La valeur du C/N_0 est identique pour chaque satellite ce qui n'est pas réaliste. En effet, la puissance de réception des signaux doit s'affaiblir en fonction de la distance qui sépare le satellite de l'utilisateur et des caractéristiques directionnelles de l'antenne;
3. La transition des satellites ne se fait pas selon les règles de l'art. En effet, parmi les satellites visibles du simulateur LACIME, ceux ayant les plus petits PRNs ont priorité sur les autres. Dans un monde réel, les satellites ayant la plus courte distance du récepteur devraient être simulés, car leur puissance de réception est meilleure. Aussi, une autre solution serait de simuler tous les satellites visibles ce qui demande beaucoup de ressources matérielles, car le simulateur doit avoir la capacité de simuler un grand nombre de satellites en même temps;
4. Il y a seulement un modèle de délais atmosphériques qui est implémenté et celui-ci n'a pas été validé ni testé correctement. Dans un simulateur commercial, il est possible de sélectionner un modèle particulier pour modéliser l'ionosphère et la troposphère;
5. Un module d'interférence logiciel et matériel ajouterait du réalisme dans une simulation. Il est très rare surtout en milieu urbain que les récepteurs soient en vue directe avec les satellites;
6. Le Position Dilution Of Precision (PDOP), Horizontal Dilution Of Precision (HDOP) et le Vertical Dilution Of Precision (VDOP) ne sont pas affichés dans le GUI du simulateur LACIME et ceux-ci déterminent la précision de la position simulée;
7. Une fonctionnalité d'enregistrement des données aiderait notamment pour déterminer l'application. Pourtant, cette fonction de base est déjà exploitée dans plusieurs simulateurs commerciaux comme le GSS7700 de la compagnie Spirent. Celui-ci enregistre les données du véhicule (position et la vitesse) et des satellites simulés (position, vitesse, fréquence Doppler et l'angle d'azimut et d'élévation);
8. Le manque de commentaires et de documentations surtout de la partie logiciel rend difficile la compréhension de certaines parties du code et demande davantage de temps pour analyser les problèmes.

Suite à une première étude du simulateur (phase de familiarisation), plusieurs problèmes ont été trouvés. Le Tableau 2.2 les résume et indique si ceux-ci ont été résolus avec les travaux effectués dans ce mémoire.

Tableau 2.2 Résumé des problèmes observés du simulateur

Descriptions des problèmes	Résolus
Perte de signal de tous les satellites	X
Test statique sur 24 heures n'est pas possible	X
Le Time To First Fix (TTFF) est trop long (11 minutes approximativement)	X
Lorsque le récepteur affiche une position, l'ajout d'un satellite fait bouger la solution au récepteur	X
Au début d'une simulation quelconque, il arrive que le récepteur détecte une quinzaine de satellites	X
Il arrive que la position des satellites ainsi que leur PRN ne correspondent pas à ce qui est affiché sur le logiciel et le récepteur u-blox	X
L'acquisition des signaux GPS pour une même simulation n'est pas constante	X
Dès que le récepteur obtient une position, une altitude négative (-1940 mètres) pour une simulation à Marseille est affichée par le récepteur	
La commande climb et descent ne fonctionne pas correctement	
L'algorithme de la génération des trajectoires n'a pas été validé	
Les C/N ₀ s du logiciel ne correspondent jamais avec celui qui est affiché par le récepteur. Ceci est valide pour une configuration avec et sans bruit	
Ajout d'une option au GUI qui permet d'activer et de désactiver le bruit	
Permettre au logiciel de démarrer plusieurs scénarios un à la suite de l'autre de manière automatisée	
Implémenter d'autres véhicules que l'avion	
L'effet de l'atmosphère, du patron de l'antenne et du Thermal noise n'a pas été observé	
Manque de documentation sur le logiciel et le code du FPGA	X

Une simulation en statique pendant 24 heures n'est pas possible. Toutefois, cette fonctionnalité est fondamentale à un simulateur. Après un certain temps (environ 30 minutes), le récepteur n'est plus capable de détecter les satellites simulés. Peu importe le scénario, après environ 5 minutes de simulation, on perd tous les signaux. Ce comportement

peut se reproduire même avant ce délai. Même si le récepteur possède 4 satellites, il n'est pas garanti qu'il puisse calculer une position en deux dimensions (2D) ou en trois dimensions (3D). Pour les tests dynamiques, le récepteur a encore plus de difficulté à trouver une position. Les algorithmes permettant de générer les trajectoires n'ont pas été validés minutieusement et certaines commandes seraient à revoir particulièrement la commande climb/descent. Avec un récepteur u-blox, il n'est pas possible de voir l'impact d'un multitrajet et de l'ajout de bruit par rapport à la valeur du C/N_0 des satellites.

2.2 Discussion sur les performances initiales du simulateur LACIME

Les résultats vus du récepteur u-blox EVK-6T d'une simulation d'une heure au Colorado avec le simulateur Spirent (Rx/Spirent) et LACIME (Rx/LACIME) sont présentés respectivement aux Figures 2.15 à 2.18. Les moyennes (μ) et les écarts-types (σ) des erreurs de position en x (μ_{errx} , σ_{errx}), y (μ_{erry} , σ_{erry}) et 2D (μ_{err} , σ_{err}) ainsi que les PDOPs (μ_{pdop} , σ_{pdop}), HDOPs (μ_{hdop} , σ_{hdop}) et VDOPs (μ_{vdop} , σ_{vdop}) ont été calculés. Le Tableau 2.3 résume les valeurs obtenues.

Le graphique de visibilité des satellites du simulateur LACIME (voir Figure 2.18) démontre que le satellite #28 n'est pas visible par le récepteur. Par contre, une simulation avec le simulateur Spirent indique le contraire : celui-ci est visible approximativement durant les 37 premières minutes. Il y a de nombreuses lignes bleues sur le graphique de la visibilité des satellites du simulateur LACIME. Ceci signifie que le récepteur éprouve de la difficulté à se synchroniser avec le signal. Ainsi, il reconnaît les PRNs des satellites simulés, mais il est dans l'impossibilité de décoder correctement les informations envoyées (Doppler, pseudo-distance etc.). À l'opposé, une ligne verte détermine que le satellite est utilisé pour calculer la position de l'utilisateur.

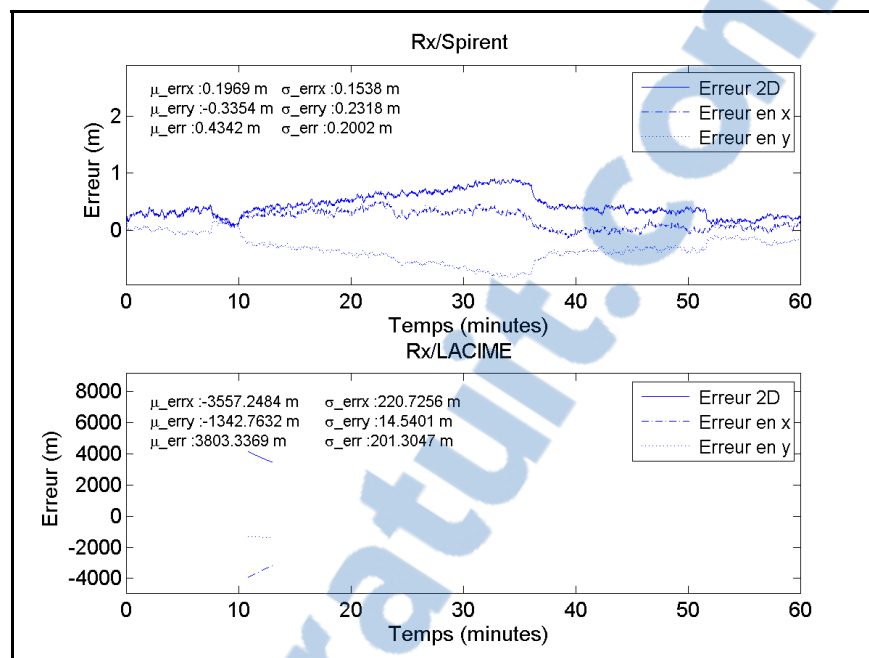


Figure 2.15 L'erreur de la position (Rx/Spirent versus Rx/LACIME)

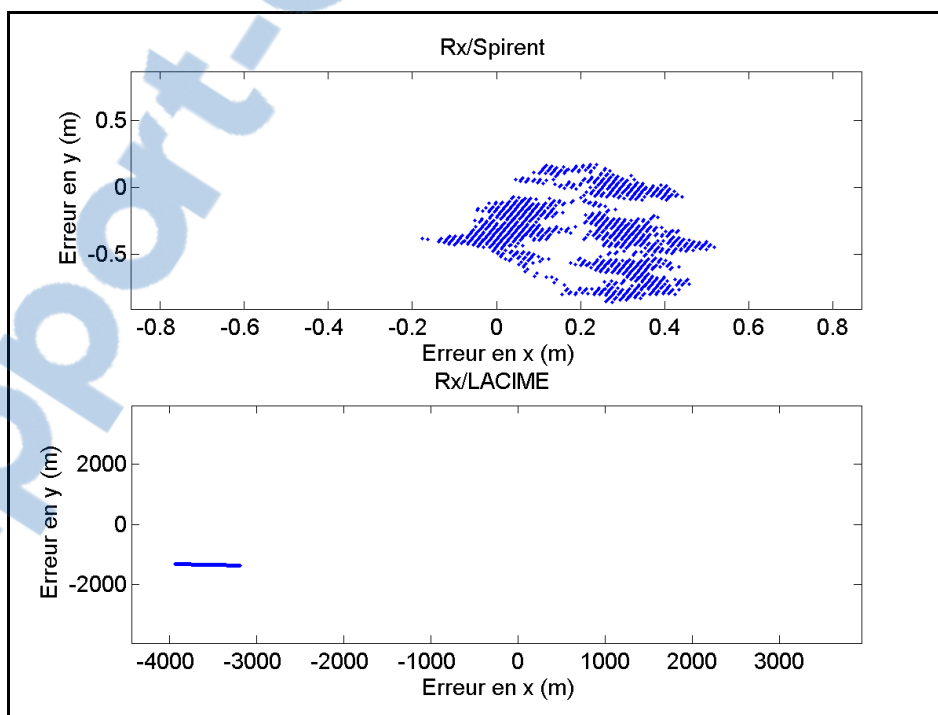


Figure 2.16 Nuage de points (Rx/Spirent versus Rx/LACIME)

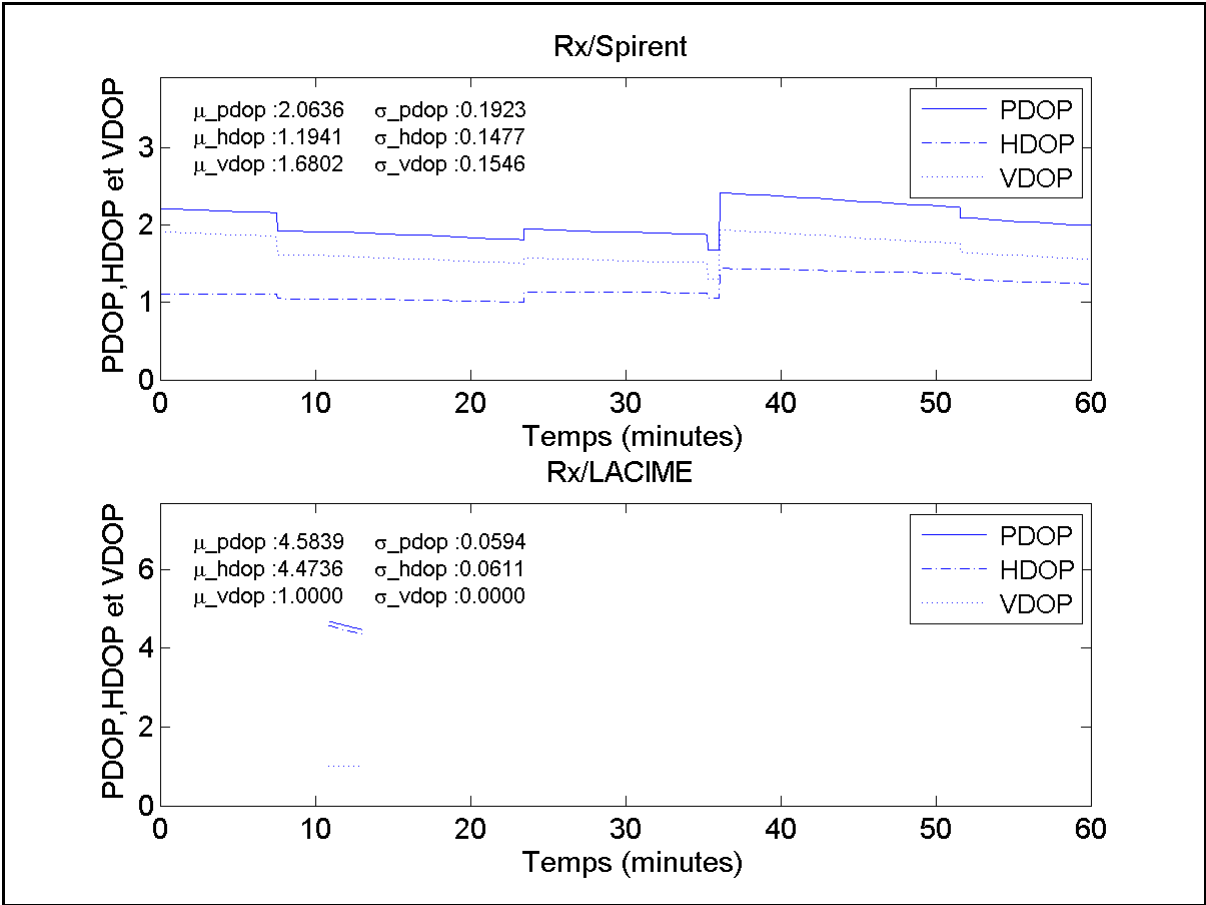


Figure 2.17 PDOP, HDOP et VDOP (Rx/Spirent versus Rx/LACIME)

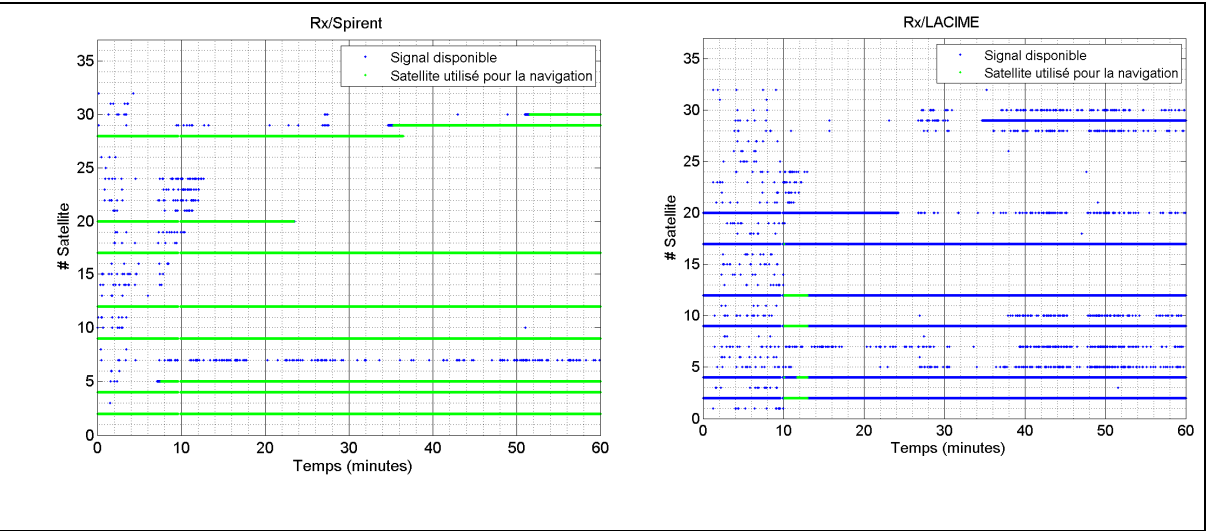


Figure 2.18 Visibilités des satellites (Rx/Spirent versus Rx/LACIME)

Tableau 2.3 Les moyennes et les écarts-types de l'erreur de position et du DOP (Rx/Spirent versus Rx/LACIME)

Paramètres	Rx/Spirent	Rx/LACIME
μ_{errx} (m)	0,1969	-3557,2484
σ_{errx} (m)	0,1538	220,7256
μ_{erry} (m)	-0,3354	-1342,7632
σ_{erry} (m)	0,2318	14,5401
μ_{err} (m)	0,4342	3803,3369
σ_{err} (m)	0,2002	201,3047
μ_{pdop}	2,0636	4,5839
σ_{pdop}	0,1923	0,0594
μ_{hdop}	2,0636	4,4736
σ_{hdop}	0,1477	0,0611
μ_{vdop}	1,6802	1,00
σ_{vdop}	0,1546	0,00

Avec le simulateur LACIME, le TTFF (Time To First Fix) obtenu est approximativement de 11 minutes. Il est trop long parce qu'une durée d'une simulation de trois minutes avec le simulateur Spirent suffit au récepteur u-blox pour trouver une position. Lorsque le TTFF a été atteint d'une simulation avec le simulateur LACIME, la position calculée par le récepteur est conservée pendant seulement 2 minutes. La moyenne de l'erreur de la position se situe à 3 803,3369 mètres occasionnant des valeurs plus élevées que la normale des paramètres PDOP, HDOP et VDOP. Elle est beaucoup trop grande ce qui évoque un manque de fiabilité du simulateur LACIME. Selon les résultats d'une simulation avec le simulateur Spirent, elle devrait être inférieure à 1 mètre.

Plusieurs problèmes peuvent causer les anomalies énumérées précédemment comme la génération du message binaire, du signal GPS etc. Ceci fera l'objet d'un examen approfondi dans les chapitres III et IV.

2.3 Conclusion

Ce chapitre est un aperçu du fonctionnement et du dysfonctionnement globaux du simulateur qui est nécessaire de connaître pour la continuité du projet. Le logiciel calcule les paramètres de navigation et génère le message binaire. Ensuite, les cartes IF et RF s'occupent respectivement à produire les porteuses IF des satellites à simuler et des transposer à leur fréquence respective. Plusieurs limitations et problèmes technologiques ont été trouvés et certains d'entre eux devront être résolus dans le but d'avoir un simulateur précis et robuste. Les résultats initiaux ont démontré qu'il y a beaucoup de travail à effectuer afin d'en venir à une solution stable pour une simulation de 24 heures. En effet, le récepteur a beaucoup de difficulté à garder une position. Les causes des problèmes observés viennent principalement du logiciel et du FPGA. Des erreurs au niveau des algorithmes de calculs doivent survenir. Aussi, il est possible que la porteuse ne soit pas bien générée, car le récepteur n'est pas capable de démoduler le signal GPS. Des investigations pour résoudre les problèmes seront nécessaires et seront faits dans les chapitres 3 et 4. Le chapitre suivant décrira les corrections apportées à la partie logicielle, plus spécifiquement les calculs des paramètres de navigation et le processus de génération des messages binaires de navigation.

CHAPITRE 3

AMÉLIORATIONS ET VALIDATION DE LA PARTIE LOGICIELLE DU SIMULATEUR DE CONSTELLATION GPS L1

Les principaux problèmes du simulateur étant connus, ce chapitre va se concentrer sur la validation de la partie logicielle qui sera présentée en deux parties. La première comprend les calculs des paramètres de navigation qui se rafraîchissent sur un GUI à toutes les secondes. La deuxième comprend la génération du message binaire de navigation. Ces deux parties comportent essentiellement des opérations arithmétiques et binaires et ils sont effectués sur un ordinateur. Les résultats des calculs sont transférés à la carte IF et cette section sera traitée au prochain chapitre.

3.1 Ajout d'une fonction d'enregistrement des mesures du simulateur

Il a été mentionné précédemment qu'une fonctionnalité d'enregistrement des données serait essentielle pour le déverminage de l'application. Alors, cette section expliquera en détail la conception de cette fonctionnalité essentielle à la démarche (voir Figure 3.1).

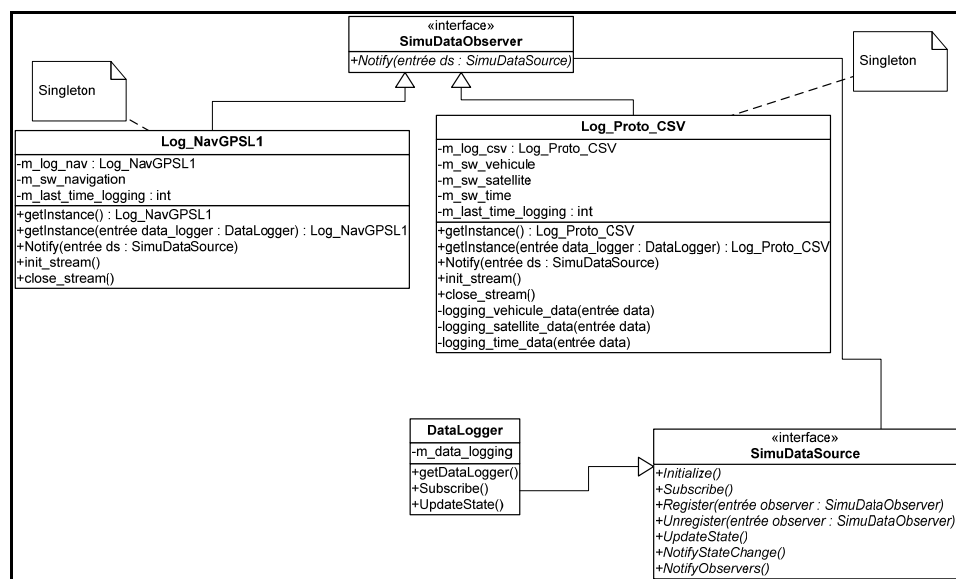


Figure 3.1 Diagramme de classes de la fonction d'enregistrement des données

Les classes `SimuDataSource` et `SimuDataObserver` étaient déjà utilisées dans le but de rafraîchir un GUI en temps réels. La classe `DataLogger` est le modèle de la fonction d'enregistrement qui encapsule les données à enregistrer des satellites, du temps, du véhicule et du message de navigation. Toutes les classes qui génèrent les fichiers d'enregistrement doivent dériver de `SimuDataObserver` et être un singleton. La classe `Log_Proto_CSV` produit trois fichiers (`sat.txt`, `vehicule.txt`, `time.txt`) CSV tandis que `Log_NavGPSL1` (`subframe_GPSL1.txt`) écrit tous les mots des sous-frames en hexadécimal. Après que des vérifications soient effectuées, la fonction d'enregistrement des mesures du simulateur fonctionne adéquatement.

3.2 Validation des algorithmes du simulateur LACIME

Maintenant que la fonctionnalité d'enregistrement des mesures du simulateur LACIME est fonctionnelle, il sera possible de vérifier la validité des algorithmes. Suite à une première validation, s'il y a lieu, des corrections seront apportées aux algorithmes de calculs. Ensuite, une revalidation sera faite afin de valider les modifications effectuées. Deux simulations ayant les mêmes configurations seront exécutées : l'une avec le simulateur LACIME et l'autre avec le simulateur Spirent. Les valeurs des paramètres de navigation des deux simulations seront comparées ensemble à l'aide de leur fonction respective d'enregistrement des données. L'erreur de ceux-ci est définie comme étant la différence entre la valeur calculée de SimGen (logiciel de simulation de Spirent) et LACIME (SimGen-LACIME).

3.2.1 Calcul des positions des satellites

Les Figures 3.2 à 3.4 montrent la position et son erreur selon l'axe des x, y et z du satellite #4 avant et après la modification apportée. D'autres graphiques sur les positions des satellites sont à l'ANNEXE II.

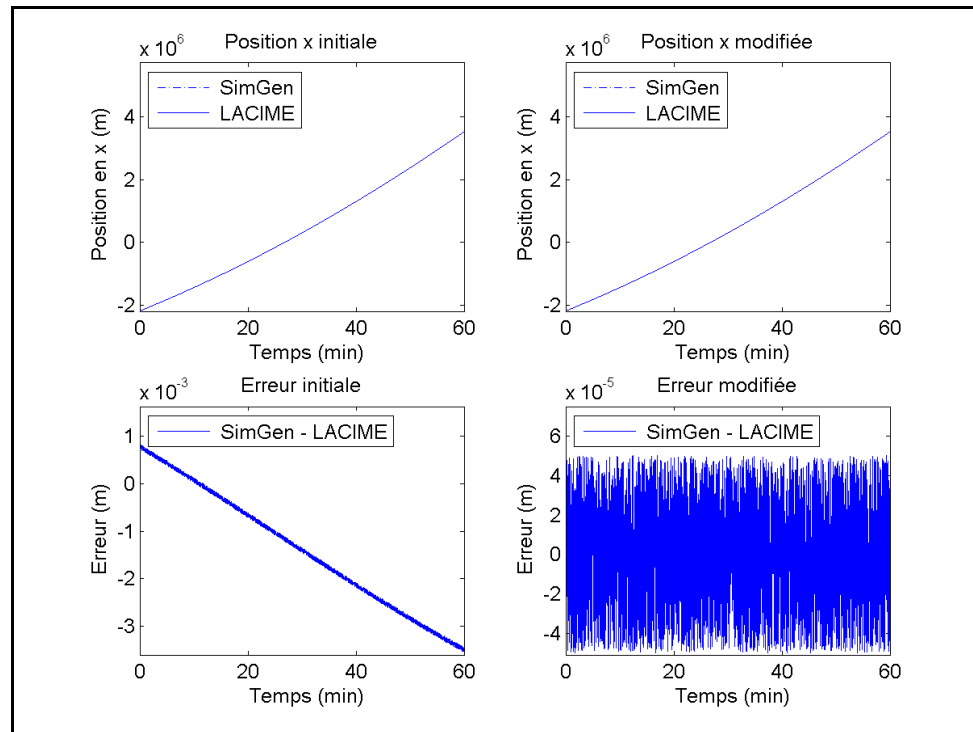


Figure 3.2 Position en x et son erreur initiale et modifiée du satellite #4

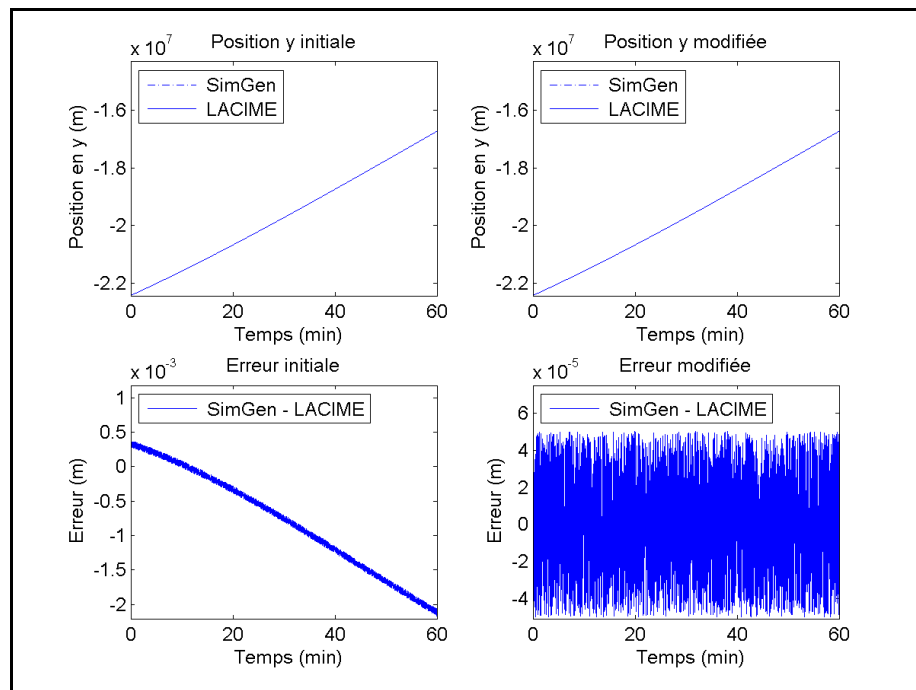


Figure 3.3 Position en y et son erreur initiale et modifiée du satellite #4

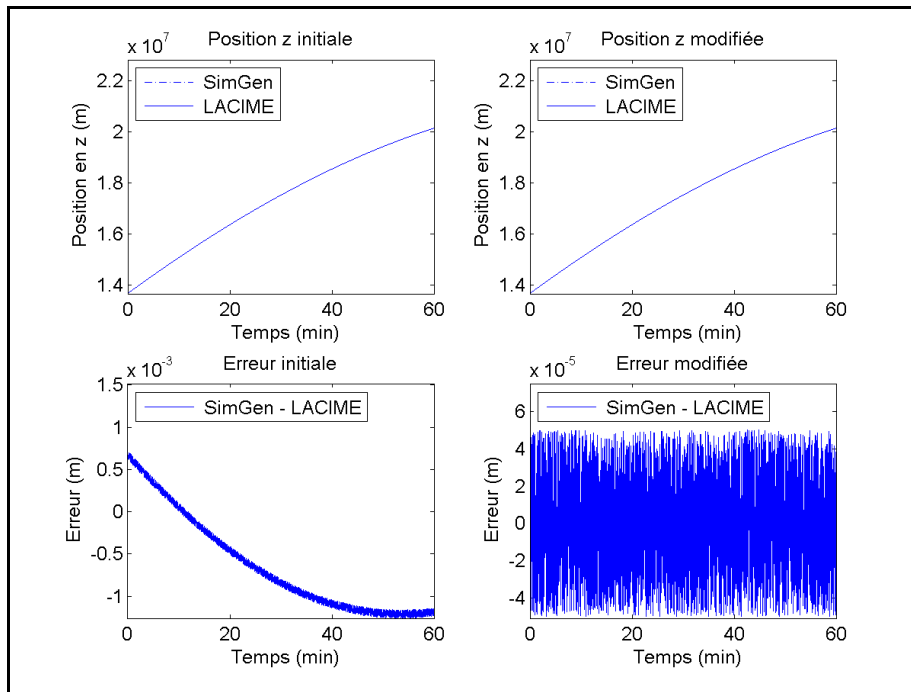


Figure 3.4 Position en z et son erreur initiale et modifiée du satellite #4

Le positionnement des satellites doit être le plus précis possible, car il est pris en considération lors de la génération des signaux. Son erreur initiale correspond à une fonction linéaire qui est causée par le temps de transition et il était évalué à partir de l'emplacement précédent du satellite. Pour corriger l'inexactitude, il faudrait le trouver à partir de la position courante du satellite. Pour pallier ce problème, l'emplacement des satellites est établi à tous les 100 ms et le temps de transition est estimé par une extrapolation des deux dernières positions du satellite (il y a 200 ms et 100 ms). L'algorithme permettant de calculer la position des satellites est détaillé à l'ANNEXE III. Après les corrections effectuées, il y a une erreur résiduelle maximale de 0,05 mm comparativement avec les mesures obtenues de SimGen. Une erreur bruitée et inférieure à 1 mm en tout temps est suffisante pour démontrer que les positions des satellites sont correctement calculées. Alors, l'algorithme de calcul des positions des satellites fonctionne bien.

3.2.2 Calcul des vitesses des satellites

Les Figures 3.5 à 3.7 montrent la vitesse et son erreur selon l'axe des x, y et z du satellite #4 avant et après la correction apportée (voir ANNEXE II pour plus de détails).

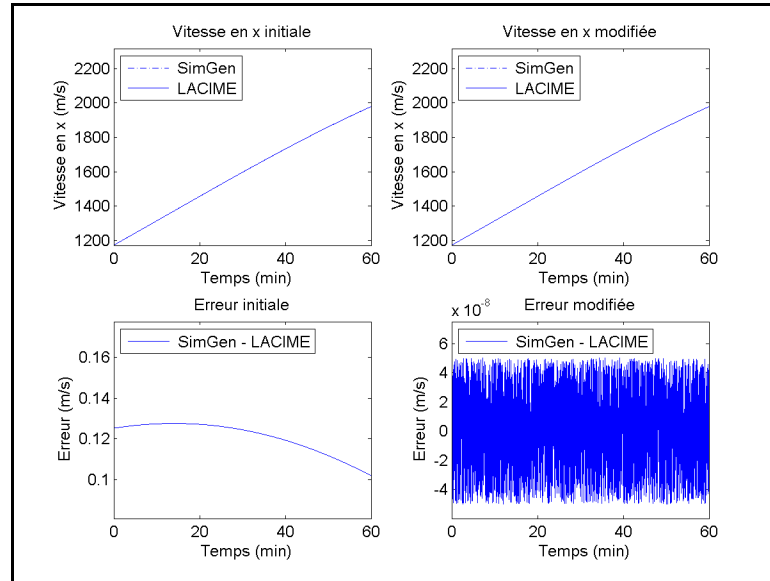


Figure 3.5 Vitesse en x et son erreur initiale et modifiée du satellite #4

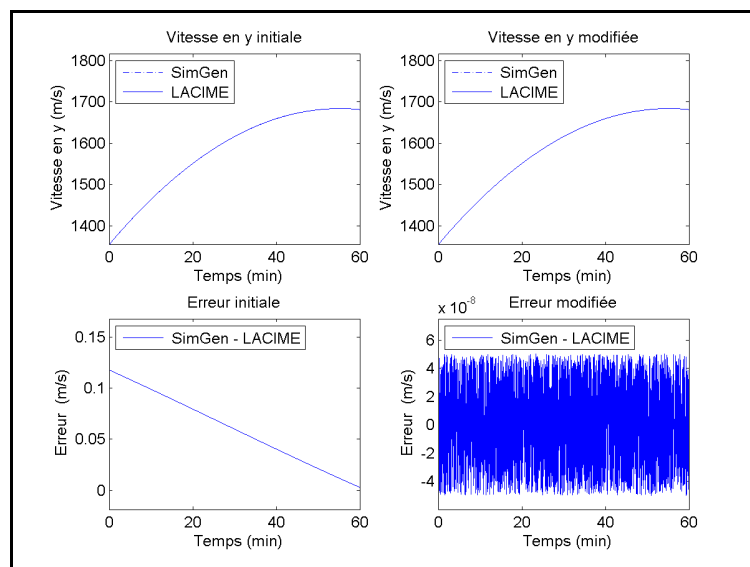


Figure 3.6 Vitesse en y et son erreur initiale et modifiée du satellite #4

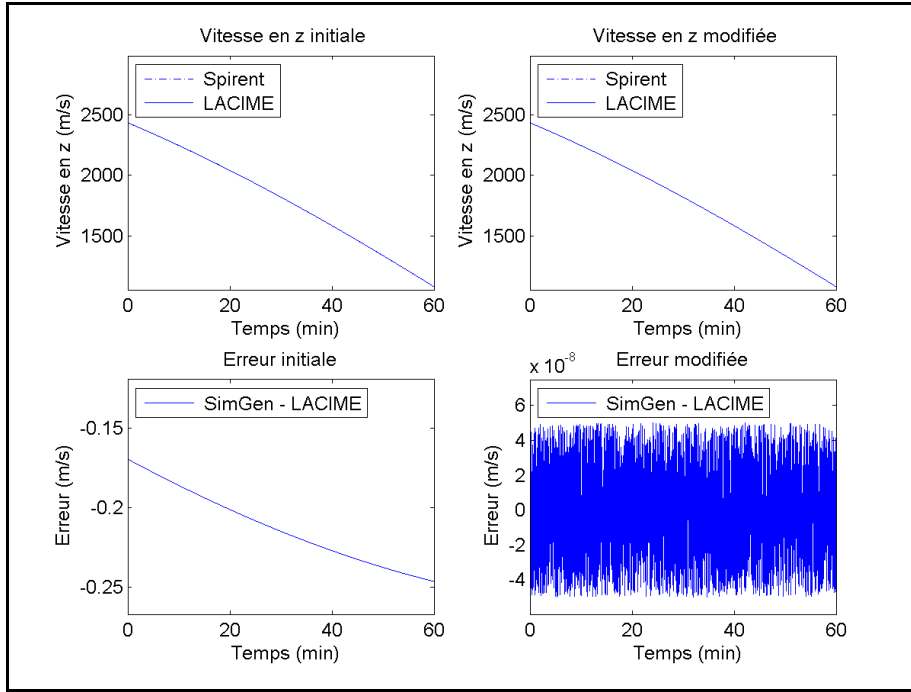


Figure 3.7 Vitesse en z et son erreur initiale et modifiée du satellite #4

L'équation (3.1) montre la formule initiale pour évaluer la vitesse des satellites selon l'axe des x :

$$V_x = \frac{x_{sat}(t) - x_{sat}(t-1)}{durée_echantillon} \quad (3.1)$$

où :

- $x_{sat}(t-1)$ est la position du satellite selon l'axe des x de l'échantillon précédent (m),
- $x_{sat}(t)$ est la position courante du satellite selon l'axe des x (m),
- $durée_echantillon$ est la durée d'un échantillon (1s).

Les vitesses des satellites selon l'axe des y et des z se calculent avec l'équation (3.1) en remplaçant les valeurs des positions des satellites selon leur axe respectif. Elles étaient approximées en supposant la linéarité de la position des satellites. Or, celles-ci ne sont pas linéaires causant une erreur sur les vitesses des satellites. Pour une meilleure précision, les vitesses doivent être calculées à partir des paramètres orbitaux et la procédure à suivre est

détaillée à l'ANNEXE IV. En suivant les directives de cet algorithme, on obtient une erreur résiduelle maximale de $5 \cdot 10^{-8}$ m/s sur les vitesses des satellites permettant de conclure qu'elles sont identiques à celles générées par SimGen.

3.2.3 Calcul de l'azimuth des satellites

L'angle d'azimuth est la mesure en degré dans un même plan horizontal de référence de l'angle entre la direction d'un satellite et le nord géographique (voir Figure 3.8).

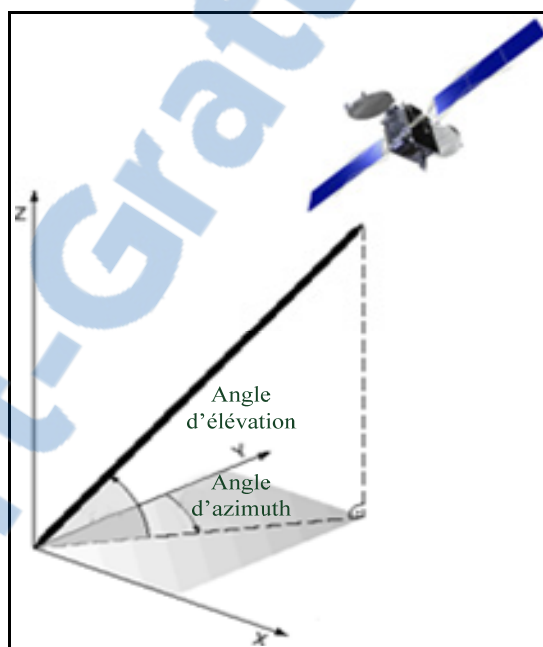


Figure 3.8 Angle d'azimuth et d'élévation d'un satellite
Adaptée de Chee, Franke, Froumentin et Yaeger (2006)

Les graphiques de la Figure 3.9 illustrent l'angle d'azimuth et son erreur pour le satellite #4 avant et après les modifications apportées (voir ANNEXE II pour les résultats de d'autres satellites).

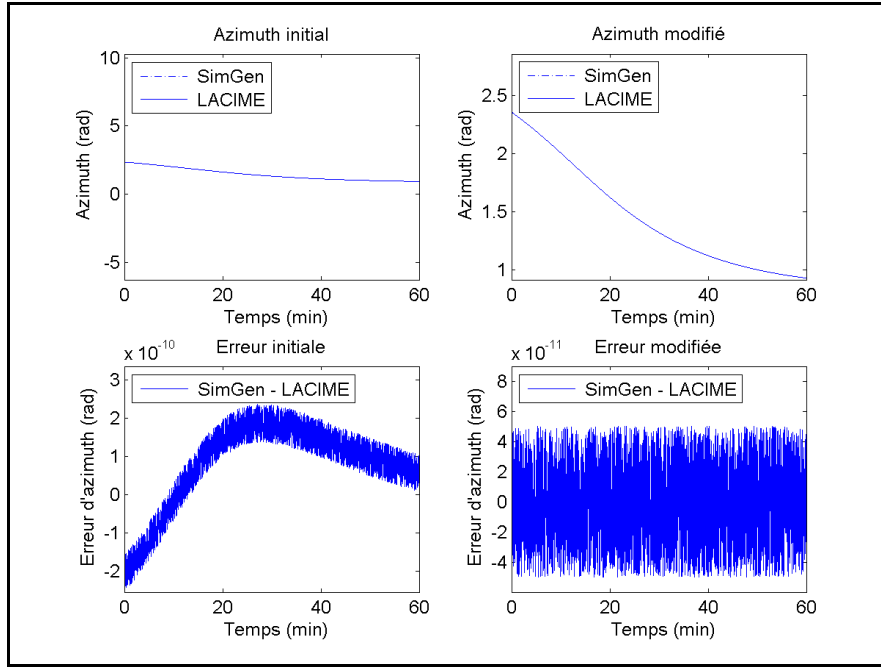


Figure 3.9 L'angle d'azimuth et son erreur initiale et modifiée du satellite #4

Initialement, il était calculé selon l'équation (3.2) :

$$azimuth(t) = \pi - \arctan 2 \left(\begin{array}{l} -\sin(long) * (x_{sat}(t) - x_{rx}(t)) + \cos(lon) * (y_{sat}(t) - y_{rx}(t)), \\ \sin(lat) * \cos(long) * (x_{sat}(t) - x_{rx}(t)) + \\ \sin(lat) * \sin(long) * (y_{sat}(t) - y_{rx}(t)) \\ -\cos(long) * (z_{sat}(t) - z_{rx}(t)) \end{array} \right) * \frac{180}{\pi} \quad (3.2)$$

où :

- $x_{sat}(t)$, $y_{sat}(t)$ et $z_{sat}(t)$ sont les coordonnées de la position du satellite en format Earth Centered Earth Fixed (ECEF) à un temps t donné,
- $x_{rx}(t)$, $y_{rx}(t)$ et $z_{rx}(t)$ sont les coordonnées du récepteur en format ECEF à un temps t donné,
- $long$ est la longitude du récepteur (rad),
- lat est la latitude du récepteur (rad).

L'erreur initiale de l'angle d'azimuth est de 2π pour certains satellites ce qui indique que l'angle d'azimuth n'est pas à sa plus simple expression c'est-à-dire entre 0 et 2π . Pour corriger cet aspect, il faut calculer celui-ci à partir des coordonnées East North Up (ENU) du satellite selon l'équation (3.3) (European Commission,2011) :

$$azimuth(t) = \arctan2 \left(\frac{diff_x(t)}{\sqrt{diff_x(t)^2 + diff_y(t)^2 + diff_z(t)^2}}, \frac{diff_y(t)}{\sqrt{diff_x(t)^2 + diff_y(t)^2 + diff_z(t)^2}} \right) * \frac{180}{\pi} \quad (3.3)$$

où :

- $diff_x(t)$, $diff_y(t)$ et $diff_z(t)$ sont les différences selon l'axe des x, y et z entre la position du satellite et du récepteur en format ENU.

En utilisant les coordonnées ENU pour trouver l'angle d'azimuth, une erreur résiduelle maximale de 0,05 nrad est obtenue. Une erreur bruitée inférieure à 0,1 nrad en tout temps est suffisante pour conclure que l'algorithme de calcul des angles d'azimuth est valide. Donc, les angles d'azimuth du simulateur LACIME correspondent bien avec ceux de SimGen.

3.2.4 Calcul de l'élévation des satellites

L'angle d'élévation est la mesure en degré de l'angle entre l'axe du satellite et l'horizon local (voir Figure 3.8). Dans le simulateur, le satellite est visible si celui-ci est supérieur ou égale au masque d'élévation qui est un paramètre configurable de la simulation. La Figure 3.10 montre l'angle d'élévation et son erreur pour le satellite #4 avant et après la modification apportée (voir ANNEXE II pour plus de détails).

L'angle d'élévation était calculé initialement selon la formule de l'équation (3.4) :

$$elevation(t) = \frac{\pi}{2} - \arccos \left(\frac{x_{rx} * (x_{sat} - x_{rx}) + y_{rx} * (y_{sat} - y_{rx}) + z_{rx} * (z_{sat} - z_{rx})}{dist_géo * \sqrt{x_{rx}^2 + y_{rx}^2 + z_{rx}^2}} \right) * \frac{180}{\pi} \quad (3.4)$$

où :

- $x_{\text{sat}}(t)$, $y_{\text{sat}}(t)$ et $z_{\text{sat}}(t)$ sont les coordonnées des positions du satellite en format ECEF à un temps t donné,
- $x_{\text{rx}}(t)$, $y_{\text{rx}}(t)$ et $z_{\text{rx}}(t)$ sont les coordonnées du récepteur en format ECEF à un temps t donné,
- dist_géo est la distance géométrique du satellite.

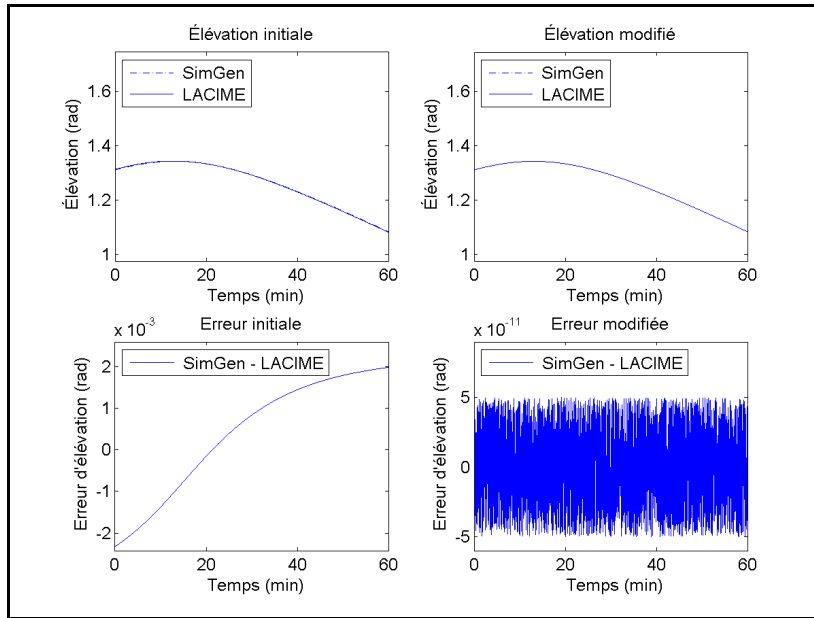


Figure 3.10 L'angle d'élévation et son erreur initiale et modifiée du satellite #4

En employant directement les positions des satellites en format ECEF, l'erreur d'élévation est beaucoup trop grande. Afin de la diminuer, l'angle d'élévation doit être estimé avec le format ENU des satellites comme l'indique l'équation (3.5) (European Commission, 2011) :

$$\text{elevation}(t) = \arcsin \left(\frac{\text{diff}_z(t)}{\sqrt{\text{diff}_x(t)^2 + \text{diff}_y(t)^2 + \text{diff}_z(t)^2}}, \frac{\text{diff}_z(t)}{\sqrt{\text{diff}_x(t)^2 + \text{diff}_y(t)^2 + \text{diff}_z(t)^2}} \right) * \frac{180}{\pi} \quad (3.5)$$

où :

- $\text{diff}_x(t)$, $\text{diff}_y(t)$ et $\text{diff}_z(t)$ sont les différences selon l'axe des x , y et z entre la position du satellite et du récepteur en format ENU.

Une erreur bruitée inférieure à 0,1 nrad en tout temps de l'angle d'élévation est suffisante pour conclure qu'il est correctement calculé. Avec l'équation (3.5), l'angle d'élévation a une erreur maximale de 0,05 nrad permettant de conclure qu'il est calculé correctement sur le simulateur LACIME.

3.2.5 Calcul de la distance géométrique des satellites

La distance géométrique (Γ) est la distance théorique exacte entre le récepteur et le satellite. Les délais de propagation seront ajoutés dans le calcul de la pseudo-distance. La Figure 3.11 représente Γ et son erreur (Γ_{err}) initiale et modifiée du satellite #4 (voir ANNEXE II pour plus de détails).

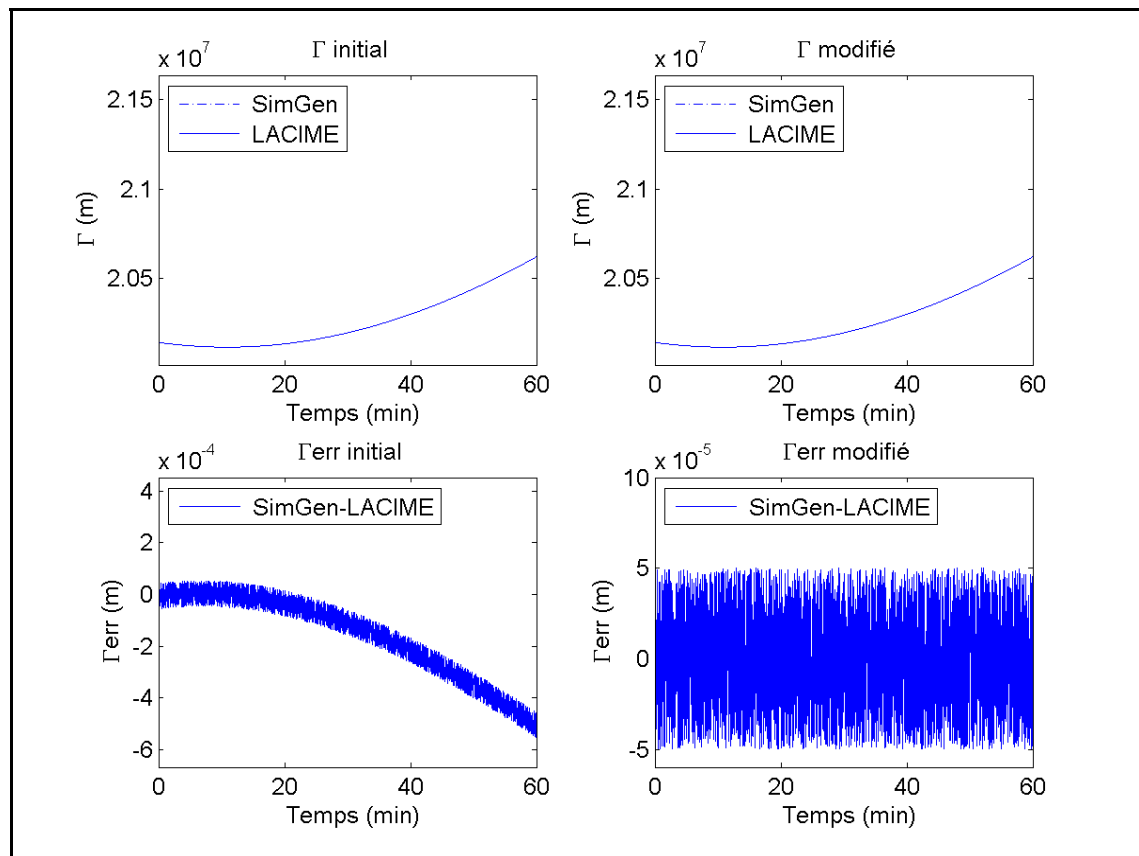


Figure 3.11 Γ et Γ_{err} initiale et modifiée du satellite #4

On trouve Γ en utilisant l'équation :

$$\Gamma(t) = \sqrt{(x_{sat}(t) - x_{rx}(t))^2 + (y_{sat}(t) - y_{rx}(t))^2 + (z_{sat}(t) - z_{rx}(t))^2} \quad (3.6)$$

où :

- $x_{sat}(t)$, $y_{sat}(t)$ et $z_{sat}(t)$ sont les coordonnées ECEF du satellite à un temps t donné,
- $x_{rx}(t)$, $y_{rx}(t)$ et $z_{rx}(t)$ sont les coordonnées ECEF du récepteur à un temps t donné.

Cette fonction a été validée et en corrigeant les erreurs de positions des satellites à la section 3.2.1, la valeur maximale de Γ_{err} se fixe à 0,05 mm. Aucune correction n'a été effectuée à la formule qui calcule Γ . Donc, les distances géométriques sont maintenant valides.

3.2.6 Calcul de la pseudo-distance des satellites

La pseudo-distance (ρ) est la distance entre le récepteur et le satellite en prenant compte des délais d'horloge. La Figure 3.12 illustre ρ et son erreur (ρ_{err}) du satellite #4 avant et après la modification apportée (voir ANNEXE II pour plus de détails).

ρ est calculé selon l'équation (3.7):

$$\rho(t) = \Gamma(t) - \text{biais_horloge}(t) * c - \text{relativite}(t) * c \quad (3.7)$$

où :

- biais_horloge est le biais d'horloge à un temps t donné (seconde),
- $\text{relativite}(t)$ est le terme de correction de la relativité à un temps t donné,
- c est la vitesse de la lumière (299792458 m/s).

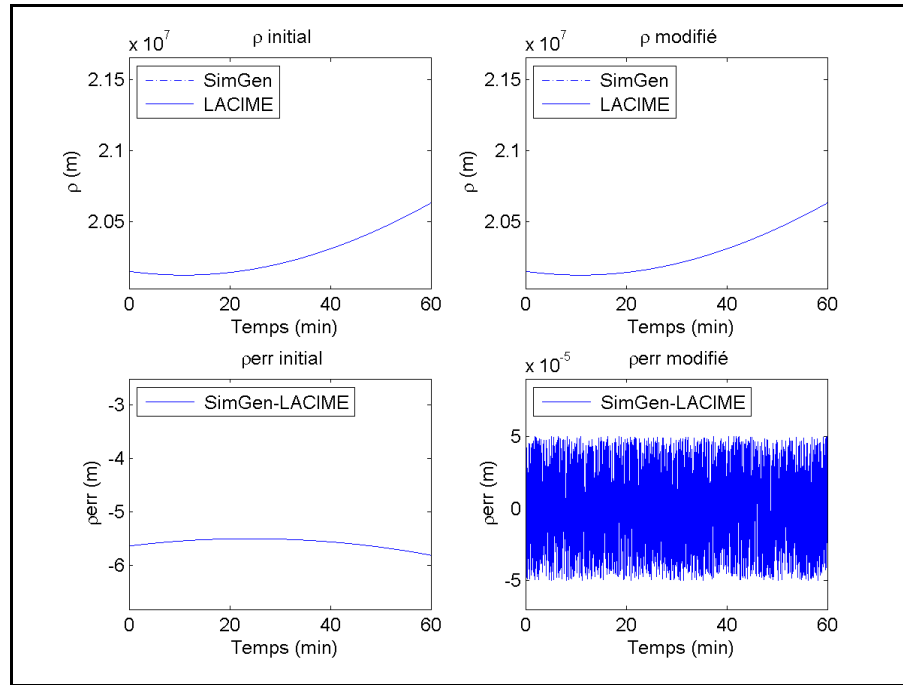


Figure 3.12 ρ et ρ_{err} initial et modifié du satellite #4

Celle-ci a été analysée en profondeur et elle a permis de conclure qu'elle est valide. Dans ce cas, les erreurs de pseudo-distance sont dues au biais d'horloge et aux corrections de l'effet relativiste des satellites.

Les satellites utilisent des horloges atomiques pour avoir une meilleure précision du temps. Comme elles sont très dispendieuses, les récepteurs emploient des horloges de moins bonnes qualités. Dans le message de navigation GPS L1, les coefficients af_0 et af_1 permettent de corriger les erreurs de temps générées par l'horloge du récepteur. Le biais d'horloge à un temps t donné est calculé selon l'équation (3.8) :

$$biais_horloge(t) = af_0 + af_1(t - t_{oc}) \quad (3.8)$$

où :

- af_0 est le coefficient du biais d'horloge d'ordre 0 (sec),
- af_1 est le coefficient du biais d'horloge d'ordre 1 (sec/sec),
- t_{oc} est le temps de référence des corrections d'horloges (sec).

Cette formule a été vérifiée et aucune modification n'a été apportée. Par contre, dans le simulateur LACIME, af_0 variait en fonction du temps ce qui n'est pas valide. Celui-ci est maintenant constant pour toute la durée de la simulation et il provient du fichier almanach.

Le terme de la correction de la relativité tient compte de la rotation de la terre il est évalué initialement avec l'équation (3.9):

$$relativite(t) = -2 * \left(\frac{x_{sat}(t) * v_{xsat}(t) + y_{sat}(t) * v_{ysat}(t) + z_{sat}(t) * v_{zsat}(t)}{c^2} \right) \quad (3.9)$$

où :

- $x_{sat}(t)$, $y_{sat}(t)$ et $z_{sat}(t)$ sont les coordonnées ECEF du satellite à un temps t donné,
- $v_{xsat}(t)$, $v_{ysat}(t)$ et $v_{zsat}(t)$ sont les vitesses respectives selon l'axe des x, y et z du satellite à un temps t donné,
- c est la vitesse de la lumière (299792458 m/s).

Après vérification de la validité de la formule, il a été établi que celle-ci ne correspond pas avec les valeurs de SimGen. En utilisant la définition pour évaluer la relativité, l'équation (3.9) a été remplacée par l'équation (3.10) (ARINC Engineering Services, 2004) :

$$relativite(t) = F * e * \sqrt{A} * \sin(E_k) \quad (3.10)$$

où :

- F est constant à $-4,442807633 * 10^{-10} \frac{\text{sec}}{\sqrt{\text{mètre}}}$,
- e est l'excentricité de l'orbite du satellite,
- a est le demi-grand axe (m),
- E_k est l'anomalie excentrique (voir ANNEXE III pour plus de détail).

En corrigeant le biais d'horloge et la relativité des satellites, on obtient un ρ_{err} maximum de 0,05 mm. Alors, les pseudo-distances du simulateur LACIME sont valides.

3.2.7 Calcul de la fréquence Doppler des satellites

La fréquence Doppler (Δf) est le décalage fréquentiel entre l'émission et la réception d'une onde électromagnétique. Elle est influencée par les mouvements du véhicule et des satellites. La Figure 3.13 montre Δf et son erreur (Δf_{err}) du satellite #4 avant et après la correction appliquée (voir ANNEXE II pour plus de détails).

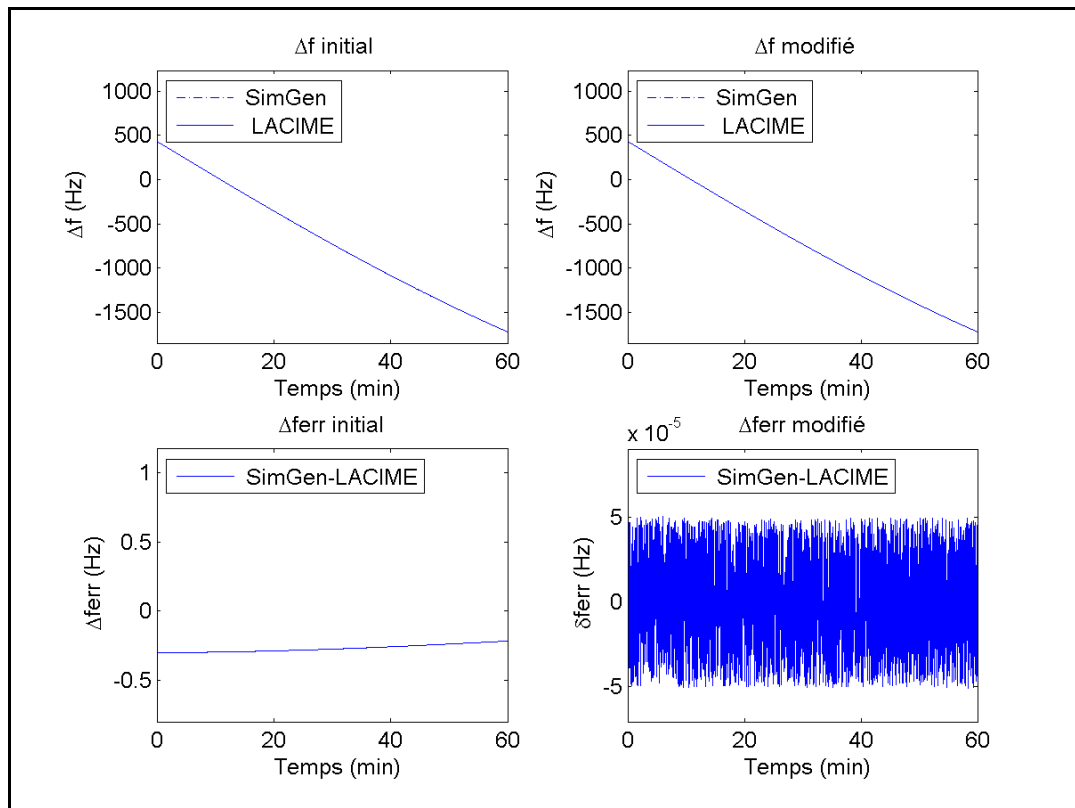


Figure 3.13 Δf et Δf_{err} initial et modifié du satellite #4

Δf est actuellement calculée selon l'équation (3.11) :

$$\Delta f(t) = \left(\frac{\rho(t) - \rho(t-1)}{\text{Durée d'un échantillon}} \right) * \left(\frac{F}{c} \right) \quad (3.11)$$

où :

- $\rho(t)$ est la pseudo-distance à l'instant t donné (m),
- $\rho(t-1)$ est la pseudo-distance de l'échantillon précédent (m),
- Durée d'un échantillon est l'intervalle de temps entre deux calculs successifs (1 sec).
- F est la fréquence porteuse du signal RF (1575,42 MHz).
- c est la vitesse de la lumière (299792458 m/s).

Après vérification, la formule établie initialement s'est avérée valide. Par contre, la valeur des paramètres est invalide. En effet, il y a toujours une plus grande erreur sur la valeur initiale du Doppler. Comme $\rho(t-1)$ n'a pas été initialisé pour $t = 0$, on a un Doppler erroné. Pour corriger ce problème, les calculs des paramètres de navigation s'effectuent durant la phase d'initialisation avec un TOW inférieur à une seconde du TOW du début de la simulation. Ainsi, cette étape d'initialisation qui n'était pas présente initialement permet à $\rho(t-1)$ d'avoir une valeur valide.

La durée d'un échantillon d'une seconde n'est pas suffisante pour avoir un Doppler valide. Pour que celui-ci soit acceptable, Δf_{err} doit être bruité avec une valeur maximale de 0,01 Hz. Initialement, cela n'a pas été respecté. Donc, la durée d'un échantillon a été fixée à 100 ms. En apportant les corrections appropriées sur le Doppler des satellites, la valeur maximum de Δf_{err} est de 51,614 uHz. Alors, les Dopplers du simulateur LACIME correspondent bien avec ceux de SimGen.

3.3 Validation du message binaire GPS L1 du simulateur LACIME

Cette sous-section permettra de vérifier et de faire les corrections qui s'imposent sur la génération du message binaire GPS L1 du simulateur LACIME. Chaque bit de chaque sous-trame produite pour un même TOW sera comparé avec celui du simulateur Spirent pour une simulation d'une heure. Lorsqu'ils sont différents entre eux, il y a une erreur. Le Tableau 3.1 présente l'erreur initiale du message binaire de navigation pour le satellite #4 (voir ANNEXE VI pour plus de détails).

La génération des messages binaires de navigation n'est pas fonctionnelle, car le pourcentage d'erreur est souvent supérieur à 40% peu importe le satellite. Dans les sections suivantes, ce processus sera analysé plus en profondeur afin de régler le problème.

Tableau 3.1 Erreur initiale du message binaire GPS L1 du simulateur LACIME pour le satellite #4

# Sous-trames	Numéro de pages	Erreur (%)
1	-	61,88
2	-	21,36
3	-	14,93
4	1,6,11,16,21	43,81
	2,3,4,5,7,8,9,10	41,13
	12,19,20,22,23,24	43,61
	13	43,27
	14,15	44,77
	17	40,27
	18	42,60
	25	42,67
5	1 à 24	49,96
	25	52,67

3.3.1 Amélioration de l'architecture de la génération du message de navigation

Une trame du message de navigation GPS est composée de 25 pages de 5 sous-trames de 300 bits. Sachant que la fréquence d'envoi d'une sous-trame est de 50 Hz, il faut douze secondes et demie pour transmettre une trame. La Figure 3.14 illustre brièvement la composition du message de navigation GPS L1.

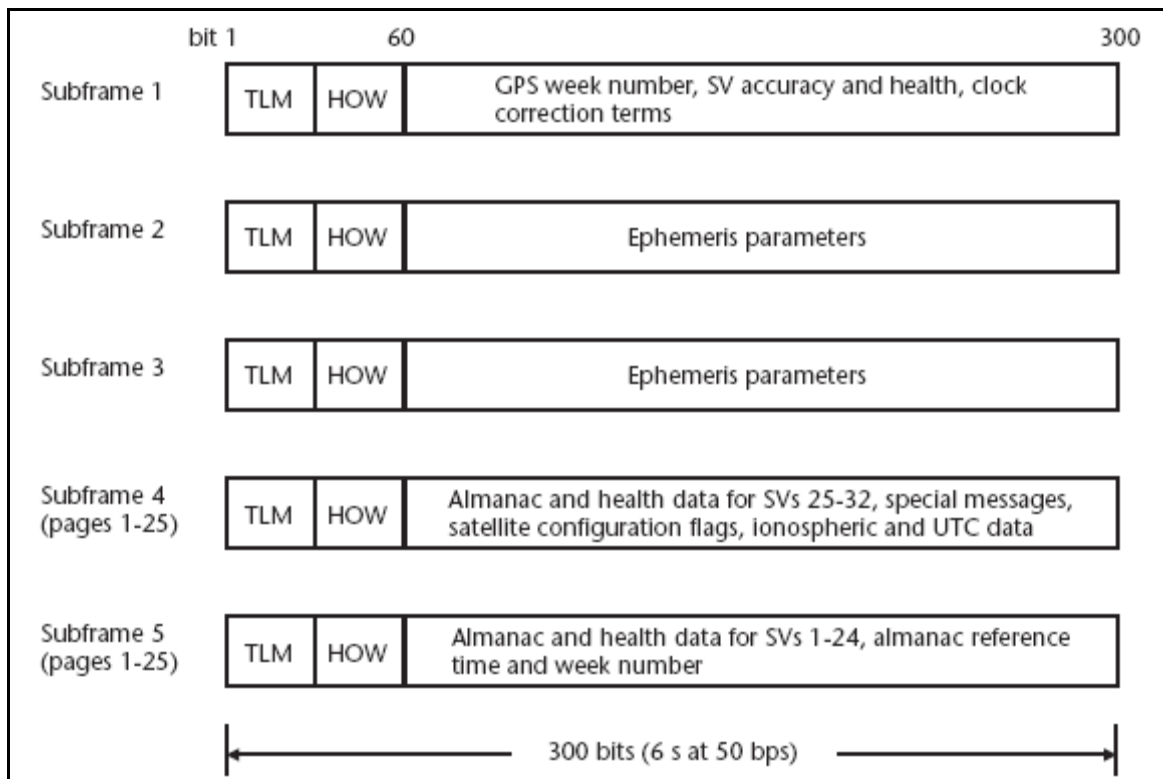


Figure 3.14 Format du message de navigation GPS L1
Tirée de Kaplan et Hegarty (2006, p. 143)

Initialement, dans le simulateur LACIME, il y a trois fonctions principales qui permettent de générer les messages de navigation : `UpdateConstParam()`, `UpdateVarParam()` et `GenerateMsg()`. La première et la seconde fonction affectent respectivement les constantes et les variables de la simulation (voir l'ANNEXE V pour leur description). `GenerateMsg()` est la fonction maître qui produit tous les messages binaires de navigation et elle contient plusieurs milliers de lignes de code. Pour faciliter le déverminage et sa lisibilité, celle-ci a été découpée en d'autres fonctions dont chacune d'entre elles génère un mot d'une sous-trame. Le morphogramme de la Figure 3.16 montre la structure modifiée de la génération des messages binaires. L'opération modulo 30 ($\text{mod } 30$) utilisée par la fonction `GenerateMsg()` représente le temps requis en seconde pour envoyer les 5 sous-frames d'une page.

Quelques ajustements ont été faits à la fonction `GenerateMsg()` pour respecter la fréquence du message de navigation qui est de 50 Hz (voir Figure 3.14). En effet, elles étaient produites en retard de six secondes. Pour régler ce problème, la variable locale `temps_courant` (voir Figure 3.16) a été incrémentée de 6 secondes parce qu'il faut les générer en avance par rapport au temps courant de la simulation. La Figure 3.15 montre l'état de la génération du message binaire de navigation.

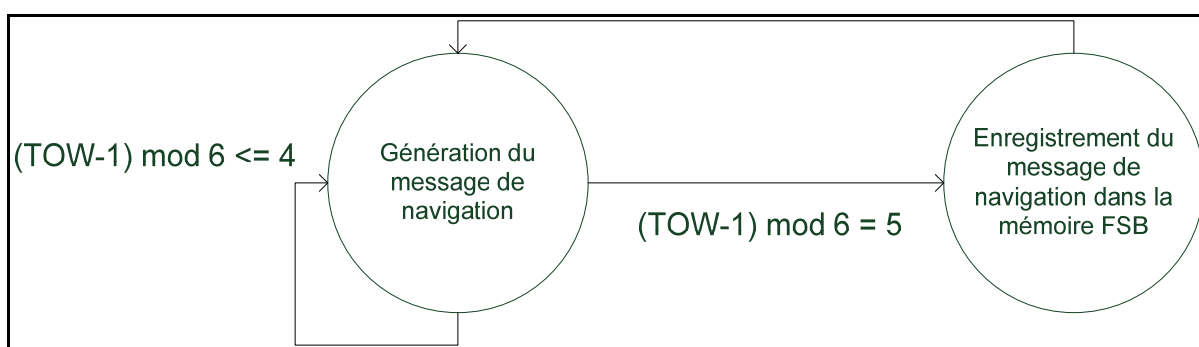


Figure 3.15 Diagramme d'état de la génération du message binaire de navigation GPS L1

La fonction `GenerateMsg()` est active pendant les 5 premières secondes de la génération d'une sous-trame. À partir de la cinquième seconde, une nouvelle sous-trame est produite et le processus #2 (voir Figure 2.3) l'enregistre dans la mémoire FSB. Auparavant, la mémoire FIFO était désynchronisée parce qu'elle était vide. Alors, la nouvelle façon de procéder l'empêche de se vider complètement. Les modifications apportées à la fonction `GenerateMsg()` ont été testées permettant de conclure qu'il n'y a plus de délai sur la génération du message binaire de navigation. Donc, la fonction `GenerateMsg()` est maintenant fonctionnelle.



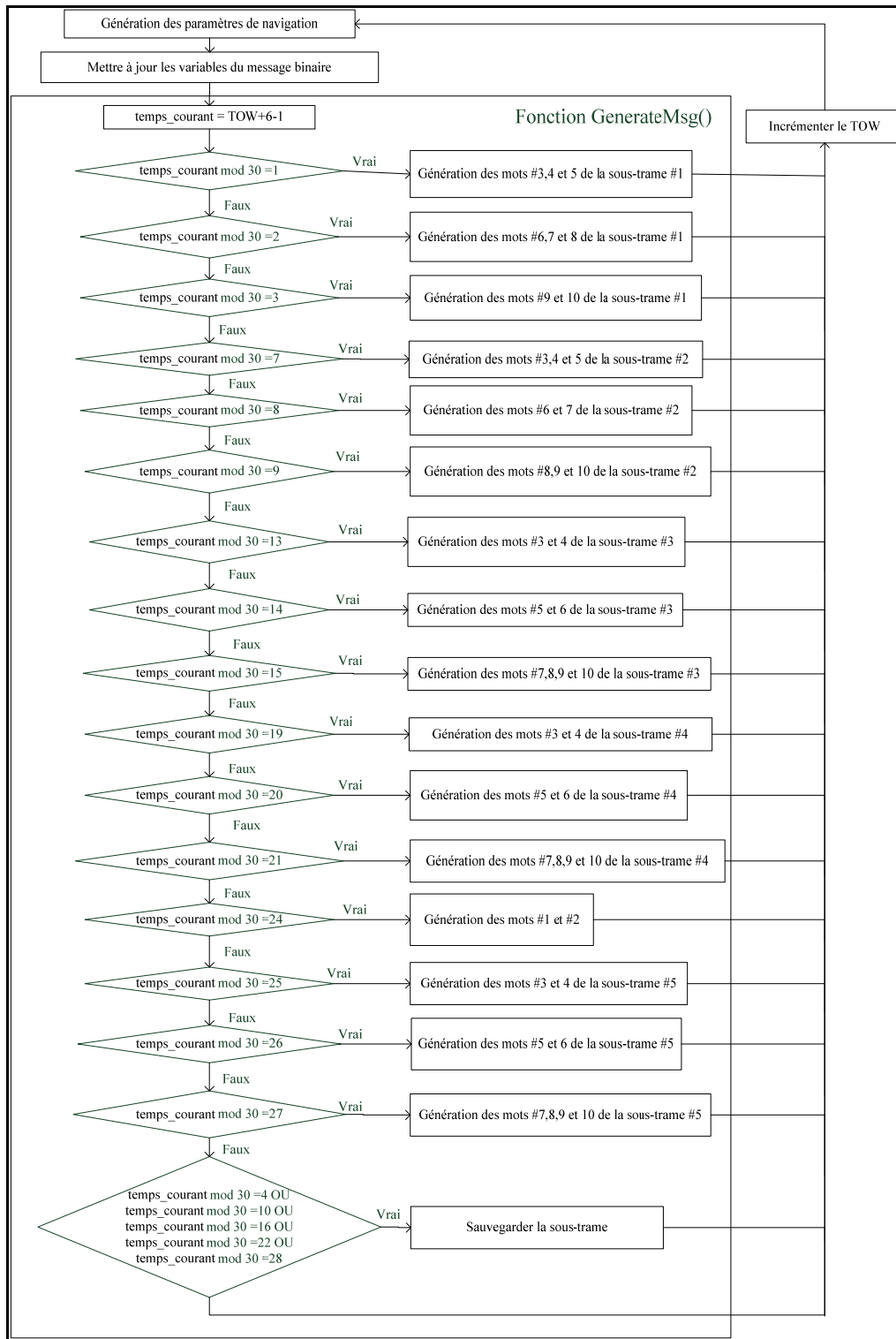


Figure 3.16 Morphogramme modifié de la génération des messages binaires de navigation (GPS L1)

3.3.2 Calcul de l'indicateur de disponibilité de la correction du message de navigation

L'indicateur de disponibilité de la correction du message de navigation (NMCT) détermine si le récepteur doit corriger les pseudo-distances mesurées. Si cette option est active, les pseudo-distances corrigées correspondront à la différence entre les pseudo-distances mesurées et le Estimated Range Deviation (ERD) (ARINC Engineering Services, 2004). L'indicateur de disponibilité du NMCT doit être défini à la page 13 du mot #3 de la sous-trame #4 du message de navigation. Le Tableau 3.2 énumère la signification des différentes valeurs de celui-ci.

Tableau 3.2 Disponibilité de la table de correction du message de navigation
Tirée de ARINC Engineering Services (2004, p. 116)

<u>AI</u>	<u>Navigation Message Correction Table Availability</u>
00	The correction table is unencrypted and is available to both authorized and unauthorized users.
01	The correction table is encrypted and is available only to authorized users (normal mode).
10	No correction table available for either authorized or unauthorized users.
11	Reserved.

Initialement, il était à '00'. Pourtant, il n'y aucune table de correction du message de navigation qui est employée. Alors, la valeur de l'indicateur a été remplacée par '10' indiquant au récepteur que la NMCT et les ERDs ne sont pas disponibles et par le fait même les pseudo-distances ne seront pas corrigées.

3.3.3 Calcul du Estimated Range Deviation (ERD)

Le ERD (6 bits) correspond à l'estimation de l'erreur de la pseudo-distance du satellite. Il est calculé par le segment de contrôle selon l'équation (3.12) (ARINC Engineering Services, 2004) :

$$ERD = err_éphéméride - c * err_clk \quad (3.12)$$

où :

- $err_éphéméride$ est la composante radiale de l'erreur d'éphémérides du satellite (m),
- c est la vitesse de la lumière (299792458 m/s),
- err_clk est l'erreur d'horloge du satellite (sec).

Dans une simulation, il serait possible de le faire varier pour ajouter une erreur volontaire dans le calcul des paramètres d'éphémérides. Pour rendre le ERD disponible, il faut activer la table de corrections du message de navigation (voir section 3.3.2). Initialement, le ERD était à '000000' ce qui signifie qu'il y a une erreur sur les paramètres d'éphémérides. Pourtant, dans le code, il n'est jamais employé pour les évaluer. Alors, comme la table de correction du message de navigation n'est pas active, les 6 bits du ERD dans le message de navigation doivent être alternés de '10' pour garder la synchronisation à la réception des signaux et par le fait même le rendre indisponible. La nouvelle valeur du ERD confirme que les bons paramètres d'éphémérides des satellites sont transmis. Ainsi, un récepteur sera en mesure de positionner correctement les satellites dans le ciel.

3.3.4 Statut de l'antibrouillage et de la configuration des satellites

À la page 25 de la sous-trame #4 du message de navigation GPS L1, 4 bits sont nécessaires pour spécifier le statut d'antibrouillage des satellites et leur configuration. Le bit le plus significatif contient un '0' pour désactiver l'option d'antibrouillage. Les autres bits représentent la configuration des satellites. L'état d'antibrouillage n'est pas pris en compte par le récepteur puisqu'il sert pour la corrélation du code P (Rizos, 1999). Le simulateur

LACIME simule seulement le code C/A. Le code de configuration des satellites spécifie au récepteur comment calculer le signal GPS en phase et en quadrature (ARINC Engineering Services, 2004). Il concorde au Tableau 3.3 selon les caractéristiques de la simulation.

Tableau 3.3 Code pour la configuration des satellites
Adaptée de ARINC Engineering Services (2004, p. 125)

Code	SV Configuration
001	“Block II/IIA/IIR” SV
010	“Block IIR-M” SV
011	“Block IIF” SV

Celui-ci était à ‘000’ ce qui permet de conclure qu’il est invalide. Il est maintenant à ‘001’ ce qui assure que le signal GPS en phase et en quadrature sera bien reçu.

3.3.5 Calcul de l’index User Range Accuracy (URA)

La variable URAIndex est un outil statistique de l’exactitude des données de navigation et elle doit être définie au mot #3 du 13^{ème} au 16^{ème} bit de la sous-trame #1. Le Tableau 3.4 présente la signification des différents indices.

Tableau 3.4 Index URA selon la précision des données de navigation
Adaptée de ARINC Engineering Services (2004, p. 84)

URA Index	URA (meters)
0	$0 < URA \leq 2,40$
1	$2,40 < URA \leq 3,40$
2	$3,40 < URA \leq 4,85$
3	$4,85 < URA \leq 6,85$
4	$6,85 < URA \leq 9,65$
5	$9,65 < URA \leq 13,65$
6	$13,65 < URA \leq 24,00$
7	$24,00 < URA \leq 48,00$
8	$48,00 < URA \leq 96,00$

URA Index	URA (meters)
9	$96,00 < URA \leq 192,00$
10	$192,00 < URA \leq 384,00$
11	$384,00 < URA \leq 768,00$
12	$768,00 < URA \leq 1536,00$
13	$1536,00 < URA \leq 3072,00$
14	$3072,00 < URA \leq 6144,00$
15	$61440,00 < URA$ (or no accuracy prediction is available – unauthorized users are advised to use the SV at their own risk.)

Initialement, l'index URA était à 2 et il a été remplacé par 0, car aucune erreur n'a été introduite dans les données de navigation. Dans le futur, il serait possible d'en ajouter une volontaire dans les paramètres d'éphémérides pour varier l'index URA. Ainsi, cela aurait un impact pour le récepteur au moment d'estimer la position des satellites dans le ciel.

3.3.6 Calcul des paramètres Issue Of Data Clock (IODC) et Issue Of Data Ephemeris (IODE)

La variable IODC indique un changement sur les données de l'horloge (a_{f0} , a_{f1} , a_{f2} et t_{oc}). Lorsque cet événement survient, il faut incrémenter l'IODC. C'est le même principe pour la variable IODE à l'exception que celle-ci concerne les éphémérides (sous-trame #2 et #3). Au début de la simulation, les variables IODC et IODE sont initialisées à 2 et elles s'incrémentent de 1 à tous les 2 heures ce qui représente la durée de validité des éphémérides. Initialement, ces variables étaient constantes à 2 dans le simulateur LACIME pour toute la durée de la simulation. Donc, la correction qui a été effectuée sur IODC et IODE va permettre au récepteur de détecter convenablement une variation des informations de l'horloge et des éphémérides.

3.3.7 Calcul du temps de référence de l'horloge et des éphémérides

T_{oe} et T_{oc} sont respectivement le temps de référence des éphémérides et de l'horloge. Elles sont équivalentes et à toutes les deux heures elles s'incrémentent de 7200 secondes. La durée de validité des éphémérides a été fixée à deux heures. Initialement, dans le simulateur

LACIME, le temps de référence était inférieur au temps courant de la simulation. Or, selon les données de navigation recueillies par le simulateur Spirent, celui-ci devrait être plutôt en avance sur le temps de la simulation. L'équation (3.13) corrige ce problème sur les valeurs de Toe et Toc.

$$Toe(t) = 7200 * (\text{entier}) \left(\frac{TOW}{7200} \right) + 7200 \quad (3.13)$$

Le paramètre TOW est le nombre de secondes écoulées depuis le début de la semaine et il est à 0 tous les dimanches à minuit.

3.3.8 Calcul du numéro courant de la sous-trame

La variable SubFrameID indique le numéro de la sous-trame courante à générer. Elle est évaluée en fonction du TOW au lieu du temps écoulé de la simulation. L'équation (3.14) illustre comment calculer le numéro de la sous-trame à générer.

$$SubFrameID = \left(\frac{TOW + 6}{6} \right) \bmod(5) + 1 \quad (3.14)$$

Le chiffre 6 au dénominateur identifie le temps en seconde qu'une nouvelle sous-trame doit être produite.

3.3.9 Calcul du paramètre Time Of Week (TOW)

La variable TOW_DEC représente le temps tronqué (17 bits) de la semaine qui sera utilisé à tous les mots #2 des sous-frames. En exécutant plusieurs scénarios, il a été possible de constater qu'elle pouvait devenir négative à cause que la fonction IEEERemainder() définie dans le langage C++ ne calculait pas la bonne valeur du modulo. Donc, dans le code, elle a été remplacée par l'opérateur modulo (%). Les équations (3.15) et (3.16) sont respectivement les formules C++ avant et après la modification à propos du calcul du TOW.

$$TOW_DEC = IEERemainder\left(\left(\left(\frac{TOW+6}{6}\right)+1\right), 100800\right) \quad (3.15)$$

$$TOW_DEC = \left(\left(\frac{TOW+6}{6}\right)+1\right) \% 100800 \quad (3.16)$$

3.3.10 Calcul de la semaine de référence de l'almanach

Au mot #3 de la sous-trame #5 (page 25 du message de navigation), il faut spécifier le numéro de la semaine de référence de l'almanach qui est représenté par la variable WN_a . Selon les spécifications du GPS L1, WN_a doit être représenté par modulo 256 ce qui n'était pas le cas dans le simulateur. Voici la formule de l'équation (3.17) qui corrige le problème :

$$WN_a = \text{WeekNumber} \bmod 256 \quad (3.17)$$

où :

- WeekNumber est le numéro de la semaine courante.

3.3.11 Synchronisation des sous-frames

Chaque sous-trame a la même structure des mots #1 et #2 du message de navigation (voir Figure 3.14). Ceci est très important à respecter afin d'en assurer leur synchronisation ce qui n'avait pas été le cas. Alors, chaque sous-trame commence avec 8 bits de préambule (0x8B) suivi de 16 bits alternés de '1' et '0'. Dans le simulateur LACIME, les 16 derniers bits étaient à '0'. Donc, la correction qui a été faite assure la synchronisation entre chacune des sous-frames lorsqu'elles seront reçues.

3.3.12 Spécification des bits réservés

Certains mots des sous-frames contiennent des bits réservés où il est possible de les voir à partir du Tableau 3.5.

Tableau 3.5 Bits réservés des sous-trames du message binaire de navigation

Sous-trame #	Page #	Mot #
1	-	4,5,6,7
4	1,6,11,16,21	3,4,5,6,7,8,9
	12,19,20,22,23,24	3,4,5,6,7,8,9
	14,15	3,4,5,6,7,8,9
	18	10
	25	8
5	25	10

Pour générer un mot ayant des bits réservés, il est nécessaire d'alterner successivement '1' et '0' pour que le récepteur ne perde pas le rythme de fonctionnement des signaux GPS. Avant d'effectuer cette modification, un '0' était envoyé pour un bit réservé ce qui faisait perdre la synchronisation du récepteur sur le message de navigation.

3.3.13 Génération des messages spéciaux

La sous-trame #4 de la page 17 du message de navigation GPS L1 est réservée pour émettre des messages alphanumériques. Ceux-ci peuvent être utilisés par exemple pour envoyer de l'information à l'utilisateur. Ceci est à la discrétion de l'opérateur GPS. Le Tableau 3.6 présente les messages spéciaux expédiés par Spirent.

Tableau 3.6 Messages spéciaux envoyés

Mot #	Message #	Caractères
3	1	M
	2	A
4	1	D
	2	E
	3	□

Mot #	Message #	Caractères
5	1	B
	2	Y
	3	□
6	1	G
	2	S
	3	S
7	1	□
	2	L
	3	I
8	1	M
	2	I
	3	T
9	1	E
	2	D
	3	.
10	1	□
	2	□

3.3.14 Particularité de la parité des mots #10

L'algorithme de détection d'erreur du GPS L1 est un code de Hamming. Il est très important qu'il n'y est pas d'erreur sur les bits de parité. Sinon, un récepteur pourrait être mis en erreur lors de la détection et de la correction des bits du message de navigation. Le message de navigation GPS L1 oblige que les deux derniers bits de chaque sous-trame doivent être à '0'. Cette condition est nécessaire afin que chaque sous-trame commence par la même séquence, c'est-à-dire les 8 bits de préambule. Cela implique que les deux derniers bits des sous-frames avant la parité doivent être générés selon ce contexte ce qui n'a pas été considéré auparavant. L'approche permettant de les trouver est très simple. Il y a 4 possibilités pour ceux-ci (00,01,10,11). Donc, pour chacune d'entre elles, on vérifie en obtenant les bits de parité si les deux derniers bits de la sous-trame sont '0'. Ainsi, la modification apportée sur la parité des mots #10 assure que les erreurs détectées et corrigées par le récepteur seront les bonnes.

3.3.15 Problème de perte des signaux de tous les satellites

Peu importe la simulation, il a été mentionné à la section 2.1.4 que le récepteur perdait après cinq minutes de simulation les signaux de tous les satellites. En effet, le simulateur LACIME transmettait une rafale de '0' dans le message de navigation pour les paramètres d'almanach d'un satellite qui n'est pas présent dans la constellation. Ainsi, le récepteur perdait la synchronisation. Pour la garder, il faut envoyer une succession de '1' et '0' pour les paramètres d'almanach d'un satellite absent (Parkinson et Spilker, 1996).

3.4 Discussion relative aux résultats finaux de la partie logiciel

Les résultats comprenant la correction des paramètres de navigation peuvent être consultés à l'ANNEXE II. Les paramètres de navigations correspondent bien avec les valeurs calculées SimGen. La petite différence observée est due à la précision des paramètres. En effet, le simulateur LACIME a beaucoup plus de chiffres après la virgule. Le Tableau 3.7 illustre pour chaque paramètre le nombre de décimales des deux simulateurs.

Tableau 3.7 Nombre de décimales des paramètres de navigation (LACIME versus SimGen)

Paramètres de navigation	Nombre de décimales de SimGen	Nombre de décimales du simulateur LACIME
Position des satellites	4	7
Vitesse des satellites	7	11
Azimuth	10	14
Élévation	10	15
Distance géométrique	4	7
Pseudo-distance	4	7
Fréquence Doppler	4	11

Le Tableau 3.8 montre les erreurs restantes du message de navigation du satellite #4 après les corrections apportées (voir l'ANNEXE VI pour les autres satellites).

Les autres erreurs restantes dans le message de navigation sont causées par les bits indiquant la santé des satellites. Les différentes valeurs possibles sont indiquées dans le Tableau 3.9.

Tableau 3.8 Erreur modifiée du message binaire de navigation du satellite #4

# Sous-trames	Numéro de pages	Erreur initiale (%)	Erreur modifiée (%)
1	-	61,88	0,00
2	-	21,36	0,00
3	-	14,93	0,00
4	1,6,11,16,21	43,81	0,00
	2,3,4,5,7,8,9,10	41,13	6,42
	12,19,20,22,23,24	43,61	0,00
	13	43,27	0,00
	14,15	44,77	0,00
	17	40,27	0,00
	18	42,60	0,00
5	25	42,67	20,00
	1 à 24	49,96	0,00
	25	52,67	0,00

Tableau 3.9 Santé des satellites de l'almanach
Tirée de ARINC Engineering Services (2010, p. 30-1)

Health setting	Six bit health word in NAV message	SEM and YUMA numerical representation of a 6 bit health word
ACTIVE	000000	000
BAD	111100	060
DEAD	111111	063

Le simulateur Spirent n'utilise pas le bon mot pour la santé d'un satellite qui n'est pas opérationnel. En effet, il emploie la valeur 0x33 ('110011') pour le satellite #32 qui n'est plus disponible. Ceci a un impact à tous les endroits du message de navigation où les 6 bits de la santé d'un satellite sont utilisé : sous-trame #4 pages 2 à 5, 7 à 10 et 25. Dans le simulateur LACIME, la santé du satellite #32 a été remplacée par '111111'.

3.5 Conclusion

Les Tableaux 3.10 à 3.12 récapitulent les corrections qui ont été apportées aux paramètres de navigation et au message de navigation GPS L1.

Tableau 3.10 Résumé des corrections apportées aux paramètres de navigation

Paramètres de navigation	Corrections apportées
Position des satellites	<ul style="list-style-type: none"> Extrapolation du temps de transit La durée d'un échantillon est de 100 ms
Vitesse des satellites	La vitesse se calcule avec les paramètres orbitaux
Azimuth	L'azimuth se calcule avec le format ENU
Élévation	L'élévation se calcule avec le format ENU
Pseudo-distance	<ul style="list-style-type: none"> La correction de l'effet relativiste s'effectue à l'aide des paramètres orbitaux Le paramètre af_0 est constant pour le calcul du biais d'horloge
Fréquence Doppler	<ul style="list-style-type: none"> La durée d'un échantillon est de 100 ms Ajout d'une phase d'initialisation logicielle pour initialiser correctement ρ de l'échantillon précédent

Tableau 3.11 Résumé des corrections apportées aux variables de navigation

Variables	Corrections apportées
AI_SVS	Ne pas utiliser la table de correction du message de navigation
AntiSpoof_SVs	La configuration des satellites est à Bloc II/IIA/IIR
ERD_SVs	Le ERD est à 0x2A
IODC	La variable s'incrémente à tous les deux heures
IODE	La variable s'incrémente à tous les deux heures
SubFrameID	Calculée en fonction de l'heure courante de la simulation
t_oc	La variable change de valeurs à tous les deux heures
t_oe	La variable change de valeurs à tous les deux heures
TOW_Dec	Modification de la formule pour empêcher qu'elle devienne négative
URAIndex	Elle a été fixée à 0
WN_a	Représenté par modulo 256

Tableau 3.12 Résumé des corrections apportées au message binaire de navigation

Problèmes rencontrés	Corrections apportées
La fonction GenerateMsg() est trop longue et difficile à déverminer	La fonction GenerateMsg() a été découpée en sous-fonction dont chacune d'entre elles génèrent un mot d'une sous-trame
Mauvaise synchronisation des sous-frames	Chaque sous-trame commence par 0x22EAAA8A
Les bits réservés sont à '0'	Alterner par '1' et '0' chaque bit réservé consécutif
Aucun message spécial n'est envoyé	Transmettre le message spécial suivant au récepteur : MADE BY GSS LIMITED.
Les deux derniers bits avant la parité des mots #10 ne sont pas correctement générés	Les deux derniers bits avant la parité des mots #10 sont produits en fonction que les deux derniers bits de la sous-trame soient '00'
Perte des signaux de tous les satellites	Alterner successivement par '1' et '0' tous les paramètres d'almanach d'un satellite absent de la constellation
Les sous-frames sont générées en retard	La génération des sous-frames s'effectue six secondes plus tôt qu'auparavant

Ce chapitre a permis de décrire les corrections logicielles qui ont été effectuées sur les paramètres et le message binaire de navigation. Aussi, les formules permettant de calculer les paramètres de navigations ont été énoncées et corrigées. Avec les modifications apportées, les paramètres de navigation et le message binaire de navigation sont valides. Les paramètres de navigation enregistrés du simulateur LACIME ont une meilleure précision que SimGen ce qui explique qu'il y a une erreur bruitée. SimGen ne spécifie pas correctement l'état d'un satellite dysfonctionnel causant une différence dans le message de navigation du simulateur Spirent et LACIME.

Maintenant que les problèmes logiciels ont été résolus, le chapitre suivant va traiter des problèmes de la IF.

CHAPITRE 4

VALIDATION DE LA PARTIE IF DU SIMULATEUR LACIME

Ce chapitre permet d'identifier et de résoudre les problèmes IF du simulateur LACIME. Pour faire cela, il faut récupérer les données brutes décodées par le récepteur u-blox pour une simulation avec le simulateur Spirent et LACIME. Ensuite, le biais d'horloge et les C/N_0 s des satellites ont été comparés par la valeur obtenue d'une simulation Spirent versus LACIME. À propos des mesures de Doppler et de pseudo-distances reçues par le récepteur, elles ont été analysées par celles recueillies du logiciel SimGen de Spirent.

Les problèmes traités dans ce chapitre seront : l'Universal Time Coordinated (UTC), la génération de l'horloge principale, les NCOs, le sinus et le cosinus de la porteuse, le calcul du code et du bit courant du message de navigation.

4.1 Discussion des données brutes initiales décodées par le récepteur u-blox

Les résultats initiaux pour une simulation avec le simulateur Spirent (Rx/Spirent) et LACIME (Rx/LACIME) vus par le récepteur u-blox sont présentés à l'ANNEXE VII. La dérive du biais d'horloge est beaucoup trop grande rendant impossible pour un récepteur de calculer une position. Cette mesure indique la variation en Hz entre les horloges des satellites (qui est dans notre cas l'horloge du simulateur) et celle intégrée au récepteur u-blox. Elle varie d'un récepteur à l'autre dépendamment de sa qualité d'horloge employée. Il y a trois facteurs qui influencent la dérive d'une horloge : la précision de l'horloge principale du simulateur, le calcul du code et du bit courant du message de navigation. Aussi, le biais d'horloge (t_u) contient des sauts et ceci n'est pas un comportement attendu. La dérive de t_u et t_u feront l'objet d'une étude plus approfondie dans les prochaines sections du chapitre.

Il y a une trop grande différence entre Δf calculé par le récepteur et celui par SimGen. Comme la valeur de Δf a été corrigée au chapitre précédent, un problème matériel occasionne cette anomalie. L'horloge principale peut en être la cause ainsi que le NCO de la porteuse

qui a été conçu dans le FPGA du simulateur LACIME. Ce phénomène sera étudié plus minutieusement aux sections 4.4 et 4.5.

Le récepteur u-blox ne détecte aucun signal (C/N_0) du satellite #28. Pourtant, la fonctionnalité d'enregistrement indique que celui-ci est visible et simulé par la IF dès le début de la simulation. Il est possible que le simulateur LACIME n'ait pas le bon PRN rendant impossible le décodage de l'information transmise.

4.2 Problème de délai UTC

Il a été observé que l'UTC du simulateur LACIME est en avance de 14 secondes avec celui qui est affiché au récepteur. Pourtant, dans le message de navigation (sous-trame #4, page 18), il a été spécifié qu'il ne doit pas y en avoir. Selon la Figure 2.3, la condition « temps_simulation ≥ 14 » indique qu'aucune donnée n'est écrite dans la mémoire pendant les 13 premières secondes de la simulation ce qui cause un décalage dans le temps. Pour corriger ce problème, il a été nécessaire de modifier le processus #2 ainsi que d'ajouter une phase d'initialisation matérielle au simulateur. La Figure 4.1 est le morphogramme modifié du processus #2.

Avec le processus #2 modifié, dès qu'une donnée est disponible, elle est écrite immédiatement dans la mémoire. Il existait dans le simulateur LACIME deux mécanismes permettant de compter le temps écoulé : une minuterie logicielle et un compteur PC en VHDL. Le premier ne garantit pas que le temps configuré soit respecté. Il est possible qu'il y ait une différence de quelques millisecondes comparativement au compteur PC qui s'incrémente à chaque coup d'horloge. Pour avoir une meilleure précision, la minuterie logicielle a été supprimée et le temps de simulation est synchronisé à l'aide du compteur PC.

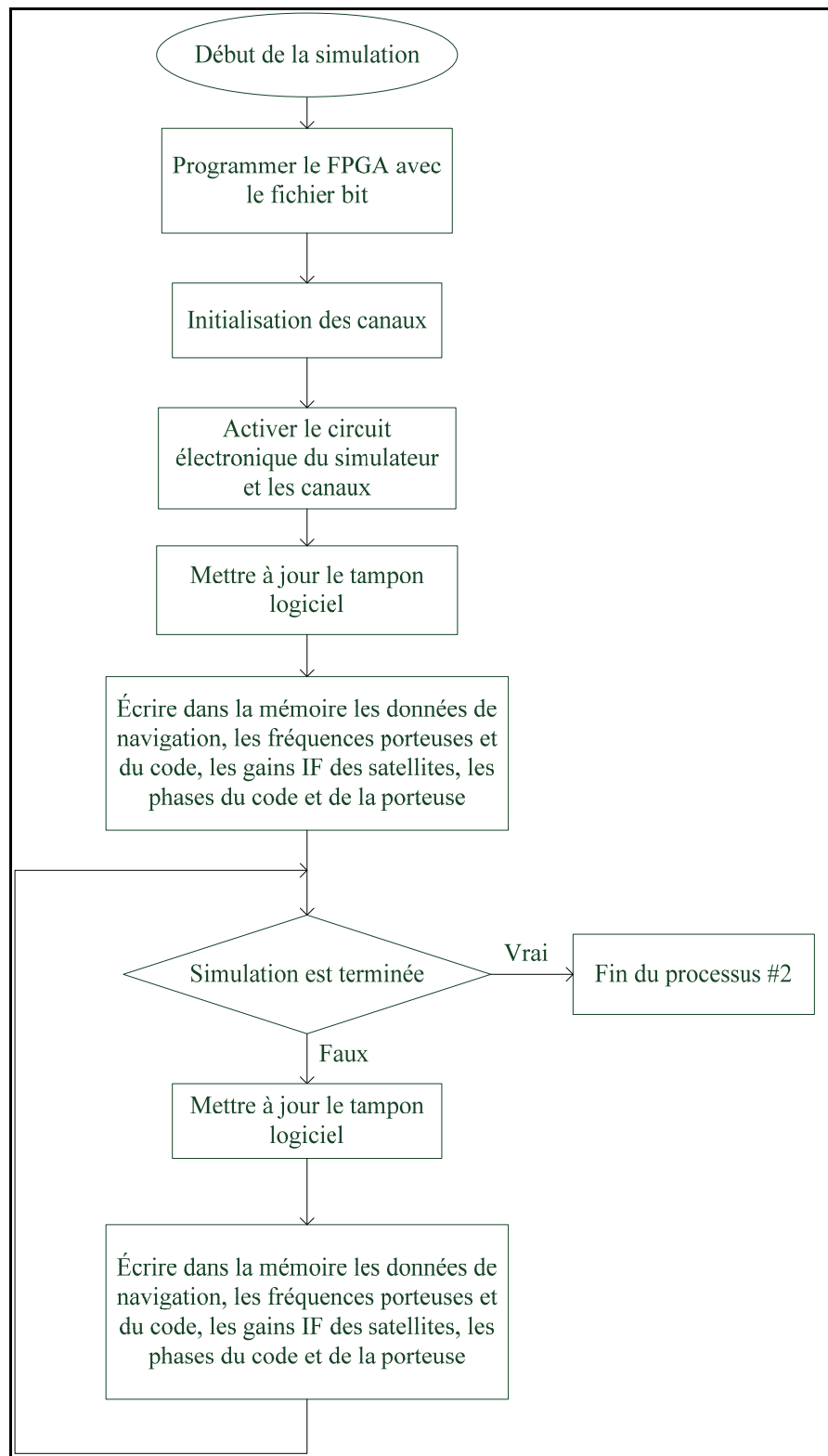


Figure 4.1 Morphogramme modifié du processus #2

Initialement, il n'y avait pas d'état d'initialisation en VHDL. Pourtant, cette opération devient nécessaire afin de donner assez de temps au logiciel d'enregistrer les premières données de navigation dans la mémoire et de permettre au FPGA de les récupérer. Alors, l'étape d'initialisation qui a été ajoutée dure pendant la première seconde de la simulation. Durant cette période, aucun signal de navigation n'est produit afin de laisser le temps approprié au système de s'initialiser. Ainsi, cela assure que tous les canaux seront synchronisés entre eux.

Donc, en modifiant le processus #2, en synchronisant le temps de simulation avec le compteur PC et en implémentant un état d'initialisation au FPGA, l'UTC du simulateur LACIME est identique à celui du récepteur u-blox.

4.3 Délais sur la génération des adresses XMEM

La mémoire XMEM enregistre les données utilisées par les NCOs du simulateur. Une valeur qui n'est pas lue au bon moment aura un impact important sur les phases produites. La mémoire XMEM est composée de deux modules principaux : le premier (maître) génère les adresses mémoires et le second (esclave) (voir Figure 4.2) mémorise dans un registre la fréquence courante utilisée par le NCO.

Initialement, le générateur d'adresse XMEM produisait les adresses en fonction du numéro de l'échantillon courant. Il y avait un décalage de quelques périodes d'horloge sur le changement de fréquences des NCOs. Pour régler ce problème, la génération d'adresses doit s'effectuer en avance d'un échantillon. Ainsi, lorsqu'une donnée expire (à toutes les 10 ms), la suivante sera déjà disponible par le biais du registre temporaire du module esclave.

Cette modification a été testée et il n'y a pu de délais lors du changement des fréquences du NCO.

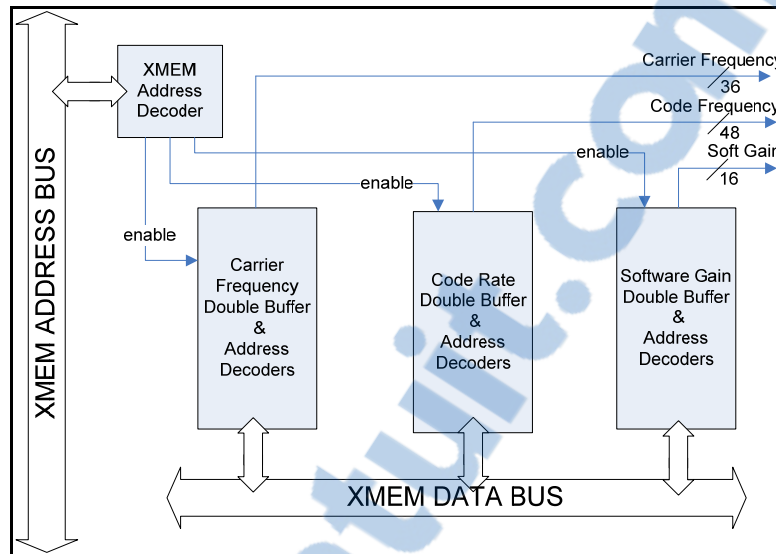


Figure 4.2 Représentation du module esclave de la mémoire XMEM
Tirée de Sauriol (2007, p. 6)

4.4 Problème lors de la génération de l'horloge principale

Une des fonctions de la carte RF est de produire l'horloge principale du système (100 MHz). La Figure 4.3 montre la forme de l'onde de l'horloge principale avant et après la modification.

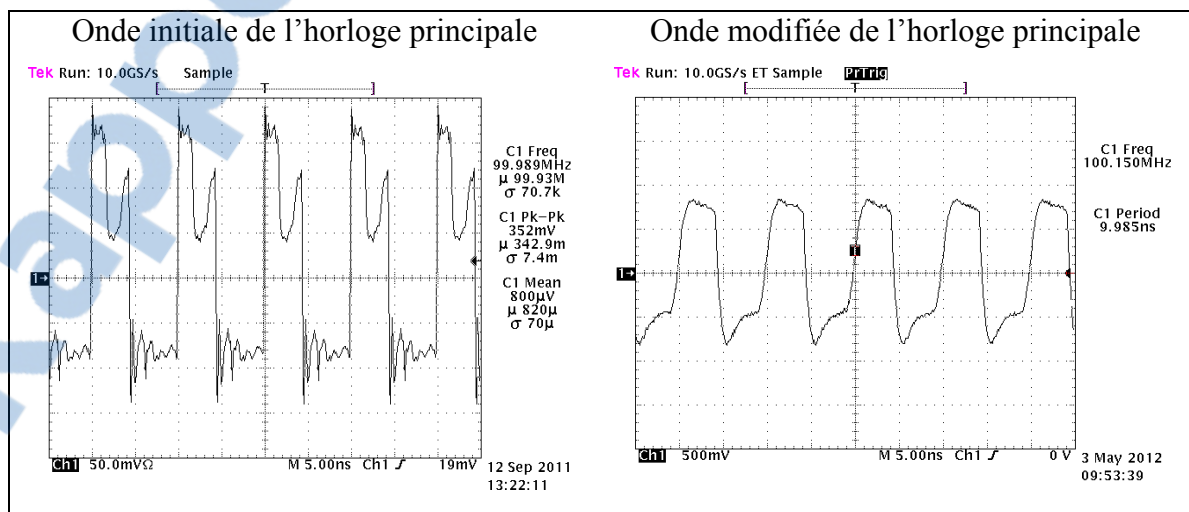


Figure 4.3 L'onde initiale et modifiée de l'horloge principale

L'horloge initiale comporte de grandes oscillations en amplitude ce qui affecte le biais d'horloge du récepteur. Cela est dû à un problème de réflexion sur la carte RF causant des décalages d'impédance. Un récepteur évalue p à partir des délais de propagation qui sont produits par le simulateur ayant comme référence une horloge. Ainsi, une mauvaise génération de l'horloge principale peut empêcher un récepteur de trouver une position.

Comme il a été mentionné à la section 2.1.3.3, il est nécessaire de calibrer manuellement l'OCXO pour avoir une fréquence adéquate de 100 MHz. Ceci a été effectué il y a quelques années et l'usure des composantes électroniques a pu décalibrer l'horloge. Pour régler temporairement le problème, on emploie une horloge externe. De cette manière, on s'assurera de sa stabilité pour cette phase de déverminage. La Figure 4.4 représente le montage de l'horloge principale.

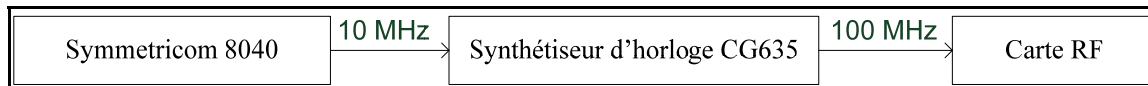


Figure 4.4 Montage de la génération de l'horloge

Le Symmetricom 8040 est un synthétiseur de fréquence au rubidium et il fournit seulement une onde sinusoïdale de 10 MHz. Le synthétiseur d'horloge CG635 se synchronise avec la sortie du synthétiseur de fréquence pour produire une horloge de 100 MHz. La nouvelle onde de l'horloge a été vérifiée à l'aide d'un oscilloscope (voir Figure 4.3) et celle-ci a bien une fréquence de 100 MHz avec aucune oscillation importante en amplitude. Alors, l'horloge générée à partir du synthétiseur d'horloge est de meilleure qualité et elle sera une solution temporaire au problème de l'OCXO.

4.5 Présentation et correction des erreurs sur l'architecture des NCOs

La Figure 4.5 démontre l'interaction initiale entre le logiciel et le FPGA du simulateur LACIME.

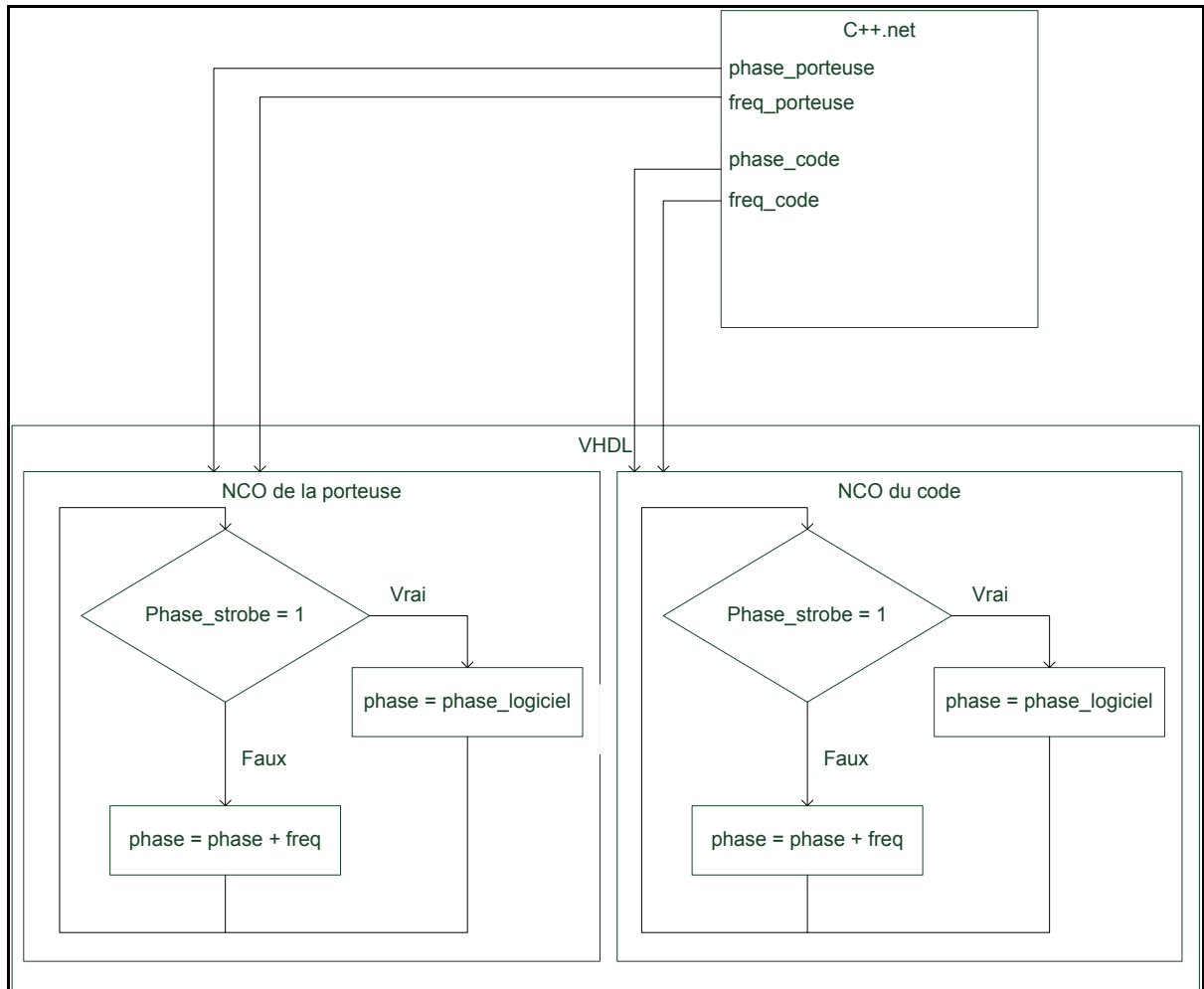


Figure 4.5 Interaction initiale des NCOs entre le logiciel et le FPGA

Les NCOs sont des intégrateurs qui calculent la phase d'un signal en radian dont elle se situe entre 0 et 2π . À chaque front montant de l'horloge principale, la phase s'incrémente selon l'équation (4.1) :



$$\varphi_{NCO}^+ = \text{mod}(\varphi_{NCO}^- + \Delta f * \Delta t, 2\pi) \quad (4.1)$$

où :

- φ_{NCO}^+ est la phase du NCO au temps courant en radian,
- φ_{NCO}^- est la phase au dernier front montant de l'horloge en radian,
- Δf est la fréquence Doppler en Hz,
- Δt est le temps de rafraîchissement de Δf en seconde (1 sec).

Un cycle est complété lorsque la phase égale à 2π . Les fréquences à syntoniser par les NCOs sont évaluées par le logiciel et elles sont rafraîchies à toutes les secondes, mais l'adressage de la mémoire XMEM où elles sont enregistrées permettrait de les mettre à jour à toutes les 10 ms. Comme la phase a un nombre de bits limités, il est certain que la position simulée dévie au fil du temps (Belzile, 2006). Afin d'éviter ce problème, à toutes les secondes, le drapeau Phase_strobe est à '1' pour indiquer qu'une correction de la phase est nécessaire. Ainsi, la phase corrigée qui est calculée par le programme C++ et écrite dans la mémoire FSB vient écraser l'ancienne valeur.

4.5.1 Erreur de précision du NCO

À la section 4.5, il a été mentionné que les NCOs accumulent une erreur avec le temps, car leur nombre de bits est restreint. La Figure 4.6 représente la phase ($\varphi_{\text{porteuse}}$) obtenue d'une simulation ModelSim et son erreur ($\varphi_{\text{err_porteuse}}$) pour une porteuse de 30 MHz échantillonné à 100 MHz ($F_s=100$ MHz).

La droite de régression indique que l'erreur de la phase de la porteuse sera approximativement de 1,8286 mrad après une seconde (période de correction des phases). Comme les 15 premiers bits de 36 bits au total de la phase sont employés pour trouver la valeur du sinus et du cosinus de la porteuse, il faut s'assurer que $\varphi_{\text{err_porteuse}}$ ne les affectera pas. Ainsi, pour trouver l'erreur maximale permise, il suffit de mettre en binaire tous les bits de la phase du 15^{ème} au 36^{ème} à '1' et de convertir en radian le nombre obtenu en divisant par $2^n/2\pi$ où n est le nombre de bits de la phase de la porteuse (36 bits). En effectuant ce calcul,

une erreur maximale de 0,19175 mrad est obtenue. Pour décroître $\phi_{\text{err_porteuse}}$, il devient nécessaire de corriger les phases plus souvent. Il existe déjà un système d'adressage de la mémoire qui rafraîchit les valeurs à toutes les 10 msec. Celui-ci sera réutilisé afin de ne pas dupliquer ce qui existe. Les corrections de phase seront enregistrées dans la mémoire XMEM au lieu de FSB suivant l'adressage de la Figure 2.9. Les numéros d'identification du tampon pour la phase de la porteuse et du code sont respectivement 3 et 4. Une fréquence de correction des phases de 100 Hz aura une erreur résiduelle maximale de 0,018286 mrad pour une porteuse de 30 MHz selon la droite de régression de la Figure 4.6.

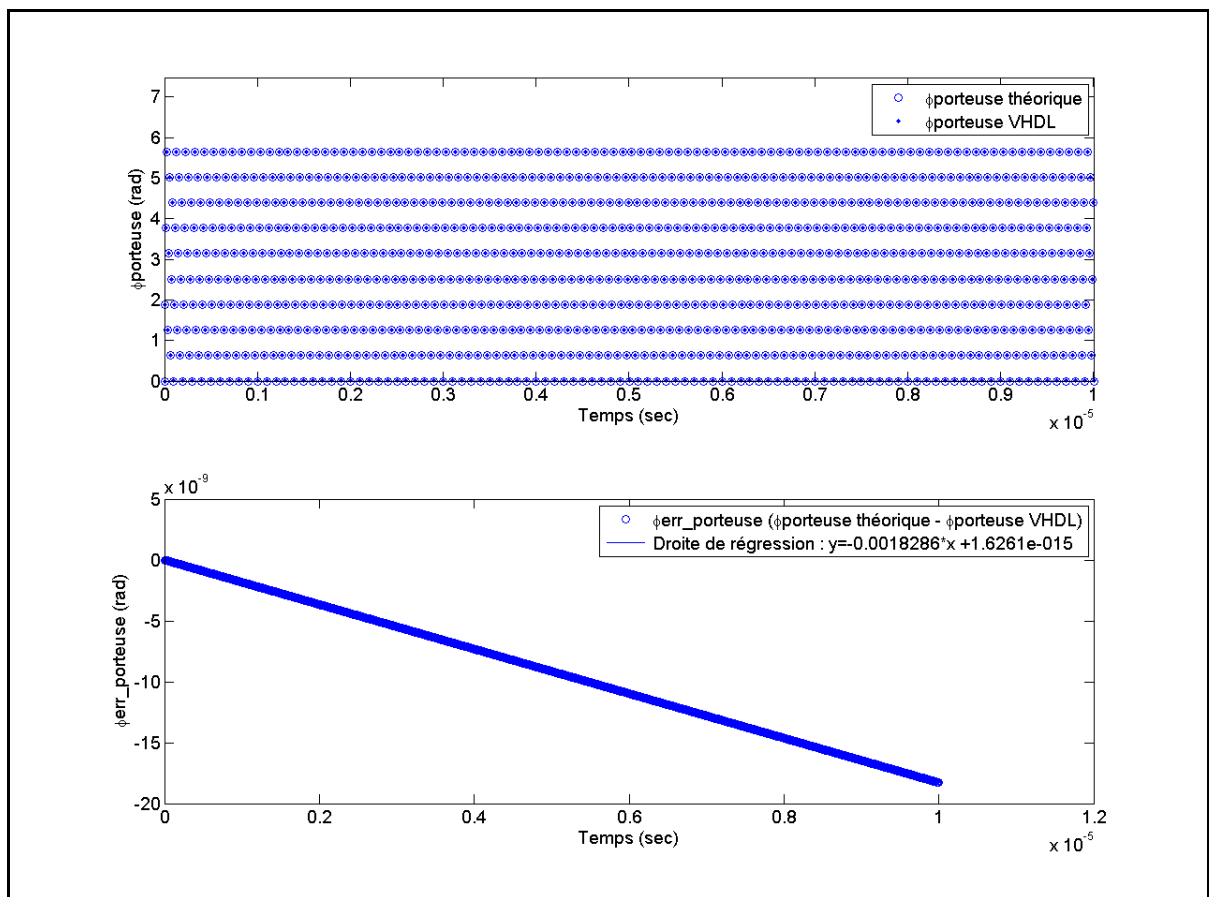


Figure 4.6 ϕ_{porteuse} et $\phi_{\text{err_porteuse}}$ d'une porteuse de 30 MHz avec $F_s=100$ MHz

4.5.2 Calcul des phases corrigées du simulateur LACIME

La Figure 4.7 montre la phase corrigée calculée en C++.NET de la porteuse, du code (ϕ_{code}) et du message de navigation (ϕ_{mess}) du satellite #4 avant et après la modification apportée. La fréquence d'échantillonnage est de 100 MHz pour ϕ_{porteuse} et ϕ_{code} et de 1000 Hz pour ϕ_{mess} .

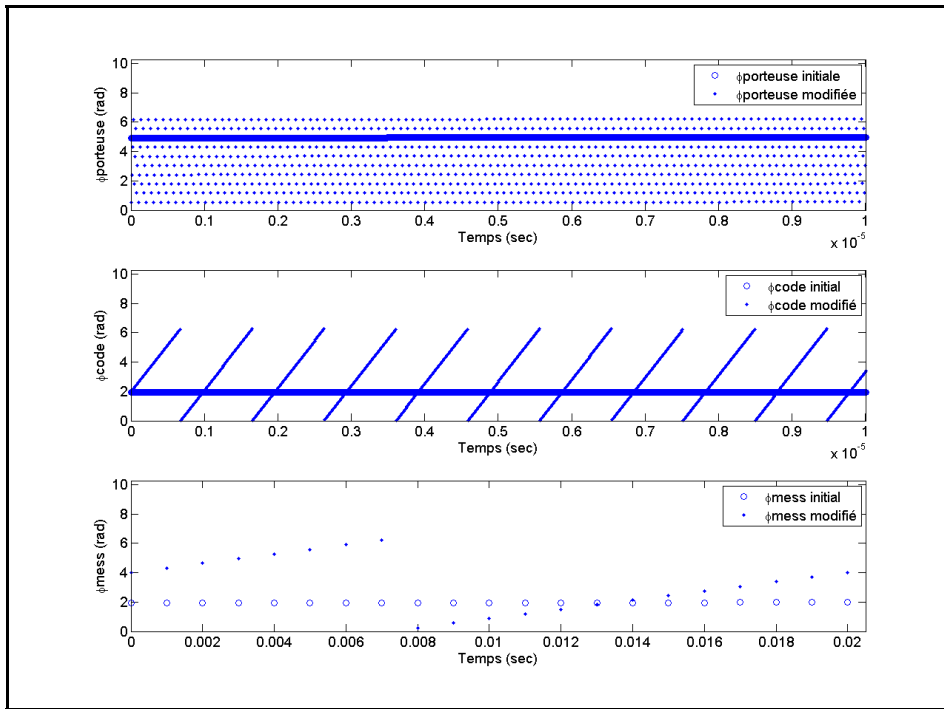


Figure 4.7 ϕ_{porteuse} , ϕ_{code} et ϕ_{mess} du satellite #4 calculé par le logiciel avec $F_s=100$ MHz (ϕ_{porteuse} et ϕ_{code}) et $F_s=1000$ Hz (ϕ_{mess})

Les phases initiales démontrent que la génération des périodes ne se fait pas correctement. En effet, les fréquences centrales ne sont pas représentées. Les satellites émettent le code à 1,023 MHz tandis que la porteuse IF se propage à 30 MHz. À chaque 20 ms, ϕ_{mess} doit compléter un cycle ce qui n'est pas le cas dans la phase initiale du message de navigation. Un récepteur va recevoir les signaux GPS à la fréquence émise par les satellites \pm la fréquence Doppler générée par le satellite et l'utilisateur en mouvement. Alors, l'équation (4.2) (Petovello et O'Driscoll, 2010) tient compte de ces deux paramètres pour calculer les phases dont :

$$\varphi(t) = \text{mod} \left(f_0 * tow - \frac{\rho(t)}{\lambda}, 1.0 \right) * 2\pi \quad (4.2)$$

où :

- f_0 est la fréquence d'émission sans le décalage fréquentiel (30 MHz pour la porteuse, 1,023 MHz pour le code et 50 Hz pour le message de navigation),
- tow est le temps de la semaine en secondes,
- $\rho(t)$ est la pseudodistance du satellite en mètre,
- λ est la longueur d'onde en mètre (0,1902936728 m pour la porteuse, 293,0522561095 m pour le code et 6 000 000 m pour le message de navigation).

En utilisant l'équation (4.2) pour calculer les phases, toutes les fréquences émises par les satellites et les Dopplers sont bien représentées.

4.5.3 Calcul des fréquences porteuses et du code

Maintenant que la phase de la porteuse et du code sont calculées correctement, elles seront utilisées pour valider les phases des NCOs. Ainsi, il sera possible de déterminer si les fréquences des NCOs sont bien calculées. $\varphi_{\text{err_porteuse}}$ et l'erreur de phase du code ($\varphi_{\text{err_code}}$) correspondent à la différence entre la phase calculée du logiciel et la phase du NCO avant de la corriger ($\varphi_{\text{logiciel}} - \varphi_{\text{NCO}}$). Un module de test en VHDL a été développé pour stocker temporairement dans la mémoire FSB φ_{NCO} juste avant d'appliquer la correction de la phase. À toutes les secondes, le logiciel enregistre φ_{NCO} dans un fichier.

Les Figures 4.8 et 4.9 montrent $\varphi_{\text{err_porteuse}}$ et $\varphi_{\text{err_code}}$ du satellite #4 avec $F_s=1\text{Hz}$. L'ANNEXE VIII contient d'autres résultats de $\varphi_{\text{err_porteuse}}$ et $\varphi_{\text{err_code}}$ de d'autres satellites.

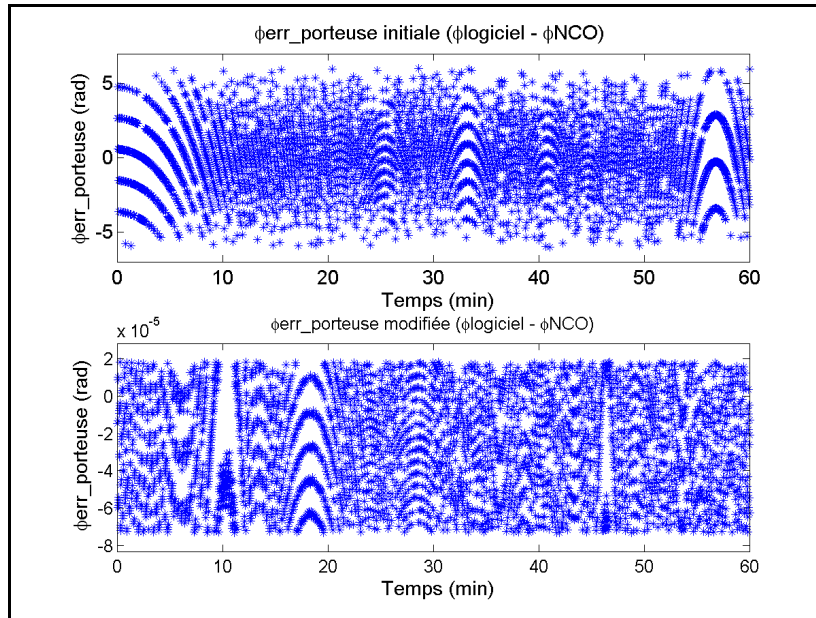


Figure 4.8 $\phi_{err_porteuse}$ initiale et modifiée du satellite #4 avec $F_s=1\text{Hz}$

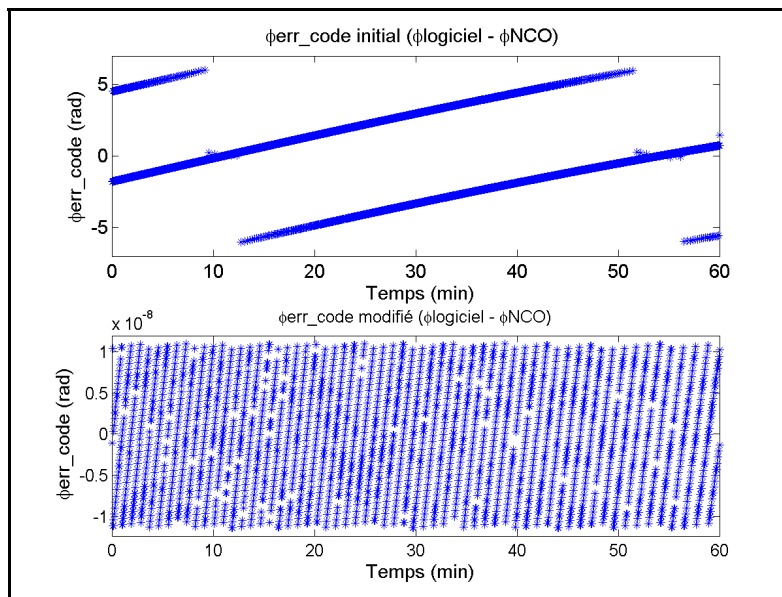


Figure 4.9 ϕ_{err_code} initial et modifié du satellite #4 avec $F_s=1\text{Hz}$

Les phases initiales de la porteuse et du code du NCO ne correspondent pas avec celle calculée du logiciel. Donc, les fréquences syntonisées par les NCOs ne sont pas les bonnes.

Initialement, celles-ci étaient le Doppler du satellite. L'équation (4.3) montre comment calculer la fréquence du NCO pour le code et la porteuse :

$$f_{NCO} = \frac{Nb_cycle_2 - Nb_cycle_1}{\frac{f_{clk}}{f_{echan_IF}}} = \frac{Nb_cycle_2 - Nb_cycle_1}{1000000} \quad (4.3)$$

où :

- $Nb_cycle_2 - Nb_cycle_1$ est le nombre de cycle que le NCO doit effectuer entre deux échantillons IF,
- f_{clk} est la fréquence de l'horloge (100 MHz),
- f_{echan_IF} est la fréquence des échantillons IF (100 Hz).

L'équation (4.3) démontre qu'il est nécessaire de trouver entre deux échantillons successifs (10 ms) le nombre de cycles que le NCO doit atteindre durant la période de validité d'un échantillon. On le trouve en réutilisant l'équation (4.2), mais sans l'opération modulo (mod). Puisque le logiciel calcule $p(t)$ à tous les 100 ms, une interpolation linéaire de celui-ci devient essentielle pour obtenir une valeur au 10 ms. Comme $p(t)$ ne suit pas une fonction linéaire, son interpolation introduira une erreur sur sa valeur exacte. En choisissant un petit intervalle d'interpolation (100 ms), l'erreur ainsi générée devient négligeable. Dans le but de bien suivre le Doppler des satellites, il est nécessaire de mettre à jour les fréquences des NCOs à tous les 10 ms pour avoir respectivement une erreur maximale de la position et du Doppler de 10 mm et de 5,26 mHz (Belzile, 2006). Cela se faisait toutes les secondes et ceci a été changé pour respecter cette spécification.

En augmentant la fréquence de mise à jour des NCOs et en calculant les fréquences des NCOs selon l'équation (4.3), on obtient une valeur résiduelle maximale tout satellite confondu de 73,128 μ rad pour la phase de la porteuse et de 11,362 nrad pour la phase du code. Donc, ces modifications ont permis de diminuer considérablement ϕ_{err_code} et $\phi_{err_porteuse}$.

4.5.4 Calcul de la phase du message de navigation

La phase du message de navigation représente une fraction d'un bit du message de navigation. Cette composante est essentielle dans un simulateur afin de changer au bon moment le bit simulé du message de navigation. Aussi, elle permet de calculer le temps GPS exact en se basant sur la structure du message.

Afin de vérifier le bon fonctionnement de la génération de la phase IF du message de navigation, il faut calculer l'erreur de la phase du message de navigation ($\phi_{\text{err_mess}}$). Celle-ci est la différence entre la phase calculée par le logiciel (ϕ_{logiciel}) et celle du FPGA ($\phi_{\text{mess_FPGA}}$). Voici $\phi_{\text{err_mess}}$ initiale et modifiée du satellite #4 avec $F_s=1\text{Hz}$ (voir ANNEXE VIII pour plus de détails).

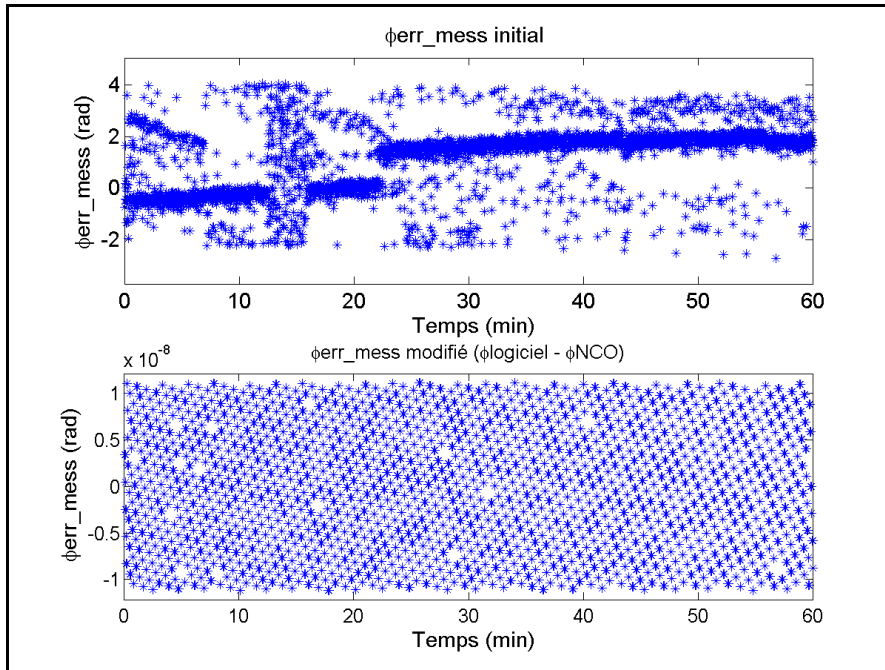


Figure 4.10 $\phi_{\text{err_mess}}$ initiales et modifiée du satellite #4 avec $F_s=1\text{Hz}$

La phase du logiciel du message de navigation ne coïncidait pas avec celle du FPGA. Dans le FPGA, ϕ_{mess} s'incrémentait de 1 lorsqu'il y avait un changement de bit du code. Quand celui-ci atteignait le ratio de la fréquence du code sur la fréquence du message de navigation

(20 460), un cycle de ϕ_{mess} venait de se terminer. Cette manière de fonctionner n'était pas la bonne. Afin de produire correctement la phase IF du message de navigation, il est nécessaire d'ajouter un NCO et sa fréquence à syntoniser est trouvée à l'aide de l'équation (4.3). Les corrections des phases du message de navigation et les fréquences du NCO seront enregistrées dans la mémoire XMEM selon l'adressage de la Figure 2.9. Leur numéro de tampon est respectivement 5 et 6.

En ajoutant un NCO pour la génération en matérielle de la phase du message de navigation, on obtient une erreur maximale de 11,158 nrad permettant de conclure que ϕ_{mess} est généré correctement.

4.6 Calcul de l'index de la position courante du code du satellite

Cette partie permettra de valider l'index du code courant du satellite calculé par le logiciel et par le FPGA. Celui-ci est compris entre 0 et 1023, car le code des satellites contient 1024 bribes. La Figure 4.11 démontre la position du code courant initiale et modifiée calculée par le logiciel pour le satellite #4.

La fréquence du code (1.023 MHz) n'a pas été respectée selon le graphique du code initial de la Figure 4.11. En effet, pendant une période de répétabilité des 1023 bribes qui doit durer approximativement 1 ms, le code n'a pas changé. Donc, les équations et évaluent respectivement l'index du code courant avant et après la corrections apportées.

$$code(t) = (\text{arrondir}) \bmod \left(-\frac{\rho(t)}{\lambda_{\text{code}}}, 1023 \right) \quad (4.4)$$

$$code(t) = (\text{troncation}) \left(\bmod \left(f_{\text{rép_bribe}} * tow - \frac{\rho(t)}{\lambda_{\text{rép_bribes}}}, 1.0 \right) * 1023 \right) \quad (4.5)$$

où :

- $f_{\text{rép_bribes}}$ est la fréquence de répétabilité des 1023 bribes (1000 Hz),
- t_{ow} est le temps de la semaine (sec),
- λ_{code} est la longueur d'onde du code (293,0525 m),
- $\lambda_{\text{rép_bribes}}$ est la longueur d'onde de la répétabilité des 1023 bribes (299792,458 m).

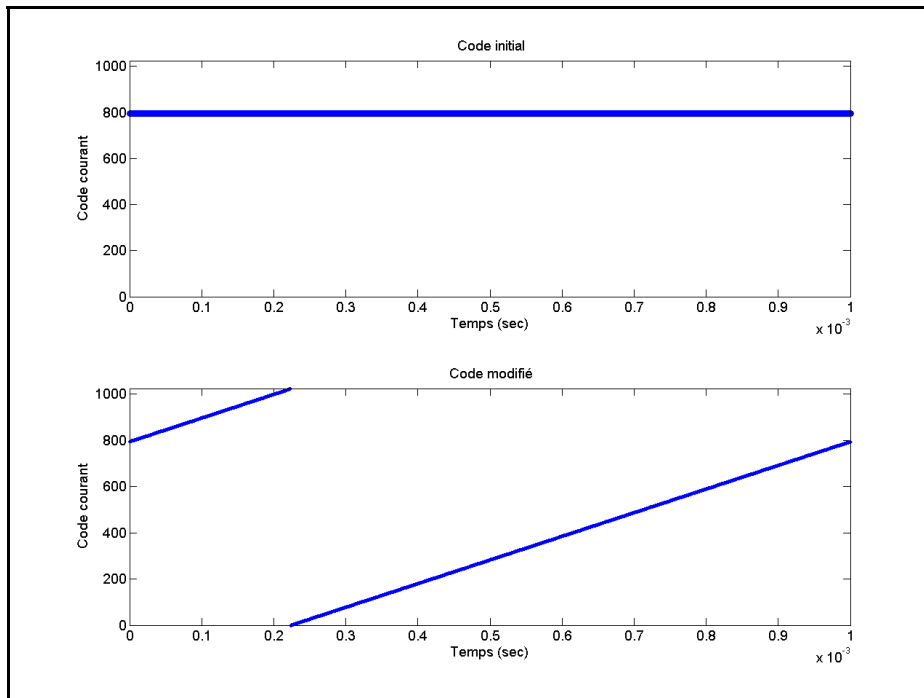


Figure 4.11 L'index du code courant calculé par le logiciel pour le satellite #4

Puisque maintenant l'index du code courant du satellite est correctement calculé par le logiciel, cette valeur sera utilisée pour valider l'index du code courant du FPGA. L'erreur de l'index du code (err_code) est la différence entre l'index de la position du code courant calculé par le logiciel et celui du FPGA ($\text{Code logiciel} - \text{Code FPGA}$). La mesure Code FPGA est obtenue par un module de test en VHDL qui écrit à toutes les secondes dans la mémoire FSB la valeur de l'index du code. Celle-ci est lue et enregistrée dans un fichier par le logiciel. La Figure 4.12 montre err_code initiale et modifiée avec $F_s=1$ Hz (voir ANNEXE IX pour plus de détails).

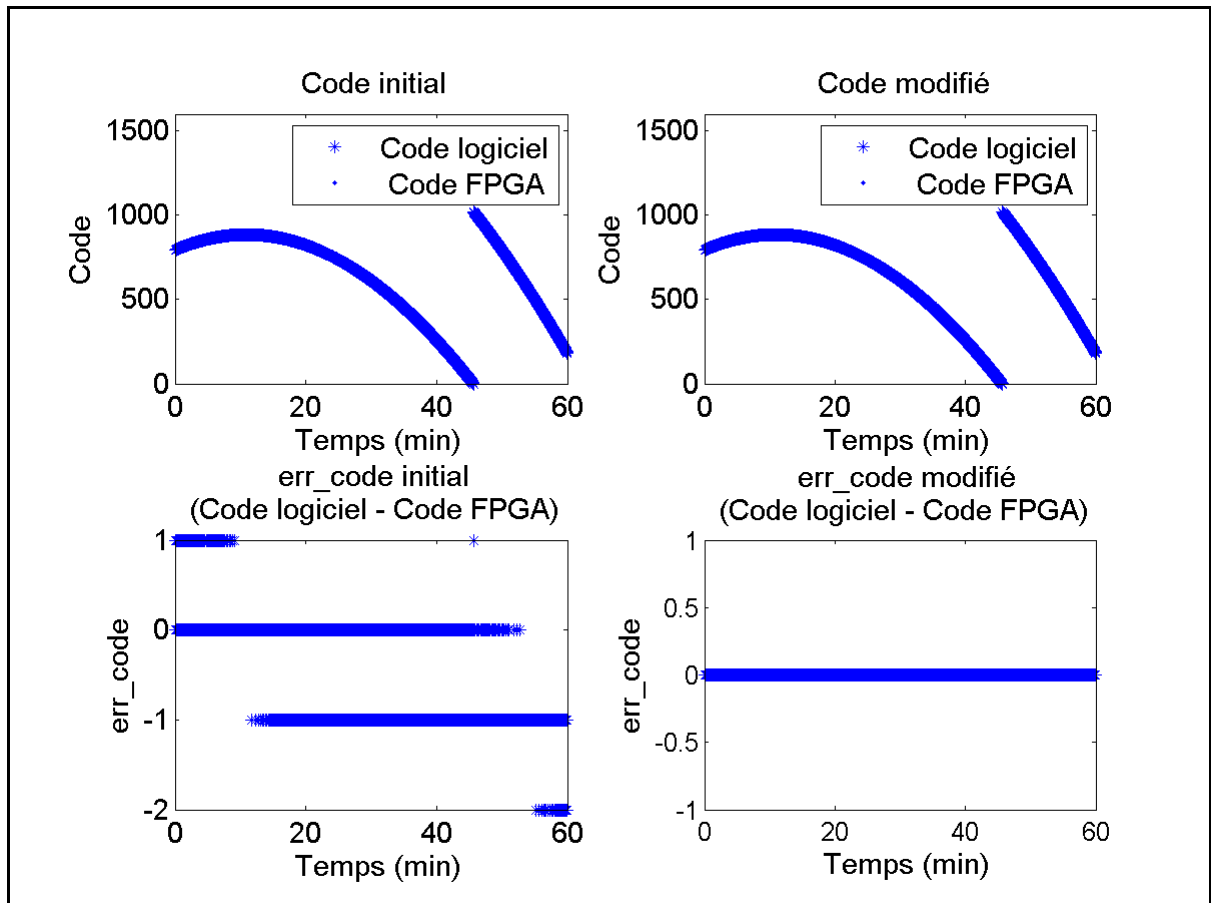


Figure 4.12 err_code initiale et modifiée du satellite #4 avec $F_s=1\text{Hz}$

Initialement, le code logiciel n'a aucune correspondance avec celui du FPGA. Durant l'initialisation d'un canal, le logiciel calcule et stocke dans la mémoire FSB le code du satellite à simuler. Ensuite, le FPGA change le bit courant du code lorsque le NCO du code a complété un cycle. Le module VHDL PCLK_debouncer détecte les débordements de la phase du code. Cette fonctionnalité est défectueuse et causait un décalage sur le code. PCLK_debouncer a été remplacé par une sortie d'un bit au module NCO. Celle-ci est à '1' lorsque la phase du NCO a subi un débordement sinon '0'.

En utilisant l'équation pour calculer l'index du code, en ajoutant une sortie qui détecte les débordements sur la phase au module NCO et en corrigeant les erreurs sur la phase du code énuméré aux sections 4.5.2 et 4.5.3, il n'y a plus d'erreur sur la position courante du code.



4.7 Calcul de l'index de la position courante du bit du message de navigation

Cette partie permettra de vérifier si les changements de bit du message de navigation surviennent au bon moment. Le calcul logiciel et FPGA de l'index du bit courant du message de navigation seront vérifiés. La Figure 4.13 montre l'index du bit courant du message de navigation du satellite #4 calculé par le logiciel initialement et après la modification. Comme il y a 300 bits dans le message de navigation, celui-ci est compris entre 0 et 299.

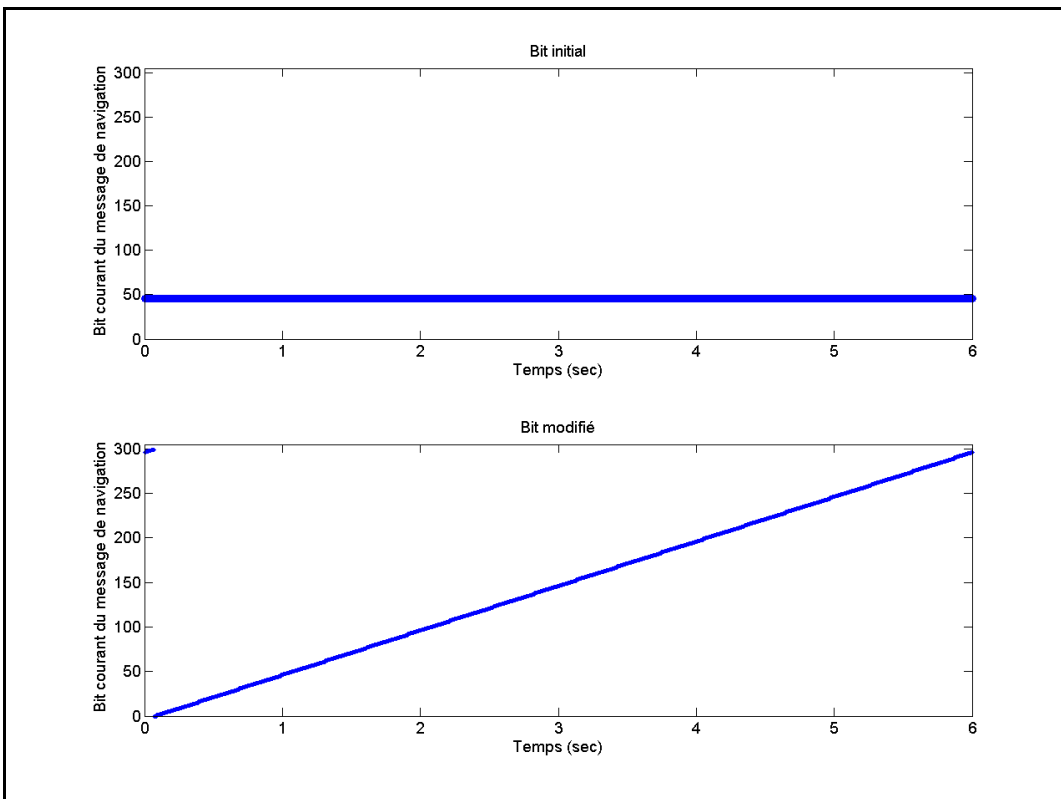


Figure 4.13 L'index de la position courante du bit du message de navigation du satellite #4

La courbe initiale de la position courante du bit du message de navigation est constante à 46. Cela n'est pas valide parce que les changements du bit du message de navigation surviennent à tous les 20 ms. L'équation (4.6) permet de le corriger :

$$bit_nav(t) = (troncation) \left(\text{mod} \left(f_{s\text{ous-trame}} * tow - \frac{\rho(t)}{\lambda_{s\text{ous-trame}}}, 1.0 \right) * 300 \right) \quad (4.6)$$

où :

- $f_{s\text{ous-trame}}$ est la fréquence d'une sous-trame (approximativement 0,166667 Hz, 6 secondes),
- tow est le temps de la semaine (sec),
- $\lambda_{s\text{ous-trame}}$ est la longueur d'onde de la sous-trame (1798754748 m).

Pour vérifier l'index du bit courant du message de navigation du FPGA, la valeur calculée par le logiciel sera utilisée puisqu'elle a été validée précédemment. L'erreur du bit du message de navigation (err_mess) est définie comme étant la soustraction entre la valeur calculée par le programme C++.NET (Mess logiciel) et celle obtenue du FPGA (Mess FPGA). La Figure 4.14 montre err_mess initiale et modifié du satellite #4 avec $F_s=1$ Hz (voir ANNEXE VIII pour plus de détails).

Il a été mentionné à la section 4.5.4 que la phase du message de navigation n'était pas générée correctement. Dans ces circonstances, il est tout à fait normal d'avoir une erreur sur la position du bit courant du message de navigation. Initialement, le module VHDL GNSS_Clock_Tree permettait d'indiquer à quel moment un changement de bit est nécessaire. Puisque ϕ_{mess_FPGA} est calculé par un NCO, ce module n'est plus utilisable. Il a été remplacé par une sortie $o_overflow$ du module NCO et celle-ci est à '1' lorsqu'on doit incrémenter le pointeur du bit courant du message de navigation.

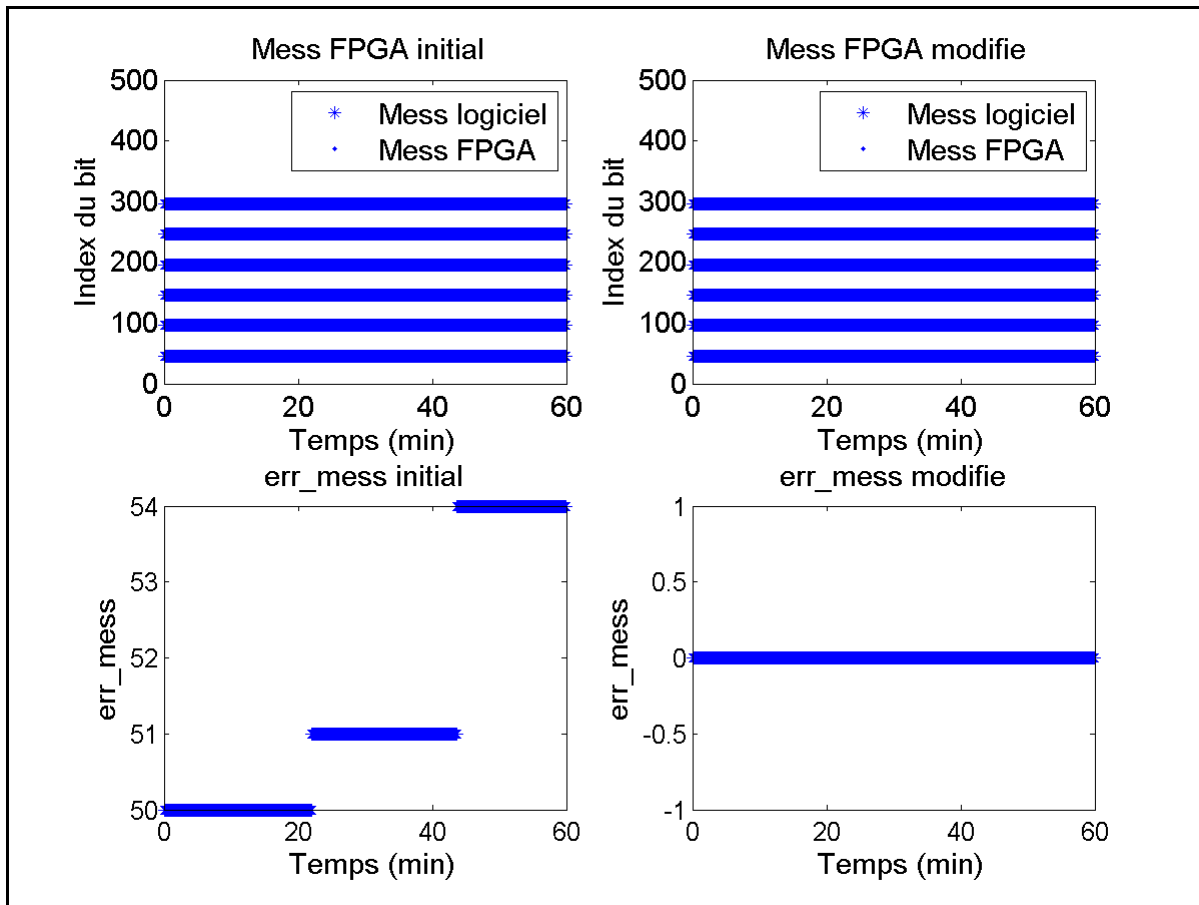


Figure 4.14 err_mess initiale et modifiée du satellite #4 avec $F_s=1$ Hz

En effectuant les corrections sur φ_{mess} énumérées à la section 4.5.2 et 4.5.4, en changeant le bit du message de navigation lorsqu'un cycle vient de se terminer et en utilisant l'équation (4.6) pour évaluer la position courante du bit du message de navigation, il n'y a aucune erreur sur l'index du bit courant du message de navigation.

4.8 Calcul du sinus et du cosinus de la porteuse

Les sinus et les cosinus de la porteuse sont calculés à partir des 15 bits les plus significatifs de $\varphi_{\text{porteuse}}$ et ils permettent de générer I et Q (voir Figure 2.7). Ceux-ci sont enregistrés dans une table de vérité. Le Tableau 4.1 indique les valeurs correspondantes de sinus et de cosinus de la porteuse en fonction de la phase.

Tableau 4.1 Sinus et cosinus de la phase de la porteuse
Tirée de Kaplan (1996, p.131)

Degrée	Sortie du NCO	SIN (complément à 2)	COS (complément à 2)
0	100...	000	011
45	101...	010	010
90	110...	011	000
135	111...	010	110
180	000...	000	111
225	001...	110	110
270	010...	111	000
315	011...	110	010

Les deux bits les plus significatifs de la phase déterminent le quadrant. Initialement, le quadrant choisi n'était pas le bon en fonction de la sortie du NCO (voir Tableau 4.2) et cela a été corrigé afin qu'un récepteur puisse décoder correctement la phase de la porteuse.

Tableau 4.2 Quadrant de la phase de la porteuse

Sortie du NCO (2 MSB)	Quadrant initial	Quadrant modifié
00	1	3
01	2	4
10	3	1
11	4	2

Les données enregistrées dans la table de vérité sont déjà en complément à 2. Initialement, lorsque celles-ci étaient négatives, le programme VHDL les mettait en complément à 2. La ligne de code qui convertissait ses valeurs en complément à 2 a été commentée. Alors, les valeurs de sinus et de cosinus de la porteuse sont dorénavant les bonnes.

4.9 Discussion des résultats IF finaux

Les résultats vus du récepteur u-blox avec le simulateur LACIME sont à l'ANNEXE VII. En utilisant le générateur d'horloge Symmetricom 8040 ayant une précision de $\pm 10^{-12}$, Δf_{err} décodé par le récepteur u-blox est maintenant de 300 Hz. Ainsi, la stabilité de l'horloge et la

correction de $\varphi_{\text{err_porteuse}}$ a permis d'obtenir un Doppler au récepteur qui se rapproche de celui calculé par SimGen.

Une dérive de t_u supérieur à 0,001 ppm rend impossible au récepteur u-blox de trouver une position. Les corrections qui ont été effectuées sur l'horloge et sur le calcul de la phase du code ont permis d'avoir une dérive de t_u de 0,00024 ppm. Le récepteur u-blox n'a pas trouvé correctement le temps GPS parce qu'il y avait une faute sur le code simulé du simulateur LACIME. Un err_code d'une seule position équivaut à un ρ_{err} approximatif de 291 mètres. Un ρ_{err} plus élevé que 100 kilomètres rend impossible à un récepteur de converger vers une position. Les résultats pour une simulation d'une heure démontrent que ρ_{err} modifié du simulateur LACIME est inférieur à celui du simulateur Spirent. Ainsi, la précision de la position calculée par le récepteur u-blox sera plus précise avec le simulateur LACIME. Les C/N_0 s des satellites du simulateur Spirent et LACIME ne correspondent pas entre eux, car la puissance des signaux de navigation n'a pas été calibré sur la carte RF du simulateur LACIME.

4.10 Conclusion

Le Tableau 4.3 synthétise les corrections qui ont été apportées à la IF du simulateur LACIME.

Ce chapitre a permis de régler les problèmes reliés à la IF dont la dérive d'horloge, le biais d'horloge, les fréquences Doppler et les pseudo-distances des satellites vus par un récepteur. Les résultats ainsi obtenus permettront à un récepteur de trouver une position et de la maintenir pendant plusieurs heures.

Les problèmes logiciels et matériels du simulateur LACIME sont maintenant résolus. Le chapitre suivant va se concentrer sur l'analyse des performances en statique et en dynamique du simulateur LACIME.

Tableau 4.3 Résumé des corrections apportées à la IF du simulateur LACIME

Problèmes rencontrés	Corrections apportées
L'UTC du simulateur est en avance de 14 secondes avec le récepteur	<ul style="list-style-type: none"> • Utilisation du compteur PC au lieu d'une minuterie logiciel • Stocker dans la mémoire FIFO les données de navigation dès qu'elles sont disponibles • Ajout d'un état d'initialisation au FPGA
Il y a une trop grande erreur de précision des NCOs	<ul style="list-style-type: none"> • Les corrections de la phase surviennent à toutes les 10 ms
Il y a une grande erreur des phases des NCOs du code et de la porteuse	<ul style="list-style-type: none"> • Modification de la formule permettant de calculer les phases • Les fréquences syntonisées par les NCOs sont calculées à partir des mesures de phase au lieu de la fréquence Doppler • Les fréquences syntonisées par les NCOs sont mises à jour à toutes les 10 ms
Il y a une grande erreur de la phase du message de navigation	<ul style="list-style-type: none"> • Les phases du message de navigation sont calculées avec un NCO
Le code et le bit courant du message de navigation ne sont pas correctement évalués	<ul style="list-style-type: none"> • Modification de la formule permettant de calculer le code et le bit courant du message de navigation • Ajout d'une sortie au module du NCO indiquant qu'il faut changer de bit
La sélection du quadrant pour le calcul du sinus et du cosinus de la porteuse n'a pas été faite correctement	<ul style="list-style-type: none"> • Associer les deux premiers bits de la phase de la porteuse avec un quadrant selon le Tableau 4.2
Grande oscillation en amplitude de l'horloge système	<ul style="list-style-type: none"> • Remplacer l'horloge de la carte RF par une horloge externe au rubidium

CHAPITRE 5

PERFORMANCE ET ANALYSE DU SIMULATEUR LACIME

Les problèmes majeurs du simulateur LACIME ont maintenant été corrigés. Le chapitre 5 du mémoire va se concentrer sur la caractérisation des performances du simulateur LACIME. Celui-ci est composé de deux parties : performance et analyse en statique et en dynamique. Pour chaque scénario à simuler, une simulation s'effectuera avec le simulateur Spirent et l'autre avec le simulateur LACIME. Ainsi, il sera possible de comparer les performances obtenues vues par le récepteur u-blox pour un même scénario avec les deux simulateurs.

5.1 Performance en statique du simulateur LACIME

À propos de la caractérisation des performances en statique, huit simulations d'une durée de 24 heures incluant deux d'entre elles dans chaque quart d'hémisphère seront exécutées avec le simulateur LACIME et Spirent. Ainsi, il sera possible de savoir si le simulateur est fonctionnel peu importe la position simulée. Les résultats obtenus du récepteur u-blox seront comparés pour un même scénario avec les deux simulateurs. L'analyse va se concentrer particulièrement sur les moyennes et les écarts-types de l'erreur de position et du DOP enregistrés par le récepteur. Le Tableau 5.1 montre les latitudes et les longitudes des positions simulées pour une simulation débutant le 6 janvier 2008 à 17:04.

Les Figures 5.1 à 5.4 illustrent les résultats vus par le récepteur u-blox pour la simulation de 24 heures au Colorado. Les graphiques des autres simulations en statique sont à l'ANNEXE X.

Les tableaux 5.2 et 5.3 récapitulent les moyennes et les écart-types des erreurs de position et du DOP pour les huit simulations en statique de 24 heures.

Tableau 5.1 Latitudes et longitudes des positions simulées pour l'analyse en statique

Positions simulées	Coordonnées géographiques des positions simulées
Colorado	39° 42' N 104 degré 56' O
Honduras	14° 26' N 86 degré 23' O
Marseille	43° 17' N 5 degré 22' E
Chine	31° 12' N 121 degré 26' E
Argentine	38° 24' S 63 degré 37' O
Natal	5° 47' S 35 degré 12' O
Afrique du Sud	30° 34' S 22 degré 59' E
Nouvelle-Zélande	41° 0' S 174 degré 0' E

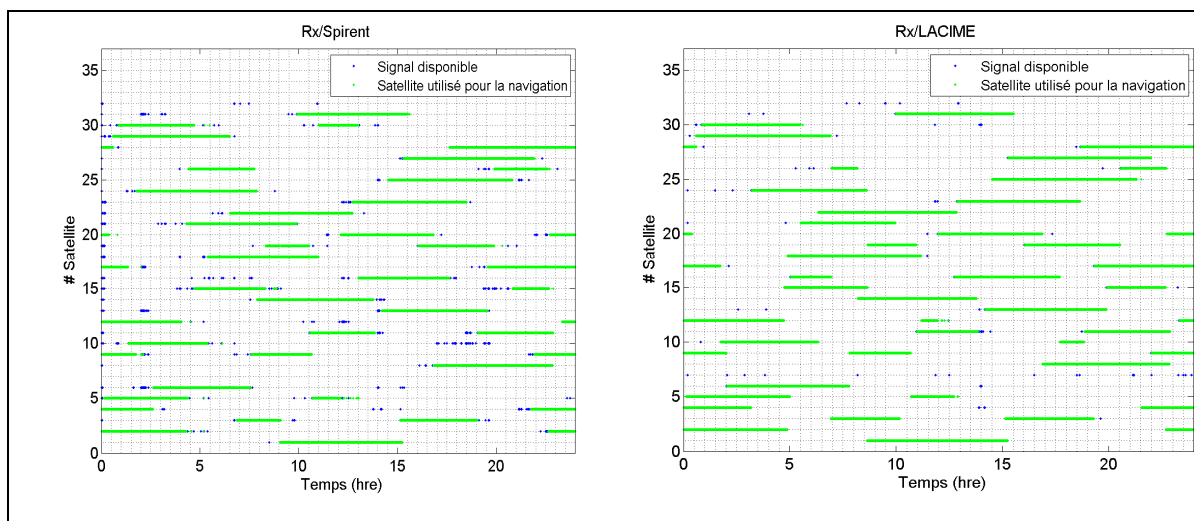


Figure 5.1 Visibilités des satellites d'une simulation au Colorado (Rx/Spirent versus Rx/LACIME)

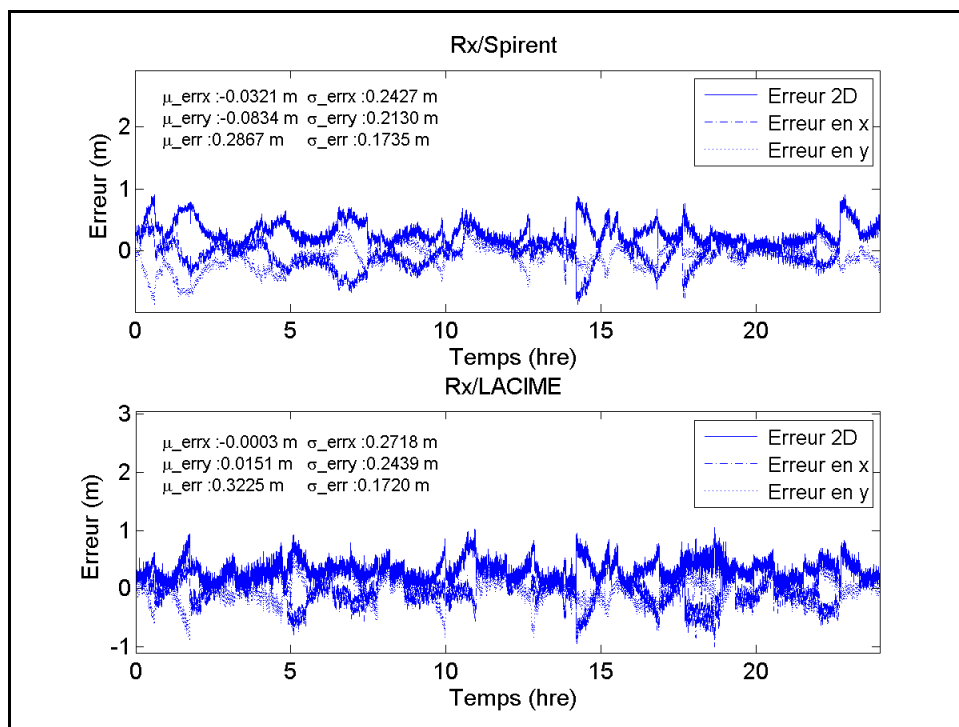


Figure 5.2 L'erreur de position d'une simulation au Colorado (Rx/Spirent versus Rx/LACIME)

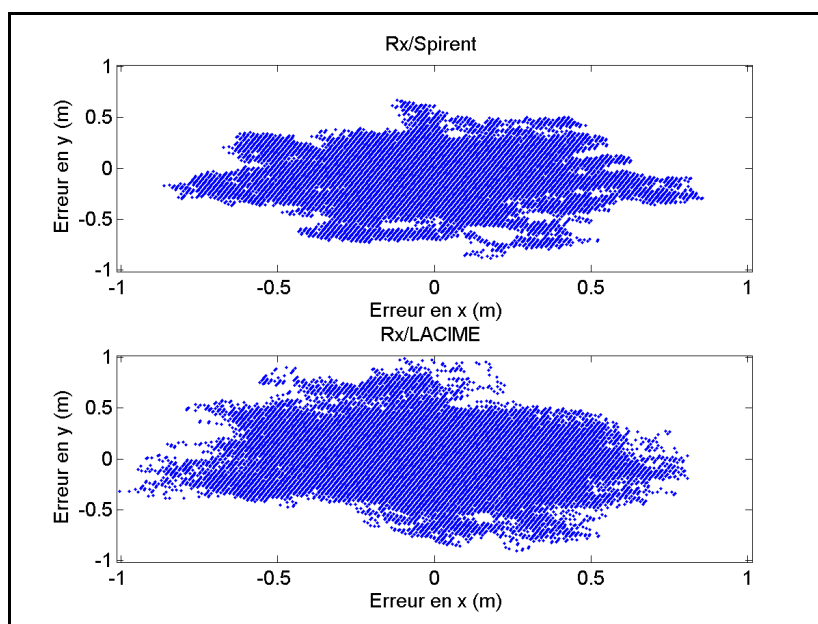


Figure 5.3 Nuage de points d'une simulation au Colorado (Rx/Spirent versus Rx/LACIME)

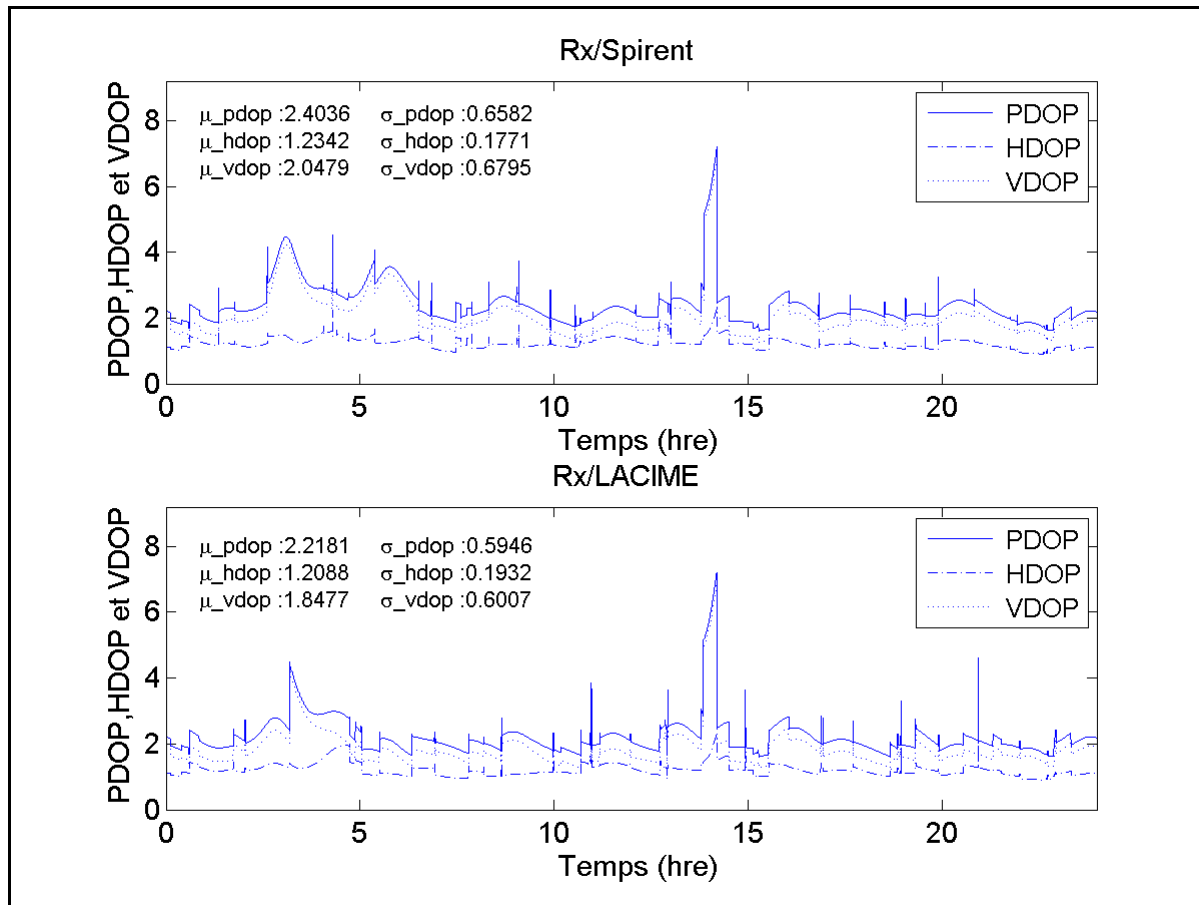


Figure 5.4 PDOP, HDOP et VDOP d'une simulation au Colorado
(Rx/Spirent versus Rx/LACIME)

Tableau 5.2 Moyennes et écarts-types de l'erreur de position en statique à différentes positions (Rx/Spirent versus Rx/LACIME)

Positions simulées	Résultats	μ_{errx} (m)	σ_{errx} (m)	μ_{erry} (m)	σ_{erry} (m)	μ_{err} (m)	σ_{err} (m)
Colorado	Rx/Spirent	-0,0321	0,2427	-0,0834	0,2130	0,2867	0,1735
	Rx/LACIME	-0,0003	0,2718	0,0151	0,2439	0,3225	0,1720
Honduras	Rx/Spirent	-0,0029	0,3239	-0,0247	0,0656	0,2592	0,2065
	Rx/LACIME	0,0581	0,5053	-0,0363	0,1082	0,4077	0,3249
Marseille	Rx/Spirent	0,0894	0,2178	0,0072	0,2317	0,2851	0,1670
	Rx/LACIME	-0,0643	0,6862	-0,0068	0,7097	0,8631	0,4836
Chine	Rx/Spirent	-0,0326	0,2414	-0,0026	0,1834	0,2670	0,1474
	Rx/LACIME	-0,1055	0,5212	-0,0374	0,5547	0,6907	0,3389
Argentine	Rx/Spirent	0,0199	0,2592	-0,1032	0,2194	0,2966	0,1960
	Rx/LACIME	-0,0334	0,3457	-0,0298	0,2553	0,3573	0,2429

Positions simulées	Résultats	μ_{errx} (m)	σ_{errx} (m)	μ_{erry} (m)	σ_{erry} (m)	μ_{err} (m)	σ_{err} (m)
Natal	Rx/Spirent	0,0216	0,1477	0,0026	0,2044	0,1970	0,1589
	Rx/LACIME	0,0357	0,1692	0,0343	0,2461	0,2347	0,1913
Afrique du Sud	Rx/Spirent	0,0634	0,1908	0,0372	0,2207	0,2553	0,1592
	Rx/LACIME	0,0068	0,2054	0,0571	0,2720	0,2970	0,1769
Nouvelle-Zélande	Rx/Spirent	0,0207	0,1934	0,0747	0,2336	0,2654	0,1661
	Rx/LACIME	0,0317	0,2771	0,0439	0,2691	0,3372	0,1960

Tableau 5.3 Moyennes et écarts-types du DOP en statique à différentes positions (Rx/Spirent versus Rx/LACIME)

Positions simulées	Résultats	μ_{pdop}	σ_{pdop}	μ_{hdop}	σ_{hdop}	μ_{vdop}	σ_{vdop}
Colorado	Rx/Spirent	2,4036	0,6582	1,2342	0,1771	2,0479	0,6795
	Rx/LACIME	2,2181	0,5946	1,2088	0,1932	1,8477	0,6007
Honduras	Rx/Spirent	2,6634	1,0107	1,2417	0,3573	2,3346	1,0288
	Rx/LACIME	2,3361	0,6280	1,1716	0,2937	2,0080	0,6200
Marseille	Rx/Spirent	2,2388	0,4279	1,2072	0,2019	1,8797	0,4050
	Rx/LACIME	2,1255	0,4009	1,1927	0,2335	1,7507	0,3695
Chine	Rx/Spirent	2,4769	0,6802	1,2130	0,2502	2,1519	0,6585
	Rx/LACIME	2,1988	0,4870	1,1567	0,2781	1,8634	0,4293
Argentine	Rx/Spirent	2,3360	0,4452	1,2124	0,1723	1,9887	0,4478
	Rx/LACIME	2,3326	0,5557	1,2736	0,3105	1,9357	0,5330
Natal	Rx/Spirent	2,6238	0,5026	1,0938	0,1684	2,3765	0,5144
	Rx/LACIME	2,2022	0,4494	1,0363	0,1936	1,9332	0,4507
Afrique du Sud	Rx/Spirent	2,5091	0,6541	1,2399	0,2342	2,1711	0,6467
	Rx/LACIME	2,2517	0,5340	1,1832	0,2252	1,9093	0,5097
Nouvelle-Zélande	Rx/Spirent	2,4263	0,7691	1,2410	0,1981	2,0710	0,7808
	Rx/LACIME	2,2771	0,6861	1,2416	0,2933	1,8922	0,6695

Pour toutes les simulations effectuées en statique, il y a une grande amélioration des performances comparativement à celles établies initialement. Tout d'abord, une simulation de 24 heures est possible ce qui était loin d'être le cas avant la correction des problèmes. Durant toute la durée de la simulation, il n'y a pas eu de perte de position ni de perte de satellites au récepteur. Les graphiques de visibilité des satellites démontrent que le satellite #32 n'est jamais visible pour les simulations avec le simulateur LACIME et Spirent. Pourtant, celui-ci le devient au cours de la simulation avec les deux simulateurs. Comme le

satellite #32 est défectueux, le récepteur reçoit le signal GPS du satellite, mais il n'en tient pas compte pour ne pas augmenter l'erreur de la position.

Avec le simulateur LACIME, on remarque que l'erreur maximale de la position est de quelques centimètres supérieures à celle du simulateur Spirent. Le choix des satellites simulé entre les deux simulateurs est différent ce qui influence la position trouvée par le récepteur. En effet, le simulateur Spirent simule les satellites les plus proches de l'utilisateur tandis que ceux ayant le plus petit PRN ont priorité pour le simulateur LACIME. Avec le simulateur Spirent, il arrive qu'un satellite ne soit plus simulé même si celui-ci est encore visible. En effet, ce phénomène arrive parce qu'un autre satellite a un ρ inférieur. Pour le simulateur LACIME, un satellite est simulé tant et aussi longtemps qu'il n'est plus visible. La façon de procéder des deux simulateurs sur la transition des satellites peut avoir pour une simulation de plusieurs heures quelques différences sur la visibilité des satellites. La Figure 5.1 l'indique où le satellite #30 est simulé entre 10 et 15 heures de la simulation avec le simulateur Spirent et pour le simulateur LACIME il n'est pas simulé durant cette période de temps. Une constellation différente entre le simulateur Spirent et LACIME entraîne des distinctions sur le DOP comme sur la Figure 5.4 à 2 heures ½ de la simulation.

5.2 Performance en dynamique du simulateur LACIME

L'étude des performances en dynamique va s'effectuer à l'aide de trois scénarios ayant la même trajectoire curviligne. Chacun d'entre eux sera exécuté avec le simulateur LACIME et Spirent en basse (5 m/s), moyenne (14 m/s) et haute (28 m/s) vitesse. Ainsi, il sera possible de déterminer si la vitesse a un impact sur les performances du simulateur. Les données recueillies par le récepteur u-blox, c'est-à-dire les positions, les vitesses et le DOP seront comparés pour une simulation avec le simulateur Spirent versus LACIME. L'erreur de la position est la différence entre la position calculée par le simulateur et celle trouvée par le récepteur u-blox. L'erreur de la vitesse est estimée selon le même principe à l'exception que c'est les mesures de vitesses du véhicule qui sont utilisées. De plus, les moyennes (μ_{vitesse}) et les écarts-types (σ_{vitesse}) des erreurs de la vitesse seront évaluées. La Figure

5.5 montre la trajectoire qui sera simulée pour l'étude des performances en dynamique du simulateur LACIME.

Les Figures 5.6 à 5.9 contiennent les résultats de la simulation de la trajectoire avec un véhicule ayant une vitesse constante de 5 m/s. D'autres résultats simulant la même trajectoire à d'autres vitesses sont disponibles à l'ANNEXE XI.

Les Tableaux 5.4 à 5.6 résument les moyennes et les écarts-types de l'erreur de la position, de l'erreur de la vitesse et du DOP pour les simulations en dynamique avec le simulateur Spirent versus LACIME.

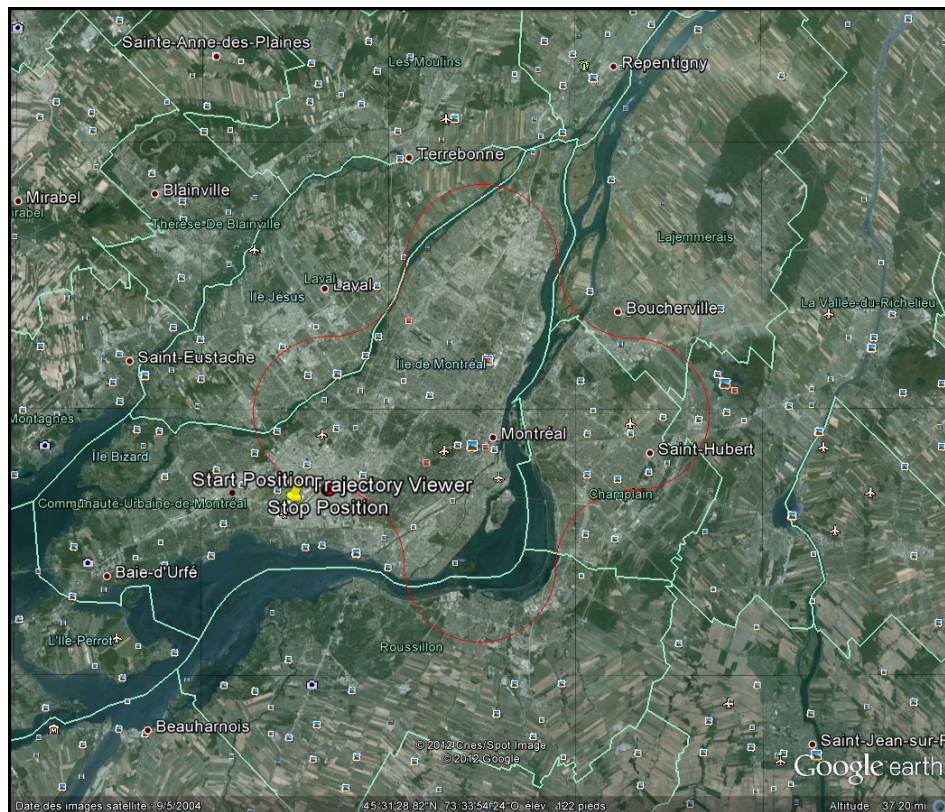


Figure 5.5 Trajectoire simulée affichée sur Google Earth

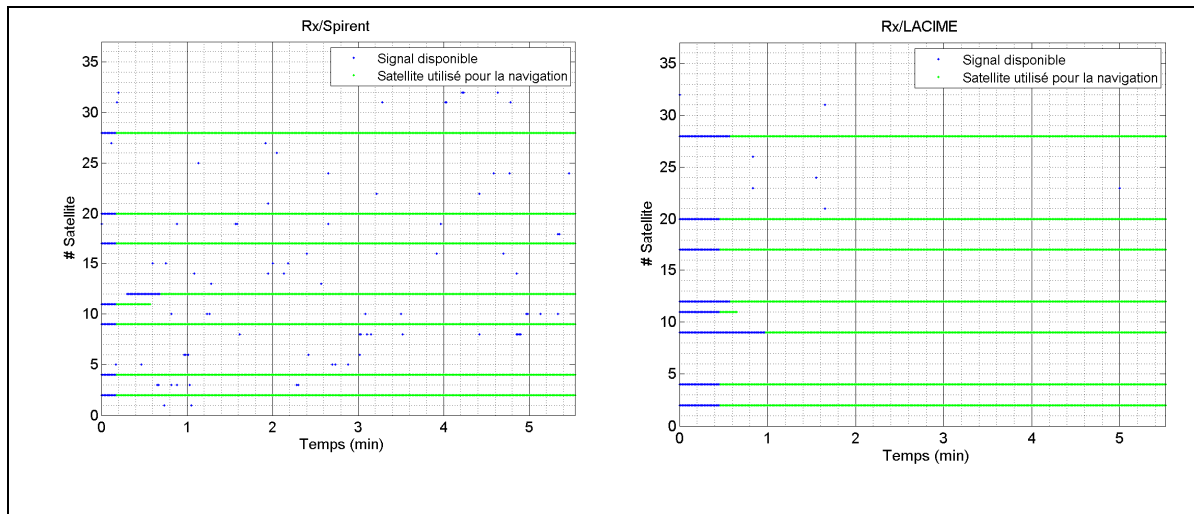


Figure 5.6 Visibilités des satellites pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)

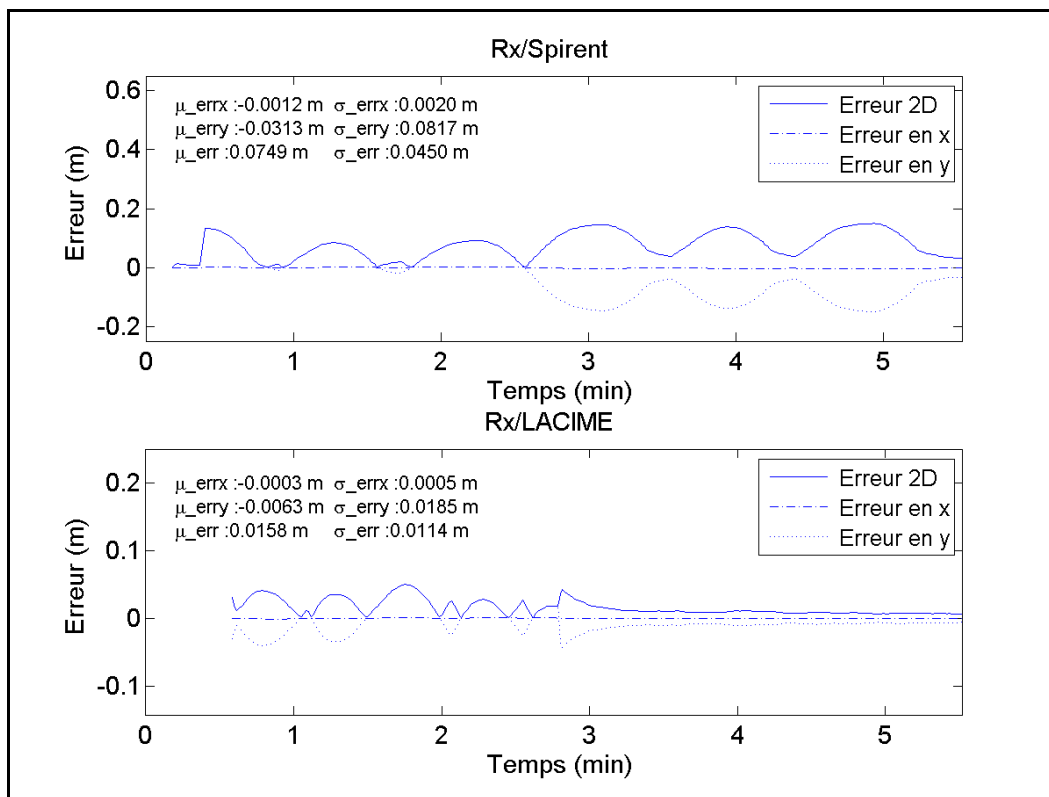


Figure 5.7 L'erreur de la position pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)

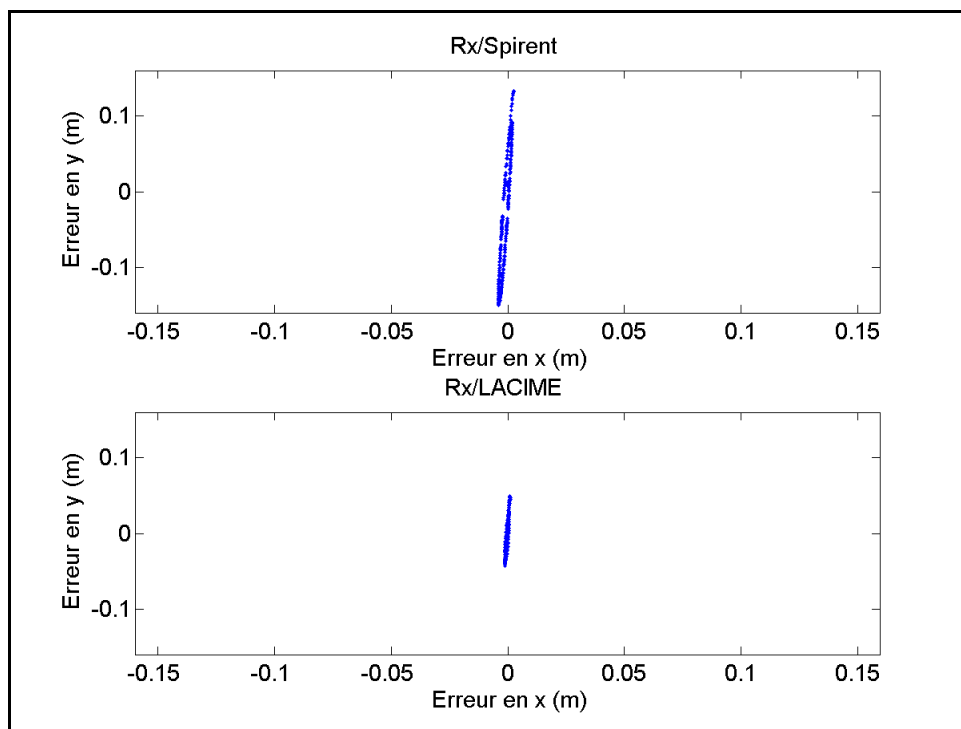


Figure 5.8 Nuage de points pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)

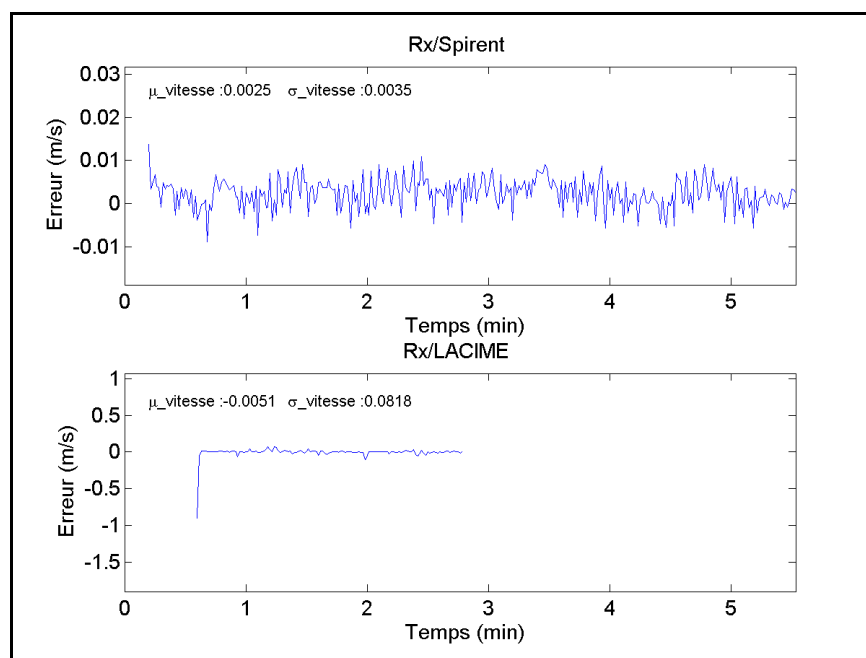


Figure 5.9 Erreur de la vitesse pour une vitesse de 5 m/s (Rx/Spirent versus Rx/LACIME)

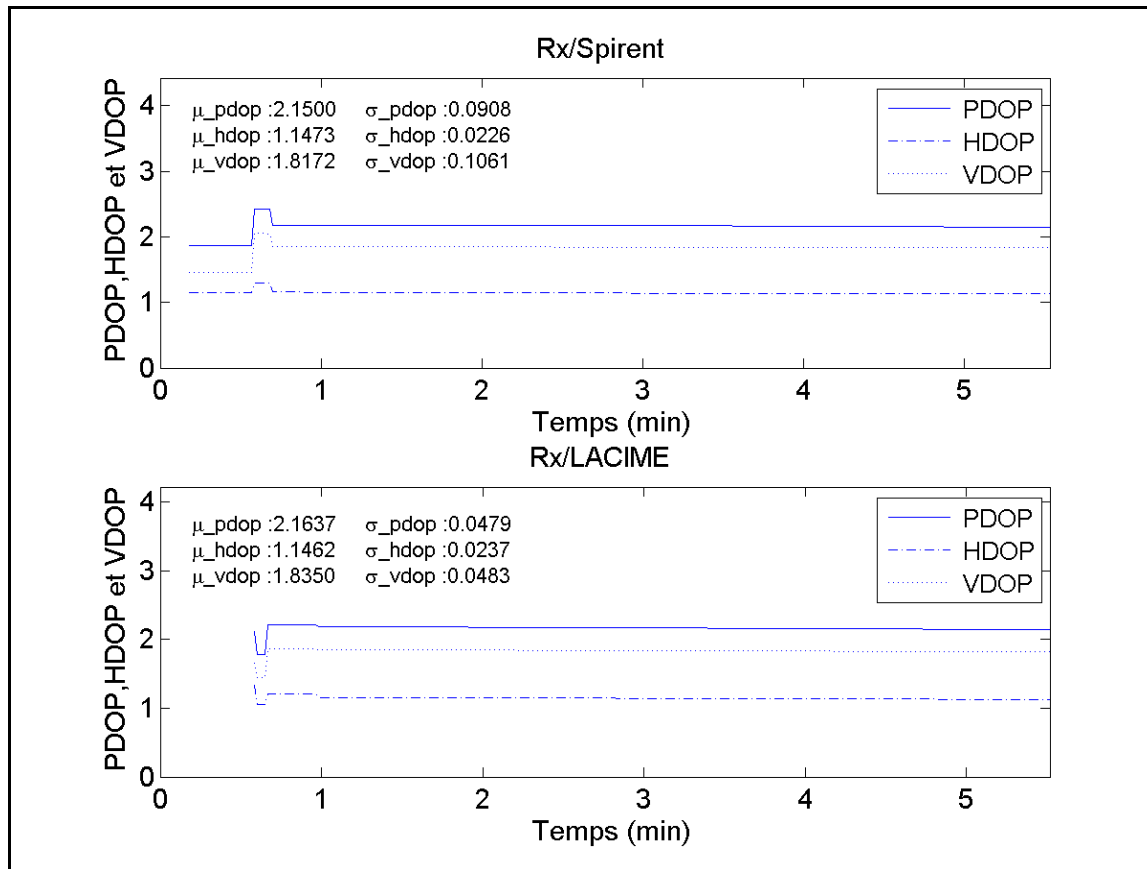


Figure 5.10 PDOP, HDOP et VDOP pour une vitesse de 5 m/s
(Rx/Spirent versus Rx/LACIME)

Tableau 5.4 Moyennes et écarts-types de l'erreur de la position en dynamique à différentes vitesses (Rx/Spirent versus Rx/LACIME)

Vites ses (m/s)	Résultats	μ_{errx} (m)	σ_{errx} (m)	μ_{erry} (m)	σ_{erry} (m)	μ_{err} (m)	σ_{err} (m)
5	Rx/Spire nt	-0,0012	0,0020	-0,0313	0,0817	0,0749	0,0450
	Rx/LACI ME	-0,0003	0,0005	-0,0063	0,0185	0,0158	0,0114
14	Rx/Spire nt	-0,0010	0,0008	-0,0247	0,0330	0,0336	0,0239
	Rx/LACI ME	-0,0016	0,0040	-0,0600	0,1818	0,1093	0,1572
28	Rx/Spire nt	-0,0062	0,0061	-0,1950	0,2545	0,2024	0,2488

	Rx/LACIME	0,0015	0,0114	0,1390	0,4360	0,3839	0,2491
--	-----------	--------	--------	--------	--------	--------	--------

Tableau 5.5 Moyennes et écarts-types de l'erreur de la vitesse en dynamique à différentes vitesses (Rx/Spirent versus Rx/LACIME)

Vitesses (m/s)	Résultats	μ vitesse (m/s)	σ vitesse (m/s)
5	Rx/Spirent	0,0025	0,0035
	Rx/LACIME	-0,0051	0,0818
14	Rx/Spirent	0,0096	0,0042
	Rx/LACIME	-0,2443	1,0724
28	Rx/Spirent	0,0214	0,0025
	Rx/LACIME	0,1353	0,8165

Tableau 5.6 Moyennes et écarts-types du DOP en dynamique à différentes vitesses (Rx/Spirent versus Rx/LACIME)

Vitesses (m/s)	Résultats	μ_{pdop}	σ_{pdop}	μ_{hdop}	σ_{hdop}	μ_{vdop}	σ_{vdop}
5	Rx/Spirent	2,1500	0,0908	1,1473	0,0226	1,8172	0,1061
	Rx/LACIME	2,1637	0,0479	1,1462	0,0237	1,8350	0,0483
14	Rx/Spirent	2,1878	0,1685	1,1610	0,1161	1,8544	0,1281
	Rx/LACIME	2,2786	0,2478	1,3016	0,3178	1,8593	0,0657
28	Rx/Spirent	2,1170	0,0741	1,1199	0,0311	1,7962	0,0757
	Rx/LACIME	2,1534	0,1540	1,1734	0,1837	1,8021	0,0577

En corrigeant les problèmes observés avec les simulations en statique, les performances en dynamique sont meilleures. En effet, le récepteur est capable de suivre la trajectoire simulée sans perdre la position ce qui était loin d'être le cas auparavant. Les moyennes et les écarts-

types des erreurs de position et des DOPs recueillis sont semblables pour une simulation avec le simulateur Spirent et LACIME peu importe la vitesse du véhicule. Par contre, il arrive que le TTFF du simulateur LACIME soit trop long (3 à 4 minutes). Effectivement, il augmente en fonction de la vitesse du véhicule. Pourtant, celui-ci devrait être de moins d'une minute peu importe la vitesse du véhicule. Les valeurs μ_{vitesse} du simulateur LACIME sont trop élevées et idéalement elles ne devraient pas dépasser 0,05 m/s comme avec le simulateur Spirent. La durée de parcours de la même trajectoire simulée avec le simulateur LACIME est plus courte qu'avec le simulateur Spirent. C'est pour cette raison que sur la Figure 5.9 il n'y a pas de vitesse du véhicule lorsque la durée de la simulation a passé 3 minutes. Un problème lors du calcul de la trajectoire pourrait avoir un impact sur le TTFF, la durée de parcours des trajectoires et la valeur de μ_{vitesse} pour une simulation. Dans ce cas, il serait adéquat d'effectuer une validation minutieuse du calcul des trajectoires.

5.3 Conclusion

Ce chapitre a permis d'établir les performances en statique et en dynamique du simulateur LACIME et de les comparer avec le simulateur Spirent. En général, les performances obtenues avec les deux simulateurs sont identiques. Pour une simulation en statique et en dynamique avec le simulateur LACIME, le récepteur u-blox garde toujours sa position lorsque le TTFF a été atteint.

Les performances actuelles du simulateur LACIME en position statique sont meilleures comparativement à celles établies initialement. Par contre, il a été observé que l'erreur maximale de la position est supérieure de quelques centimètres d'une simulation avec le simulateur Spirent. En simulant la même constellation avec les deux simulateurs, l'erreur maximale de la position diminuerait davantage.

Les performances en dynamique du simulateur LACIME sont bonnes. Par contre, il serait préférable d'obtenir un TTFF inférieur à 1 minute indépendamment de la vitesse du véhicule simulée. Cela n'est pas le cas pour une vitesse de 14 m/s et 28 m/s. Aussi, la durée de

parcours de la même trajectoire avec le simulateur Spirent et LACIME est généralement plus courte avec ce dernier. Afin d'améliorer le TTFF et obtenir la même durée de parcours d'une trajectoire qu'avec le simulateur Spirent, il serait préférable de faire une validation minutieuse de la génération des trajectoires pour s'assurer qu'elles soient bien produites. Ainsi, μ_{vitesse} se rapprochera davantage des valeurs obtenues avec le simulateur Spirent.

CONCLUSION

Avec la complexité des systèmes de navigation qui augmente d'année en année, il devient de plus en plus difficile de tester les récepteurs GNSS. En effet, il n'est pas possible de répéter un scénario de test précis, car les satellites sont toujours en mouvement. Dans cette perspective, il est nécessaire de trouver un moyen pour améliorer la procédure de vérification du bon fonctionnement des récepteurs GNSS. L'objectif principal du projet de recherche est de mettre en œuvre la génération en temps réels des signaux de navigation GNSS. Le simulateur GPS L1 déjà existant au laboratoire a été récupéré. Par contre, celui-ci comporte plusieurs lacunes importantes comparativement à un simulateur commercial qui ont été identifiées durant la première phase du projet. Une revue des caractéristiques de différents simulateurs GNSS a été établie et ceci a permis de trouver les principales fonctionnalités qu'un simulateur commercial doit posséder :

- contrôle du C/N_0 et du message de navigation,
- fonctionnalité d'enregistrement des mesures,
- modélisation des délais atmosphériques,
- simulation de multitrajet,
- générateur de bruit.

La première phase du projet consiste à faire une première étude du simulateur LACIME pour avoir une vue d'ensemble de ses performances. Il a été précisé que peu importe la configuration du scénario les mêmes problèmes apparaissent : pertes fréquentes de positions et des satellites, le récepteur n'est pas capable de démoduler le signal de tous les satellites simulés. Donc, le simulateur LACIME n'est pas fonctionnel et il n'est pas possible de faire une simulation qui dure 24 heures. En révisant le code, quelques anomalies ont été trouvées. En effet, la structure du processus IF du simulateur LACIME n'a pas été correctement programmée et elle devra être corrigée et ensuite validée.

La seconde phase du projet comprend la validation logicielle du simulateur LACIME. Les paramètres de navigation calculés ne correspondaient aucunement à celles du simulateur

Spirent. Les formules qui estiment la position, l'angle d'élévation et d'azimuth, les pseudo-distances, les Dopplers et les vitesses des satellites ont été corrigés. Quelques erreurs ont été trouvées dans la génération du message binaire de navigation. Cette opération se faisait en retard et en plus les deux premiers mots de chaque sous-trame permettant au récepteur de se synchroniser n'avaient pas les bonnes valeurs. Aussi, les bits réservés ont été corrigés pour que ceux-ci soient émis en alternant '1' et '0'. Avec les modifications apportées aux paramètres et aux messages de navigation, leur valeur est identique à ceux du simulateur commercial Spirent.

La troisième phase du projet est une validation IF du simulateur LACIME. Le délai UTC a été supprimé et l'horloge principale est une horloge externe qui sert de solution temporaire pour le déverminage du système. Les calculs des phases du code, de la porteuse et du message de navigation ont été corrigés ainsi que l'index de la position courante du code et du message de navigation. Ces deux derniers paramètres causaient de l'instabilité sur les résultats obtenus du simulateur LACIME vus du récepteur, car ses données sont employées pour calculer le biais d'horloge. Le calcul de l'adresse mémoire contenant la valeur du sinus et du cosinus de la porteuse a été modifié afin que celle-ci corresponde au bon quadrant. Les corrections effectuées à la IF ont été validées et elles permettent d'obtenir des biais d'horloge, des pseudo-distances et des Dopplers des satellites vus par le récepteur semblables aux résultats du simulateur Spirent.

La dernière phase du projet est une caractérisation du simulateur LACIME. Huit simulations de 24 heures à différentes positions en statique a permis d'obtenir une moyenne maximale sur l'erreur de position d'au plus 1 mètre. Ainsi, les performances du simulateur se rapprochent d'un simulateur commercial. Pour la caractérisation en dynamique, une trajectoire curviligne avec des vitesses constantes de 5, 14 et 28 m/s ont été simulées. Ceci a permis d'obtenir une moyenne d'au plus 0,3839 m sur l'erreur de la position simulé.

RECOMMANDATIONS

Les performances du simulateur LACIME sont nettement meilleures avec les modifications faites dans ce mémoire. En effet, il n'y a pas de perte de position du récepteur GPS et les moyennes et les écarts-types de l'erreur de la position simulée sont similaires au simulateur Spirent. Par contre, comme il y a toujours place à l'amélioration, quelques recommandations s'imposent afin que le simulateur LACIME possède les mêmes caractéristiques qu'un simulateur commercial.

Le choix de sélection des satellites simulés doit se faire plus intelligemment. Actuellement, ceux ayant le plus petit PRN ont priorité. Parmi les satellites visibles d'une simulation avec le simulateur Spirent, ils sont simulés en fonction de leur pseudo-distance ou du DOP selon la configuration de la simulation. Le simulateur LACIME pourrait implémenter ces deux options. Les C/N_0 s du simulateur LACIME sont identiques pour tous les satellites. En réalité, ceux-ci devraient s'affaiblir selon la distance qui sépare le satellite de l'utilisateur. Il serait possible de faire varier la puissance des signaux en modifiant le gain IF des satellites qui est présentement constant. Les délais atmosphériques causés par la troposphère et l'ionosphère ne sont pas modélisés. Pourtant, ceci est une fonctionnalité de base que tout simulateur commercial doit avoir (voir Tableau 1.4). À propos du simulateur LACIME, il pourrait modéliser plusieurs modèles de délai atmosphérique et l'utilisateur pourrait choisir lequel utiliser pour la simulation comme dans SimGen.

Les commandes des trajectoires n'ont pas été validées avec le simulateur LACIME. Dans la deuxième partie du chapitre #5 du mémoire, les performances en dynamiques ont été étudiées. Dans ce cas-ci, l'erreur de position est la différence entre la position calculée par le récepteur et celle générée par le simulateur à un temps donné. Cette procédure n'examine pas l'emplacement et la vitesse du véhicule selon la commande programmée. Pour le vérifier, il faudrait exécuter les mêmes commandes d'une trajectoire pour le simulateur LACIME et Spirent. Ensuite, comparer les valeurs de position du véhicule à chaque seconde

selon les axes x, y et z obtenus par la fonctionnalité d'enregistrement des mesures du simulateur. Ainsi, cela assurerait que les trajectoires sont bien produites.

Le simulateur LACIME est un système embarqué en temps réel, c'est-à-dire des contraintes de temps doivent être respectées pour ne pas nuire au bon fonctionnement. Il possède un processeur Intel Pentium 2 GHz avec 2 Go de mémoire vive. Celui-ci n'est pas récent et en exécutant des programmes demandant beaucoup de ressources durant une simulation, ceci aurait un impact sur les performances. En effet, une désynchronisation surviendrait durant la génération des phases des NCOs et le récepteur GPS perdrait sa position. Dans ses conditions, il serait préférable de choisir un ordinateur plus récent. Par exemple, en optant pour un processeur quatre cœurs, le temps d'exécution serait plus rapide avec le parallélisme. Aussi, un système d'exploitation temps réel est conçu spécialement pour les systèmes embarqués et il permettrait de garantir les délais de temps ce qui n'est pas le cas avec une version standard.

Il n'est pas possible de simuler plus que 12 satellites dus à un manque de mémoire de la carte Lyrtech (Lacasse, 2009). La mémoire FIFO qui enregistre le message de navigation a été surévaluée. En effet, chaque canal contient 16 381 bits et ceci pourrait être divisé facilement par 20 sans nuire aux performances du simulateur. Puisque chaque trame du message de navigation GPS L1 contient 300 bits, 600 bits par canal seraient suffisants. Ainsi, en réattribuant correctement la quantité de mémoire par canal, le simulateur LACIME pourra simuler d'autres canaux.

Certains récepteurs GNSS peuvent recevoir le message de navigation par le protocole RS-232 sans tenir compte des signaux de navigation. Il en serait une grande utilité si le simulateur LACIME pouvait envoyer les données sur le port RS-232. En effet, la validation du message de navigation se ferait plus facilement ainsi que l'analyse des performances sur les différents paramètres d'almanachs et d'éphémérides. Le simulateur Spirent disponible au laboratoire simule seulement la constellation GPS tandis que le simulateur LACIME simulerait plusieurs constellations dans de nombreuses années. Donc, le simulateur Spirent ne pourra plus être utilisé pour valider les résultats des autres constellations que GPS. Dans ce

cas, un simulateur GNSS permettrait de faciliter la tâche de validation des résultats comme les paramètres et le message de navigation.

La plupart des simulateurs disponibles sur le marché ont une caractéristique commune : il est nécessaire de calibrer la RF à tous les deux ans. Cette opération d'entretien est coûteuse pour l'utilisateur, car il faut payer un technicien pour l'effectuer. Par contre, le simulateur Cast-2000 permet à l'utilisateur d'effectuer cette tâche en inscrivant les valeurs des paramètres de calibration dans le fichier de configuration de l'appareil. Pour les trouver, il est nécessaire de faire plusieurs manœuvres manuelles et il n'est pas possible pour le simulateur de les calculer. Comme prochain sujet de recherche, il serait intéressant de trouver un moyen d'automatiser la calibration RF de sorte que les paramètres de calibration soient calculés par le simulateur.

ANNEXE I

PROBLÈMES RENCONTRÉS AVEC LE SIMULATEUR LACIME

Voici les problèmes rencontrés avec le simulateur LACIME suite à une première étude. Dans l'interface graphique du récepteur u-blox, une bande bleue indique que les données ne peuvent pas être démodulées. À l'opposé, une bande verte signifie que le signal est synchronisé et dans une telle situation, le récepteur est capable de lire les données qui sont transmises.

Tableau-A I-1 Résultats des tests en statique

No Requête	Descriptions
STA01	L'altitude du logiciel est différente de celui du récepteur CMC et u-blox
STA02	Pour toutes les simulations, il arrive que le récepteur perde le signal de tous les satellites pendant 10 à 15 minutes.
STA03	Test statique sur 24 heures n'est pas possible.
STA04	Au début de la simulation, lorsque 4 satellites sont poursuivis par le récepteur, l'ajout d'un 5 ^{ème} satellite fait bouger la solution du récepteur. Idem pour le 6 ^{ème} , le 7 ^{ème} et le 8 ^{ème} satellite. À vérifier en fonction du DOP qui n'est pas affiché dans le logiciel.
STA05	Test en Argentine ne peut pas être simulé, car on obtient une erreur que la coordonnée géographique ne correspond pas aux bornes minimales et maximales établies. Ce problème a été observé pour plusieurs positions de l'hémisphère Sud. Le logiciel n'accepte pas les positions à l'hémisphère Sud.
STA06	Pour la simulation à Marseille, dès que le récepteur a obtenu une position, il a indiqué une altitude négative (-1940 mètres)
STA07	Au début de la simulation Honduras, le récepteur u-blox a détecté une quinzaine de satellites. Pourtant le matériel en supporte seulement 8 satellites et ensuite il y a eu perte de tous les signaux satellites. Ce problème se répète régulièrement.
STA08	Pour la simulation Honduras, la position des satellites ainsi que leur PRN ne correspondent pas toujours à ce qui est affiché sur le logiciel et le récepteur u-blox (sky view)

No Requête	Descriptions
STA09	Pour la simulation Honduras, après 16 minutes de simulation, le récepteur n'a pas été capable d'établir une position 2D et 3D.
STA10	<p>Pour la simulation Marseille, tout au long de la simulation, il y a eu une instabilité des signaux des satellites (bande verte et bande bleue) avec le récepteur u-blox. Après 4 minutes 30 de simulation, le récepteur voit 4 satellites en bande verte et 1 satellite en bande bleue, mais il n'affiche pas de position 3D. Les satellites ne sont pas affichés dans la fenêtre sky view du récepteur. Après 6 minutes de simulation, les signaux satellites redeviennent instables. Les bandes vertes des satellites durent quelques secondes. Après 7 minutes 41 de simulation, les satellites s'affichent sur le sky view et quelques secondes plus tard la première position 3D est obtenue avec 6 satellites en bande verte, 2 satellites en bande bleue et une altitude négative, La latitude et la longitude correspondent avec quelques décimales près. Le système devient stable pour au moins 5 à 10 minutes. Par la suite, les signaux redeviennent instables comme au début de la simulation.</p> <p>Conclusion : Les signaux satellites sont instables et ,le temps d'obtention d'une position est beaucoup trop long.</p>
STA11	En démarrant une simulation en Chine, le récepteur u-blox a détecté 14 satellites et ensuite il y a eu une perte de signal de tous les satellites quelques secondes plus tard.
STA12	Pour le scénario en Chine, la position (latitude et longitude) au récepteur CMC correspond bien avec ceux du logiciel avec quelques décimales de différence. Par contre, le récepteur CMC affiche une altitude négative.
STA13	Pour la simulation en Chine avec le récepteur u-blox, en 4 minutes 30 de simulation, nous avons obtenu la première position (2D). En relançant la simulation, en 4 minutes 30 nous avons obtenu une position 3D. Pour les deux essais, dès que nous avons obtenu une position, on a quelques secondes plus tard perdu le signal des satellites. Conclusion : L'acquisition des signaux GPS est beaucoup trop lente et ceci n'est pas constant.
STA14	Pour la simulation en Chine, en obtenant une position 3D, il y a une déviation de l'ordre de quelques centaines de mètres avec le récepteur u-blox en recevant le signal de 7 satellites. Il y a une trop grande déviation pour un test statique. Elle se dirige en bas à droite.
STA15	En démarrant la simulation à Marseille, le récepteur u-blox a détecté 10 satellites et ensuite il y a eu une perte de signal de tous les satellites

No Requête	Descriptions
STA16	Pour la simulation en Chine, le récepteur u-blox voit 8 satellites (7 satellites avec bande verte et 1 satellite avec bande bleue) et tout d'un coup, le satellite avec bande bleu est disparu et réapparu quelques secondes plus tard.
STA17	En démarrant une simulation à Marseille, le récepteur u-blox a obtenu une position 2D en 5 minutes de simulation et 3D en 6 minutes de simulation. Quand il obtient une position, il la perd quelques secondes plus tard. Après 16 minutes de simulation, le récepteur u-blox a beaucoup de difficulté à garder les signaux des satellites. Le système est instable.
STA18	En démarrant la simulation au Colorado, le récepteur u-blox a détecté une dizaine de satellites (bande bleu) et ensuite il y a eu une perte de tous les signaux des satellites.
STA19	<p>En démarrant la simulation au Colorado, le récepteur u-blox a obtenu une position 3D en 6 minutes de simulation. Quand la position est obtenue, le récepteur reçoit les signaux de seulement 7 satellites (bande verte). Une altitude de 10 000 mètres est simulée et le récepteur affiche 10 022 mètres. La plus grande déviation observée est approximativement de 10 mètres.</p> <p>Conclusion : Trop grande déviation pour un test statique, erreur de 22 mètres pour l'altitude, erreur entre la latitude et longitude est de l'ordre de quelques décimales.</p>

Tableau-A I-2 Résultats des tests dynamiques

No Requête	Descriptions
DYN01	Comme les mêmes problèmes ont été aperçus pour les tests statiques (voir DYN05), il est difficile de bien valider le déplacement du véhicule s'il s'affiche correctement au récepteur. Par contre, au niveau logiciel, on aperçoit que le véhicule se déplace, car ses coordonnées x et y changent durant la simulation. Quand les tests statiques en 24 heures réussiront, il sera adéquat de refaire des tests plus approfondis sur les mouvements dynamiques du véhicule afin de valider le comportement du récepteur.
DYN02	L'algorithme logiciel implémentant le climb et le descent ne fonctionne pas bien. Des tests plus approfondis s'imposent.

No Requête	Descriptions
DYN03	Les algorithmes logiciels pour les déplacements n'ont jamais été validés minutieusement dans le logiciel (Ben Salah, 2010). Donc, cette partie serait à revoir complètement. Il est possible que certains paramètres dans la fenêtre Rx trajectory du logiciel ne soient même pas utilisés pour calculer le chemin.
DYN04	En démarrant une simulation en dynamique quelconque, le récepteur u-blox a détecté 14 satellites et ensuite il y a eu une perte de signal de tous les satellites pendant quelques secondes. Ensuite, le récepteur a détecté quelques satellites (bande bleue) et ensuite il a perdu ses signaux. Après une quinzaine de minutes de simulation, il n'y a toujours pas eu de position 3D.
DYN05	Les mêmes problèmes ont été observés qu'en position statique : Perte des signaux satellites pendant 10 à 15 minutes approximativement, détection d'une dizaine de satellites au début de la simulation (bande bleue), difficulté à obtenir une position 3D dans un temps respectable. Actuellement, les récepteurs CMC et u-blox prend de 5 à 6 minutes avant de trouver une position 3D.

Tableau-A I-3 Résultats des tests sans bruit

No Requête	Descriptions
SNRSB01	Lors du démarrage de l'application, le C/N_0 est fixé à 62 dBHz sur l'interface alors que le récepteur CMC indique des valeurs de 28 à 29 dBHz pour tous les satellites lorsque la configuration utilisée est dite fonctionnelle.
SNRSB02	Lorsque le test est dit fonctionnel, les valeurs du C/N_0 ne correspondent pas. Elles varient de 10 à 40 dBHz.
SNRSB03	<p>1) Interface : 59 dBHz CMC 26 à 27 d'autres 54 à 55 dBHz</p> <p>2) Le récepteur reçoit 15 satellites, mais en réalité il doit en recevoir un maximum de 8 (restriction du matériel)</p> <p>3) Interface : 52 dBHz CMC : 51 à 53 dBHz d'autres 23 à 24</p> <p>4) Interface : 44 dBHz CMC : 37 à 38 dBHz</p> <p>Conclusion : Le C/N_0 n'est pas constant et il diffère comparativement à ce qui est indiqué dans le logiciel. Le C/N_0 n'est pas calibré.</p>
SNRSB04	En décochant la case Automatic level control de l'interface, il y a maintenant 21 satellites visibles au récepteur CMC (en théorie seulement 8 doivent être visibles par le récepteur)

No Requête	Descriptions
SNRSB05	<p>Une position 3D a été établie au récepteur.</p> <p>1) Condition initiale : sur le récepteur u-blox le C/N_0 constant à 53 dBHz pour tous les satellites, sur l'interface graphique : C/N_0 49</p> <p>Après une minute de stabilisation, le C/N_0 au récepteur est de 52 dBHz</p> <p>2) Autre test $C/N_0 = 57$ dBHz sur l'interface graphique, Après stabilisation $C/N_0 = 52$ dBHz au récepteur</p> <p>Conclusion : Il y a une déficience au niveau de la calibration du récepteur u-blox pour le C/N_0.</p>
SNRSB06	<p>Dans l'interface graphique, en décochant la case Automatic level control et en mettant le C/N_0 à 54, le C/N_0 est devenu stable au récepteur u-blox. Après 5 minutes, le récepteur a perdu le signal satellite et il les a repris quelques secondes plus tard.</p> <p>Conclusion : le contrôle de gain automatique ne semble pas avoir d'impact sur le récepteur.</p>
SNRSB07	<p>Test effectué sur le récepteur u-blox :</p> <p>1) Élévation 0 degré, C/N_0 au récepteur : 48 dBHz</p> <p>2) Élévation 30 degré, le récepteur a perdu le signal de quelques satellites, C/N_0 au récepteur : 48 dBHz, perte de tous les signaux satellites pour quelques secondes et ensuite le récepteur a récupéré une position 3D. Le logiciel simule 6 satellites visibles, mais le récepteur a de la difficulté à récupérer le signal de deux satellites</p> <p>3) Élévation 90 degré, le récepteur a perdu le signal de tous les satellites, car le logiciel n'envoie aucune donnée de satellites.</p> <p>4) Élévation 56 degré, le logiciel envoie les données de 4 satellites et le récepteur u-blox a de la difficulté à récupérer les données de navigation (bande bleue), Le C/N_0 au récepteur est de 45 et 49 dBHz.</p> <p>Conclusion : une fois que le récepteur u-blox perd les signaux (le logiciel n'envoie pas de signal, manœuvre volontaire), il tarde à récupérer les signaux une fois que le logiciel les envoie. Après plusieurs minutes de simulation, la position n'est toujours pas récupérée. Dans une simulation réelle, l'angle d'élévation a un impact sur le rapport C/N_0. D'après ce test, ceci n'a pas d'effet. Donc, le C/N_0 n'est pas réaliste.</p>

Tableau-A I-4 Résultats des tests avec bruit

No Requête	Descriptions
SNRAB01	En changeant la valeur de la densité de bruit dans l'interface graphique, les valeurs de la colonne bruit du récepteur CMC reste toujours constantes.
SNRAB02	<p>Test effectué sur le récepteur CMC :</p> <p>1) Condition initiale : Interface $C/N_0 = 66$ dBHz, Récepteur : 43 à 44 dBHz</p> <p>2) Interface : $C/N_0 = 52$ dBHz, Récepteur CMC : 43 à 44 dBHz Quand la nouvelle valeur du C/N_0 a été appliqué, on a perdu le signal des satellites (il y a seulement trois satellites que le récepteur était capable de recevoir)</p> <p>3) Interface : $C/N_0 = 44$ dBHz , Récepteur CMC : 43 à 44 dBHz</p> <p>Conclusion : le C/N_0 ne change pas quand il y a du bruit.</p>
SNRAB03	Il serait préférable d'ajouter une option à l'interface graphique permettant d'activer et de désactiver le bruit (canal 11) plutôt que de toujours refaire la configuration du montage.

Tableau-A I-5 Problèmes divers observés

No Requête	Descriptions
DIV01	Il n'est pas possible de partir plusieurs scénarios un à la suite de l'autre de manière automatisée.
DIV02	Il y a seulement le véhicule avion qui est implémenté.
DIV03	L'obtention d'une solution 3D est longue à obtenir voir non atteignable.
DIV04	Dans le récepteur CMC, il est toujours écrit ERROR dans la colonne SV Type.
DIV05	La colonne Carrier Freq du tableau du récepteur CMC est toujours vide.
DIV06	Il serait préférable d'ajouter un type de véhicule pour la marche à pied.
DIV07	L'effet de l'atmosphère sur le simulateur n'a pas été observé.
DIV08	L'effet du patron de l'antenne et du Thermal noise sur le simulateur n'a pas été observé.
DIV09	Manque de documentation concernant le logiciel, diagramme de classe, diagramme de séquence, UML
DIV010	Manque de commentaires dans les fonctions du logiciel ainsi que pour le FPGA. Il y a très peu de commentaires pour le code du FPGA.

ANNEXE II

RÉSULTATS DES PARAMÈTRES DE NAVIGATION DU SIMULATEUR LACIME

Voici les résultats des paramètres de navigation du simulateur LACIME.

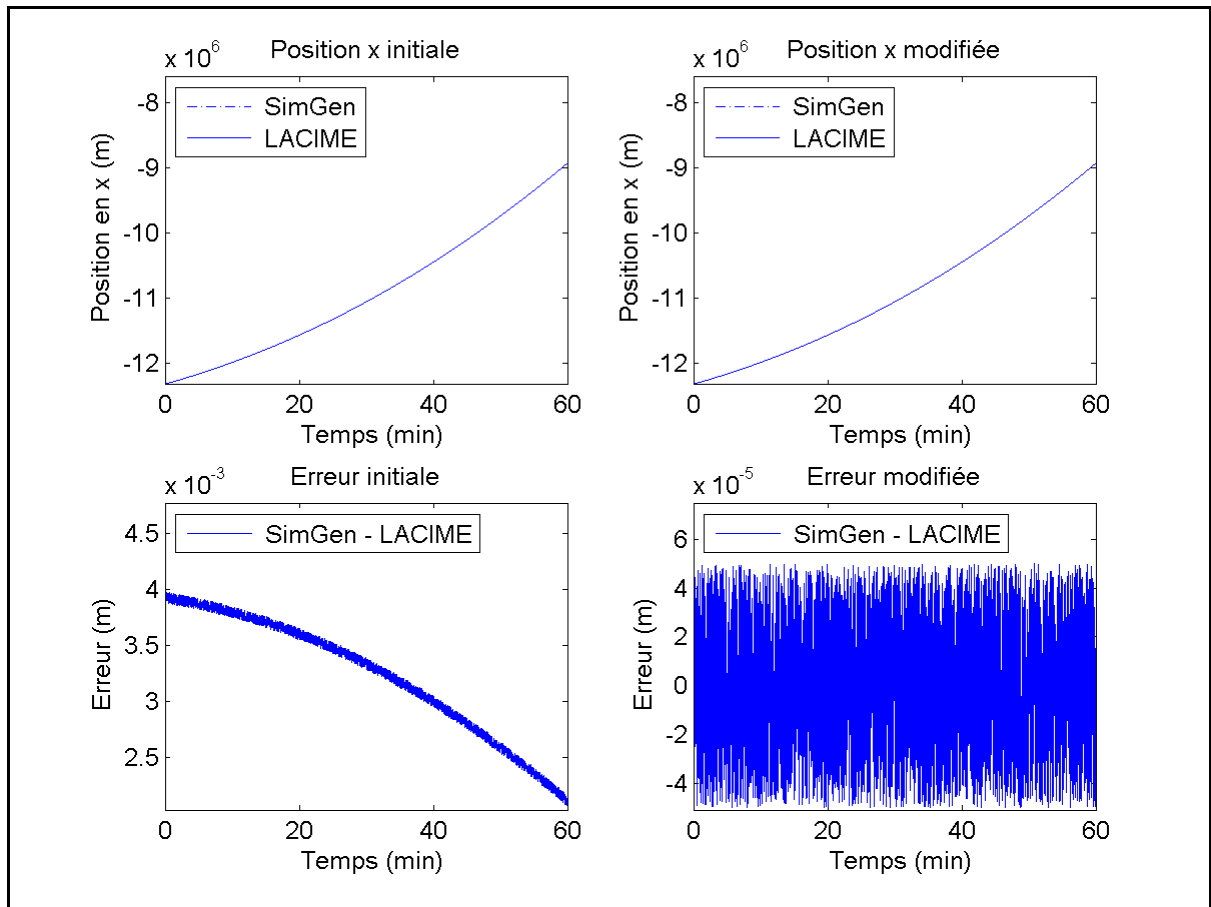


Figure-A II-1 Position en x et son erreur initiale et modifiée du satellite #2

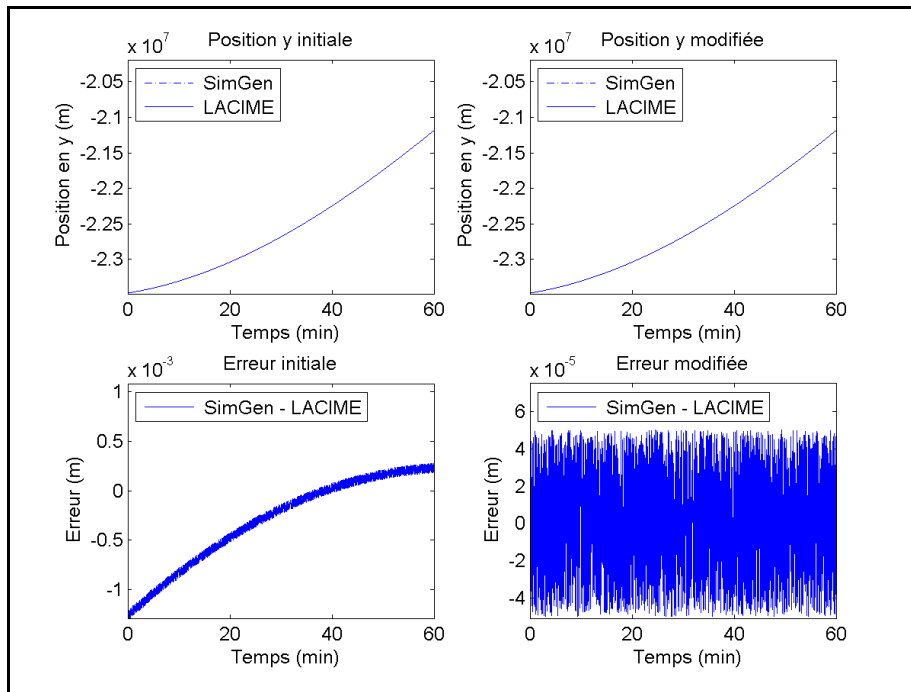


Figure-A II-2 Position en y et son erreur initiale et modifiée du satellite #2

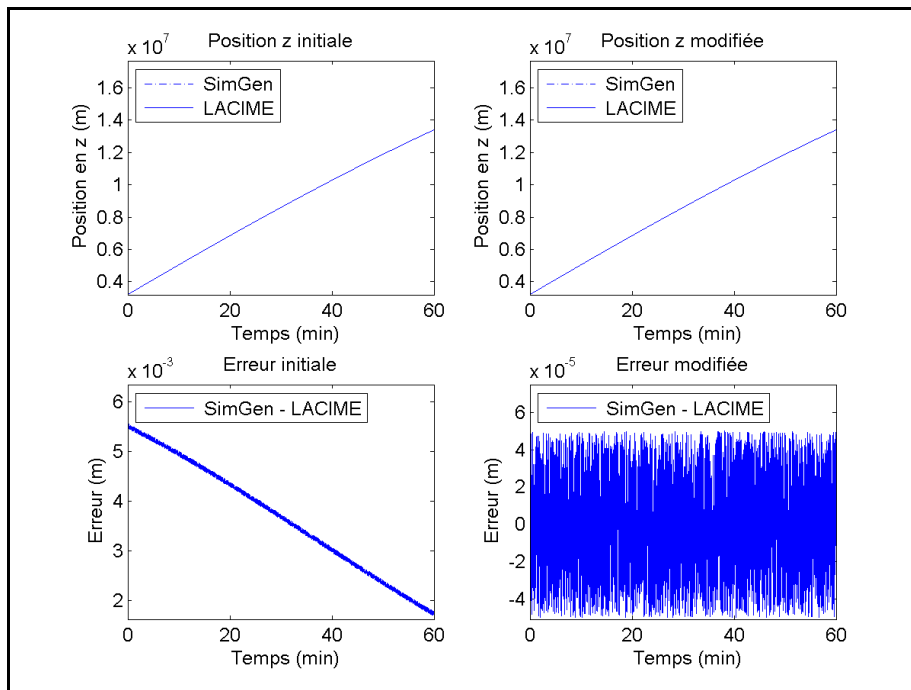


Figure-A II-3 Position en z et son erreur initiale et modifiée du satellite #2

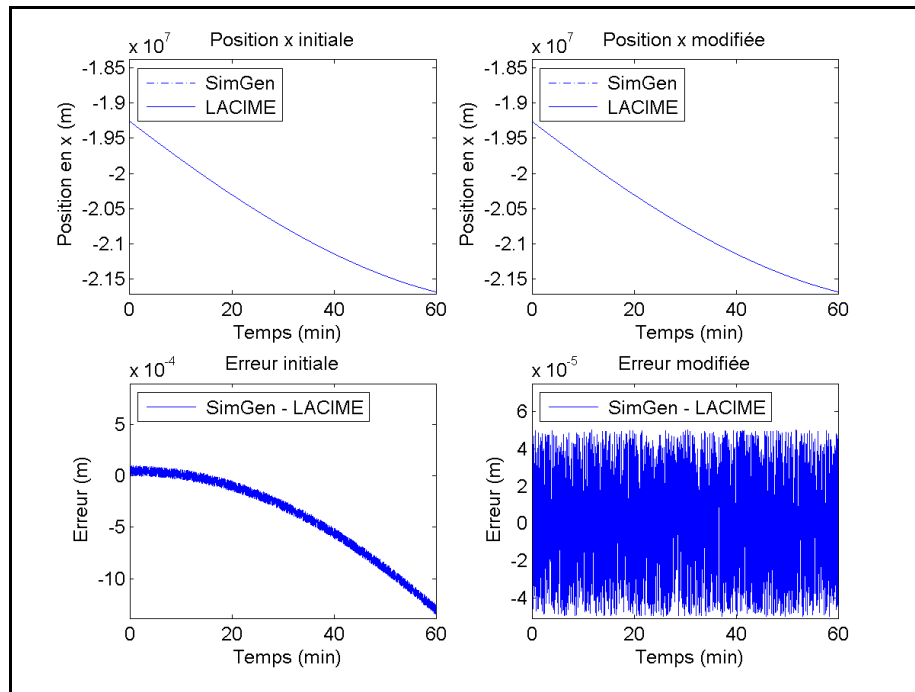


Figure-A II-4 Position en x et son erreur initiale et modifiée du satellite #9

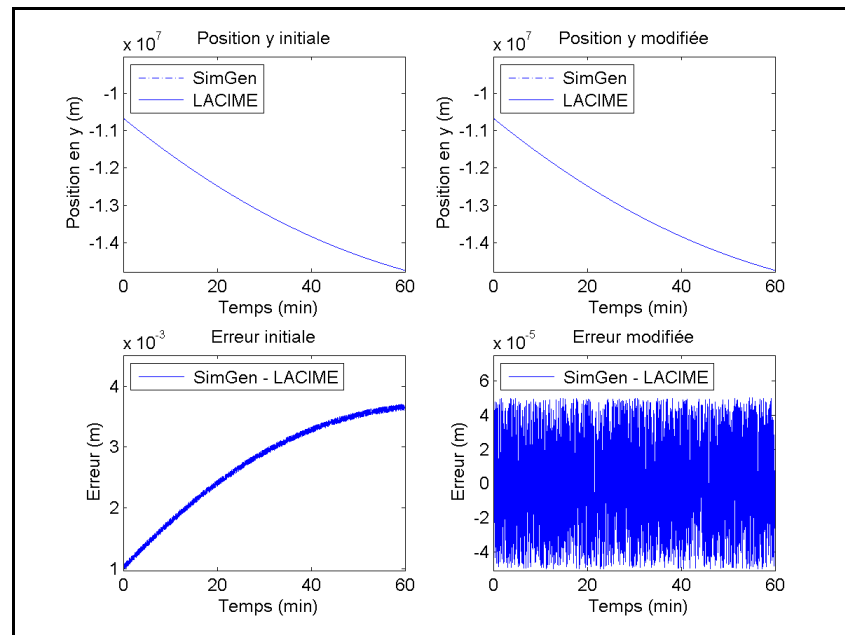


Figure-A II-5 Position en y et son erreur initiale et modifiée du satellite #9

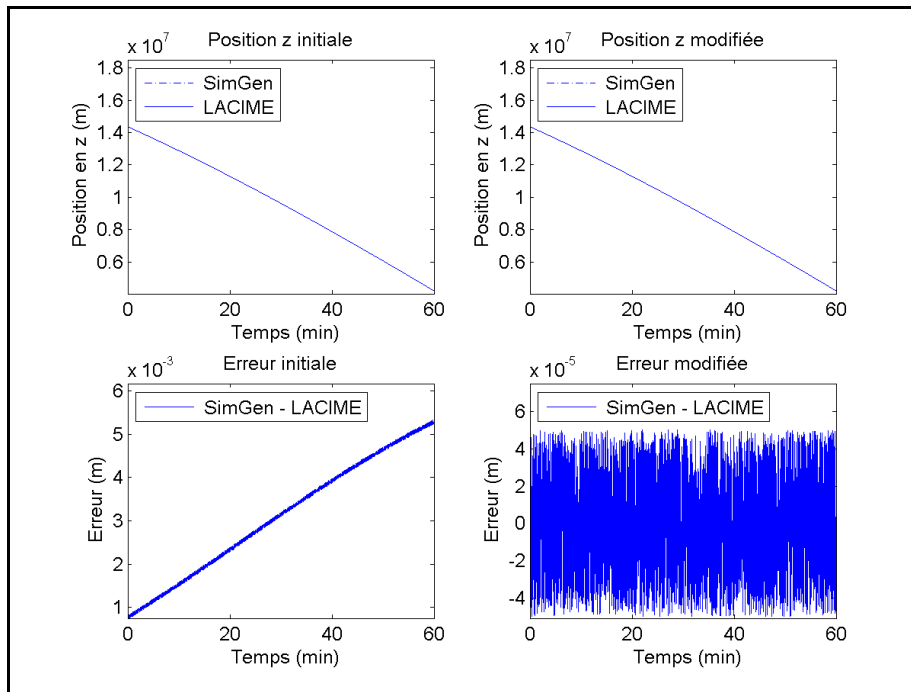


Figure-A II-6 Position en z et son erreur initiale et modifiée du satellite #9

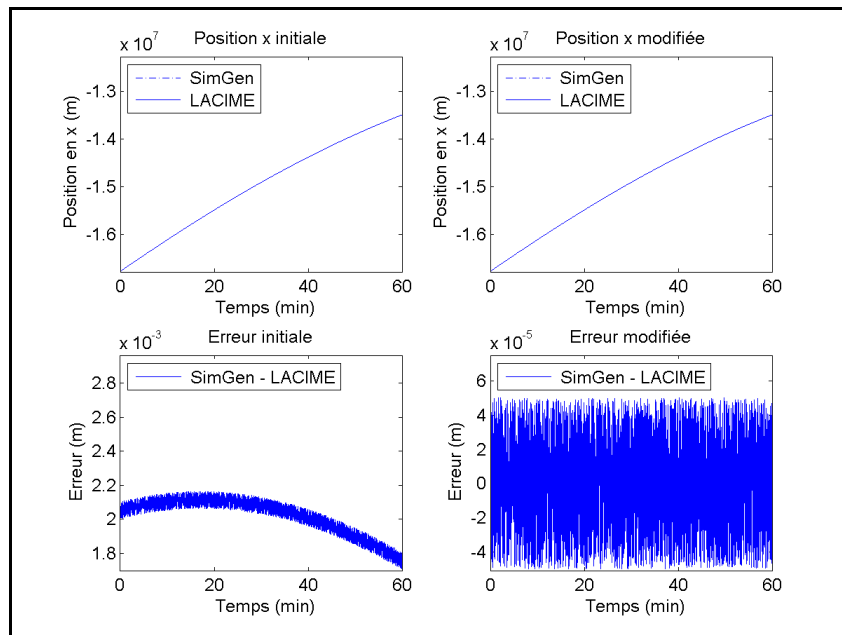


Figure-A II-7 Position en x et son erreur initiale et modifiée du satellite #12

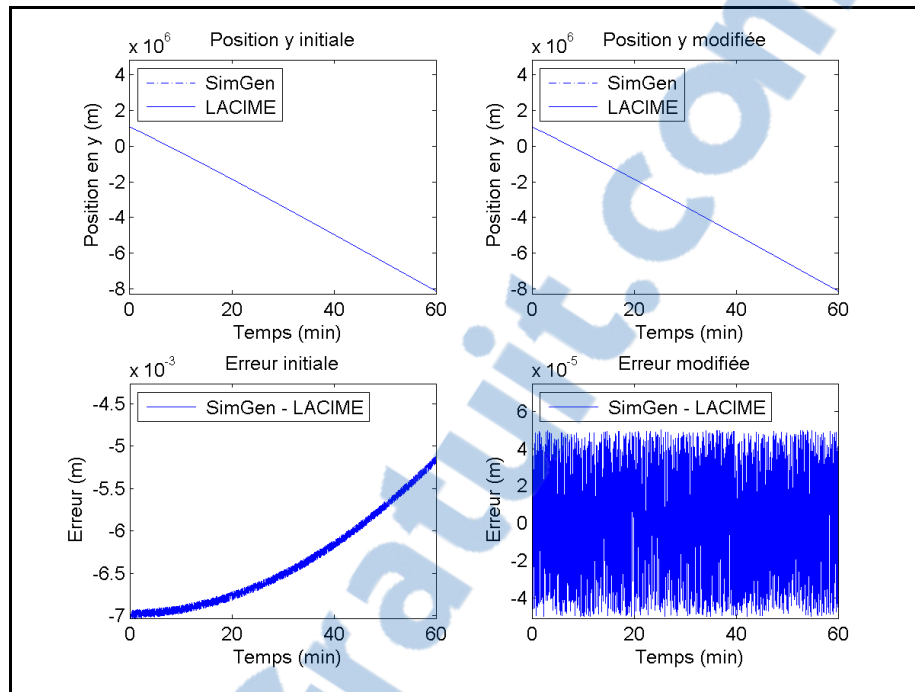


Figure-A II-8 Position en y et son erreur initiale et modifiée du satellite #12

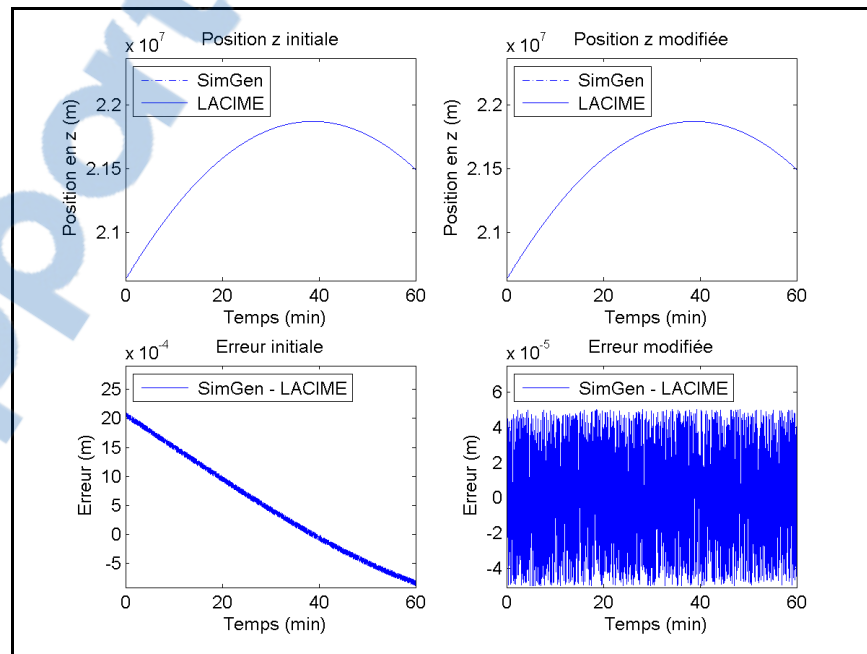


Figure-A II-9 Position en z et son erreur initiale et modifiée du satellite #12

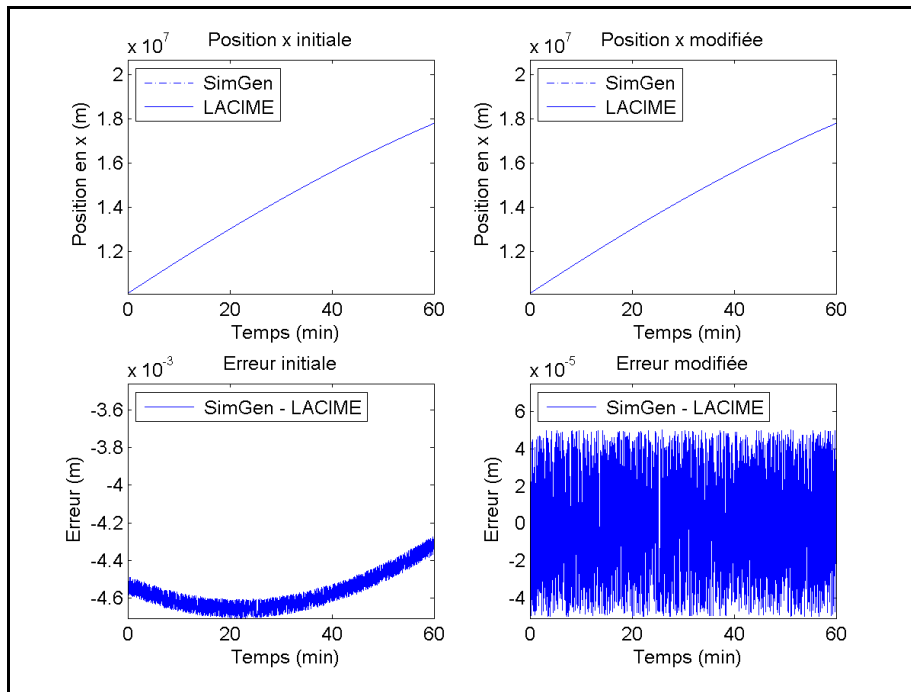


Figure-A II-10 Position en x et son erreur initiale et modifiée du satellite #17

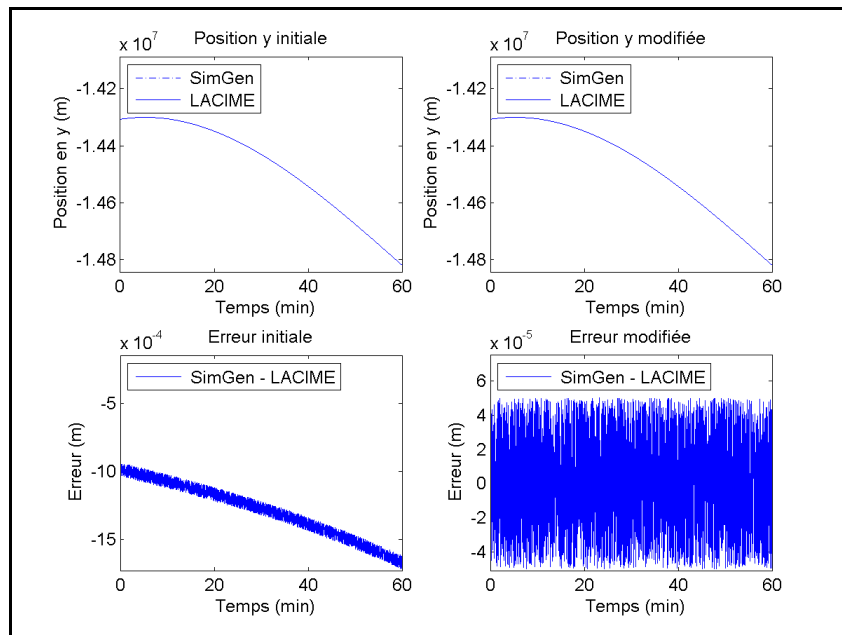


Figure-A II-11 Position en y et son erreur initiale et modifiée du satellite #17

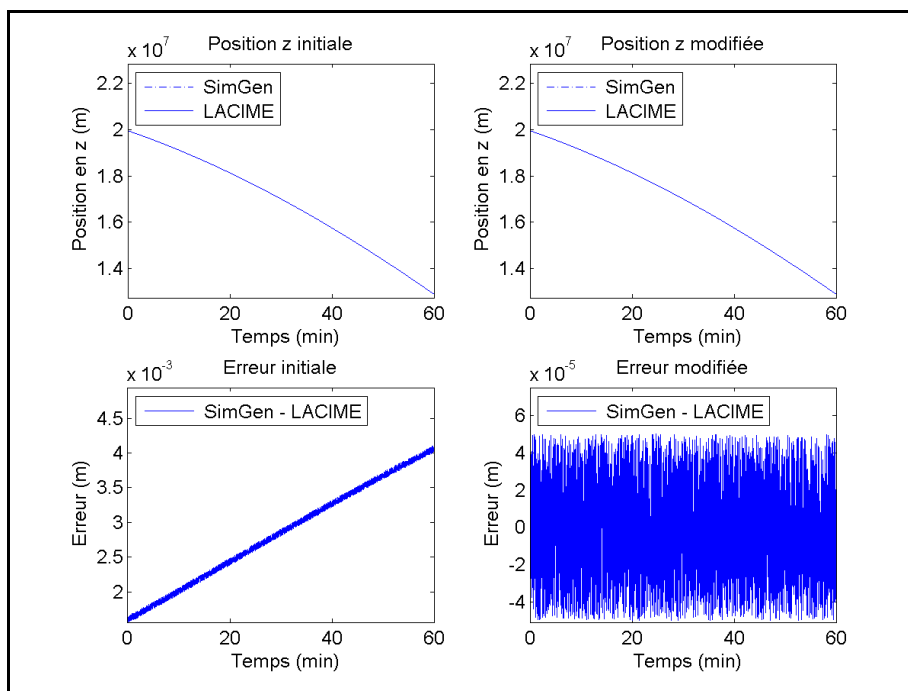


Figure-A II-12 Position en z et son erreur initiale et modifiée du satellite #17

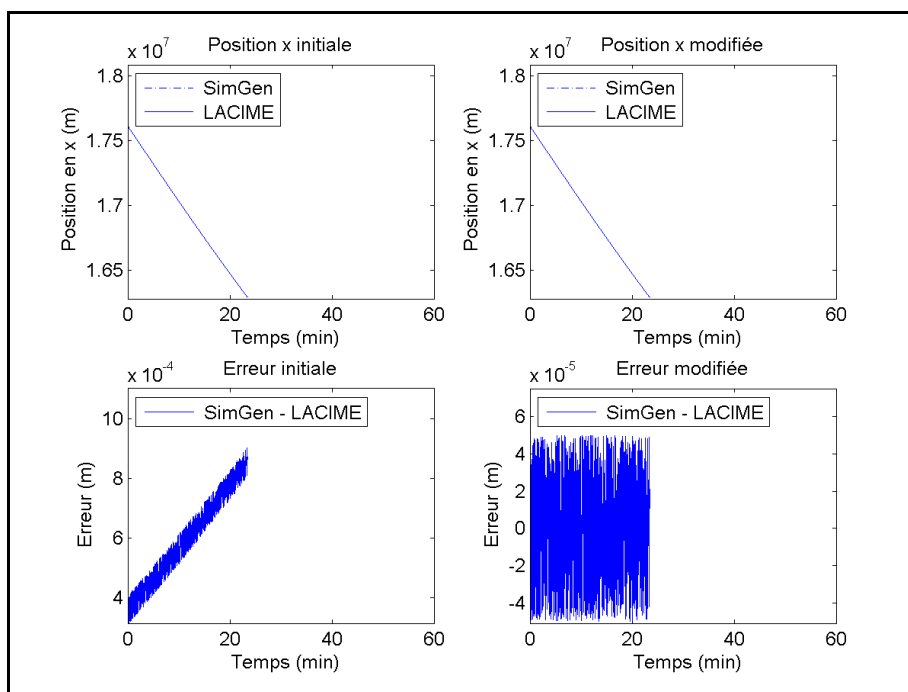


Figure-A II-13 Position en x et son erreur initiale et modifiée du satellite #20

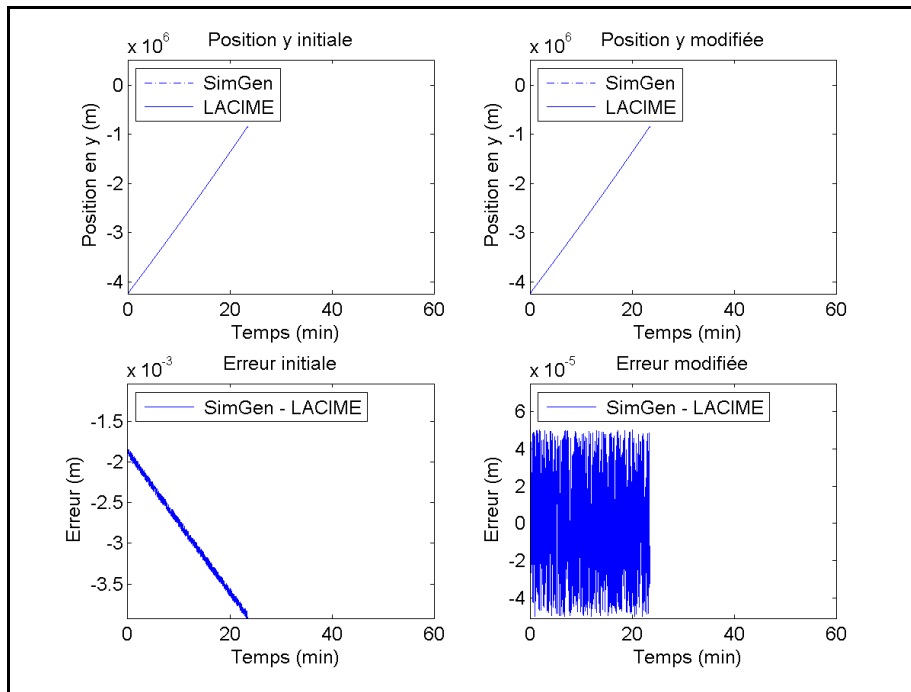


Figure-A II-14 Position en y et son erreur initiale et modifiée du satellite #20

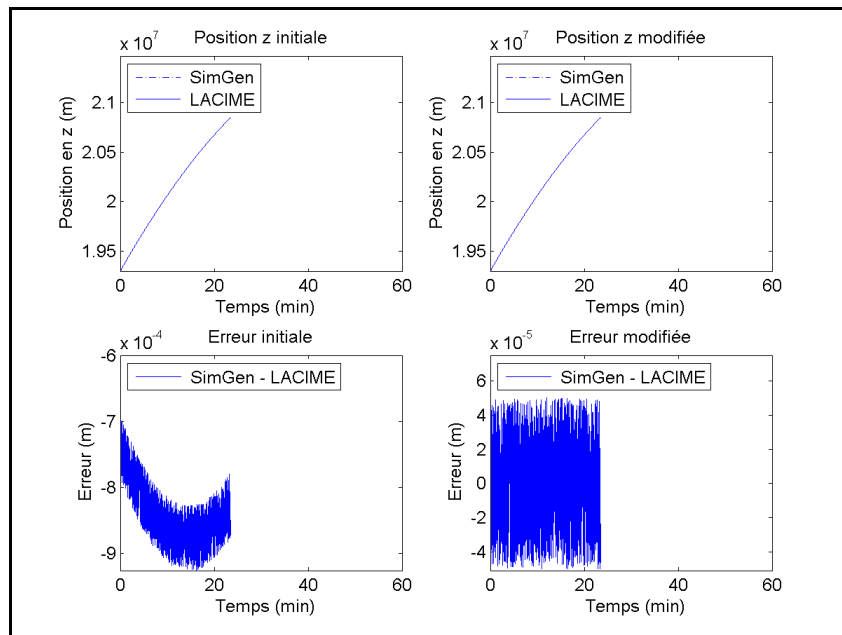


Figure-A II-15 Position en z et son erreur initiale et modifiée du satellite #20

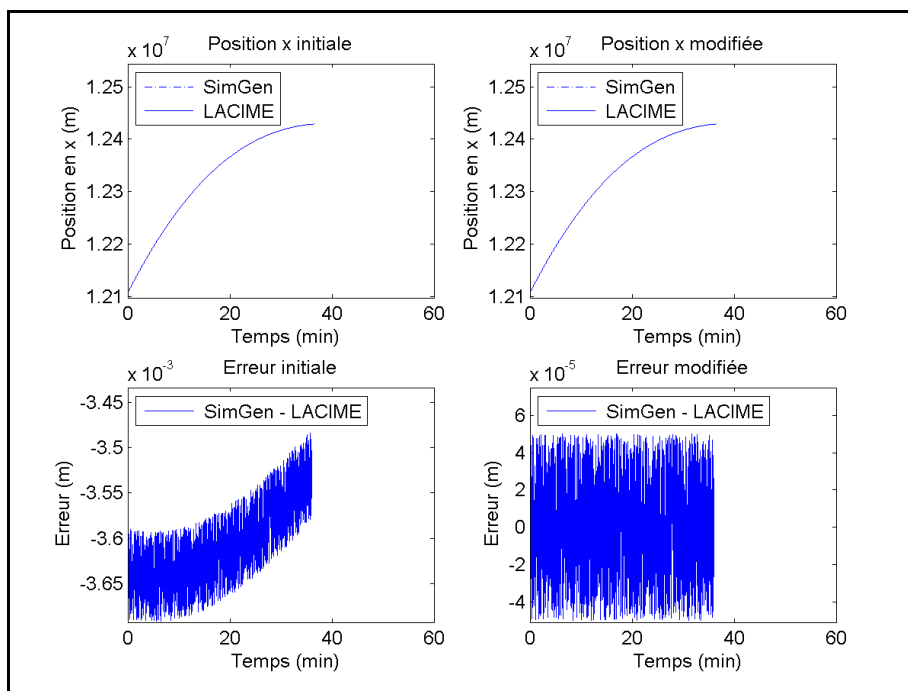


Figure-A II-16 Position en x et son erreur initiale et modifiée du satellite #28

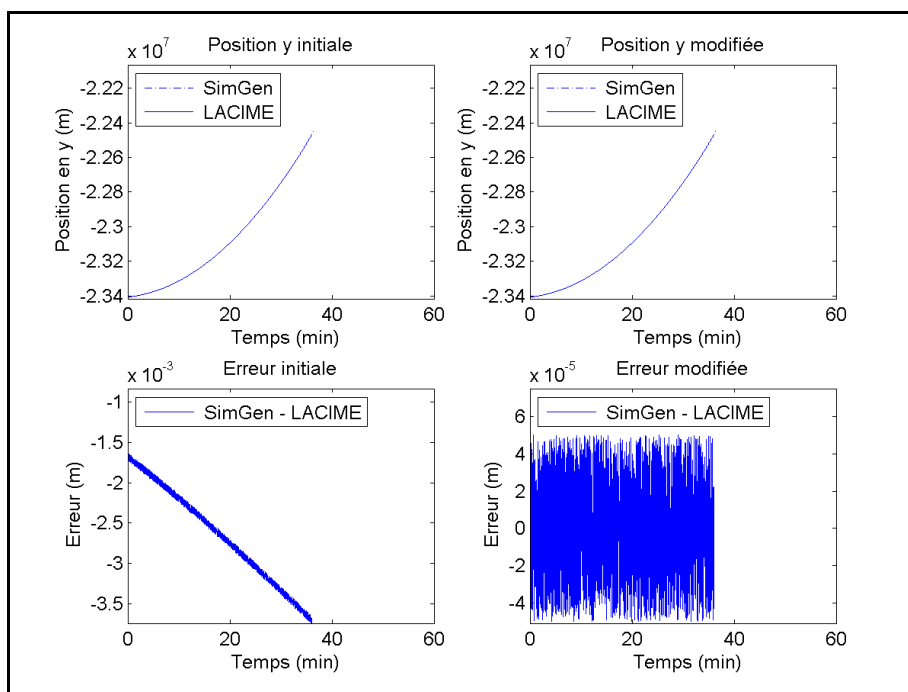


Figure-A II-17 Position en y et son erreur initiale et modifiée du satellite #28

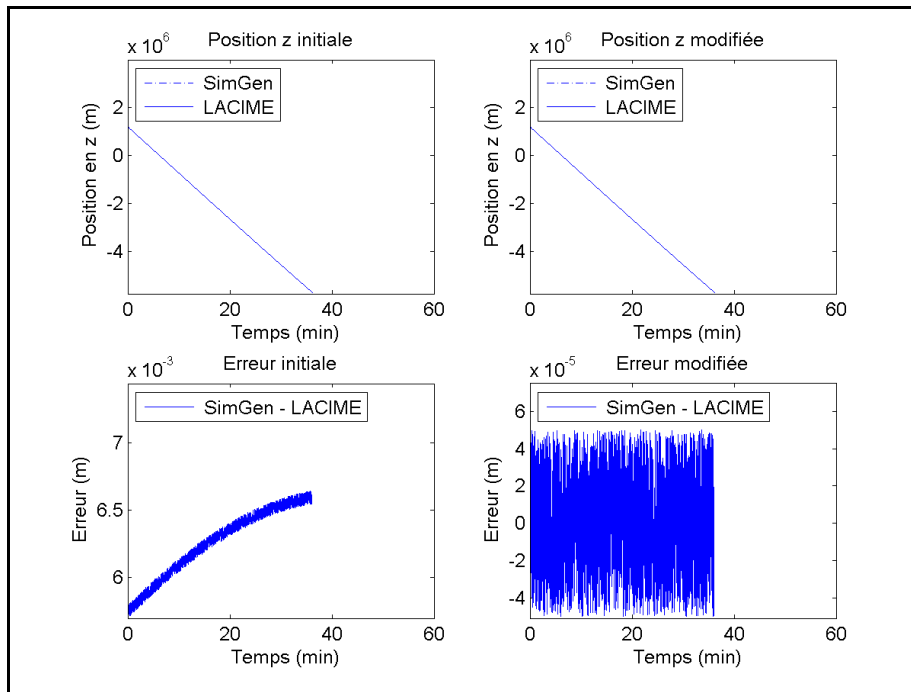


Figure-A II-18 Position en z et son erreur initiale et modifiée du satellite #28

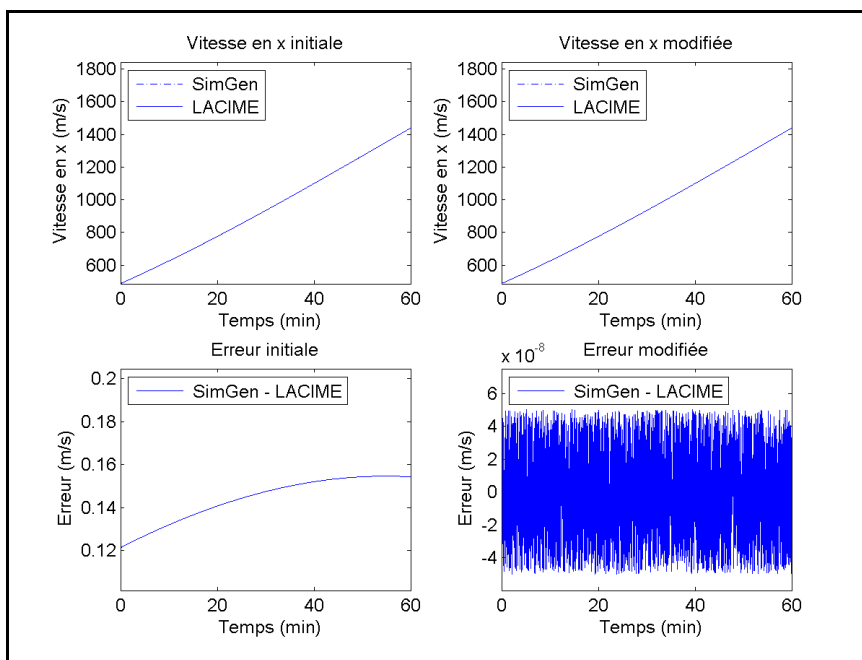


Figure-A II-19 Vitesse en x et son erreur initiale et modifiée du satellite #2

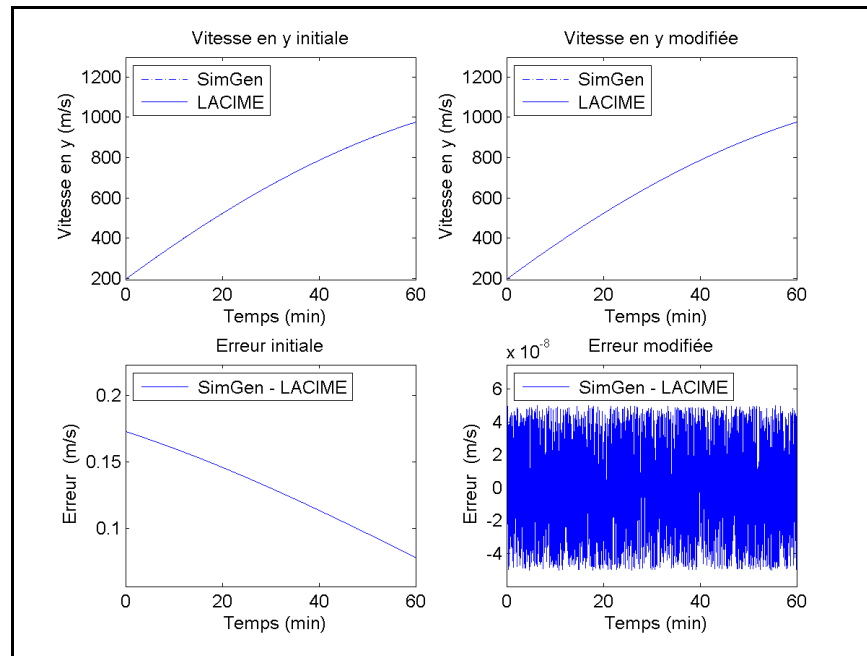


Figure-A II-20 Vitesse en y et son erreur initiale et modifiée du satellite #2

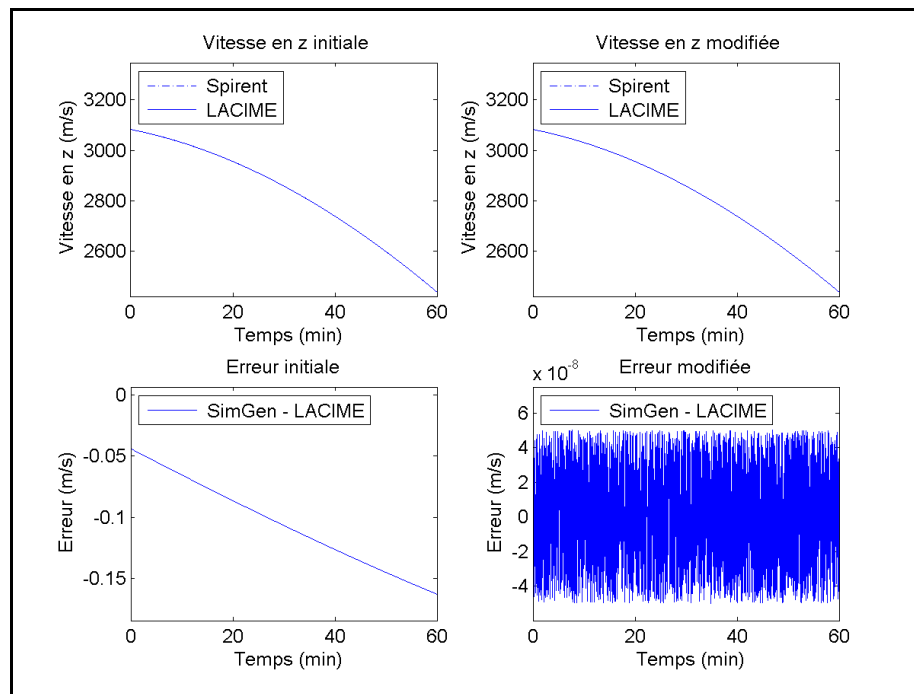


Figure-A II-21 Vitesse en z et son erreur initiale et modifiée du satellite #2

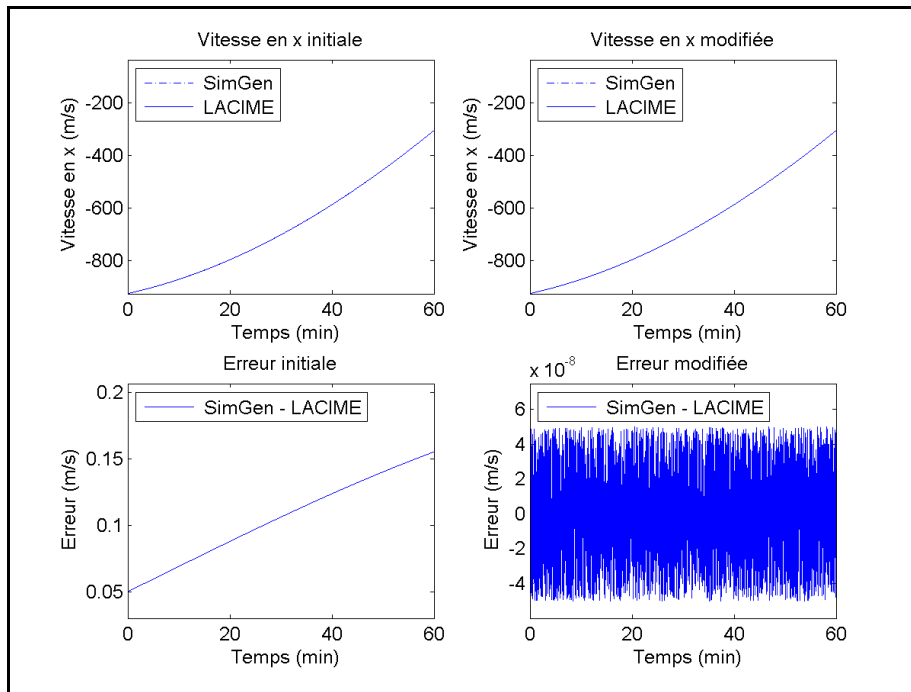


Figure-A II-22 Vitesse en x et son erreur initiale et modifiée du satellite #9

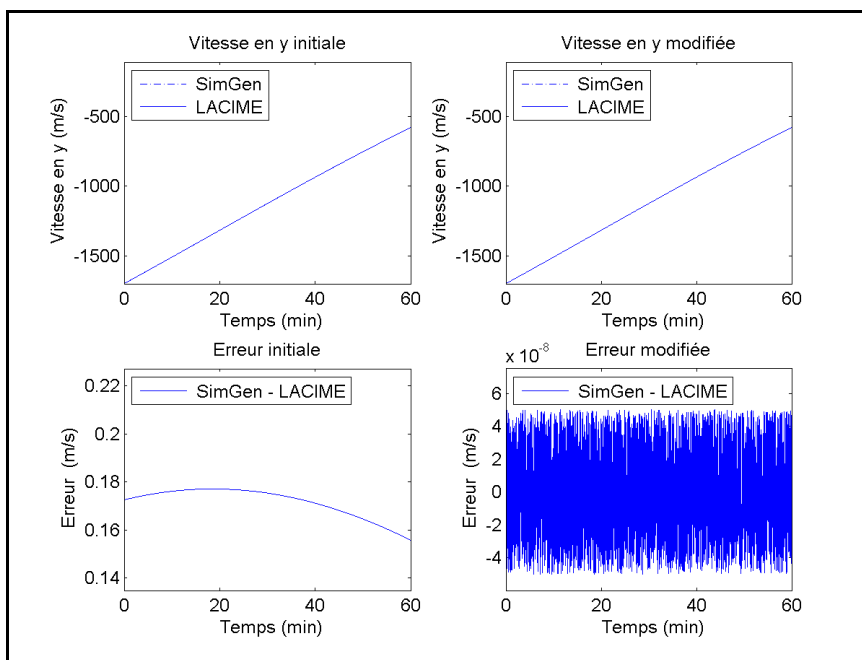


Figure-A II-23 Vitesse en y et son erreur initiale et modifiée du satellite #9

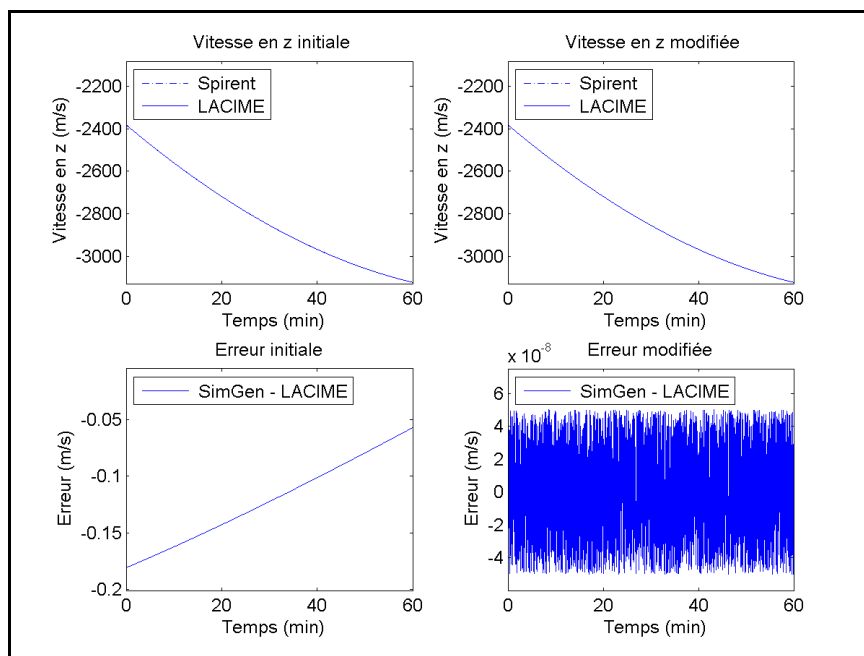


Figure-A II-24 Vitesse en z et son erreur initiale et modifiée du satellite #9

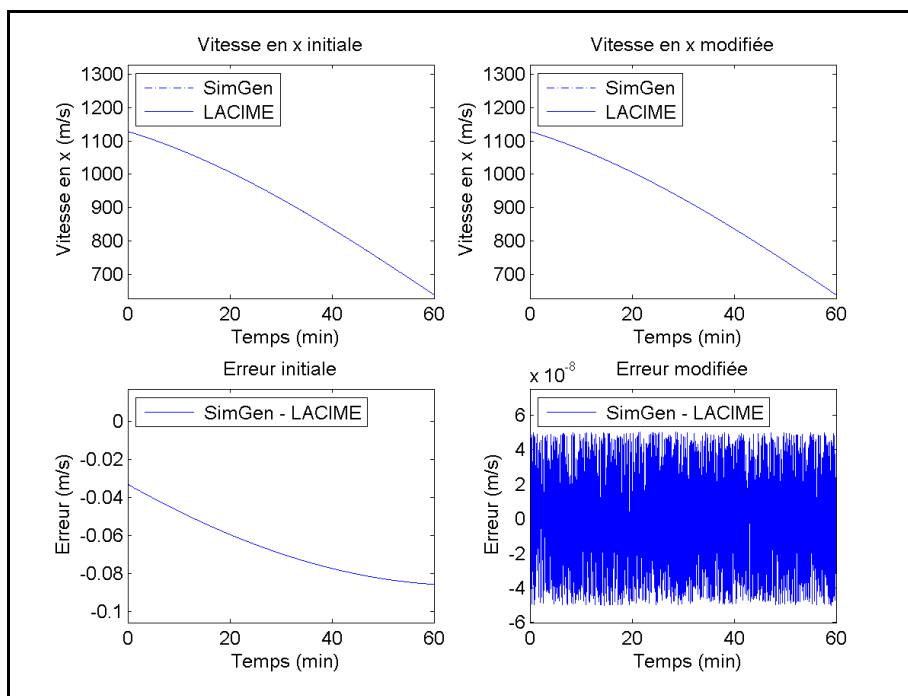


Figure-A II-25 Vitesse en x et son erreur initiale et modifiée du satellite #12

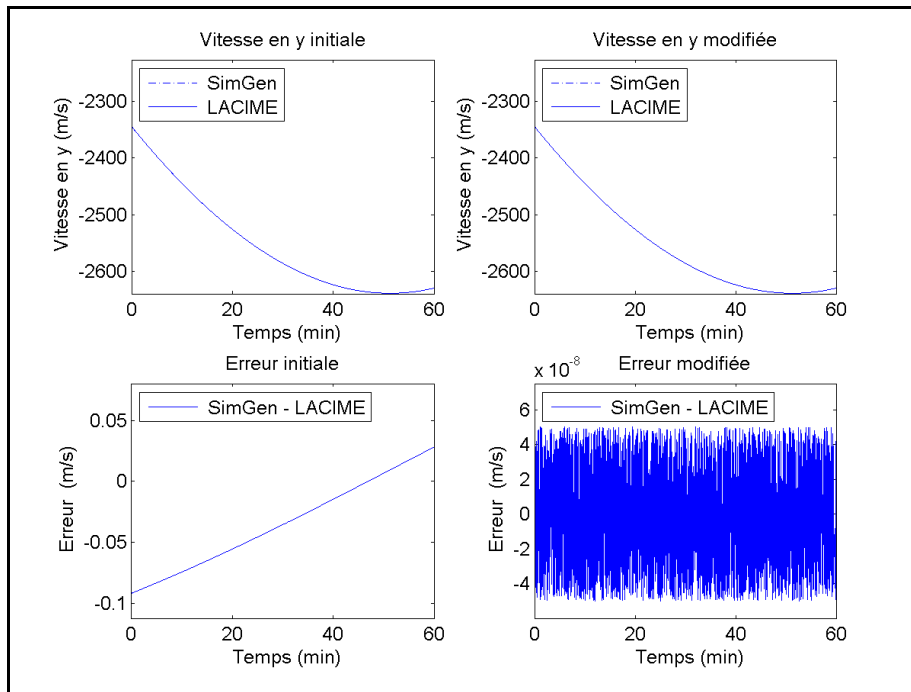


Figure-A II-26 Vitesse en y et son erreur initiale et modifiée du satellite #12

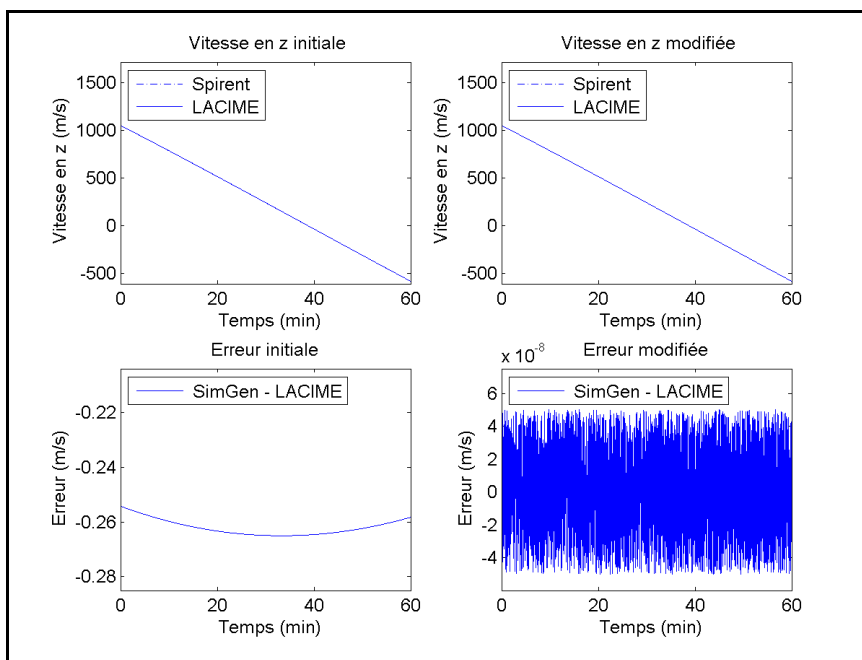


Figure-A II-27 Vitesse en z et son erreur initiale et modifiée du satellite #12

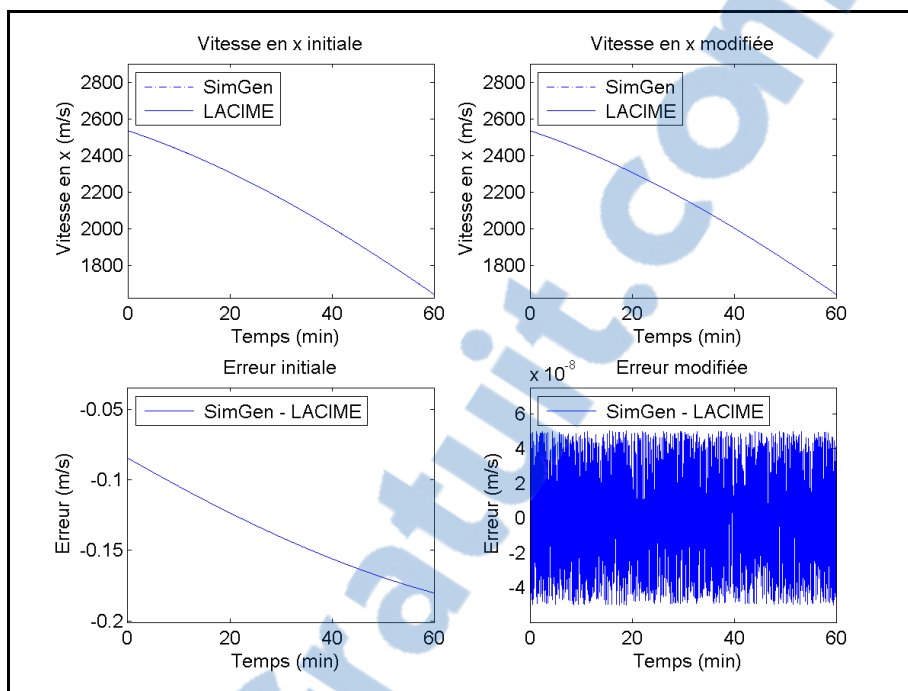


Figure-A II-28 Vitesse en x et son erreur initiale et modifiée du satellite #17

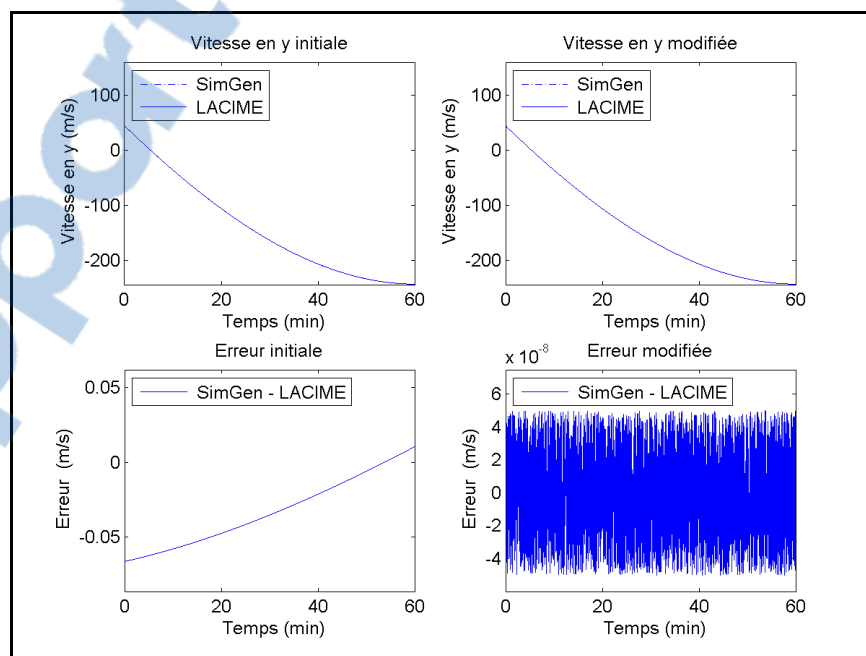


Figure-A II-29 Vitesse en y et son erreur initiale et modifiée du satellite #17

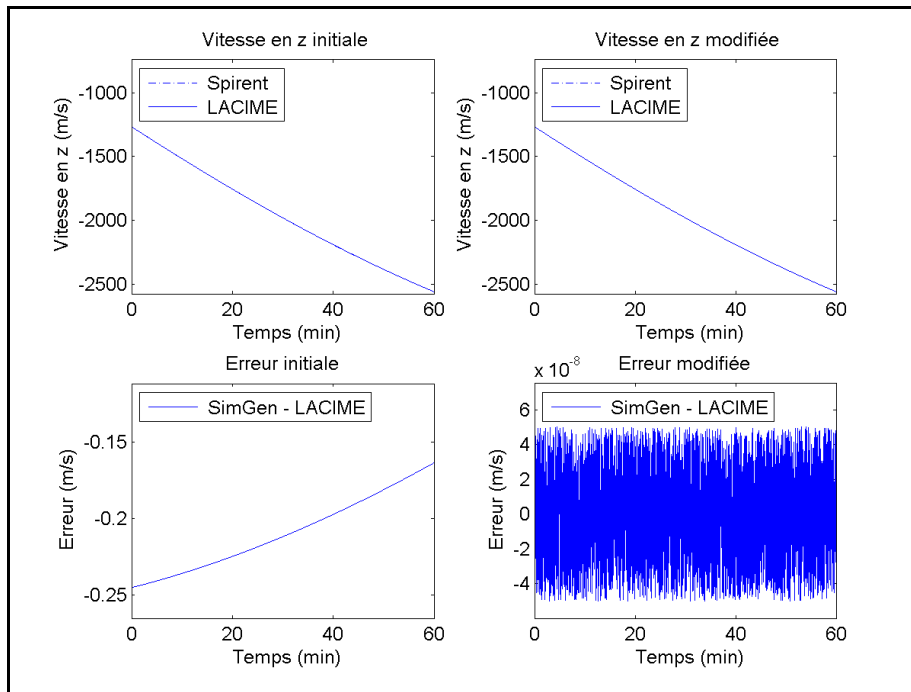


Figure-A II-30 Vitesse en z et son erreur initiale et modifiée du satellite #17

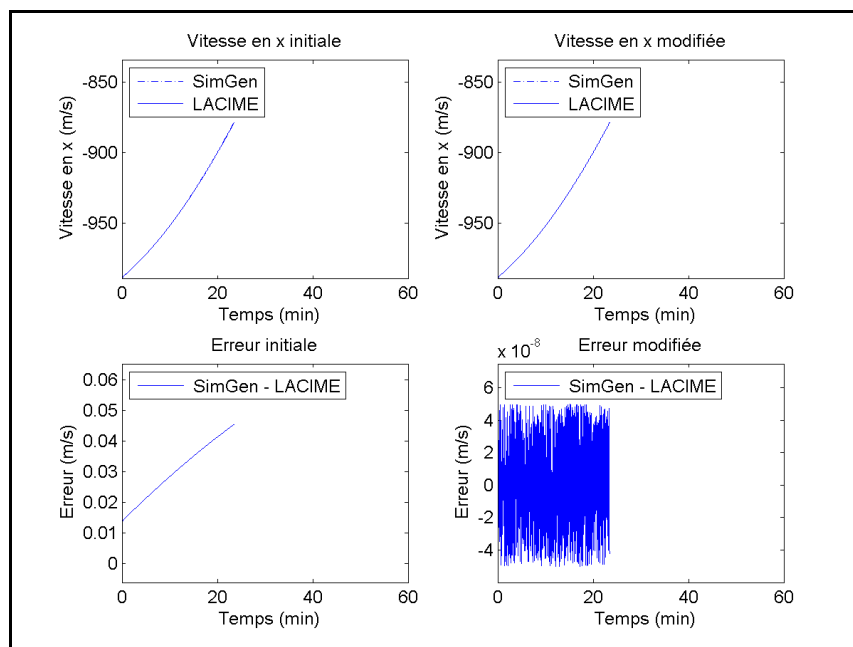


Figure-A II-31 Vitesse en x et son erreur initiale et modifiée du satellite #20

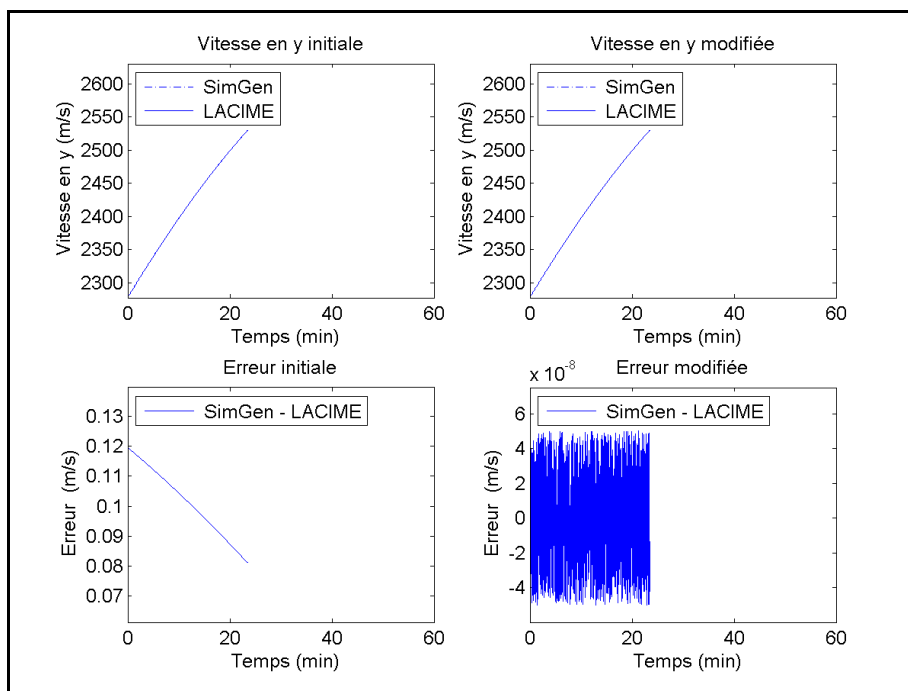


Figure-A II-32 Vitesse en y et son erreur initiale et modifiée du satellite #20

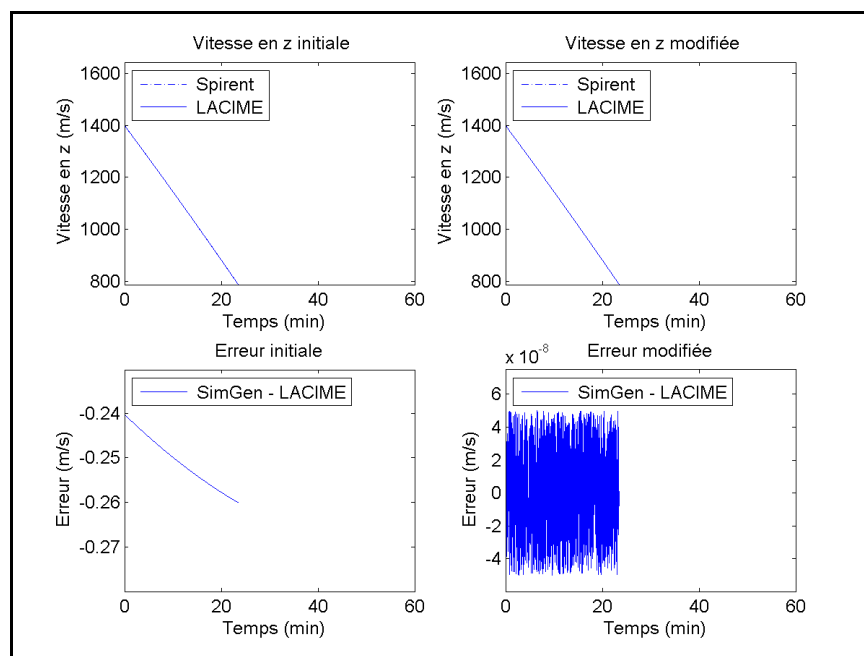


Figure-A II-33 Vitesse en z et son erreur initiale et modifiée du satellite #20

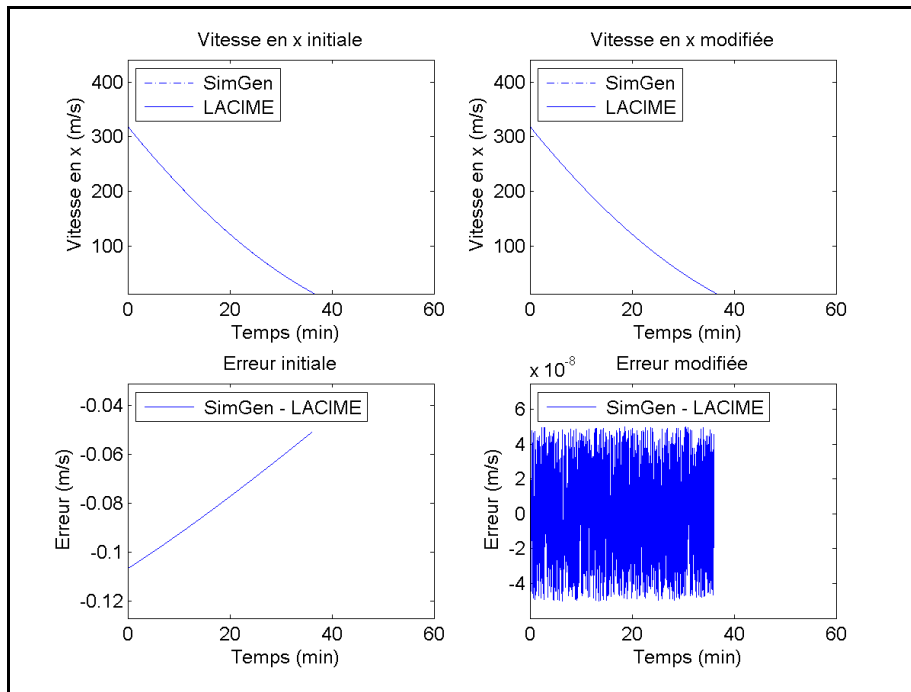


Figure-A II-34 Vitesse en x et son erreur initiale et modifiée du satellite #28

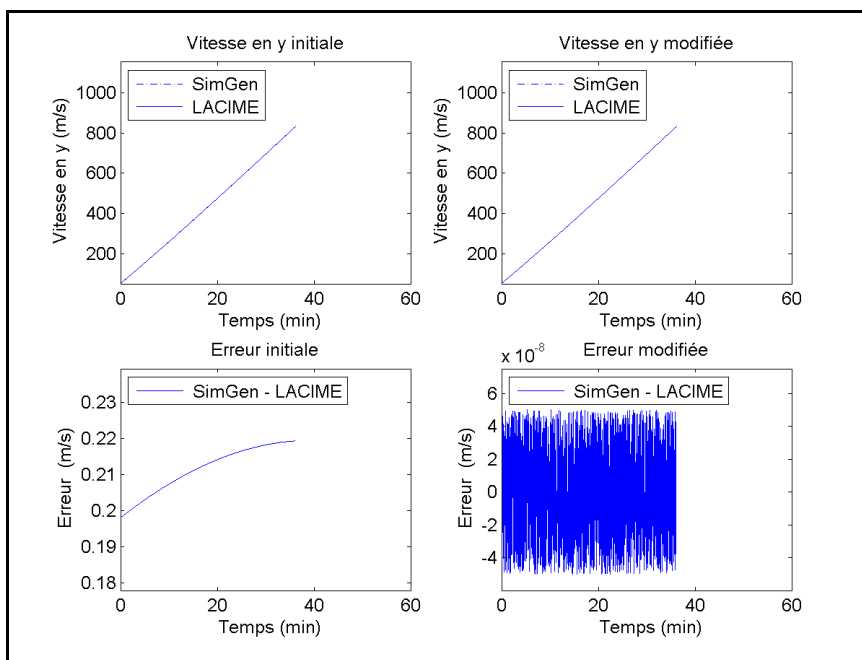


Figure-A II-35 Vitesse en y et son erreur initiale et modifiée du satellite #28

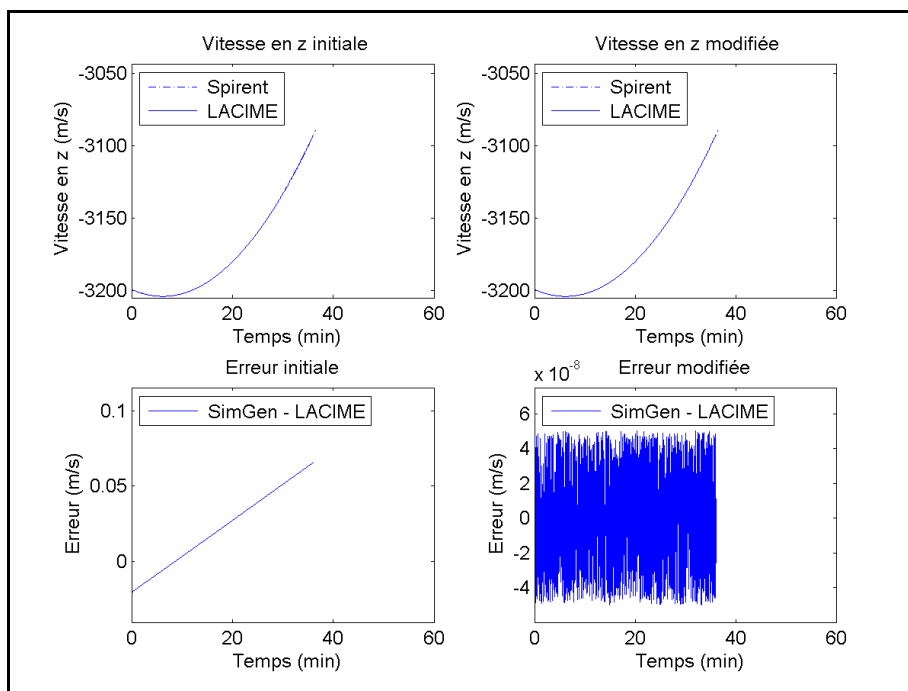


Figure-A II-36 Vitesse en z et son erreur initiale et modifiée du satellite #28

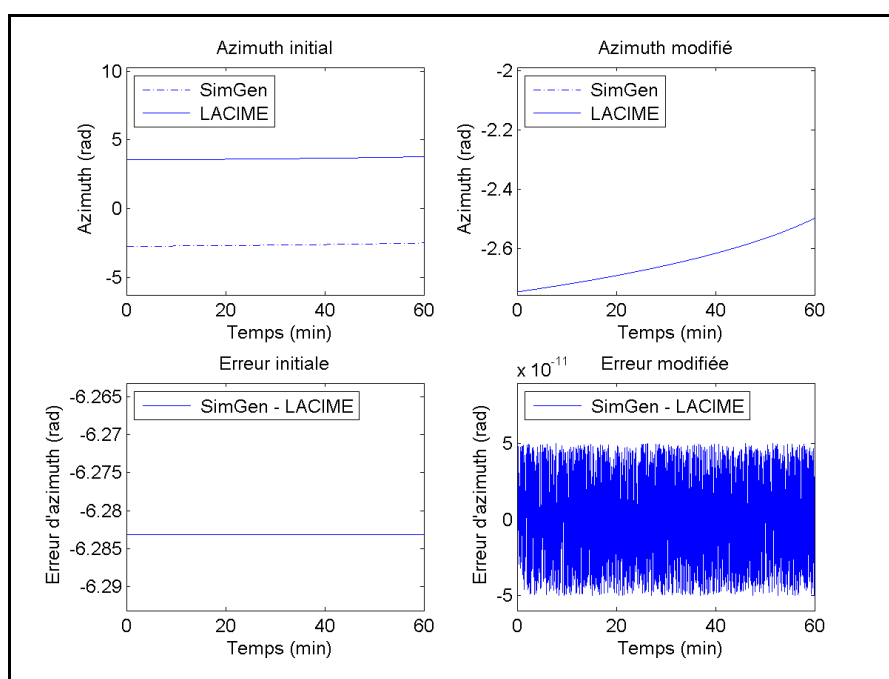


Figure-A II-37 L'angle d'azimuth son erreur initiale et modifiée du satellite #2

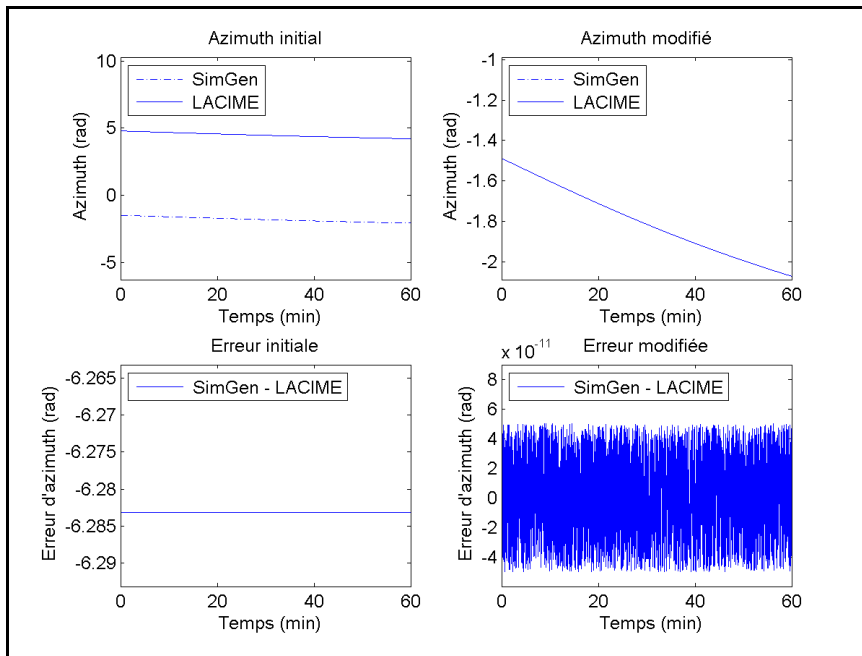


Figure-A II-38 L'angle d'azimuth son erreur initiale et modifiée du satellite #9

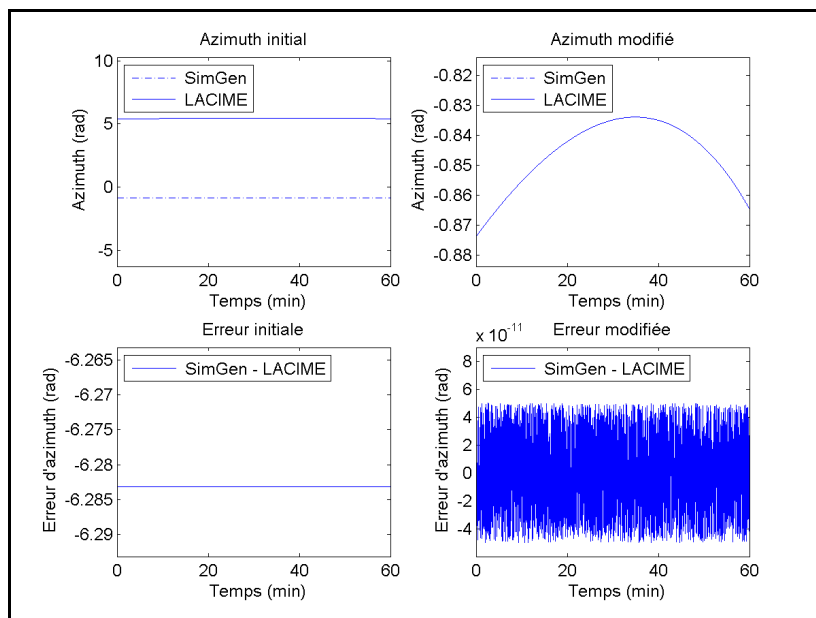


Figure-A II-39 L'angle d'azimuth son erreur initiale et modifiée du satellite #12

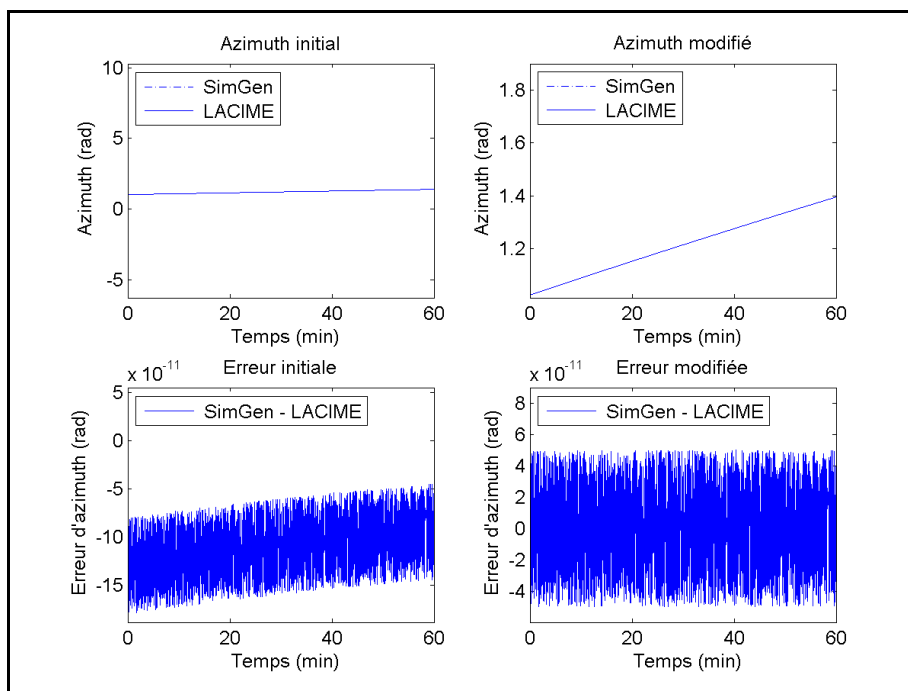


Figure-A II-40 L'angle d'azimuth son erreur initiale et modifiée du satellite #17

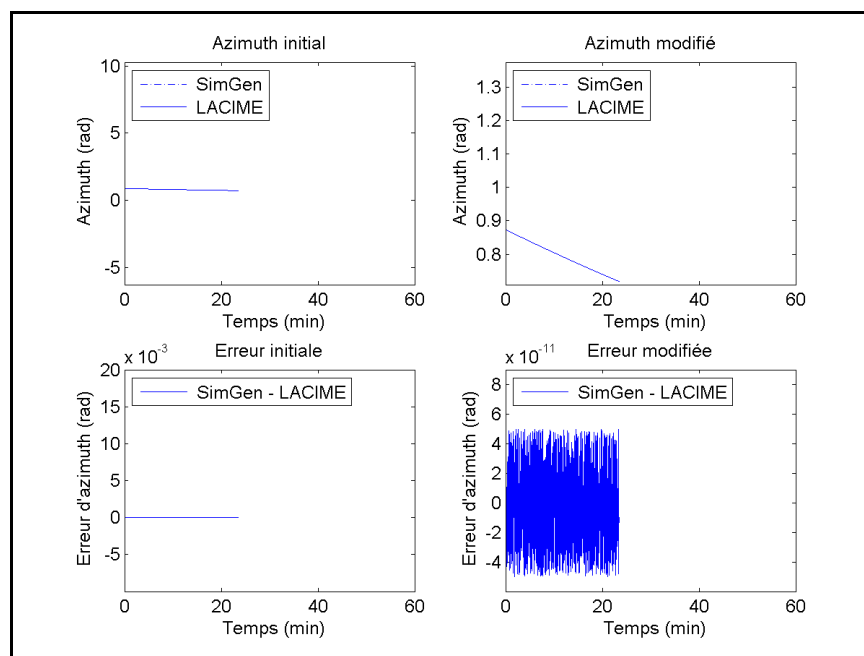


Figure-A II-41 L'angle d'azimuth son erreur initiale et modifiée du satellite #20

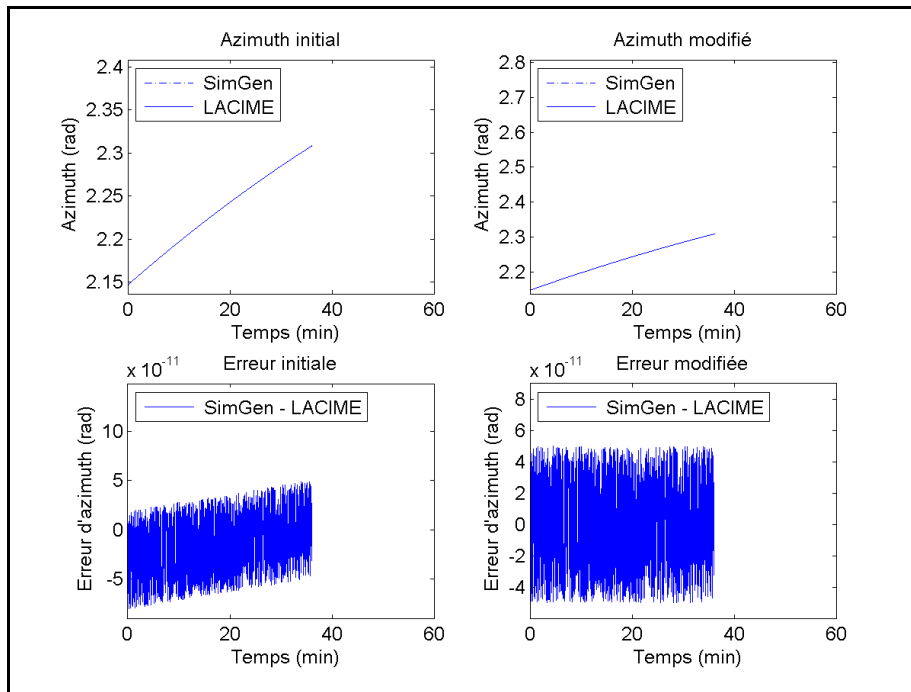


Figure-A II-42 L'angle d'azimuth son erreur initiale et modifiée du satellite #28

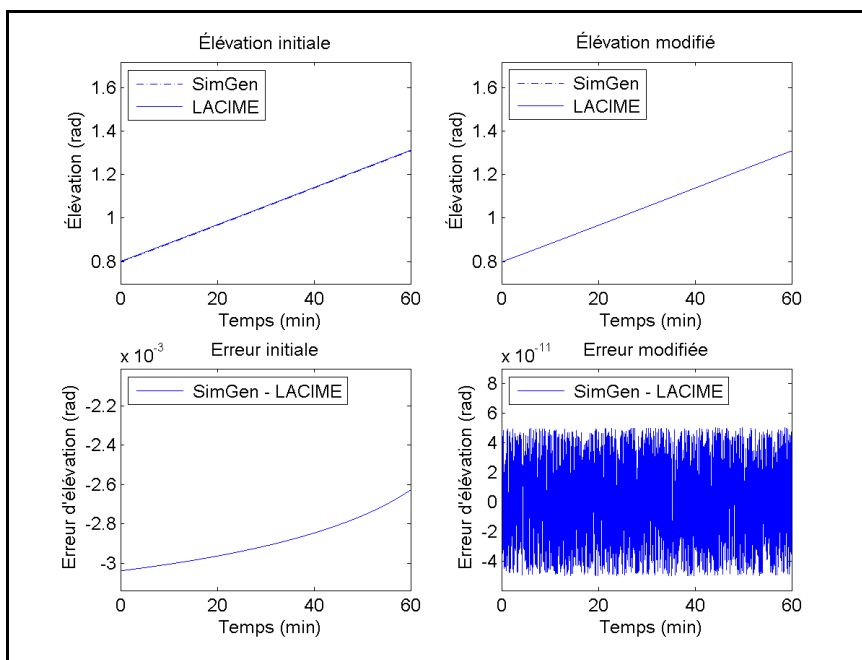


Figure-A II-43 L'angle d'élévation son erreur initiale et modifiée du satellite #2

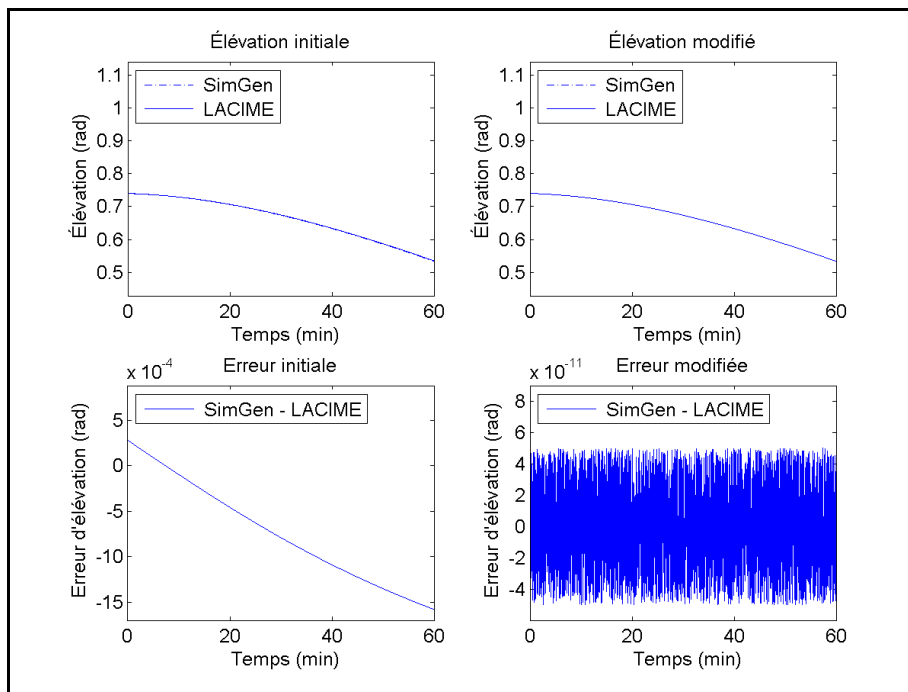


Figure-A II-44 L'angle d'élévation son erreur initiale et modifiée du satellite #9

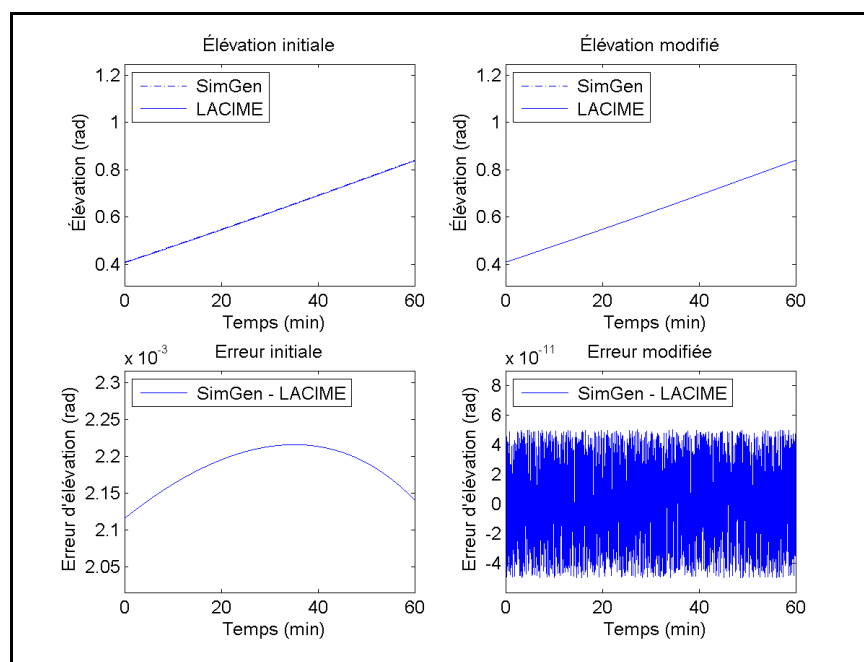


Figure-A II-45 L'angle d'élévation son erreur initiale et modifiée du satellite #12

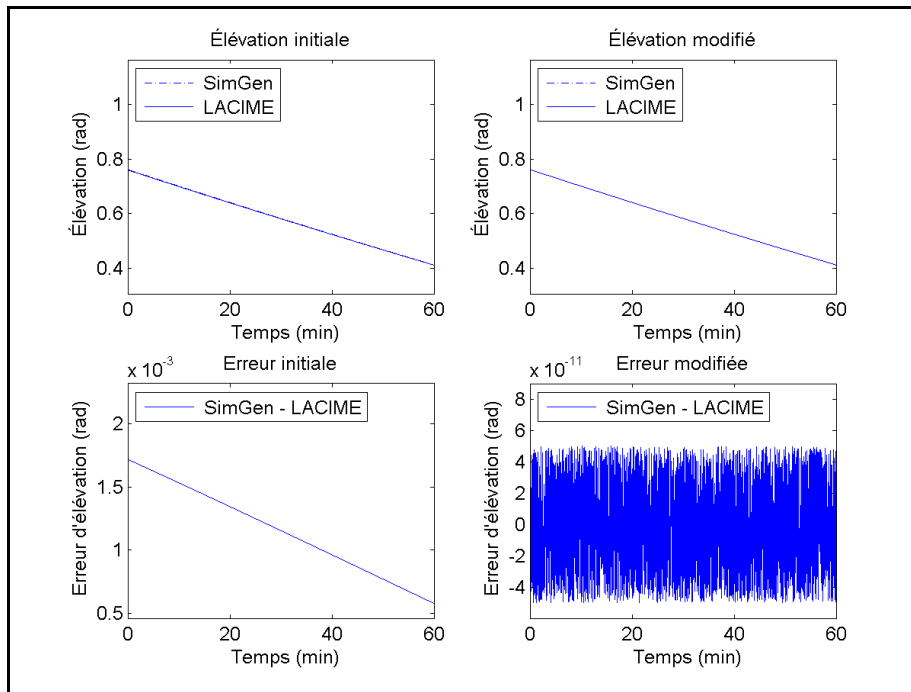


Figure-A II-46 L'angle d'élévation son erreur initiale et modifiée du satellite #17

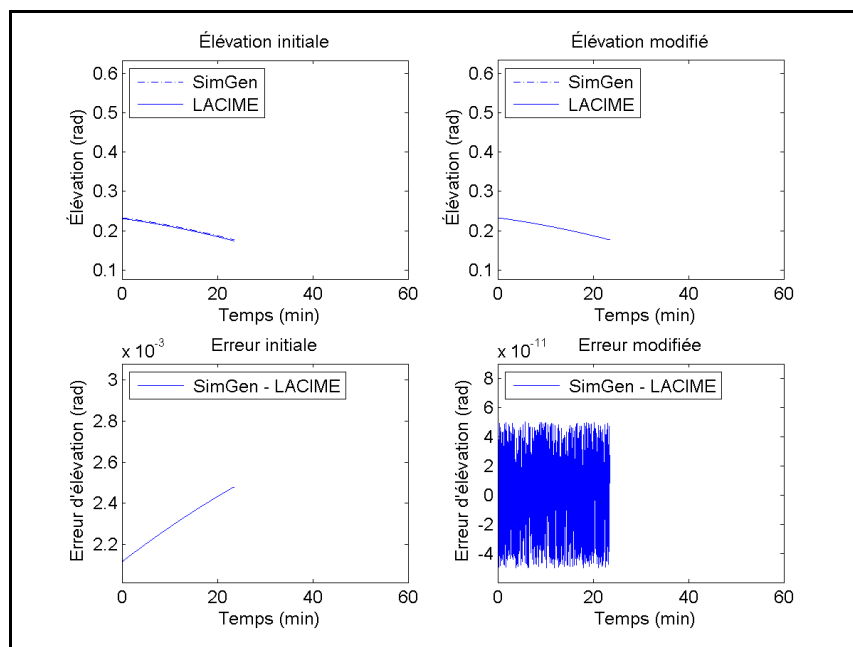


Figure-A II-47 L'angle d'élévation son erreur initiale et modifiée du satellite #20

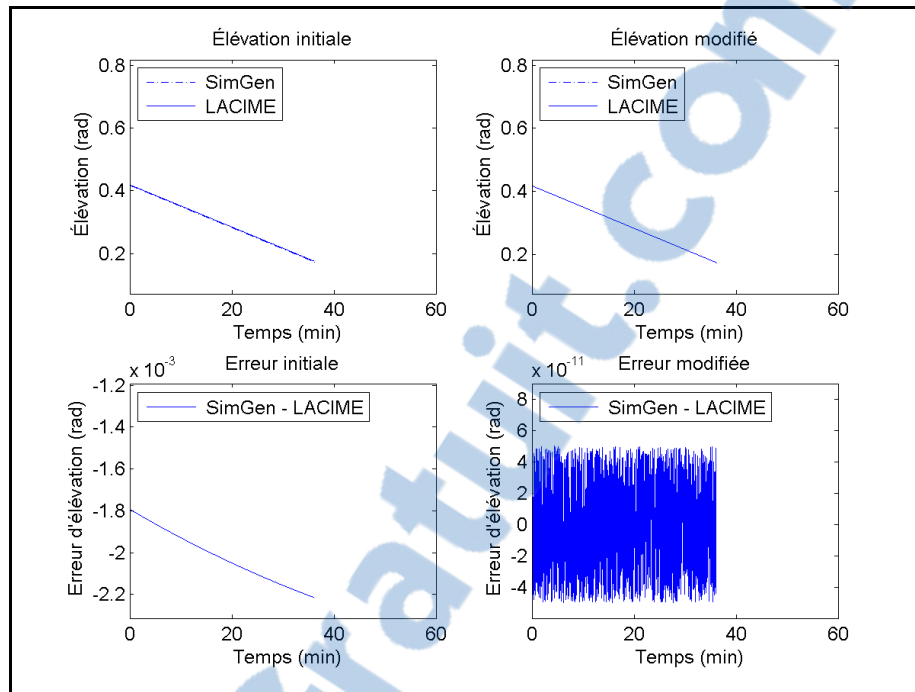


Figure-A II-48 L'angle d'élévation son erreur initiale et modifiée du satellite #28

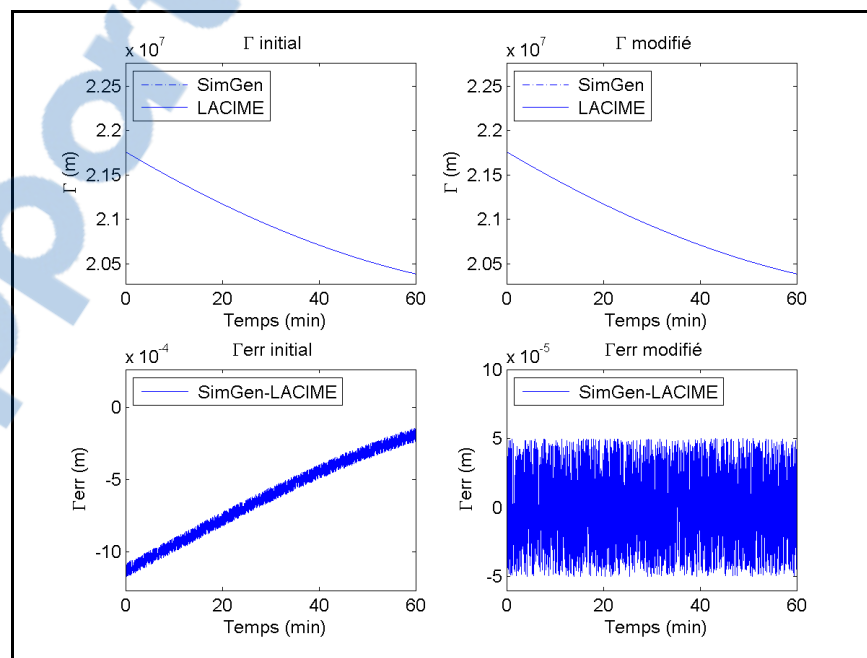
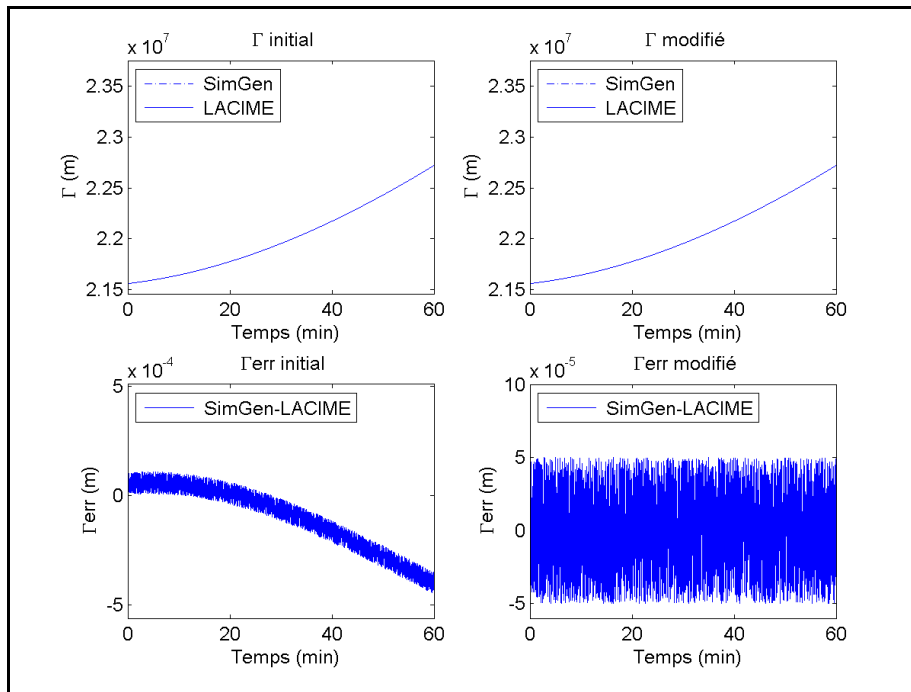
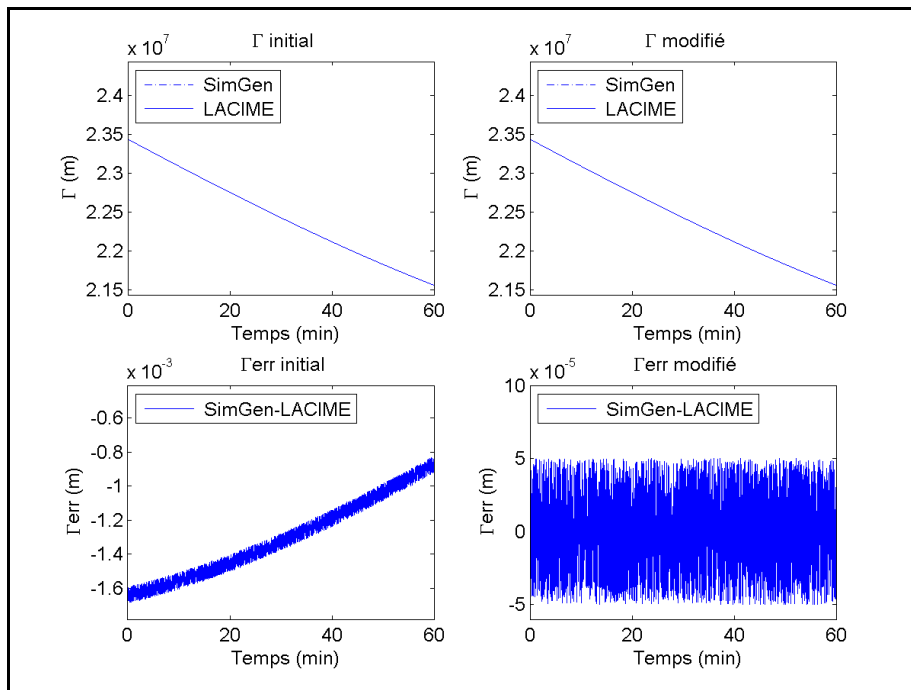
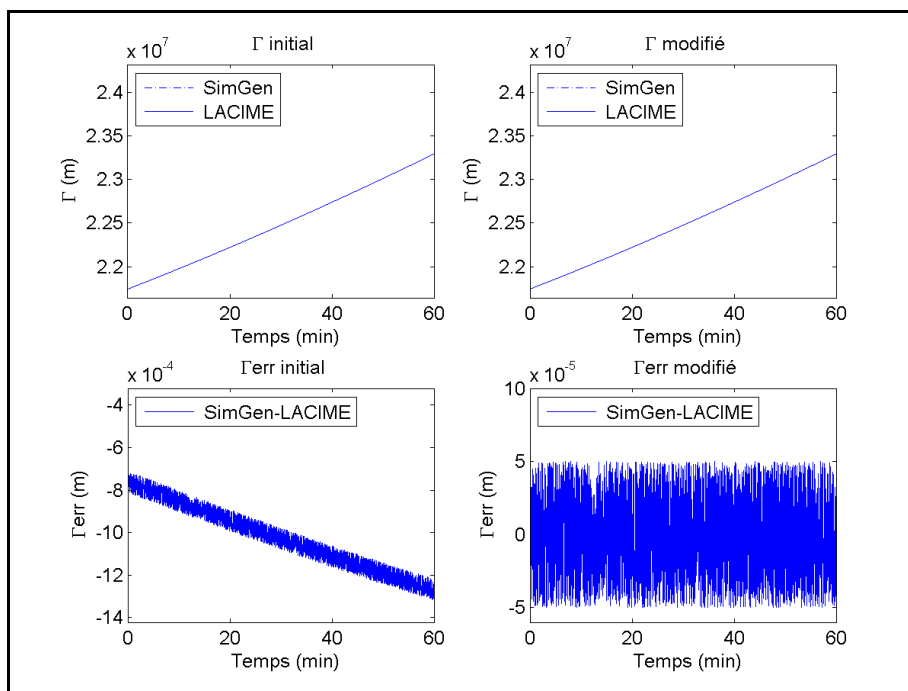
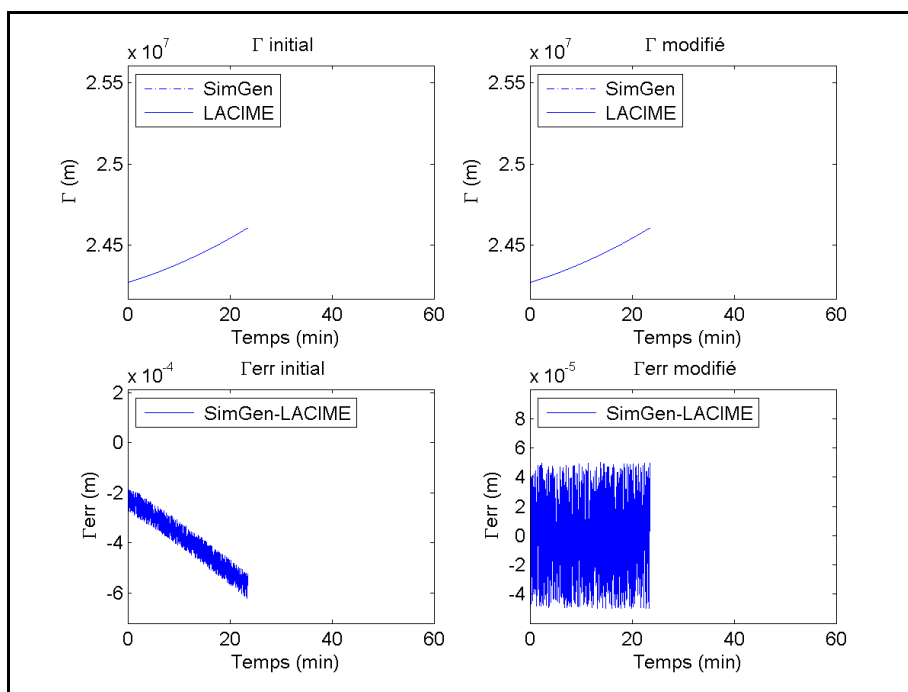
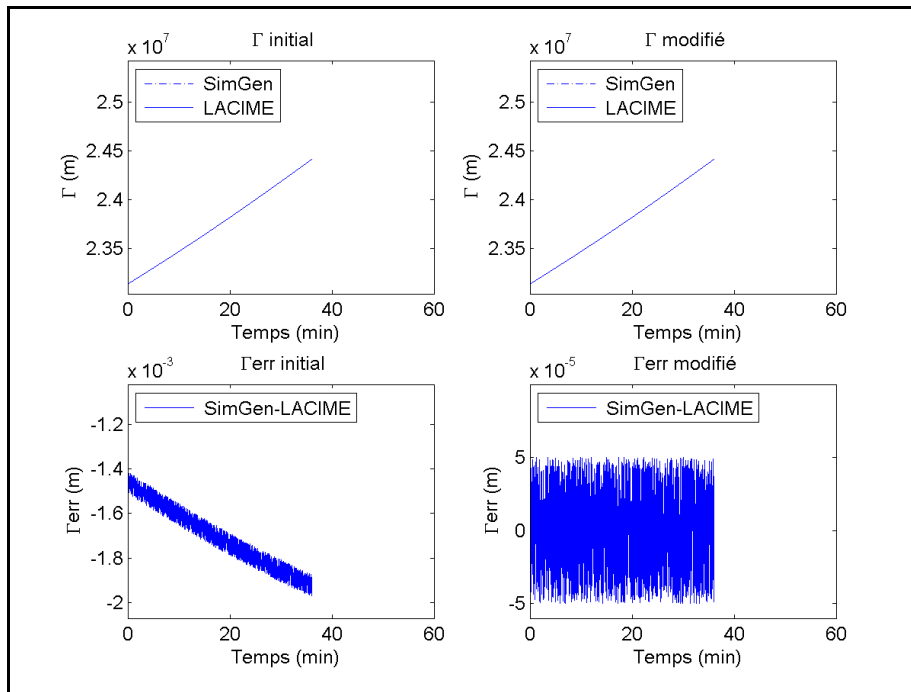
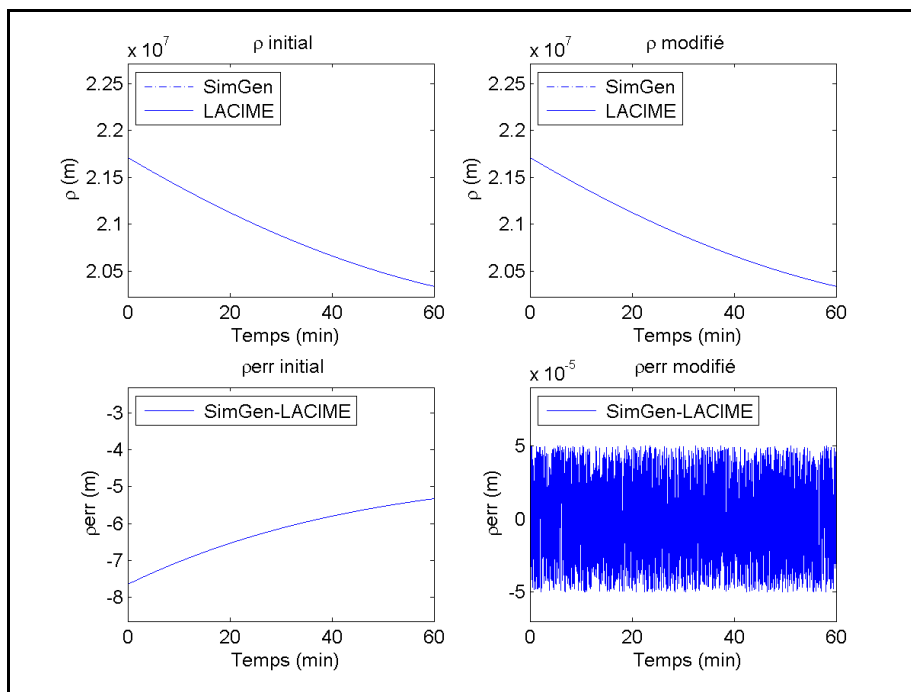
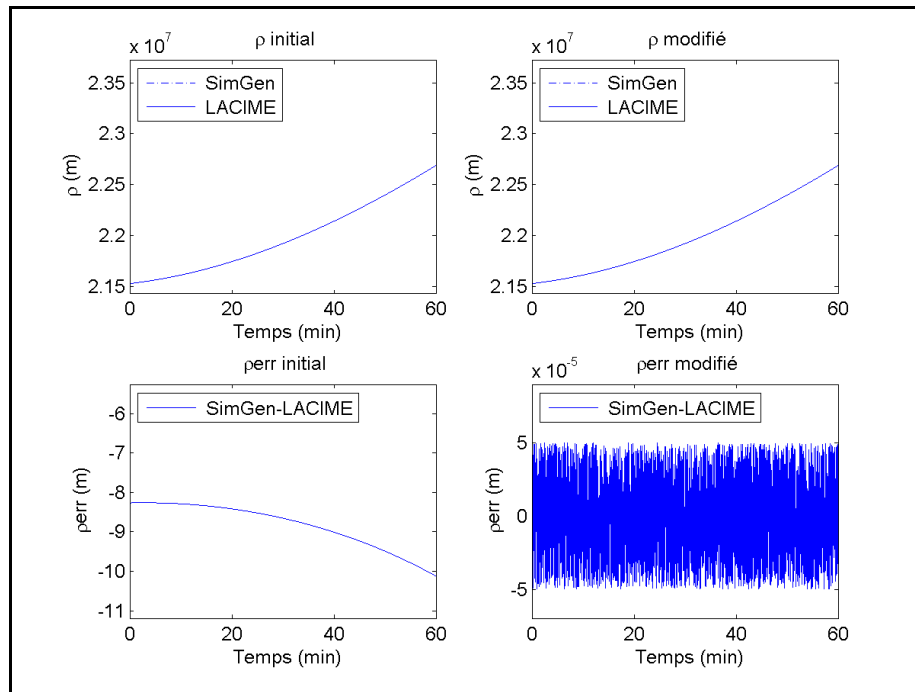
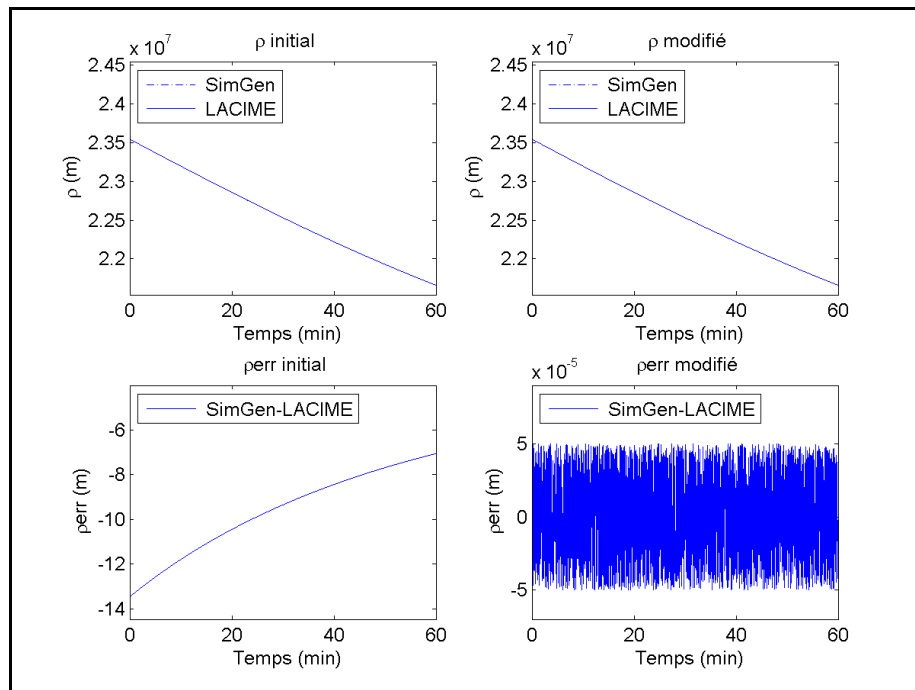


Figure-A II-49 Γ et Γ_{err} initial et modifié du satellite #2

Figure-A II-50 Γ et Γ_{err} initial et modifié du satellite #9Figure-A II-51 Γ et Γ_{err} initial et modifié du satellite #12

Figure-A II-52 Γ et Γ_{err} initial et modifié du satellite #17Figure-A II-53 Γ et Γ_{err} initial et modifié du satellite #20

Figure-A II-54 Γ et Γ_{err} initial et modifié du satellite #28Figure-A II-55 ρ et ρ_{err} initial et modifié du satellite #2

Figure-A II-56 ρ et ρ_{err} initial et modifié du satellite #9Figure-A II-57 ρ et ρ_{err} initial et modifié du satellite #12

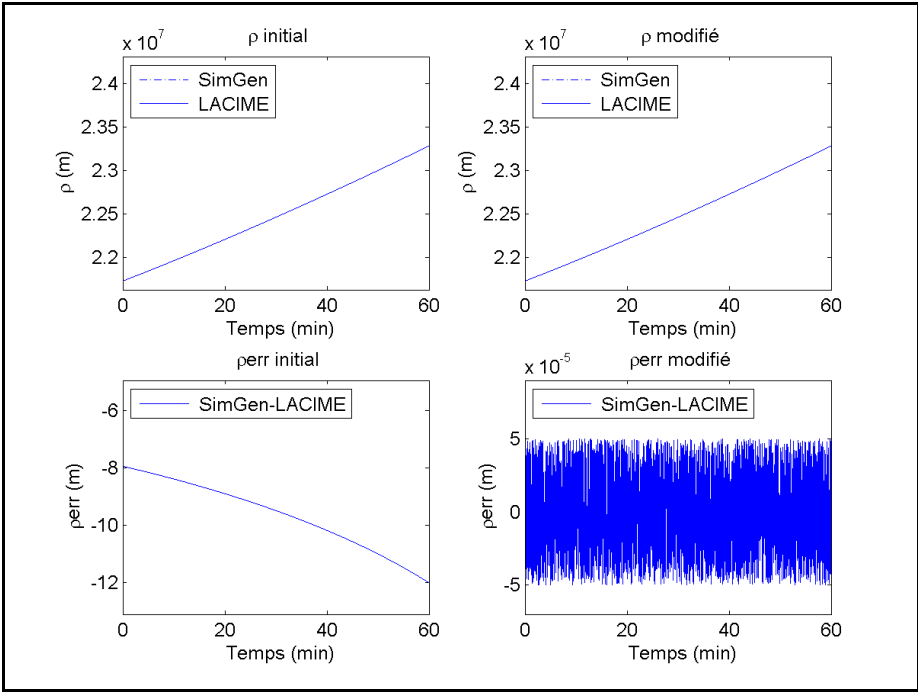


Figure-A II-58 ρ et ρ_{err} initial et modifié du satellite #17

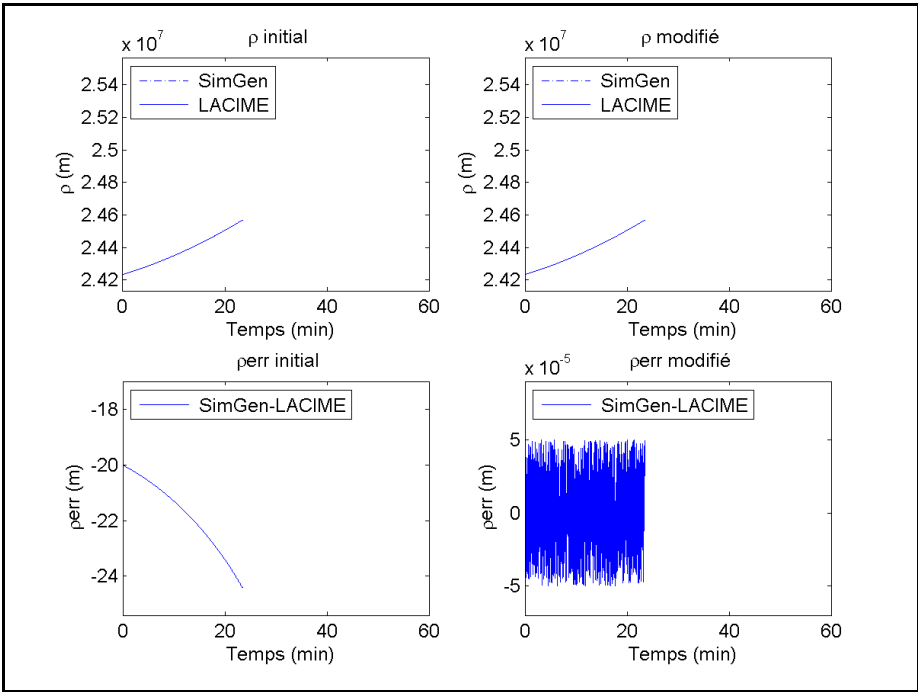
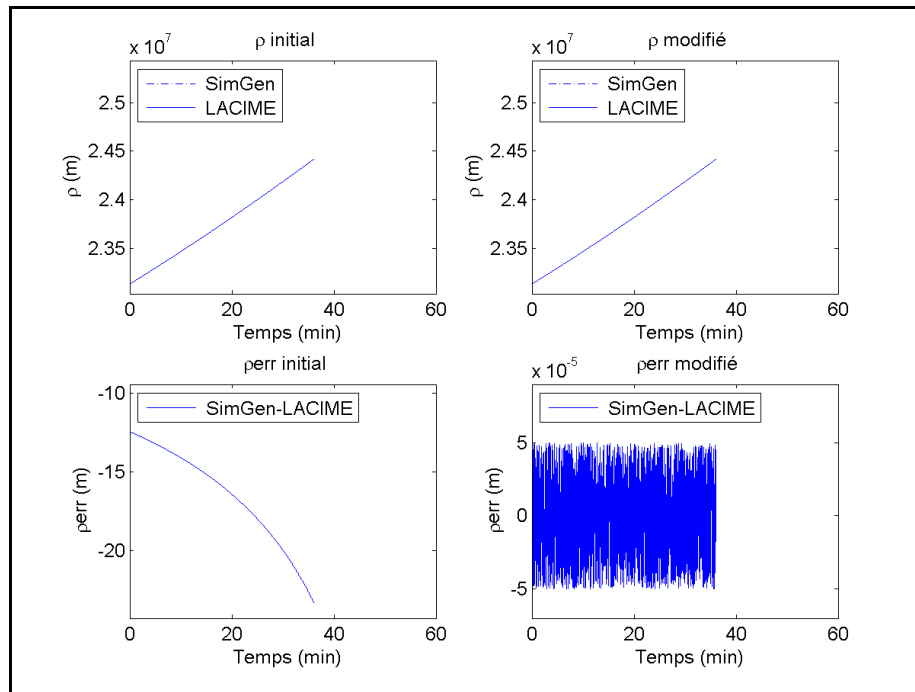
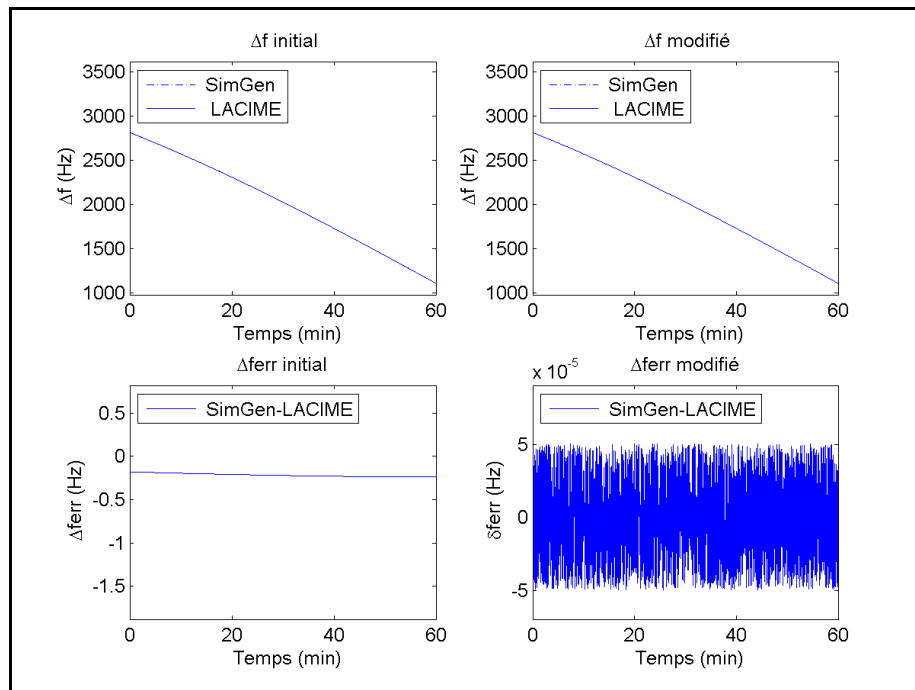
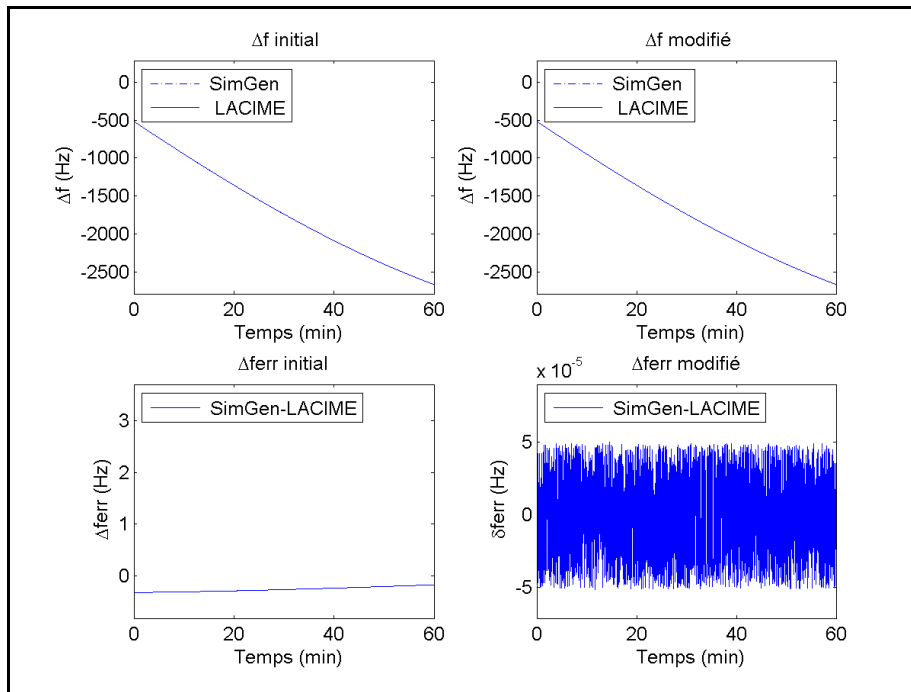
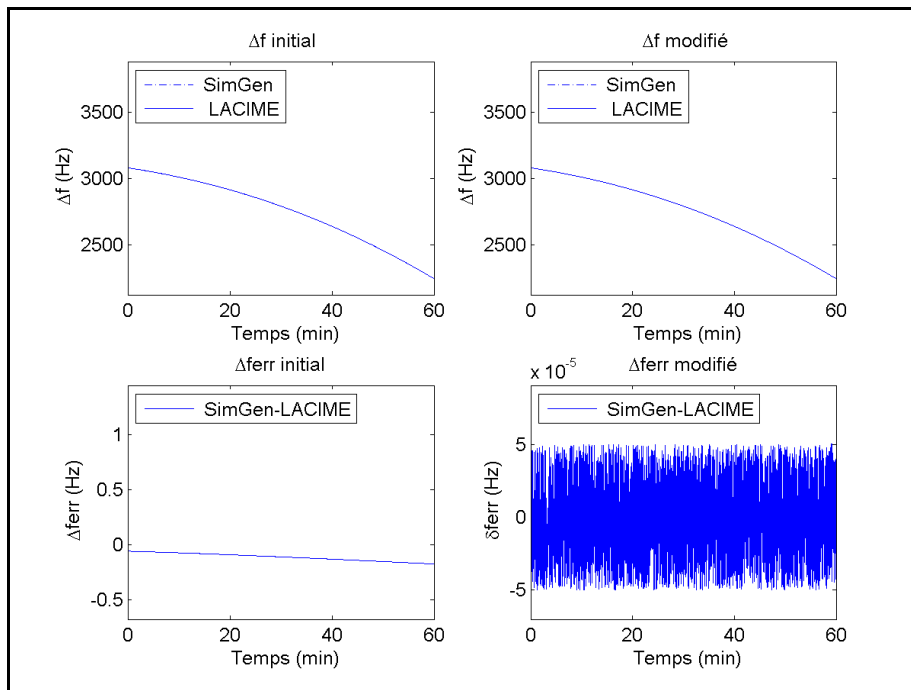
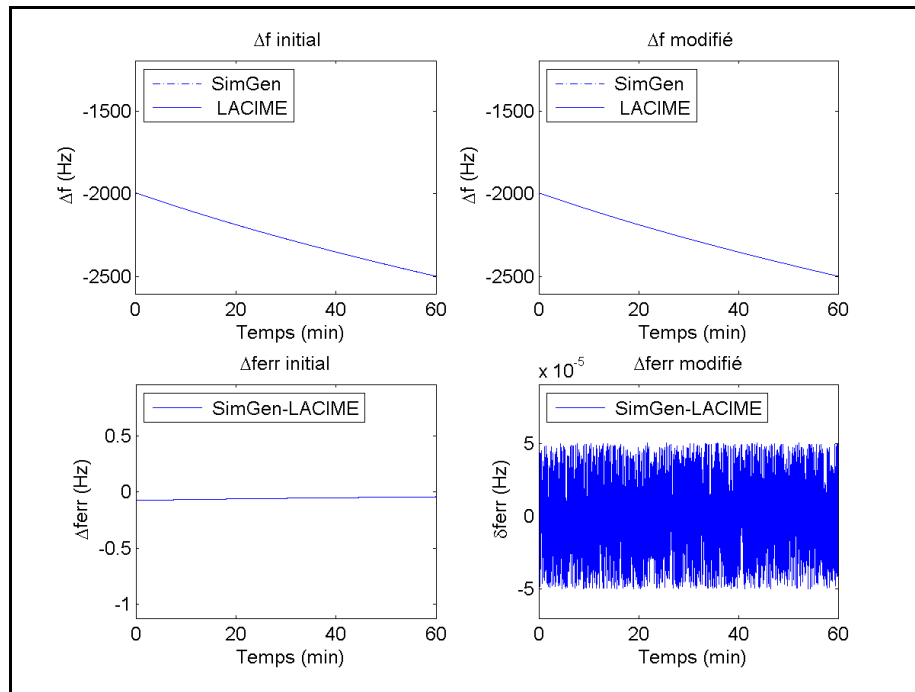
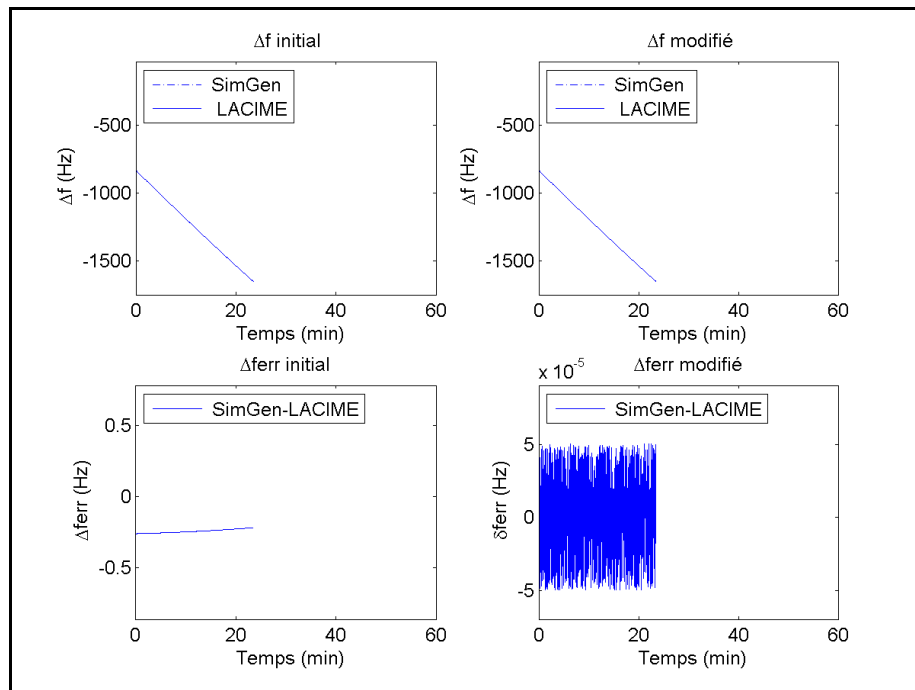


Figure-A II-59 ρ et ρ_{err} initial et modifié du satellite #20

Figure-A II-60 ρ et p_{err} initial et modifié du satellite #28Figure-A II-61 Δf et Δf_{err} initial et modifié du satellite #2

Figure-A II-62 Δf et Δf_{err} initial et modifié du satellite #9Figure-A II-63 Δf et Δf_{err} initial et modifié du satellite #12

Figure-A II-64 Δf et Δf_{err} initial et modifié du satellite #17Figure-A II-65 Δf et Δf_{err} initial et modifié du satellite #20

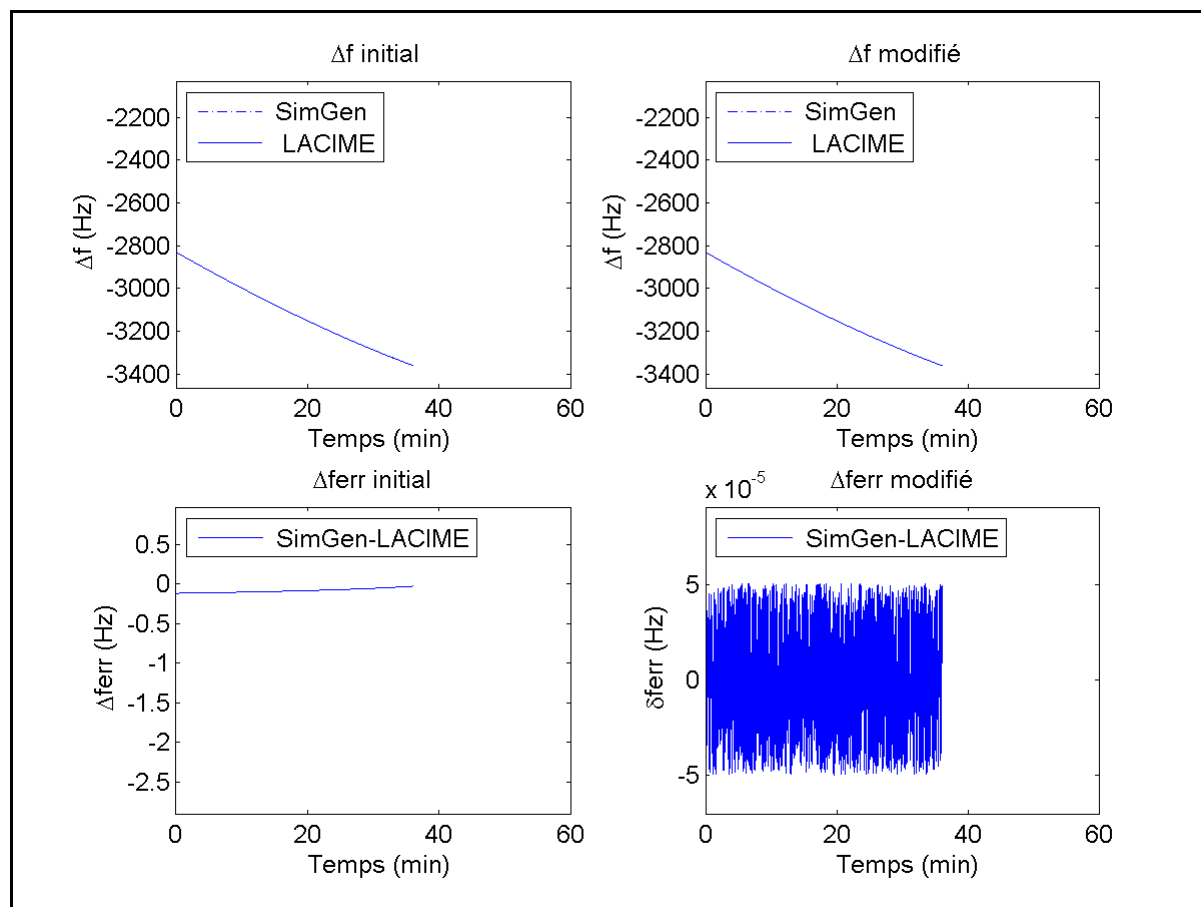


Figure-A II-66 Δf et Δf_{err} initial et modifié du satellite #28

ANNEXE III

ALGORITHME DU CALCUL DE LA POSITION DES SATELLITES

Voici l'algorithme utilisé pour calculer la position des satellites à un instant donné t (Tsui, 2005) :

1. Mouvement moyen corrigé : $n = \sqrt{\frac{\mu}{a^3}} + \Delta n$ où
 - a est le demi-grand axe de l'orbite (m),
 - μ est la constante gravitationnelle de la terre ($3,986005 \cdot 10^{14} \text{ m}^3/\text{sec}^2$).
2. Temps par rapport au temps de référence de l'éphéméride :
 $t_k = t - t_{oe} - \text{extrapolation}(\text{temps_transit}_{t-2}, \text{temps_transit}_{t-1})$ où :
 - t_{oe} est le temps de référence de l'éphéméride (sec),
 - extrapolation est la fonction qui extrapole avec $\text{temps_transit}_{t-2}$ et $\text{temps_transit}_{t-1}$.
3. Anomalie moyenne : $M_k = M_0 + n \cdot t_k$ où :
 - M_0 est l'anomalie moyenne au temps de références (rad).
4. Anomalie excentrique (E_k) : $M_k = E_k - e \cdot \sin(E_k)$ où :
 - e est l'excentricité de l'orbite.

On résout cette équation par la méthode de Newton pour trouver la valeur de E_k .

5. Anomalie vraie : $V_k = a \tan \left(\frac{\sqrt{1-e^2} \cdot \sin(E_k)}{\cos(E_k) - e} \right)$.

6. Argument de latitude : $\phi_k = V_k + \omega$ dont :
 - ω est l'argument du périée (rad).

7. Terme de correction d'orbite :

$$\delta r_k = C_{rs}\sin(2\varphi_k) + C_{rc}\sin(2\varphi_k),$$

$$\delta u_k = C_{us}\sin(2\varphi_k) + C_{uc}\sin(2\varphi_k),$$

$$\delta i_k = C_{is}\sin(2\varphi_k) + C_{ic}\sin(2\varphi_k)$$

- C_{uc} est l'amplitude du terme de correction harmonique cosinusoidal de l'argument de latitude (irad),
- C_{us} est l'amplitude du terme de correction harmonique sinusoidal de l'argument de latitude (irad),
- C_{rs} est l'amplitude du terme de correction harmonique sinusoidal du rayon orbital (m),
- C_{rc} est l'amplitude du terme de correction harmonique cosinusoidal du rayon orbital (m),
- C_{is} est l'amplitude du terme de correction harmonique sinusoidal de l'angle d'inclinaison (irad),
- C_{ic} est l'amplitude du terme de correction harmonique cosinusoidal de l'angle d'inclinaison (irad).

8. Argument de latitude corrigé : $u_k = \varphi_k + \delta u_k$.9. Rayon corrigé : $r_k = A \cdot (1 - e \cdot \cos(E_k)) + \delta r_k$.10. Inclinaison corrigée : $i_k = i_0 + \delta i_k$ où :

- i_0 est l'angle d'inclinaison (rad).

11. Longitude du nœud ascendant corrigé :

$$\Omega_k = \Omega_0 + (\Omega - \Omega_e)t_k - \Omega_e \cdot t_{oe} \text{ où :}$$

- Ω_0 est l'ascension droite au nœud ascendant (rad),
- Ω est l'argument du périégée (rad),

- Ω_e est la constante de la rotation de la terre ($7,2921151467 \cdot 10^{-5}$ rad/sec).

12. Position x du satellite sur le plan orbital : $x_p = r_k \cos(u_k)$.

13. Position y du satellite sur le plan orbital : $y_p = r_k \sin(u_k)$.

14. Coordonnée x du satellite en format ECEF : $x_{sat} = x_p \cos(\Omega_k) - y_p \cos(i_k) \sin(\Omega_k)$.

15. Coordonnée y du satellite en format ECEF : $y_{sat} = x_p \sin(\Omega_k) + y_p \cos(i_k) \cos(\Omega_k)$.

16. Coordonnée z du satellite en format ECEF : $z_{sat} = y_p \sin(i_k)$.

ANNEXE IV

ALGORITHME DU CALCUL DE LA VITESSE DES SATELLITES

Voici l'algorithme utilisé pour calculer la vitesse des satellites à un instant donné t (Korvenoja et Piché, 2000) :

1. Mouvement moyen corrigé : $n = \sqrt{\frac{\mu}{a^3}} + \Delta n$ où :
 - a est le demi-grand axe de l'orbite (m),
 - μ est la constante gravitationnelle de la terre ($3,986005 \cdot 10^{14} \text{ m}^3/\text{sec}^2$).
2. Temps par rapport au temps de référence de l'éphéméride : $t_k = t - t_{oe}$ où :
 - t_{oe} est le temps de référence de l'éphéméride (sec).
3. Dérivée de l'anomalie moyenne : $M'_k = n$.
4. Dérivée de l'anomalie excentrique : $E'_k = \frac{M'_k}{1 - e \cos(E_k)}$ où
 - e est l'excentricité de l'orbite.
5. Dérivée de l'anomalie vraie : $V'_k = \frac{\sqrt{1 - e^2} * E'_k}{1 - e \cos(E_k)}$.
6. Dérivée de l'argument de latitude : $\phi'_k = V'_k$.
7. Dérivée des termes de correction d'orbite :
$$\delta r'_k = 2\phi'_k C_{rs} \cos(2\phi_k) - 2\phi'_k C_{rc} \sin(2\phi_k),$$
$$\delta \phi'_k = 2\phi'_k C_{us} \cos(2\phi_k) - 2\phi'_k C_{uc} \sin(2\phi_k),$$
$$\delta i'_k = 2\phi'_k C_{is} \cos(2\phi_k) - 2\phi'_k C_{ic} \sin(2\phi_k) \text{ où :}$$

- C_{uc} est l'amplitude du terme de correction harmonique cosinusoidal de l'argument de latitude (irad),
- C_{us} est l'amplitude du terme de correction harmonique sinusoidal de l'argument de latitude (irad),
- C_{rs} est l'amplitude du terme de correction harmonique sinusoidal du rayon orbital (m),
- C_{rc} est l'amplitude du terme de correction harmonique cosinusoidal du rayon orbital (m),
- C_{is} est l'amplitude du terme de correction harmonique sinusoidal de l'angle d'inclinaison (irad),
- C_{ic} est l'amplitude du terme de correction harmonique cosinusoidal de l'angle d'inclinaison (irad).

8. Dérivée de l'argument de latitude corrigé : $u'_k = \varphi'_k + \delta\varphi'_k$.

9. Dérivée du rayon corrigé : $r'_k = aeE'_k \sin(E_k) + \delta r'_k$.

10. Dérivée de l'inclinaison corrigée : $i_k = \delta i'_k$.

11. Dérivée de la longitude du nœud ascendant corrigé : $\Omega'_k = \Omega - \Omega_e$ où :

- Ω est l'argument du périégée (rad),
- Ω_e est la constante de la rotation de la terre ($7,2921151467 \cdot 10^{-5}$ rad/sec).

12. Vitesse x du satellite sur le plan orbital : $x'_p = -r_k u'_k \sin(u_k) + r'_k \cos(u_k)$.

13. Vitesse y du satellite sur le plan orbital : $y'_p = r_k u'_k \cos(u_k) + r'_k \sin(u_k)$.

14. Vitesse x du satellite en format ECEF :

$$x'_{sat} = -\Omega'_k y_s + (z_s i'_k - \cos(i_k) y'_p) \sin(\Omega_k) + x'_p \cos(\Omega_k).$$

15. Vitesse y du satellite en format ECEF :

$$y'_{\text{sat}} = \Omega'_k x_s + (z_s i'_k - y'_p \cos(i_k)) \cos(\Omega_k) + x'_p \sin(\Omega_k).$$

16. Vitesse z du satellite en format ECEF : $z'_{\text{sat}} = y_p i'_k \cos(i_k) + y'_p \sin(i_k).$

ANNEXE V

DESCRIPTION DES VARIABLES DU MESSAGE DE NAVIGATION

Pour la génération du message de navigation, il y a deux fonctions qui permettent de modifier les variables et celles-ci sont définies au Tableau-A V-1 et A VIII-II.

Tableau-A V-1 Variables de la fonction UpdateNavMsgConstParam ()

Variables	Descriptions
A_0	Terme constant pour le calcul de l'UTC
A_1	Terme polynomial du premier ordre pour le calcul de l'UTC
a_f0	Coefficient du biais d'horloge d'ordre 0
a_f1	Coefficient du biais d'horloge d'ordre 1
a_f2	Coefficient du biais d'horloge d'ordre 2
AI_SVs	Indicateur de disponibilité du satellite
alpha_0	Premier coefficient d'une équation cubique représentant l'amplitude du délai dû à l'altitude (délai de l'ionosphère)
alpha_1	Deuxième coefficient d'une équation cubique représentant l'amplitude du délai dû à l'altitude (délai de l'ionosphère)
alpha_2	Troisième coefficient d'une équation cubique représentant l'amplitude du délai dû à l'altitude (délai de l'ionosphère)
alpha_3	Quatrième coefficient d'une équation cubique représentant l'amplitude du délai dû à l'altitude (délai de l'ionosphère)
AntiSpoof_SVs	Drapeau permettant d'activer/désactiver l'option antibrouillage du satellite
AODO	Compensation de la durée de vie de données
beta_0	Premier coefficient d'une équation cubique représentant la période du modèle de l'ionosphère
beta_1	Deuxième coefficient d'une équation cubique représentant la période du modèle de l'ionosphère
beta_2	Troisième coefficient d'une équation cubique représentant la période du modèle de l'ionosphère
beta_3	Quatrième coefficient d'une équation cubique représentant la période du modèle de l'ionosphère
C_ic	Amplitude du terme de correction harmonique cosinusoidal de l'angle d'inclinaison
C_is	Amplitude du terme de correction harmonique sinusoidal de l'angle d'inclinaison
C_rc	Amplitude du terme de correction harmonique cosinusoidal du rayon orbital

Variables	Descriptions
C_rs	Amplitude du terme de correction harmonique sinusoïdal du rayon orbital
C_uc	Amplitude du terme de correction harmonique cosinusoïdal de l'argument de latitude
C_us	Amplitude du terme de correction harmonique sinusoïdal de l'argument de latitude
DataID	Structure de la donnée de navigation
delta_i	Paramètre de correction de l'inclinaison
Delta_n	Différence du mouvement moyen corrigé
e	Excentricité de l'orbite
ERD_SVs	Déviations de la pseudo-distance estimée
i_0	Inclinaison orbitale
IDot	Taux de l'angle d'inclinaison
IODC	Donnée de l'horloge
IODE	Donnée de l'éphéméride
M_0	Anomalie moyenne
omega	Argument du périhélie
Omega_0	Ascension droite au nœud ascendant
OmegaDot	Vitesse d'ascension droite
sqrt_A	Racine carrée du demi-grand axe de l'orbite
SVHealth_SVs	État du satellite
T_GD	Délai de groupe différentiel estimé
t_oa	Temps de référence de l'almanach
URAIndex	Valeur de l'index URA

Voici les variables modifiées de la fonction UpdateNavMsgVarParam().

Tableau-A V-2 Variables modifiées de la fonction UpdateNavMsgVarParam ()

Variables	Descriptions
Delta_t_LS	Intervalle de temps dû au prochain saut de seconde de l'UTC
Delta_t_LSF	Intervalle de temps effectif dû au saut de seconde de l'UTC
DN	Le numéro de la journée
SubFrameID	L'index de la sous-trame courante
t_oc	Temps de référence des corrections d'horloges
t_oe	Temps de référence de l'éphéméride
t_ot	Temps de référence de l'UTC
TOW_Actual	Temps écoulé en seconde depuis le début de la semaine
TOW_Dec	Temps tronqué en seconde depuis le début de la semaine
WeekNumber	Le numéro de la semaine de simulation
WN_a	Le numéro de la semaine de référence de l'almanach

Variables	Descriptions
WN_LSF	Le numéro de la semaine de référence du numéro de la journée (DN)
WN_t	Le numéro de la semaine de référence de l'UTC

ANNEXE VI

MESURES DE L'ERREUR DU MESSAGE DE NAVIGATION GPS L1 DU SIMULATEUR LACIME

Les tableaux de cette annexe représentent les erreurs du message de navigation du simulateur LACIME avant et après les corrections apportées.

Tableau-A VI-1 Erreur initiale et modifiée du message de navigation
GPS L1 pour le satellite #2

# Sous-frames	Numéro de pages de la sous-trame	Erreur initiale (%)	Erreur modifiée (%)
1	-	60,77	0,00
2	-	9,22	0,00
3	-	12,93	0,00
4	1,6,11,16,21	54,14	0,00
	2,3,4,5,7,8,9,10	53,13	6,41
	12,19,20,22,23,24	54,84	0,00
	13	51,27	0,00
	14,15	52,77	0,00
	17	48,27	0,00
	18	50,60	0,00
	25	54,67	20,00
5	1 à 24	49,97	0,00
	25	52,67	0,00

Tableau-A VI-2 Erreur initiale et modifiée du message de navigation
GPS L1 pour le satellite #9

# Sous-frames	Numéro de pages de la sous-trame	Erreur initiale (%)	Erreur modifiée (%)
1	-	60,10	0,00
2	-	12,38	0,00
3	-	67,47	0,00

# Sous-trames	Numéro de pages de la sous-trame	Erreur initiale (%)	Erreur modifiée (%)
4	1,6,11,16,21	43,81	0,00
	2,3,4,5,7,8,9,10	41,13	6,42
	12,19,20,22,23,24	43,61	0,00
	13	43,27	0,00
	14,15	44,77	0,00
	17	40,27	0,00
	18	42,60	0,00
	25	42,67	20,00
5	1 à 24	49,96	0,00
	25	52,67	0,00

Tableau-A VI-3 Erreur initiale et modifiée du message de navigation
GPS L1 pour le satellite #12

# Sous-trames	Numéro de pages de la sous-trame	Erreur initiale (%)	Erreur modifiée (%)
1	-	62,01	0,00
2	-	21,67	0,00
3	-	26,38	0,00
4	1,6,11,16,21	47,81	0,00
	2,3,4,5,7,8,9,10	45,13	6,42
	12,19,20,22,23,24	47,61	0,00
	13	47,27	0,00
	14,15	48,77	0,00
	17	46,60	0,00
	18	46,60	0,00
	25	46,67	20,00
5	1 à 24	49,96	0,00
	25	52,67	0,00

Tableau-A VI-4 Erreur initiale et modifiée du message de navigation
GPS L1 pour le satellite #17

# Sous-trames	Numéro de pages de la sous-trame	Erreur initiale (%)	Erreur modifiée (%)
1	-	60,10	0,00
2	-	13,00	0,00
3	-	15,25	0,00
4	1,6,11,16,21	54,81	0,00
	2,3,4,5,7,8,9,10	53,13	6,42
	12,19,20,22,23,24	55,15	0,00
	13	52,87	0,00
	14,15	54,37	0,00
	17	49,87	0,00
	18	52,21	0,00
	25	54,67	20,00
5	1 à 24	49,96	0,00
	25	52,67	0,00

Tableau-A VI-5 Erreur initiale et modifiée du message de navigation
GPS L1 pour le satellite #20

# Sous-trames	Numéro de pages de la sous-trame	Erreur initiale (%)	Erreur modifiée (%)
1	-	62,01	0,00
2	-	21,04	0,00
3	-	59,53	0,00
4	1,6,11,16,21	44,14	0,00
	2,3,4,5,7,8,9,10	41,13	6,42
	12,19,20,22,23,24	43,77	0,00
	13	44,07	0,00
	14,15	45,57	0,00
	17	41,07	0,00
	18	43,40	0,00
	25	42,67	20,00
5	1 à 24	49,96	0,00
	25	52,67	0,00

Tableau-A VI-6 Erreur initiale et modifiée du message de navigation
GPS L1 pour le satellite #28

# Sous-trames	Numéro de pages de la sous-trame	Erreur initiale (%)	Erreur modifiée (%)
1	-	60,77	0,00
2	-	17,67	0,00
3	-	16,67	0,00
4	1,6,11,16,21	54,14	0,00
	2,3,4,5,7,8,9,10	53,13	6,42
	12,19,20,22,23,24	54,84	0,00
	13	51,27	0,00
	14,15	52,77	0,00
	17	48,27	0,00
	18	50,60	0,00
	25	54,67	20,00
5	1 à 24	49,96	0,00
	25	52,67	0,00

ANNEXE VII

RÉSULTATS VUS DU RÉCEPTEUR U-BLOX

Cette annexe contient les résultats initiaux et modifiés vus du récepteur u-blox d'une simulation avec Spirent (Rx/Spirent) et LACIME (Rx/LACIME).

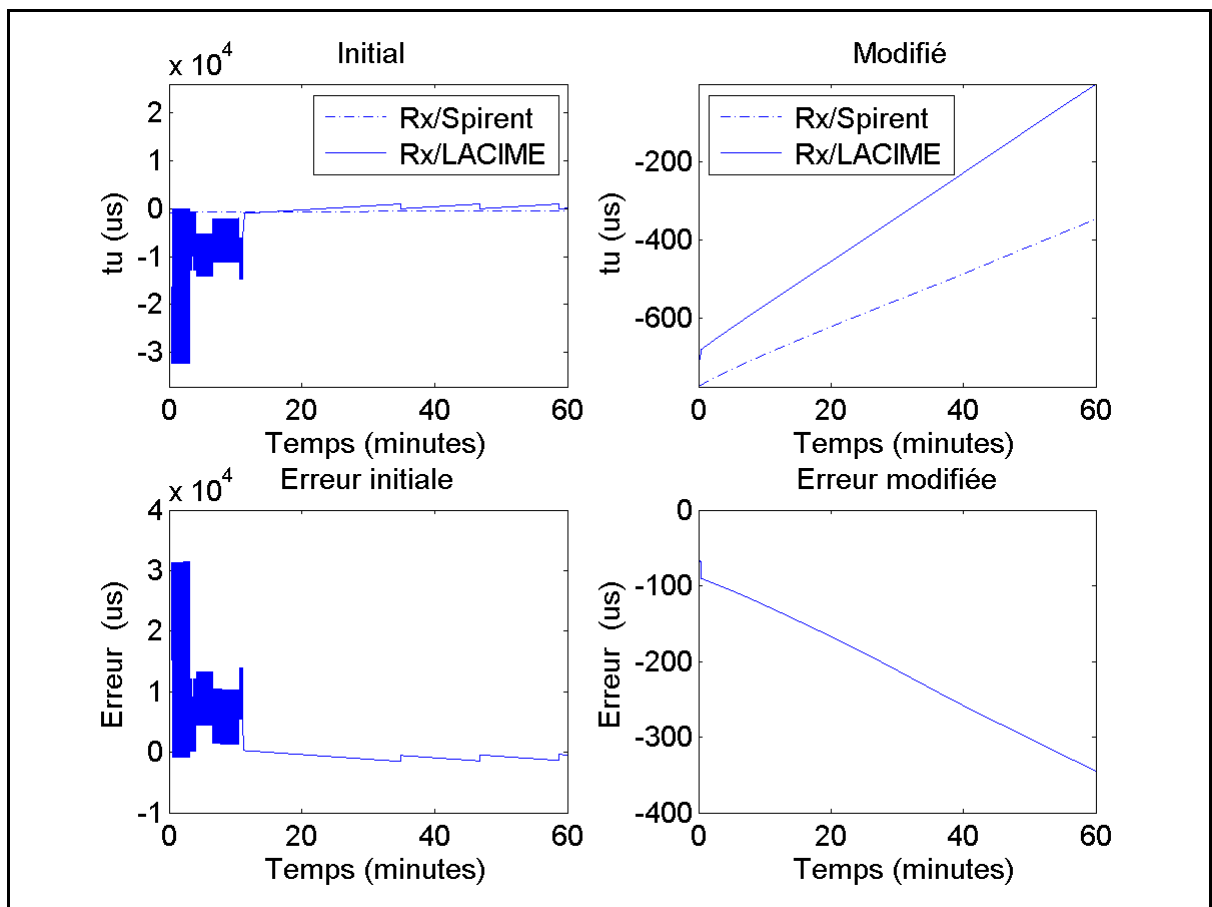


Figure-A VII-1 t_u initial et modifié ainsi que son erreur (Rx/Spirent-Rx/LACIME)

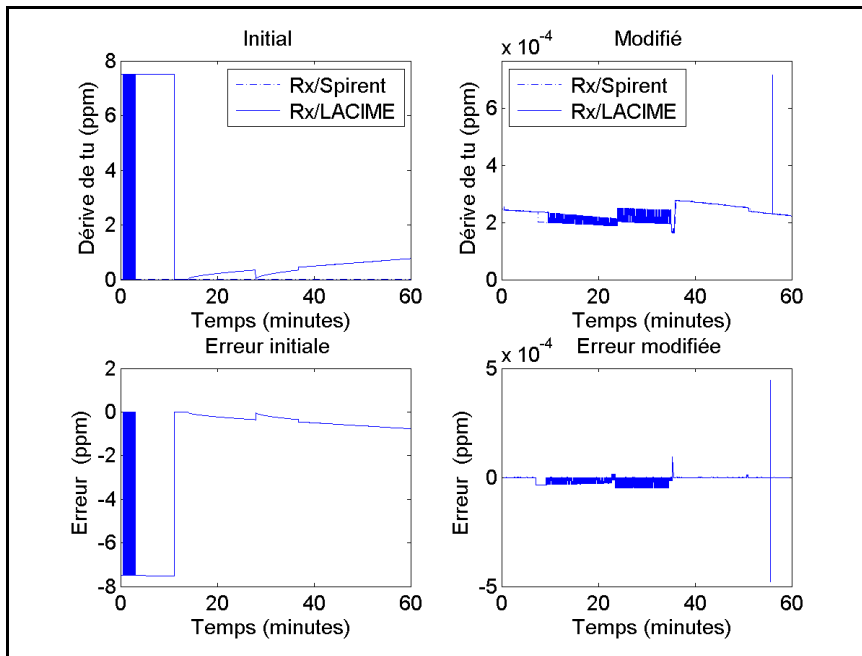


Figure-A VII-2 Dérive de t_u ainsi que son erreur (Rx/Spirent-Rx/LACIME)

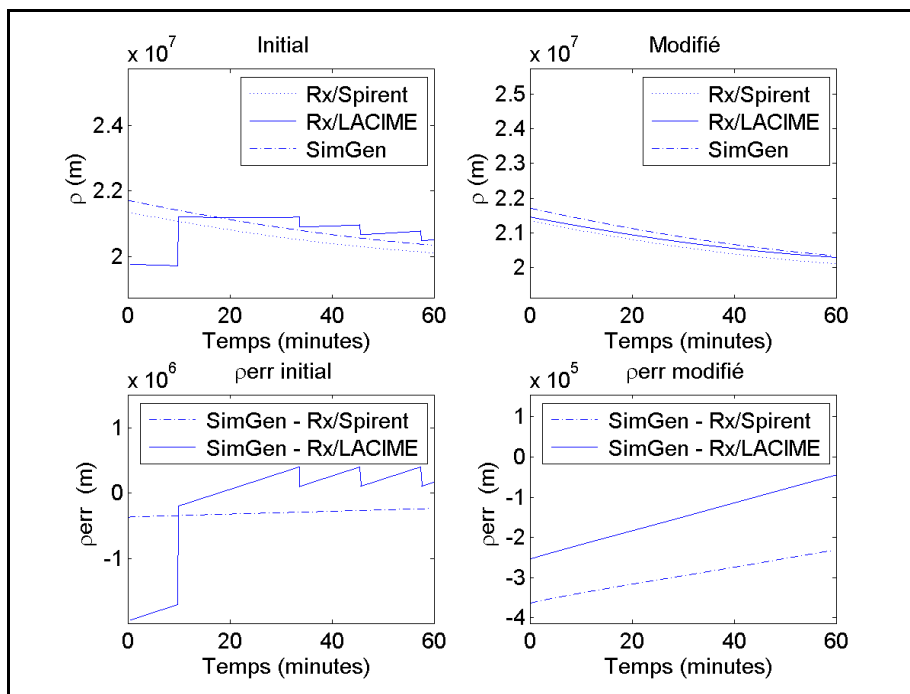
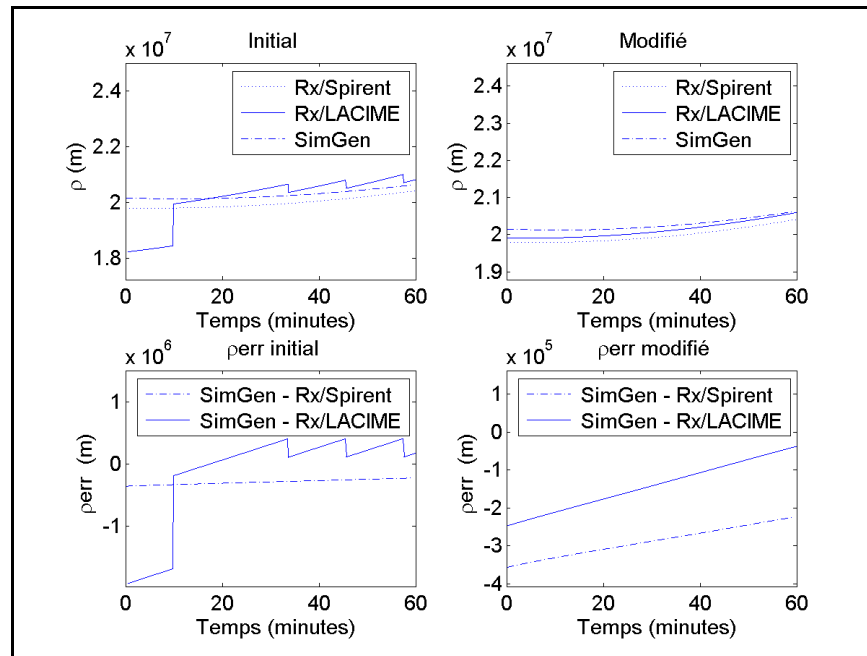
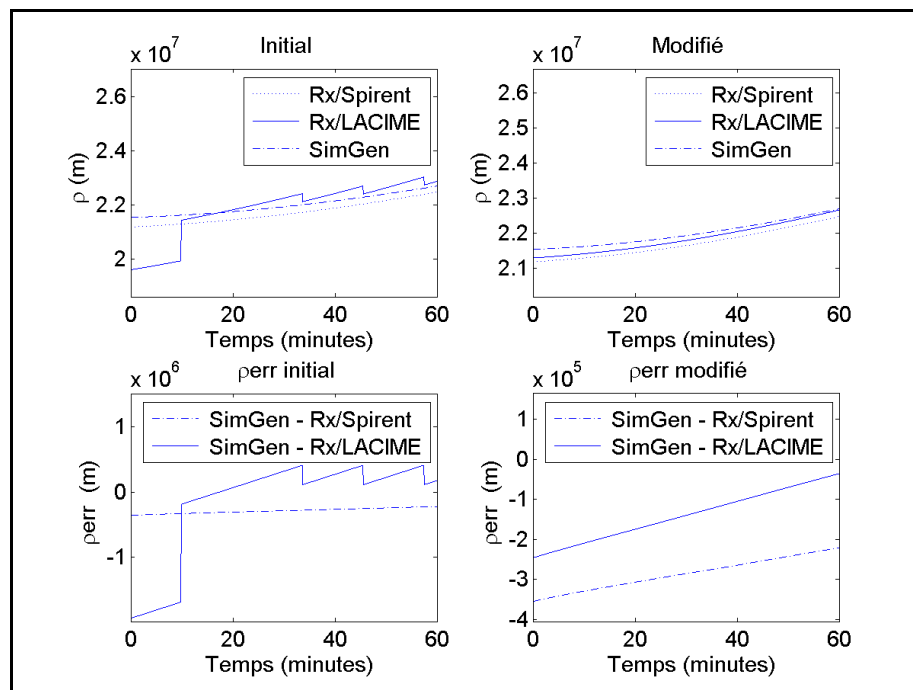
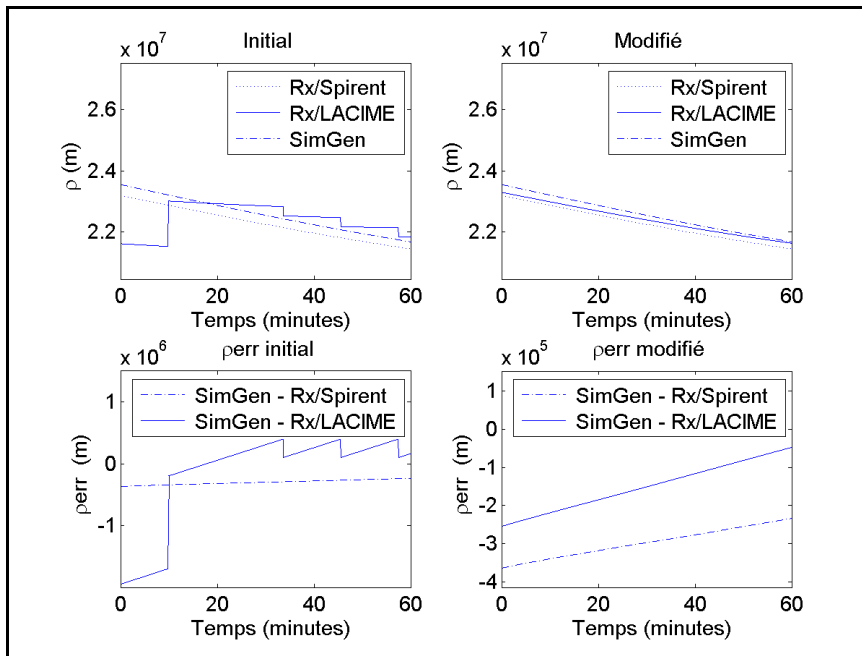
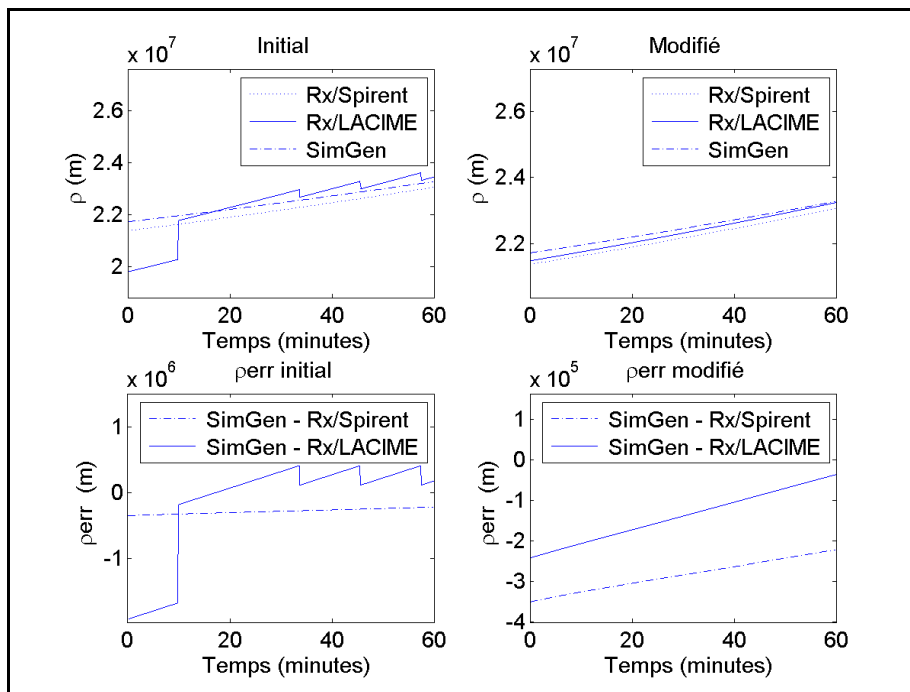
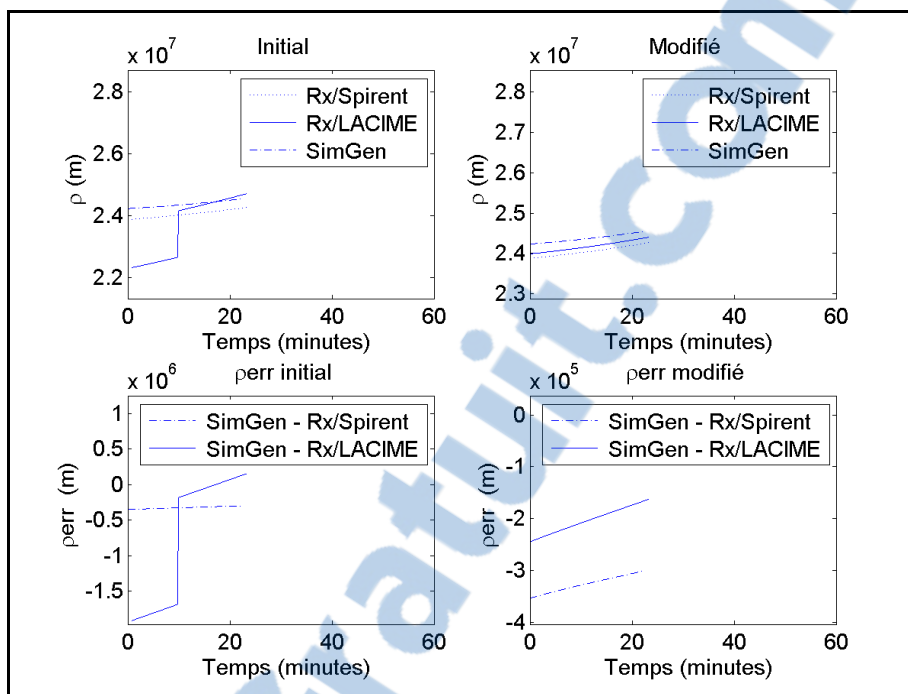
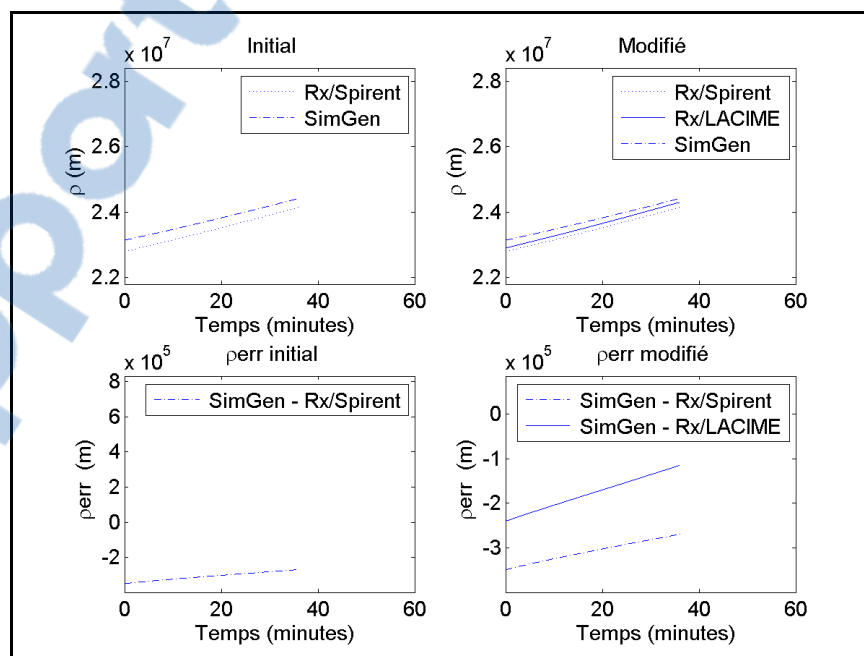
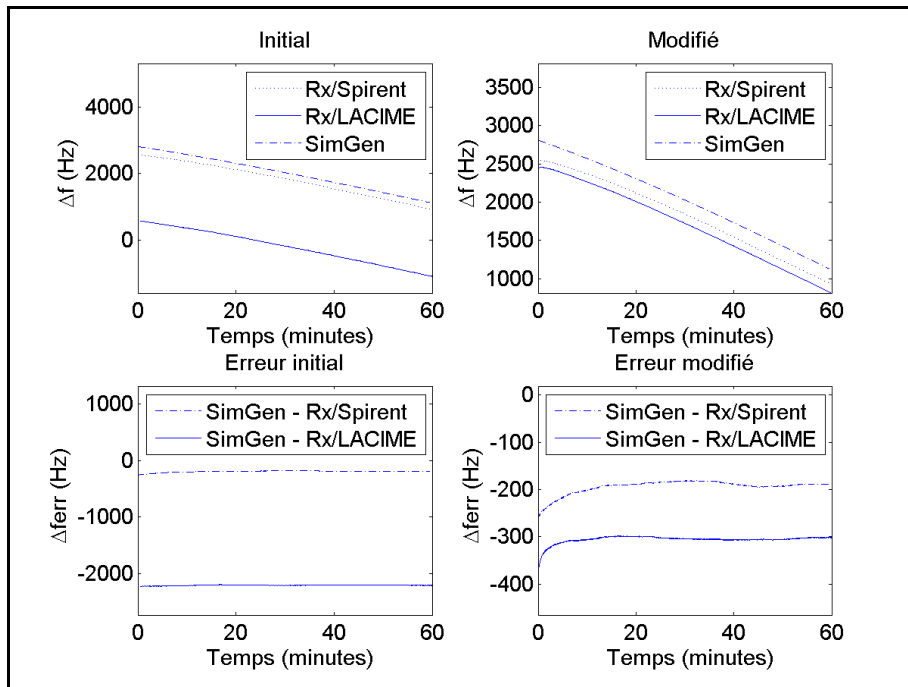
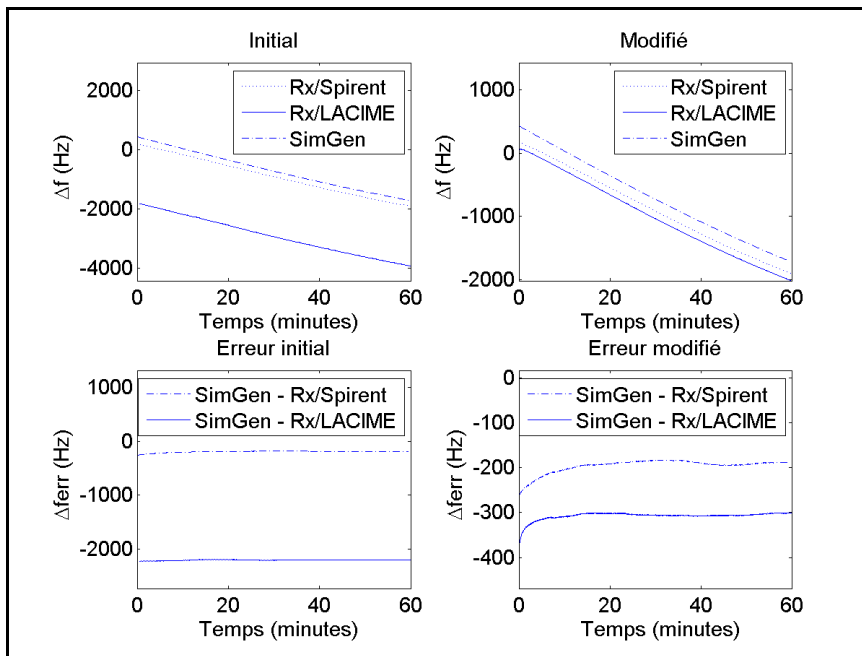


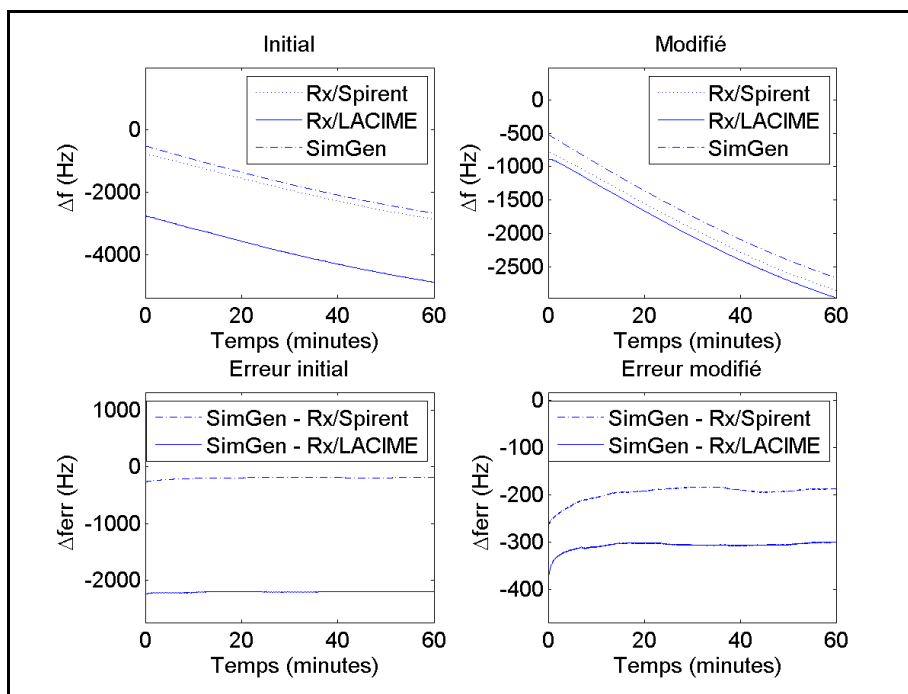
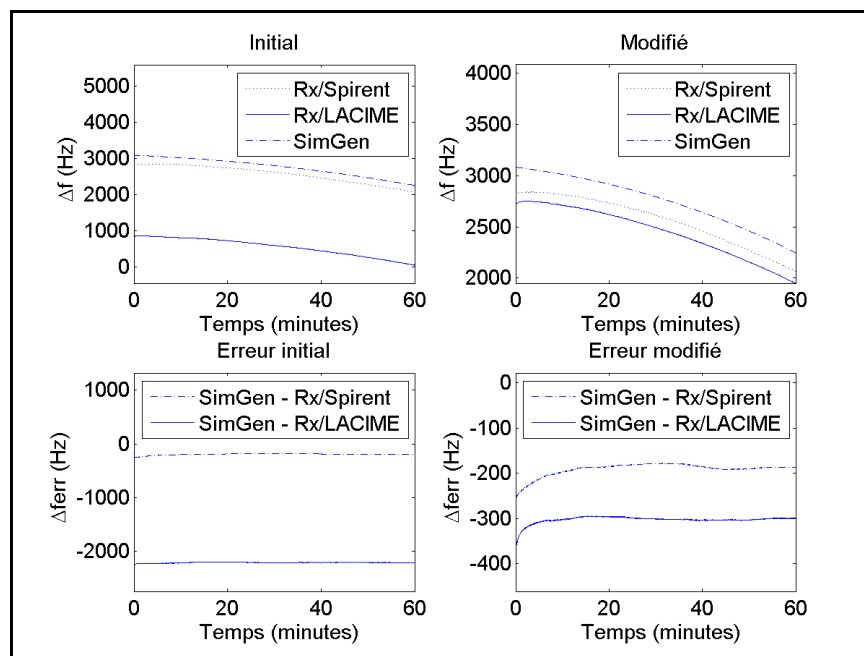
Figure-A VII-3 ρ et ρ_{err} initial et modifié du satellite #2

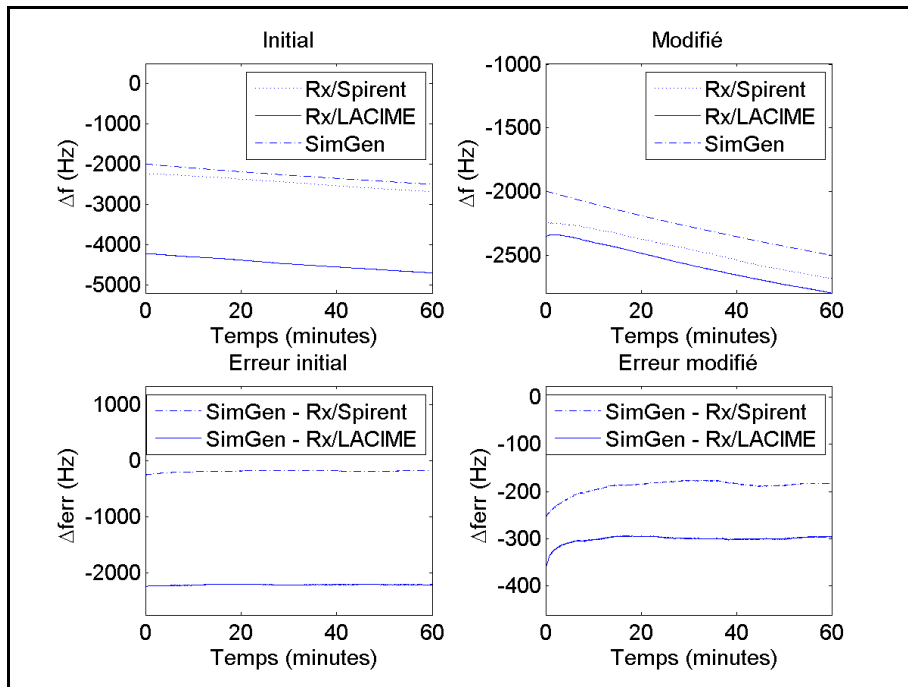
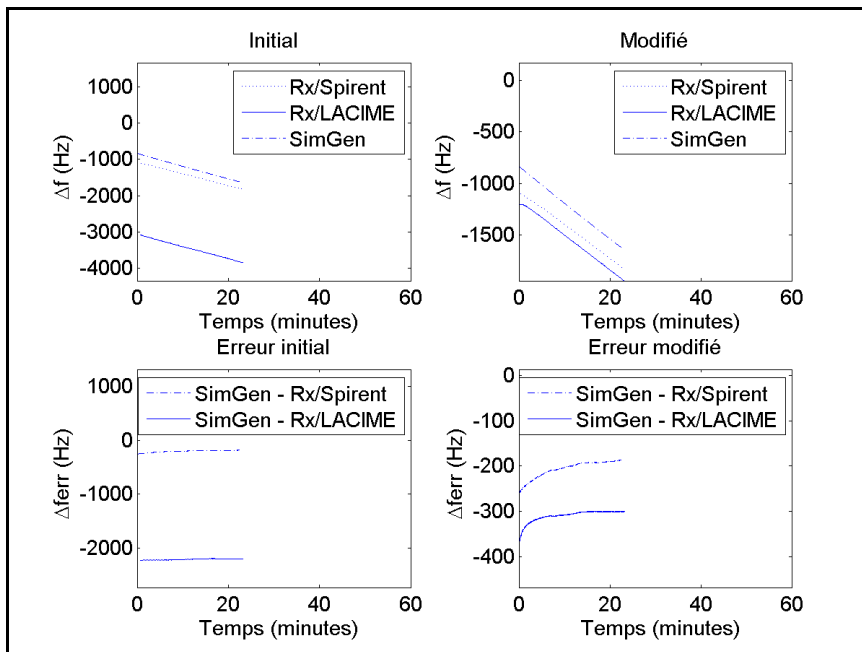
Figure-A VII-4 ρ et ρ_{err} initial et modifié du satellite #4Figure-A VII-5 ρ et ρ_{err} initial et modifié du satellite #9

Figure-A VII-6 ρ et ρ_{err} initial et modifié du satellite #12Figure-A VII-7 ρ et ρ_{err} initial et modifié du satellite #17

Figure-A VII-8 ρ et ρ_{err} initial et modifié du satellite #20Figure-A VII-9 ρ et ρ_{err} initial et modifié du satellite #28

Figure-A VII-10 Δf et Δf_{err} initial et modifié du satellite #2Figure-A VII-11 Δf et Δf_{err} initial et modifié du satellite #4

Figure-A VII-12 Δf et Δf_{err} initial et modifié du satellite #9Figure-A VII-13 Δf et Δf_{err} initial et modifié du satellite #12

Figure-A VII-14 Δf et Δf_{err} initial et modifié du satellite #17Figure-A VII-15 Δf et Δf_{err} initial et modifié du satellite #20

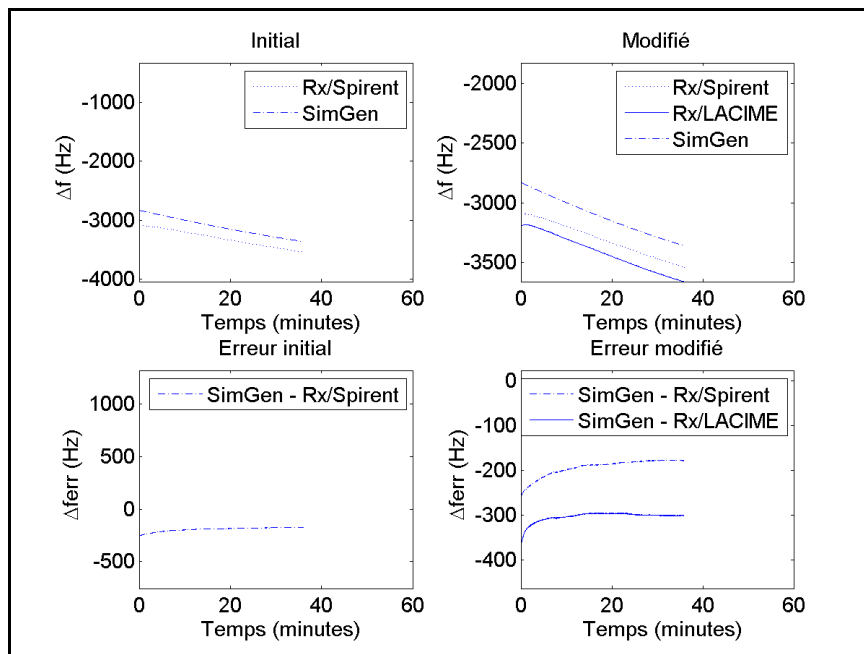


Figure-A VII-16 Δf et Δf_{err} initial et modifié du satellite #28

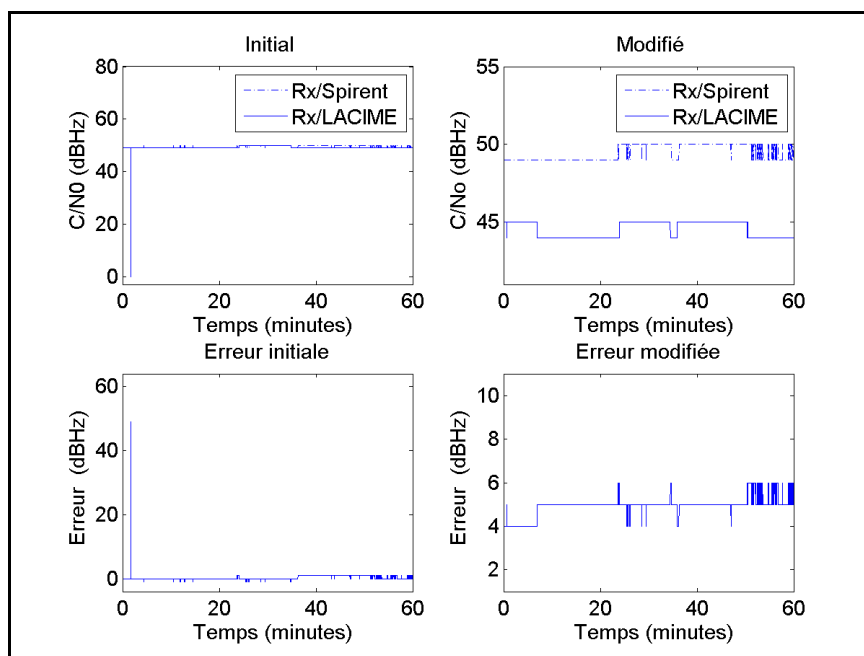


Figure-A VII-17 C/N_0 initial et modifié et son erreur (Rx/Spirent-Rx/LACIME) du satellite #2

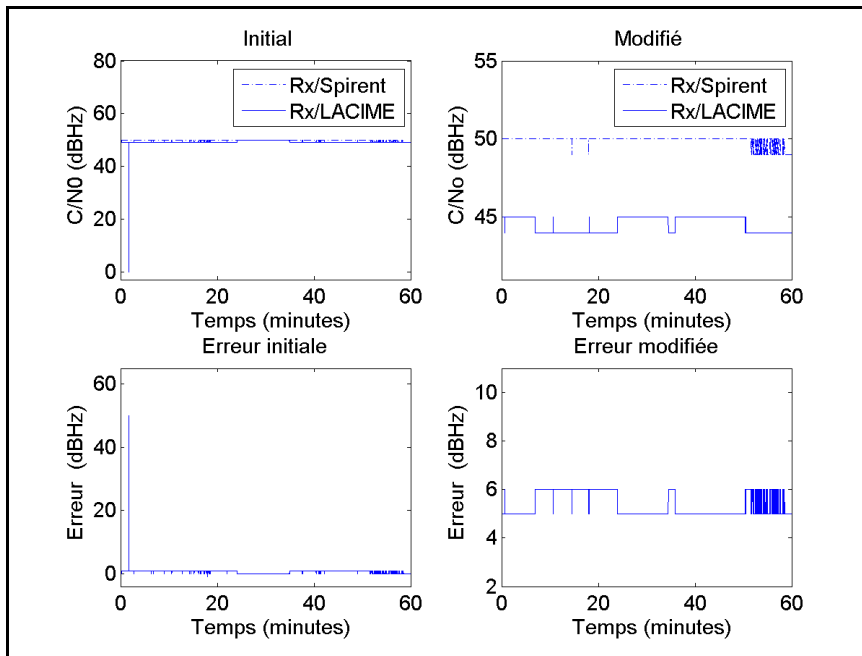


Figure-A VII-18 C/N_0 initial et modifié et son erreur (Rx/Spirent-Rx/LACIME) du satellite #4

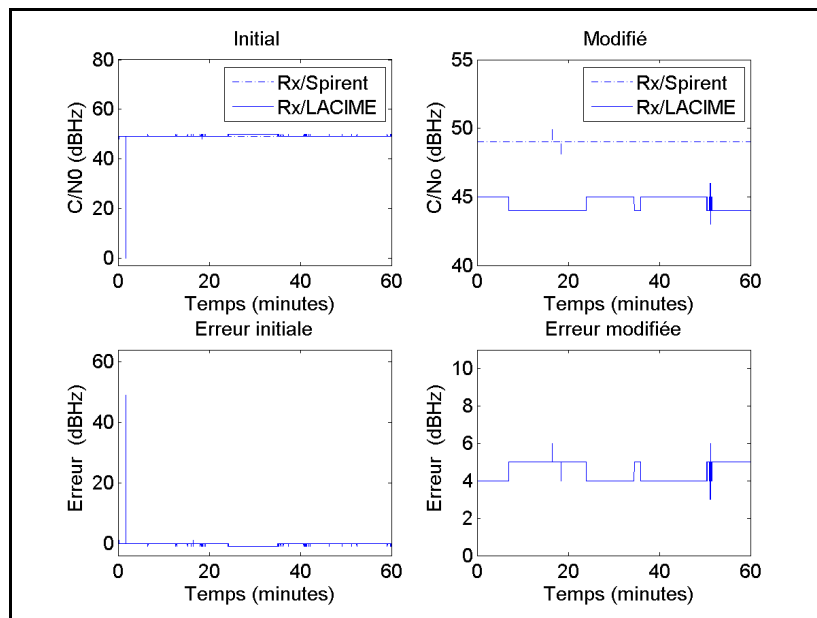


Figure-A VII-19 C/N_0 initial et modifié et son erreur (Rx/Spirent-Rx/LACIME) du satellite #9

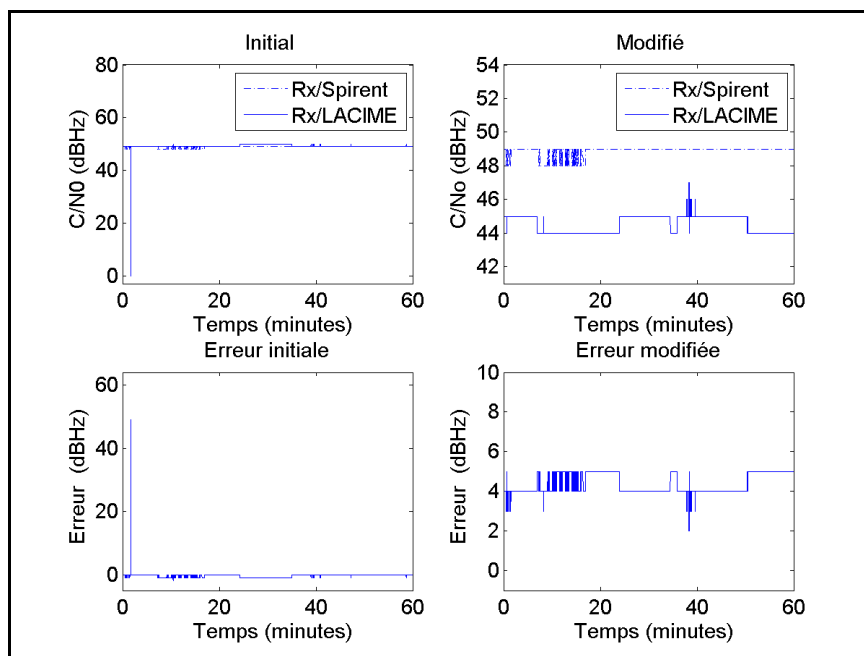


Figure-A VII-20 C/N_0 initial et modifié et son erreur (Rx/Spirent-Rx/LACIME) du satellite #12

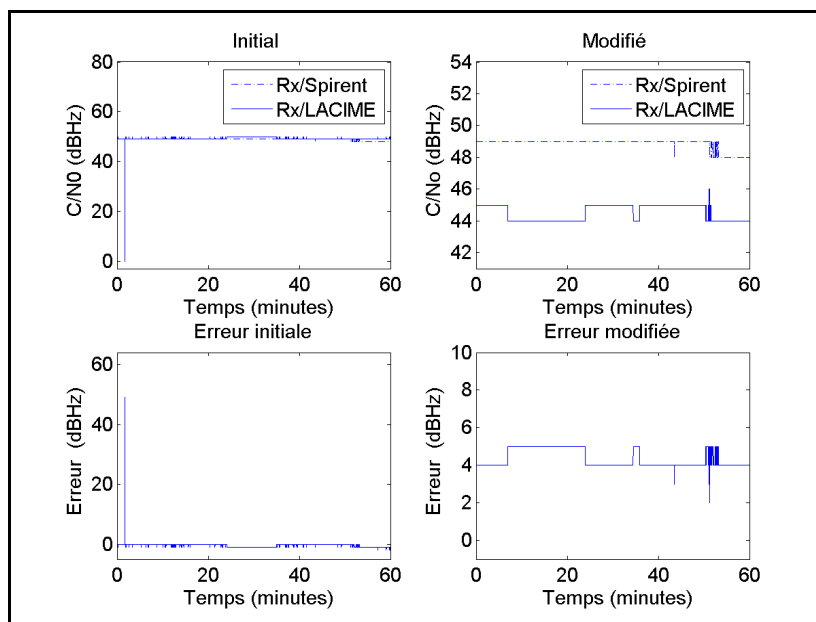


Figure-A VII-21 C/N_0 initial et modifié et son erreur (Rx/Spirent-Rx/LACIME) du satellite #17

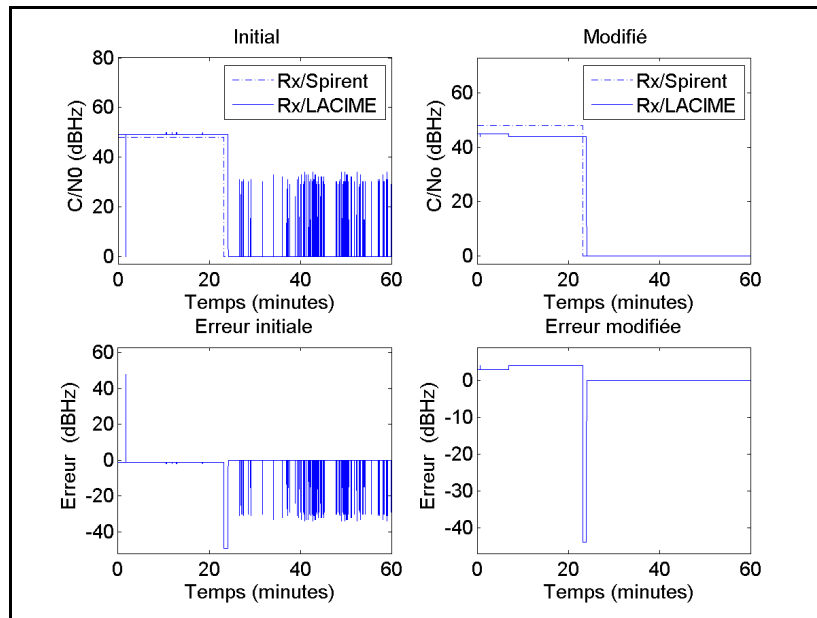


Figure-A VII-22 C/N₀ initial et modifié et son erreur (Rx/Spirent-Rx/LACIME) du satellite #20

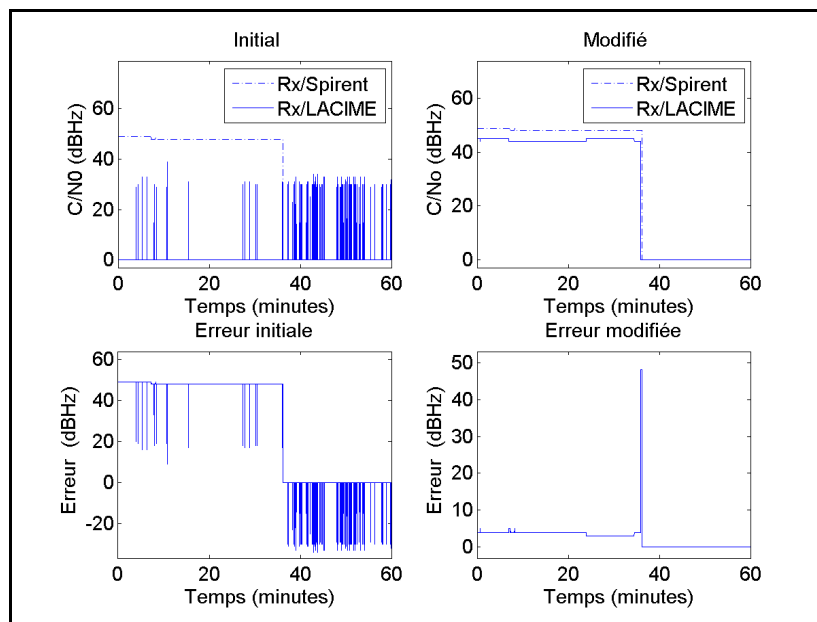


Figure-A VII-23 C/N₀ initial et modifié et son erreur (Rx/Spirent-Rx/LACIME) du satellite #28

ANNEXE VIII

RÉSULTATS DE L'ERREUR DE LA PHASE DE LA PORTEUSE, DU CODE ET DU MESSAGE DE NAVIGATION

Voici les résultats initiaux et modifiés de l'erreur de la phase de la porteuse, du code et du message de navigation du simulateur LACIME.

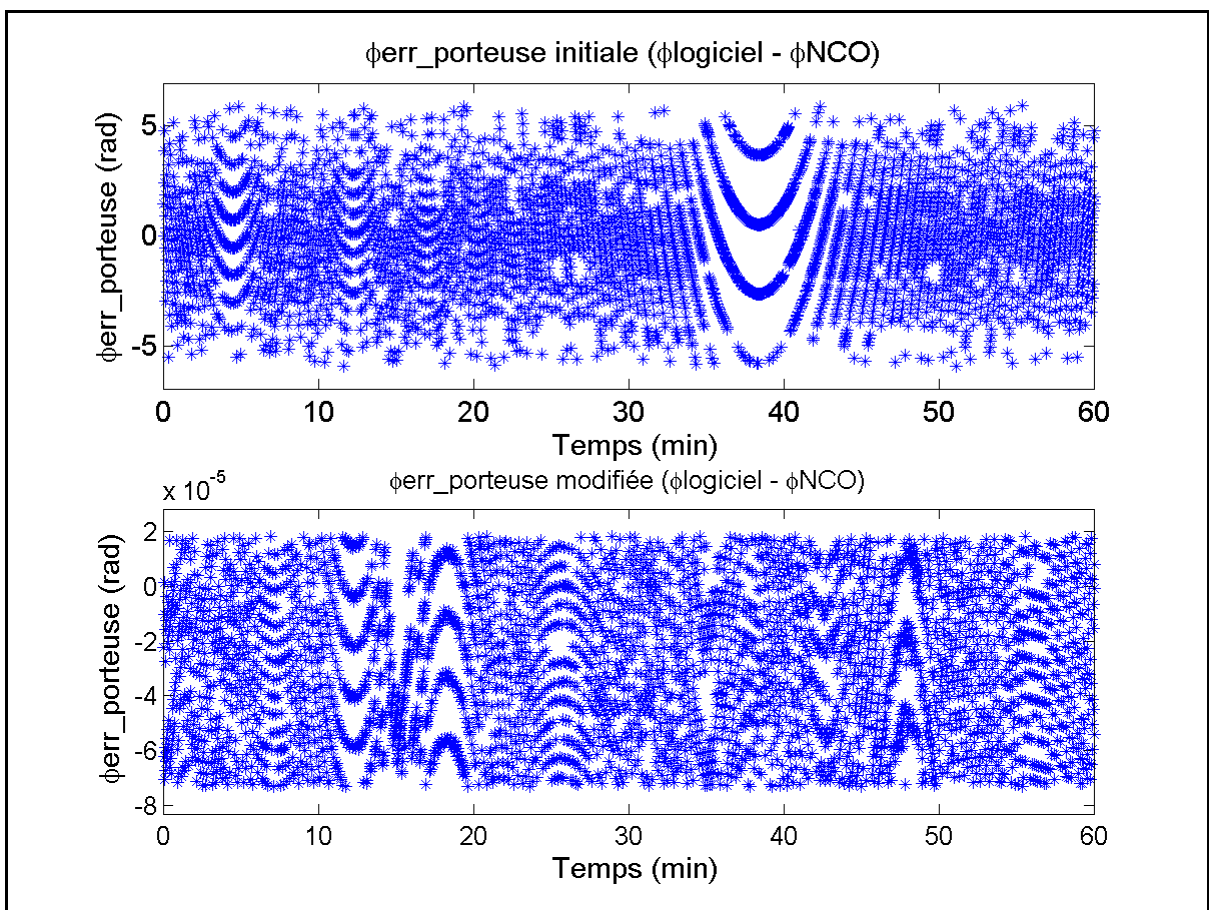


Figure-A VIII-1 $\phi_{err_porteuse}$ initiale et modifiée du satellite #2 avec $F_s=1\text{Hz}$

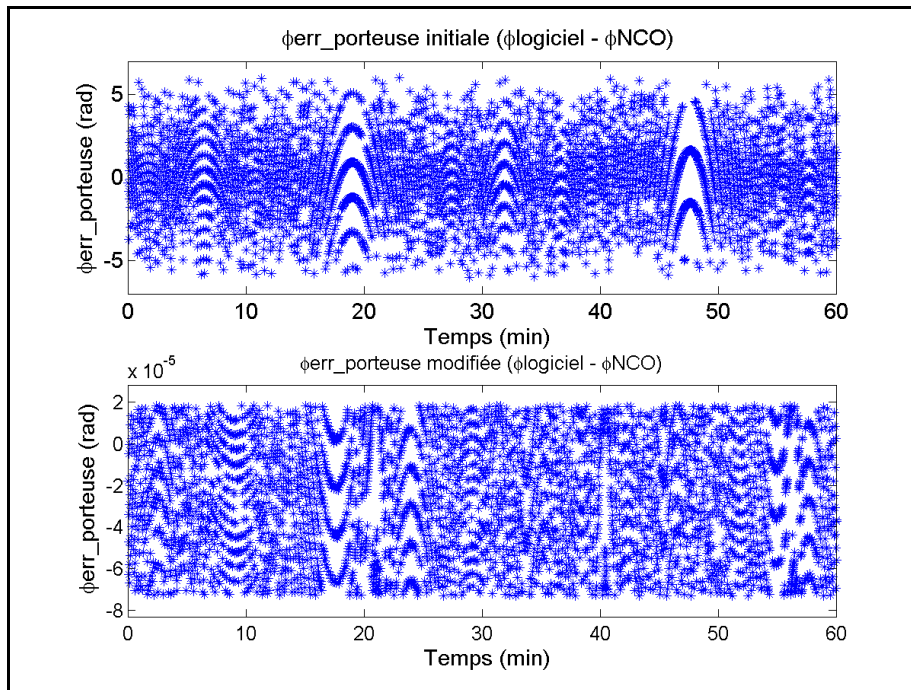


Figure-A VIII-2 $\phi_{err_porteuse}$ initiale et modifiée du satellite #9 avec $F_s = 1\text{Hz}$

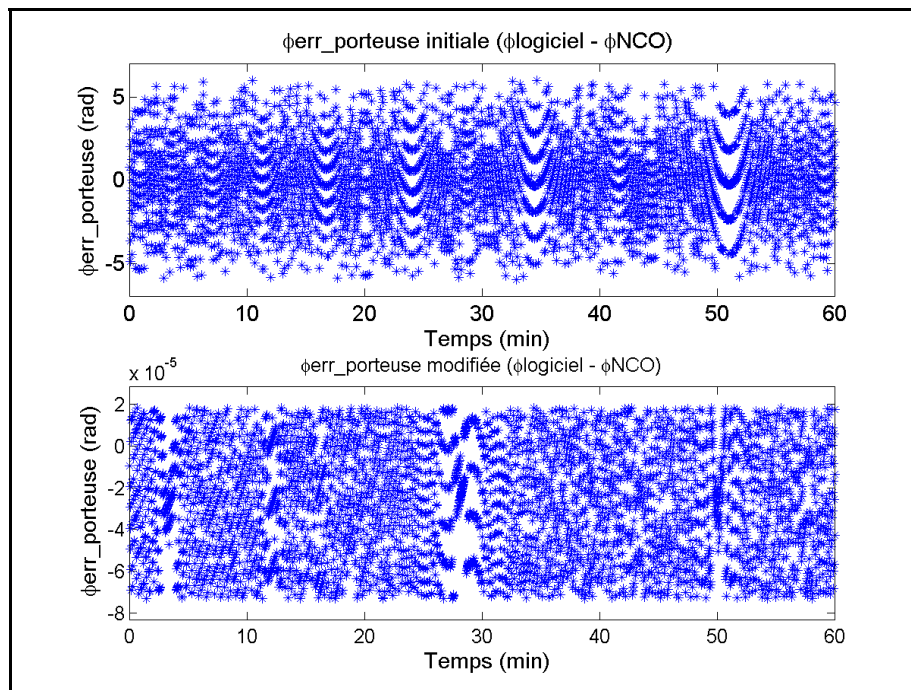


Figure-A VIII-3 $\phi_{err_porteuse}$ initiale et modifiée du satellite #12 avec $F_s = 1\text{Hz}$

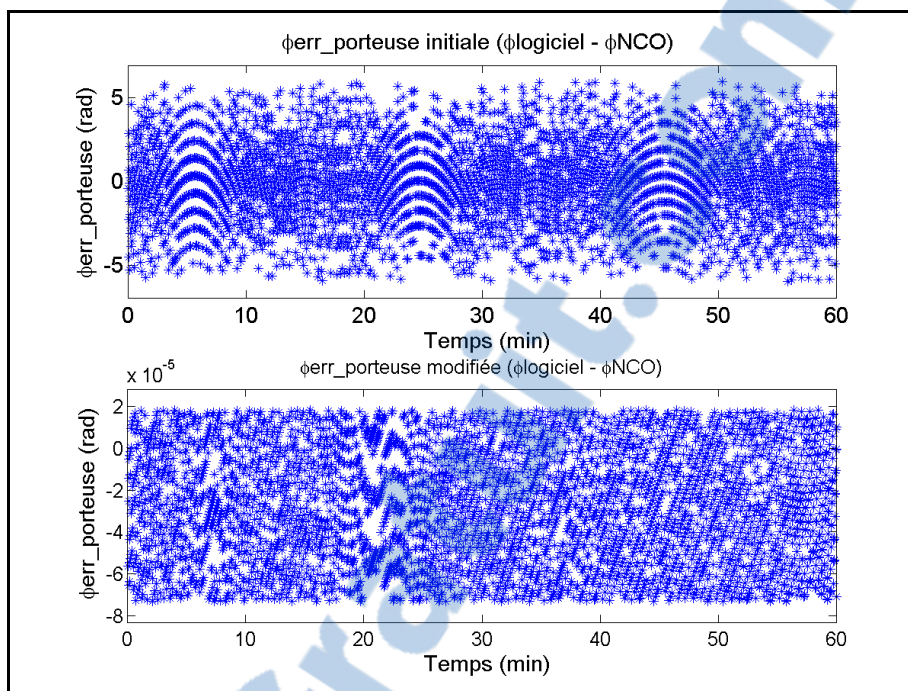


Figure-A VIII-4 $\phi_{err_porteur}$ initiale et modifiée du satellite #17 avec $F_s=1\text{Hz}$

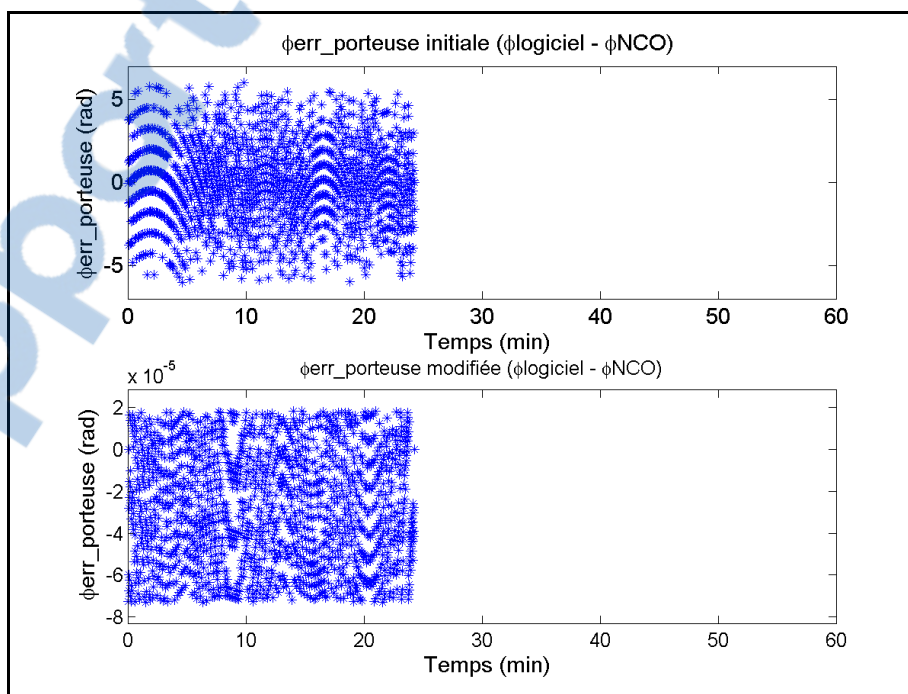


Figure-A VIII-5 $\phi_{err_porteur}$ initiale et modifiée du satellite #20 avec $F_s=1\text{Hz}$

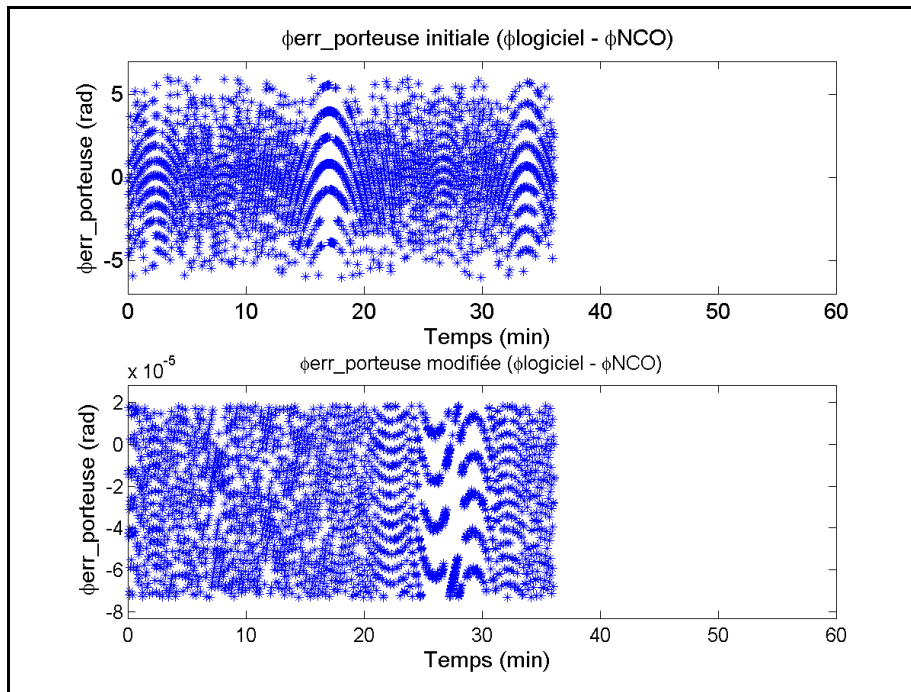


Figure-A VIII-6 $\phi_{err_porteuse}$ initiale et modifiée du satellite #28 avec $F_s=1\text{Hz}$

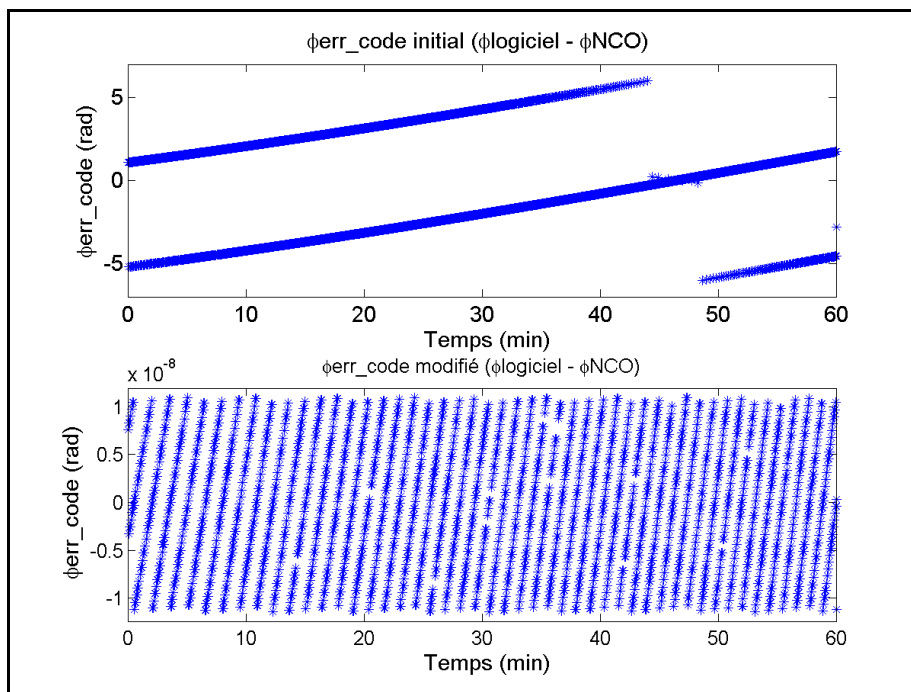


Figure-A VIII-7 ϕ_{err_code} initiale et modifiée du satellite #2 avec $F_s=1\text{Hz}$

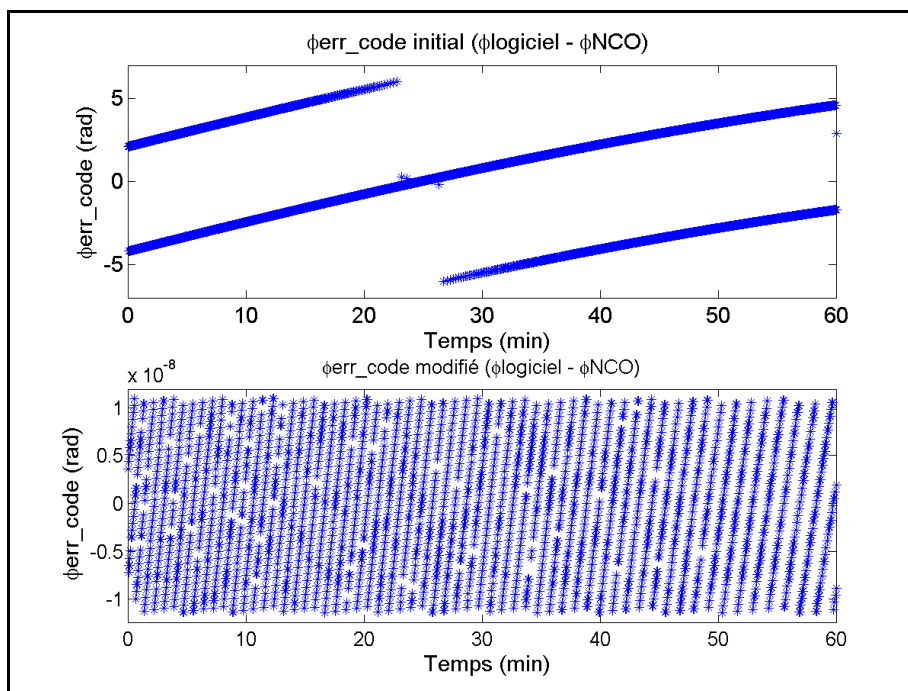


Figure-A VIII-8 ϕ_{err_code} initiale et modifiée du satellite #9 avec $F_s = 1\text{Hz}$

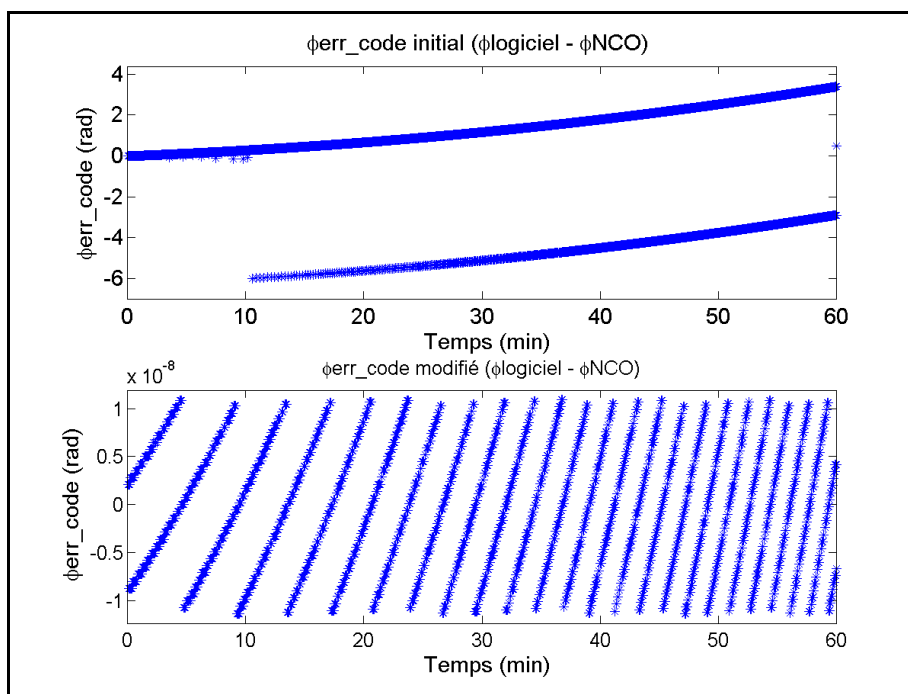
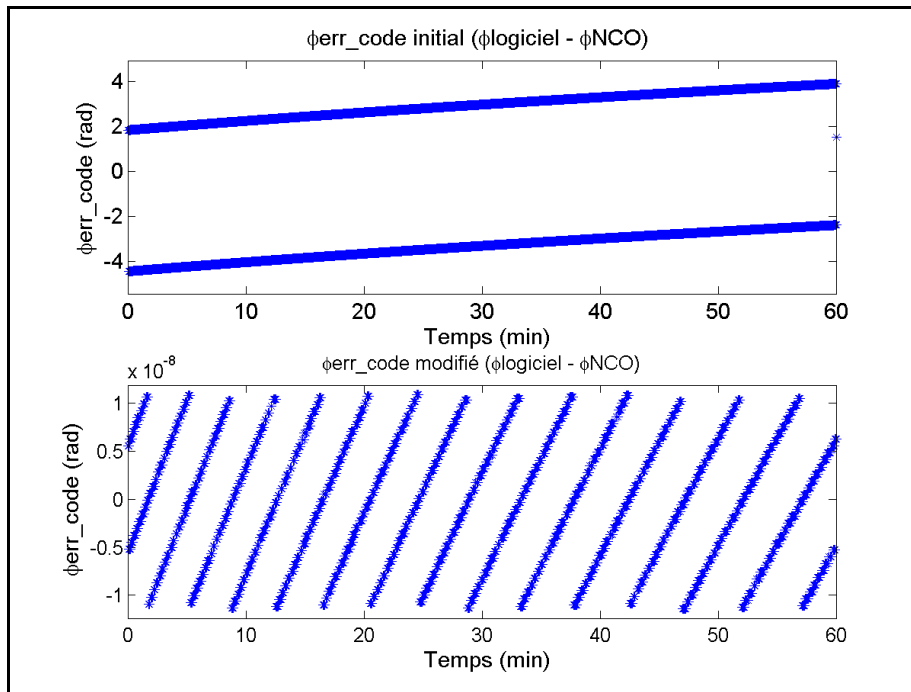
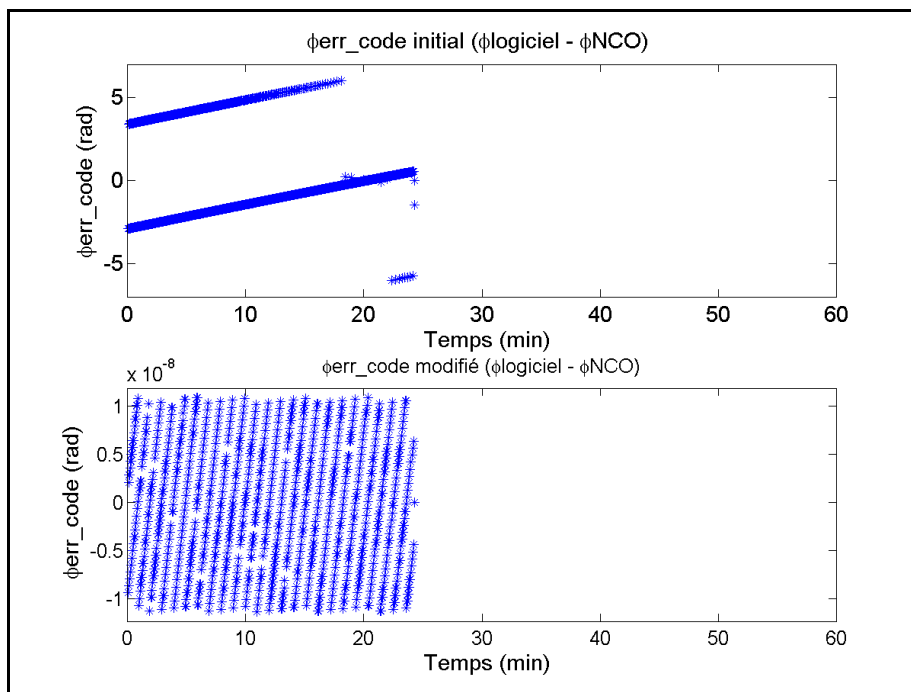


Figure-A VIII-9 ϕ_{err_code} initiale et modifiée du satellite #12 avec $F_s = 1\text{Hz}$

Figure-A VIII-10 ϕ_{err_code} initiale et modifiée du satellite #17 avec $F_s=1\text{Hz}$ Figure-A VIII-11 ϕ_{err_code} initiale et modifiée du satellite #20 avec $F_s=1\text{Hz}$

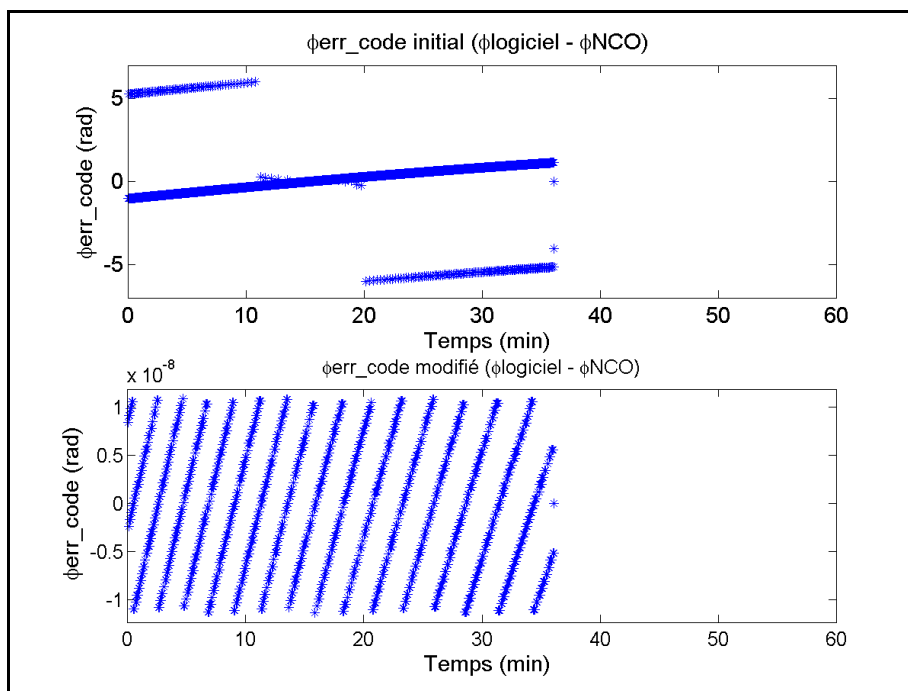


Figure-A VIII-12 ϕ_{err_code} initiale et modifiée du satellite #28 avec $F_s=1\text{Hz}$

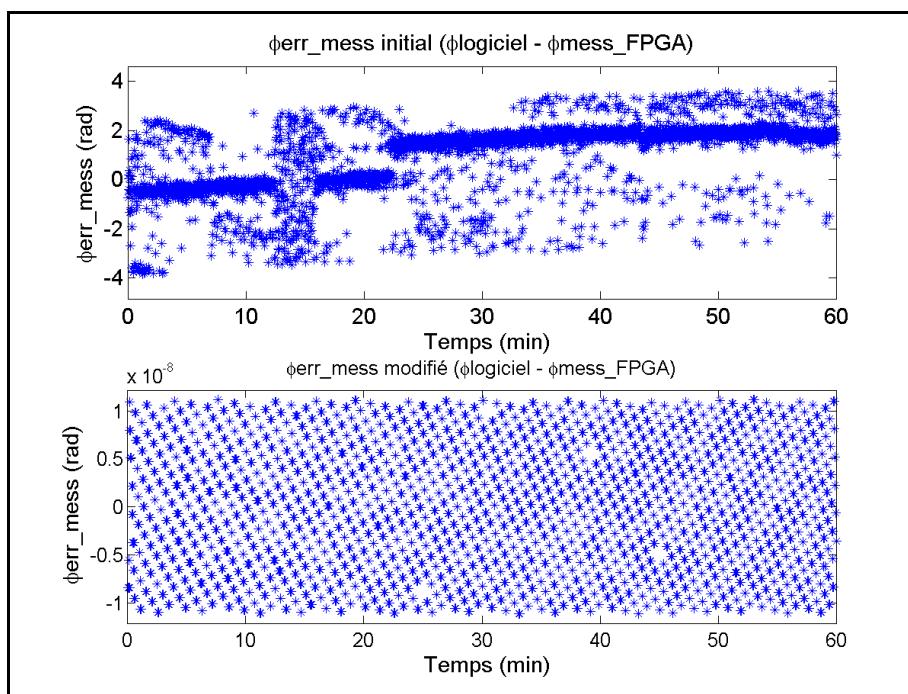


Figure-A VIII-13 ϕ_{err_mess} initiale et modifiée du satellite #2 avec $F_s=1\text{Hz}$

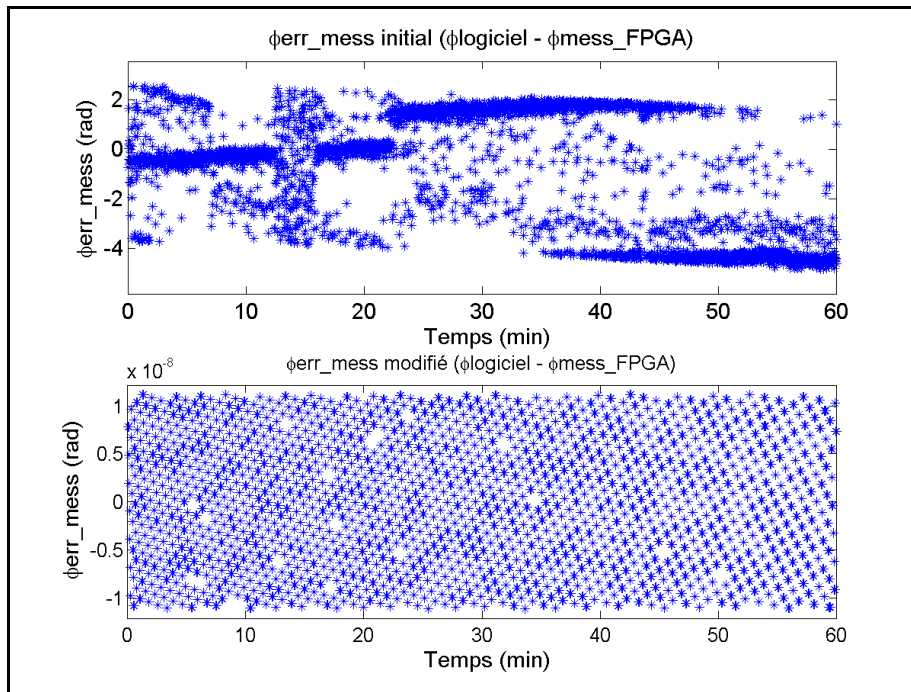


Figure-A VIII-14 ϕ_{err_mess} initiale et modifiée du satellite #9 avec $F_s=1\text{Hz}$

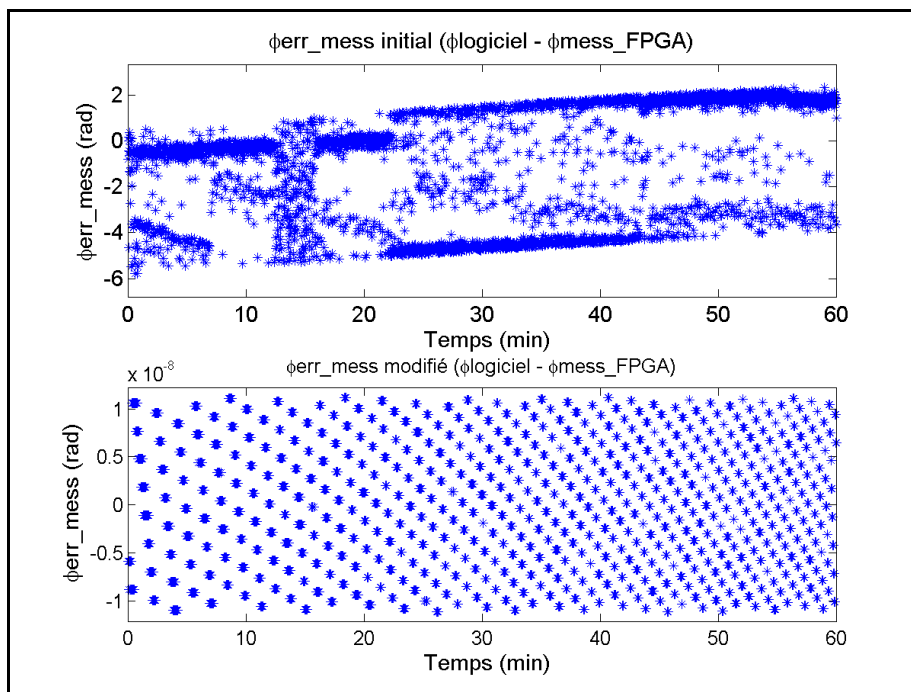


Figure-A VIII-15 ϕ_{err_mess} initiale et modifiée du satellite #12 avec $F_s=1\text{Hz}$

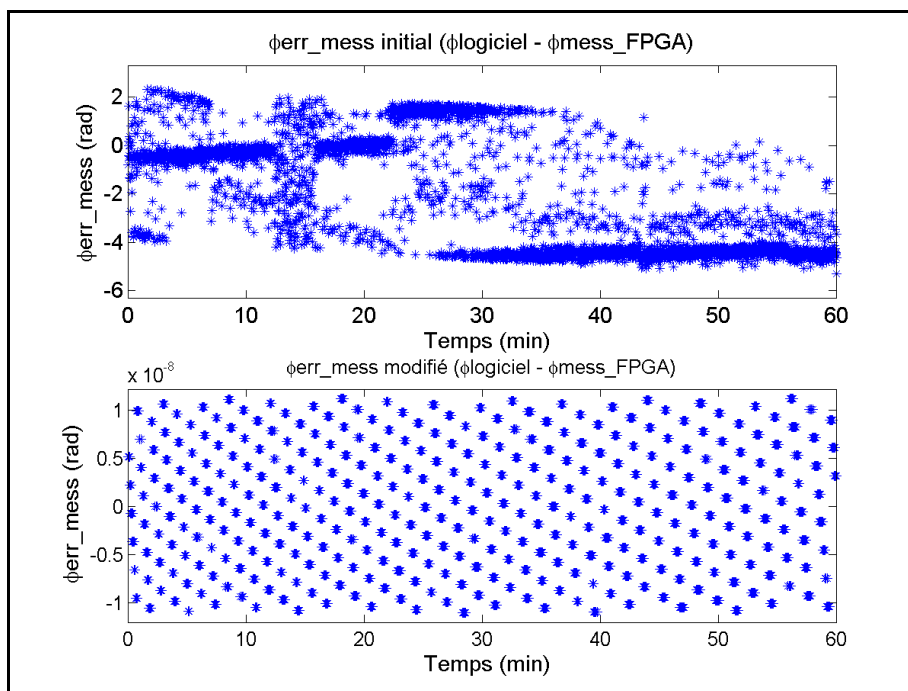


Figure-A VIII-16 ϕ_{err_mess} initiale et modifiée du satellite #17 avec $F_s=1\text{Hz}$

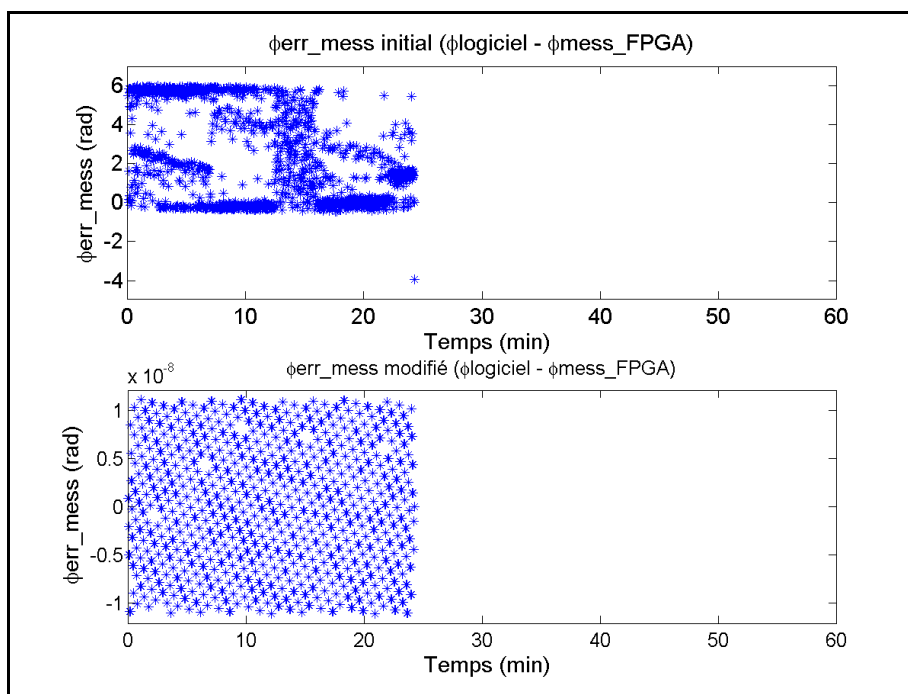


Figure-A VIII-17 ϕ_{err_mess} initiale et modifiée du satellite #20 avec $F_s=1\text{Hz}$

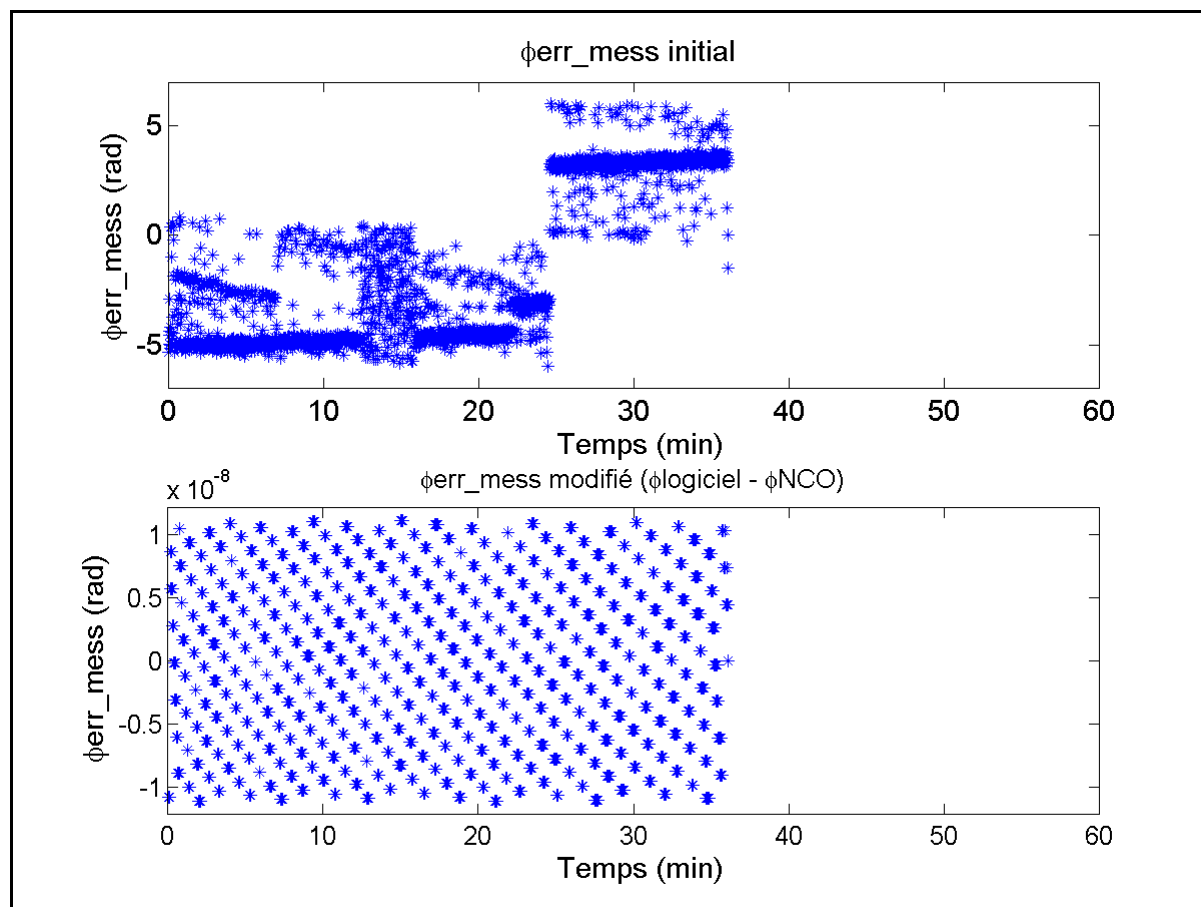


Figure-A VIII-18 ϕ_{err_mess} initiale et modifiée du satellite #28 avec $F_s=1\text{Hz}$

ANNEXE IX

RÉSULTATS INITIAUX ET MODIFIÉS DE ERR_CODE ET DE ERR_MESS

Voici les résultats initiaux et modifiés de err_code et err_mess du simulateur LACIME.

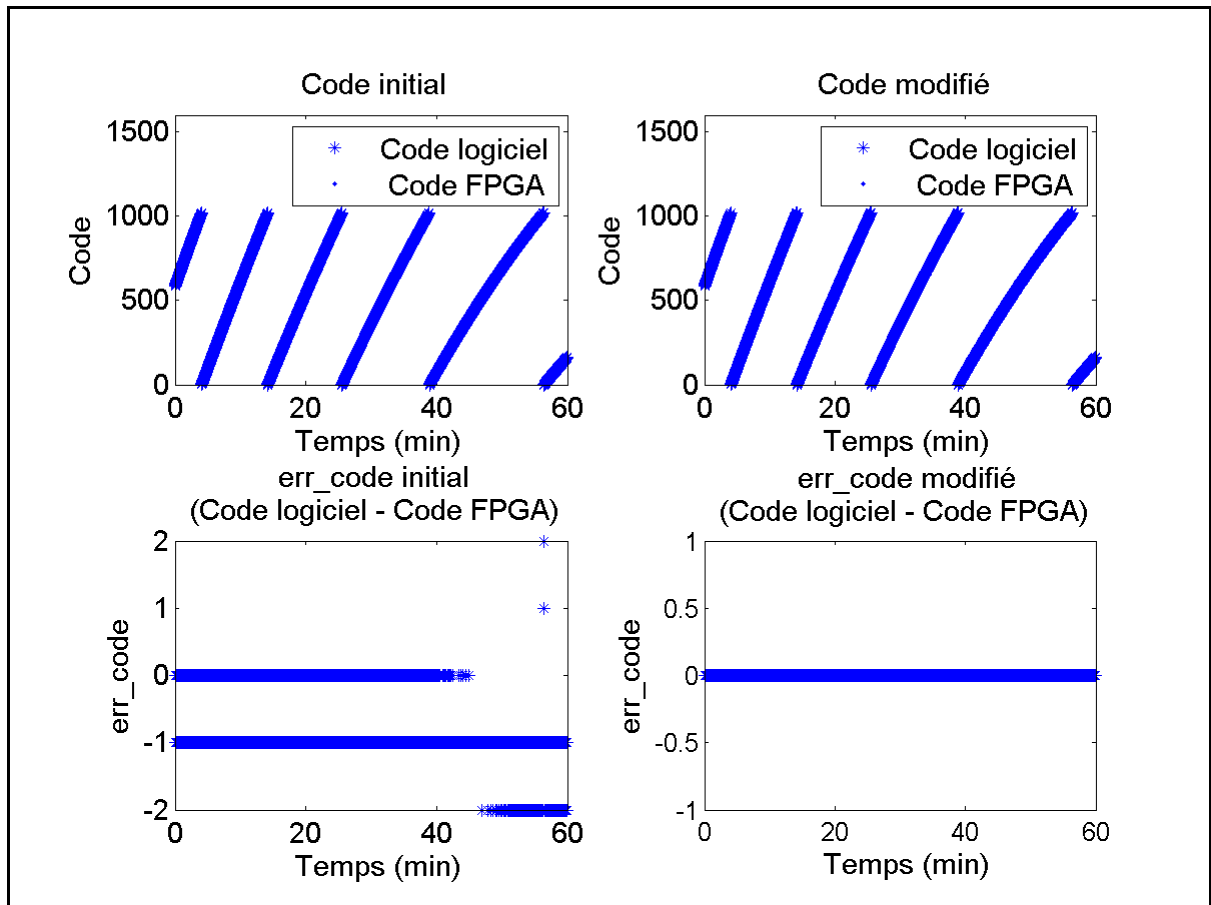
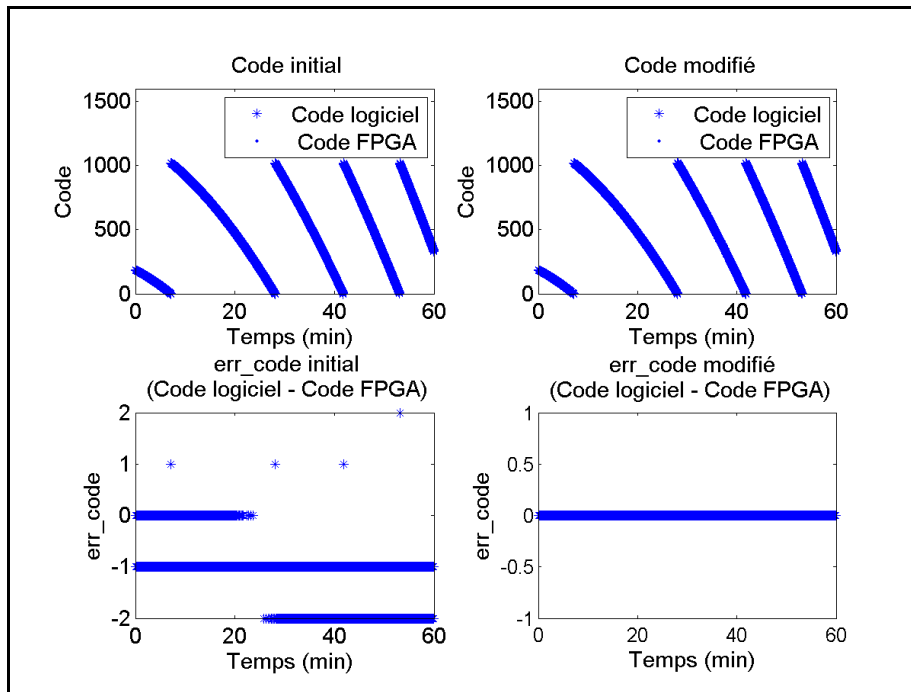
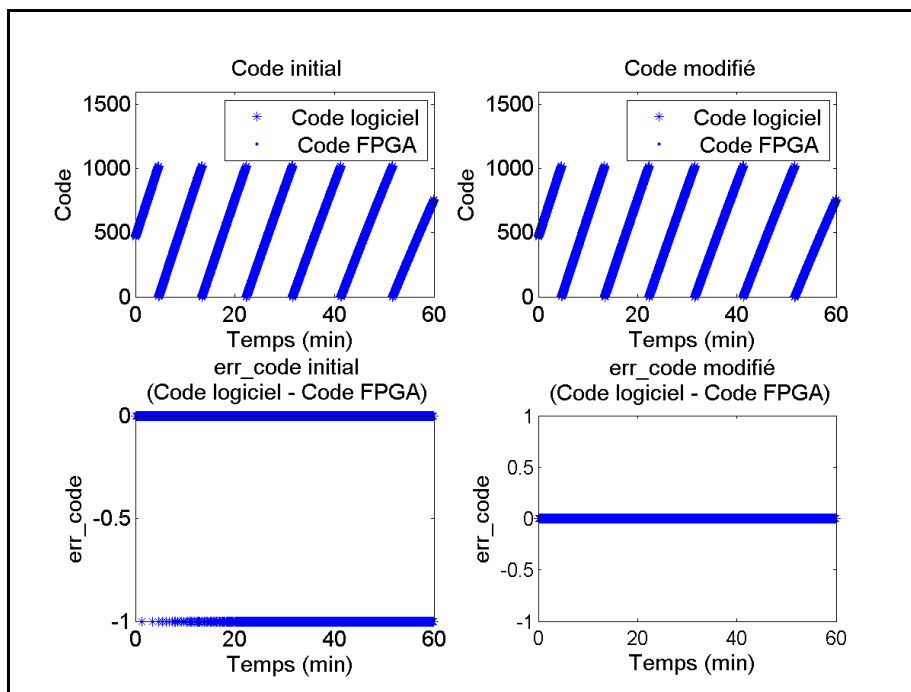


Figure-A IX-1 err_code initial et modifié du satellite #2 avec $F_s=1\text{Hz}$

Figure-A IX-2 err_code initial et modifié du satellite #9 avec $F_s=1\text{Hz}$ Figure-A IX-3 err_code initial et modifié du satellite #12 avec $F_s=1\text{Hz}$

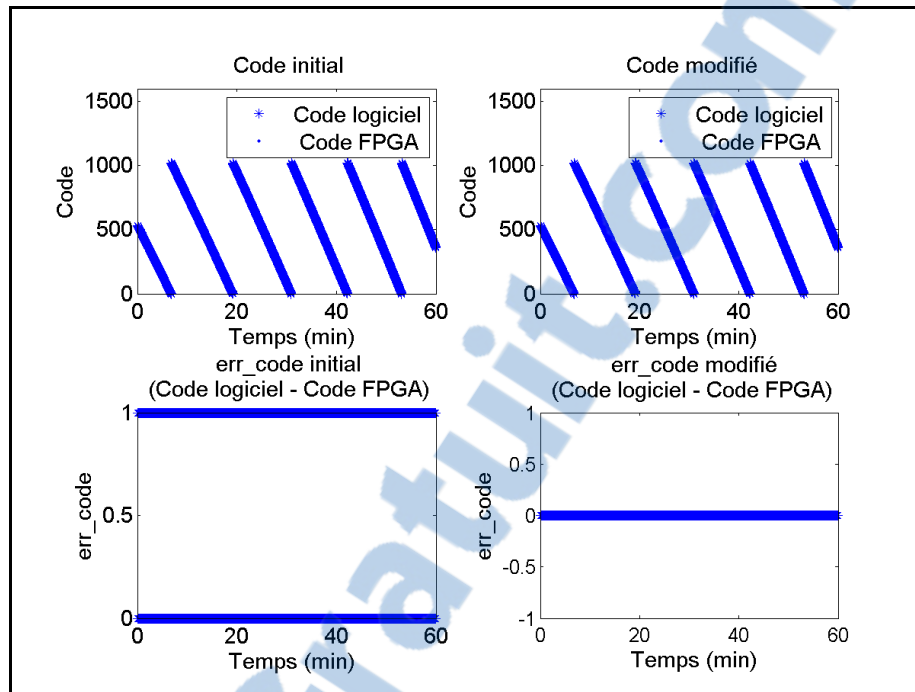


Figure-A IX-4 err_code initial et modifié du satellite #17 avec $F_s=1\text{Hz}$

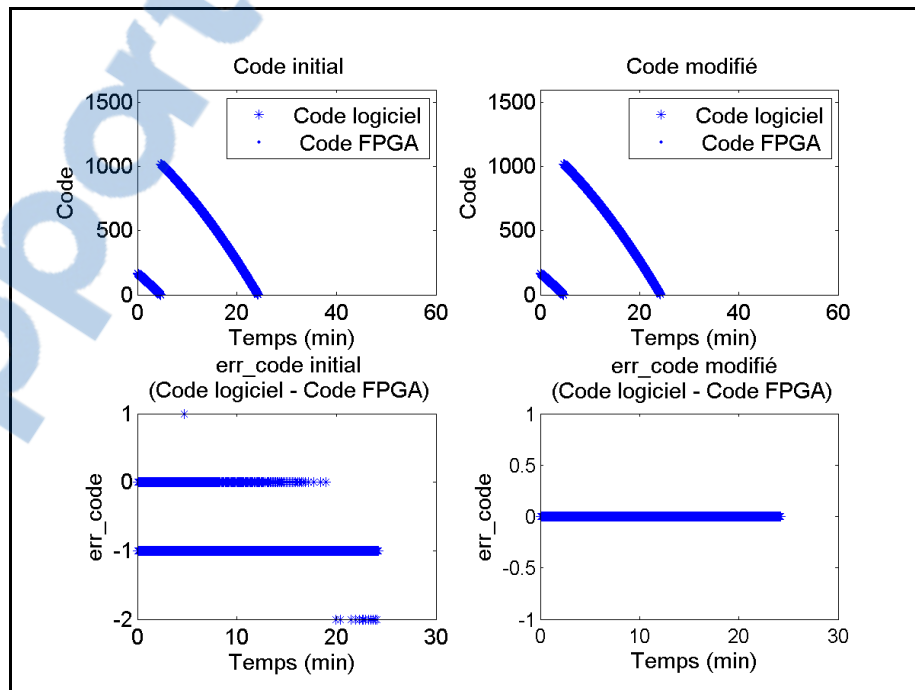


Figure-A IX-5 err_code initial et modifié du satellite #20 avec $F_s=1\text{Hz}$

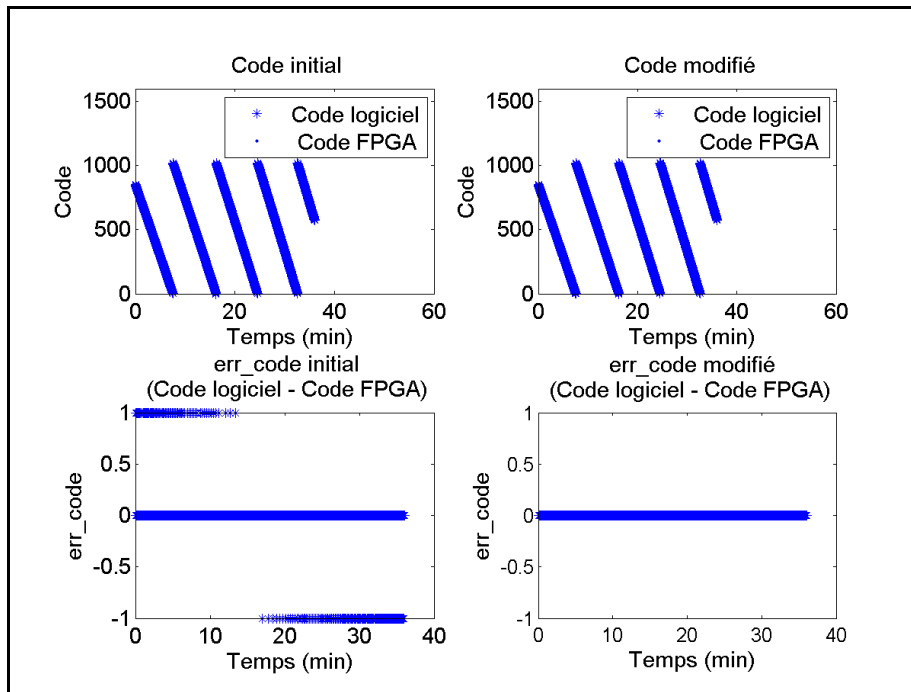


Figure-A IX-6 err_code initial et modifié du satellite #28 avec $F_s=1\text{Hz}$

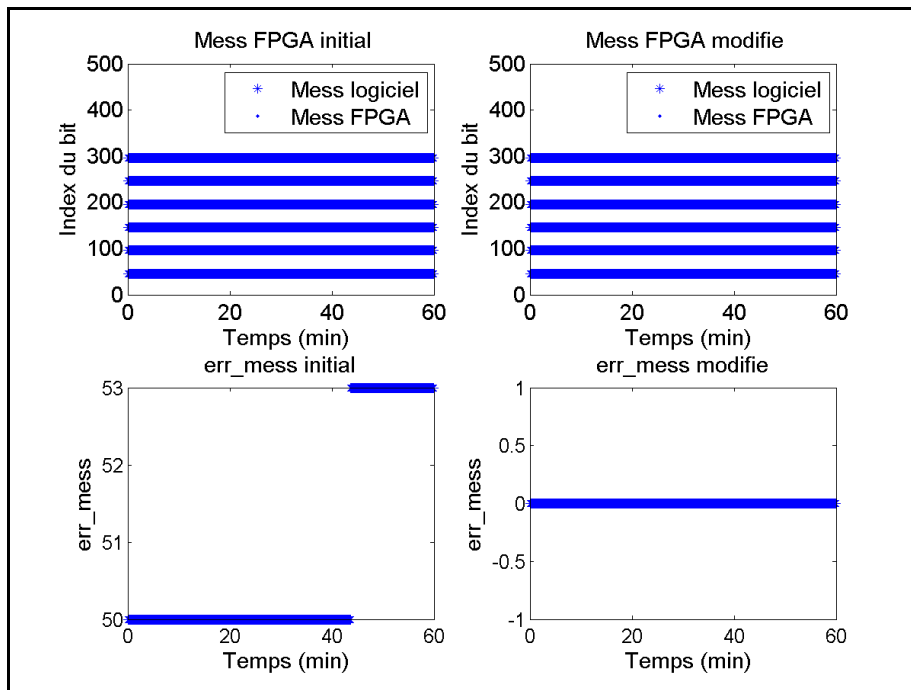


Figure-A IX-7 err_mess initial et modifié du satellite #2 avec $F_s=1\text{Hz}$

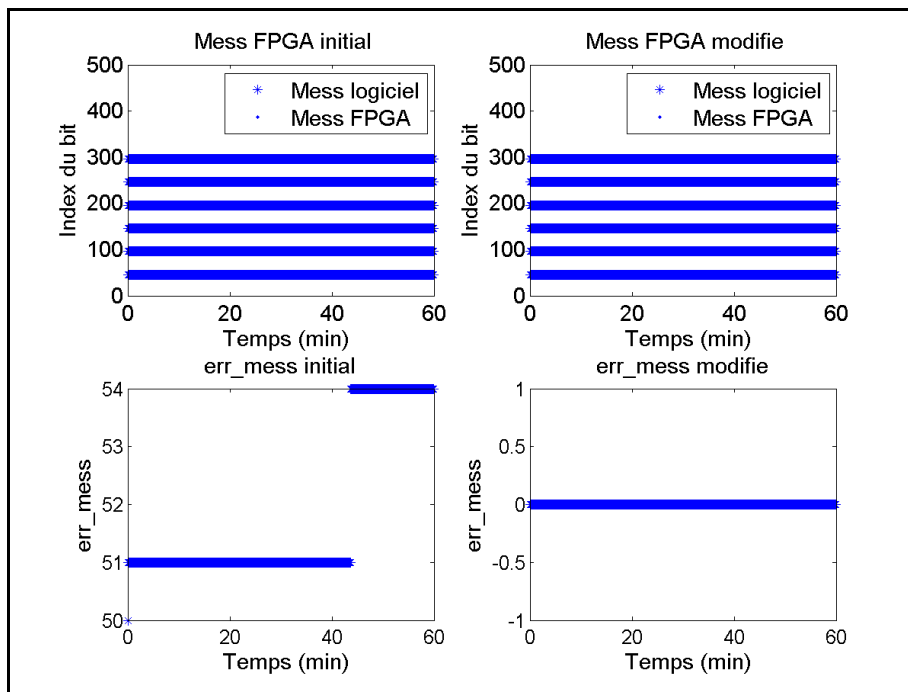


Figure-A IX-8 err_mess initial et modifié du satellite #9 avec $F_s=1\text{Hz}$

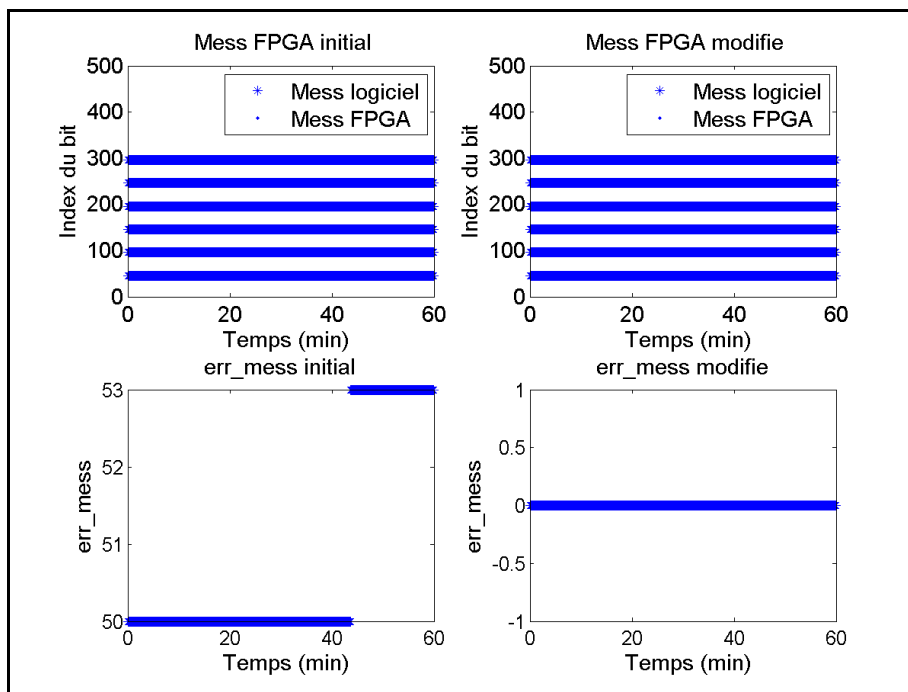


Figure-A IX-9 err_mess initial et modifié du satellite #12 avec $F_s=1\text{Hz}$

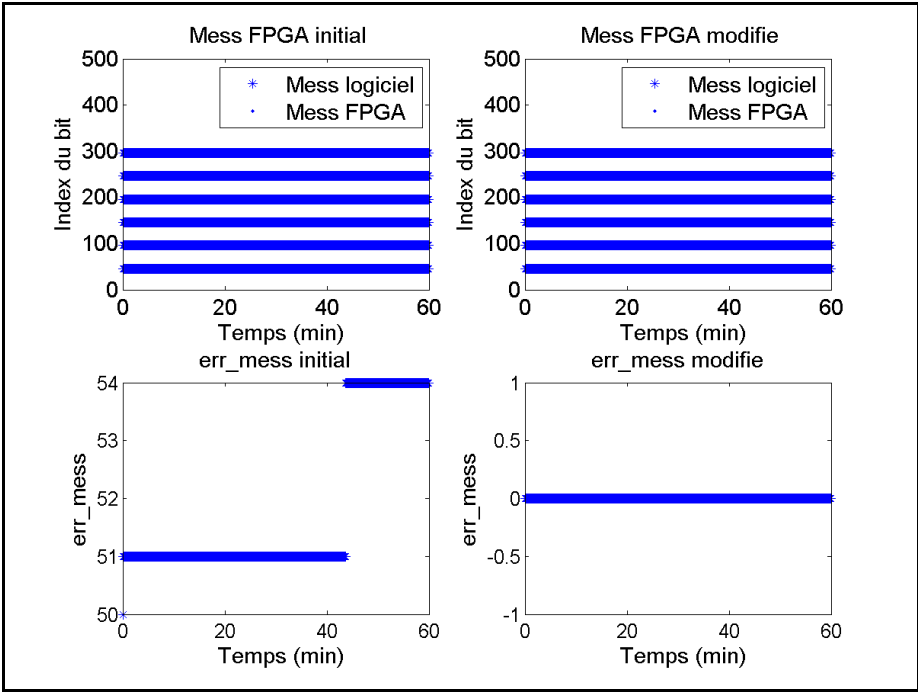


Figure-A IX-10 err_mess initial et modifié du satellite #17 avec $F_s=1\text{Hz}$

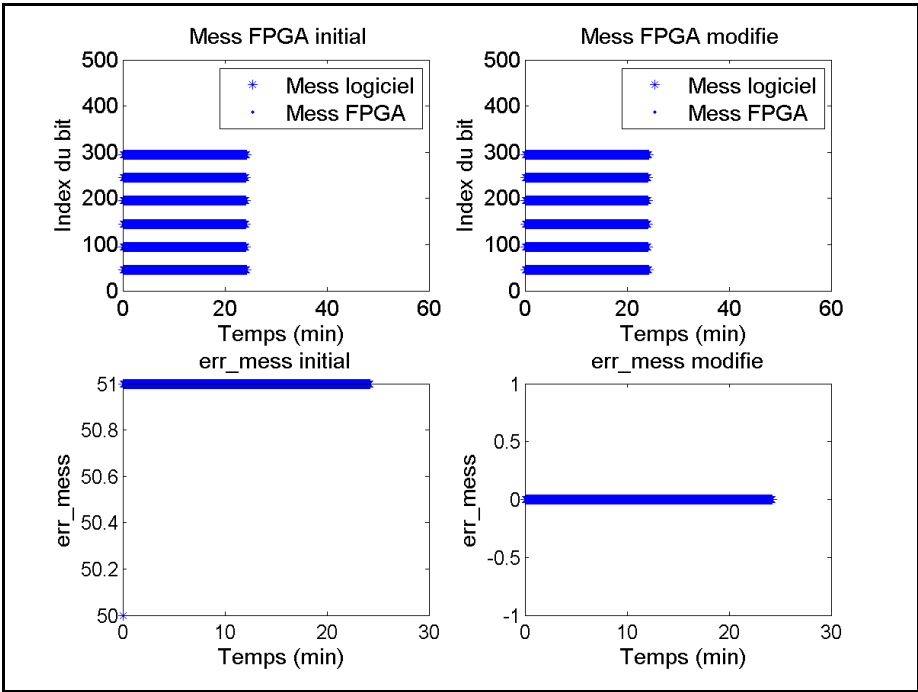


Figure-A IX-11 err_mess initial et modifié du satellite #20 avec $F_s=1\text{Hz}$

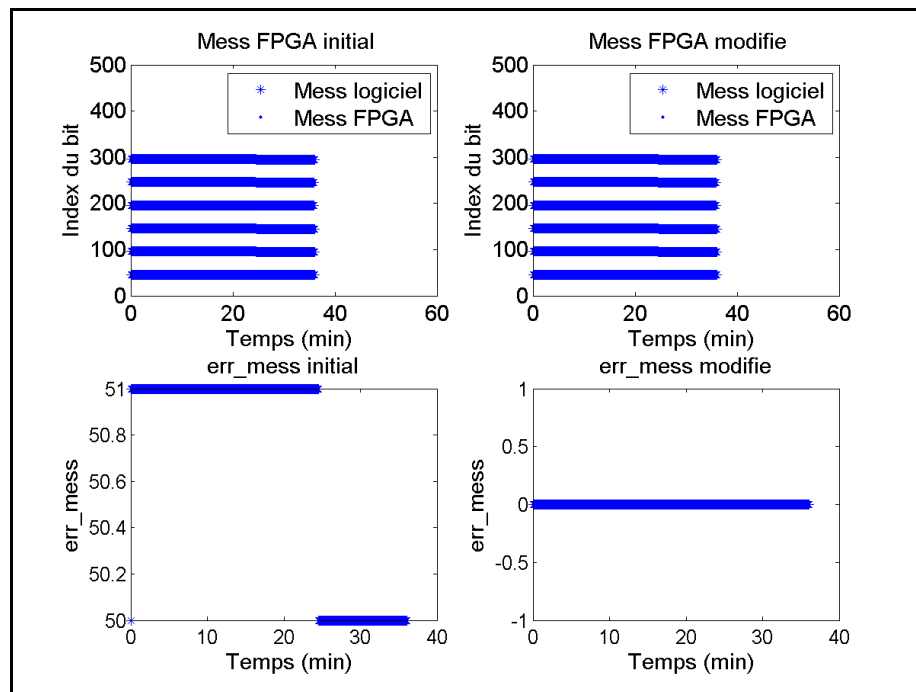


Figure-A IX-12 err_mess initial et modifié du satellite #28 avec $F_s=1\text{Hz}$

ANNEXE X

RÉSULTATS FINAUX DES SIMULATIONS EN STATIQUE

Voici les résultats des simulations en statique au Honduras, Marseille, Chine, Argentine, Natal, Afrique du Sud et Nouvelle-Zélande du simulateur LACIME.

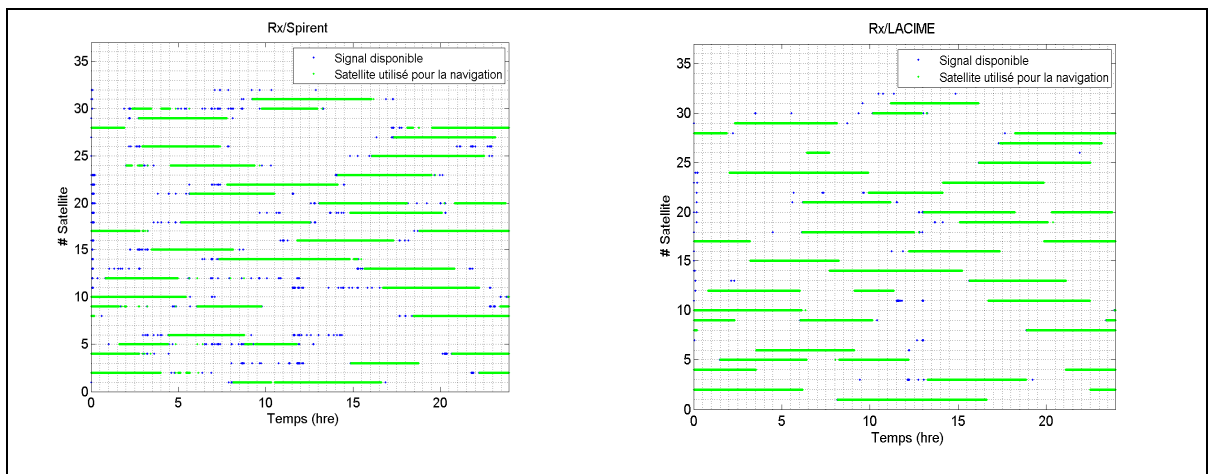


Figure-A X-1 Visibilités des satellites pour une simulation au Honduras (Rx/Spirent versus Rx/LACIME)

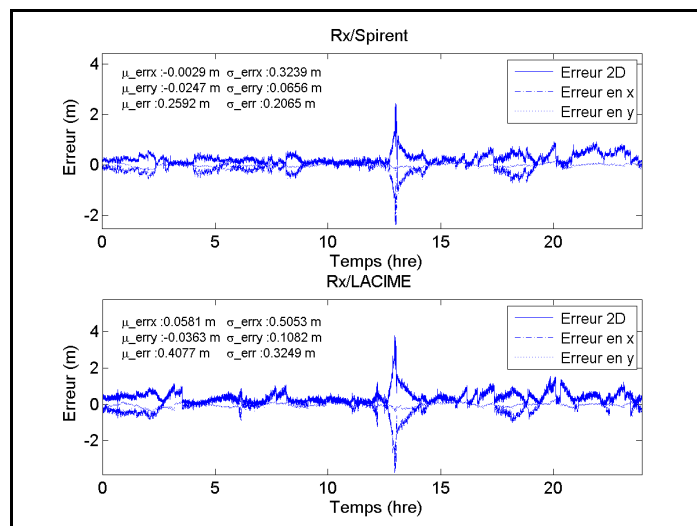


Figure-A X-2 L'erreur de position pour une simulation au Honduras (Rx/Spirent versus Rx/LACIME)

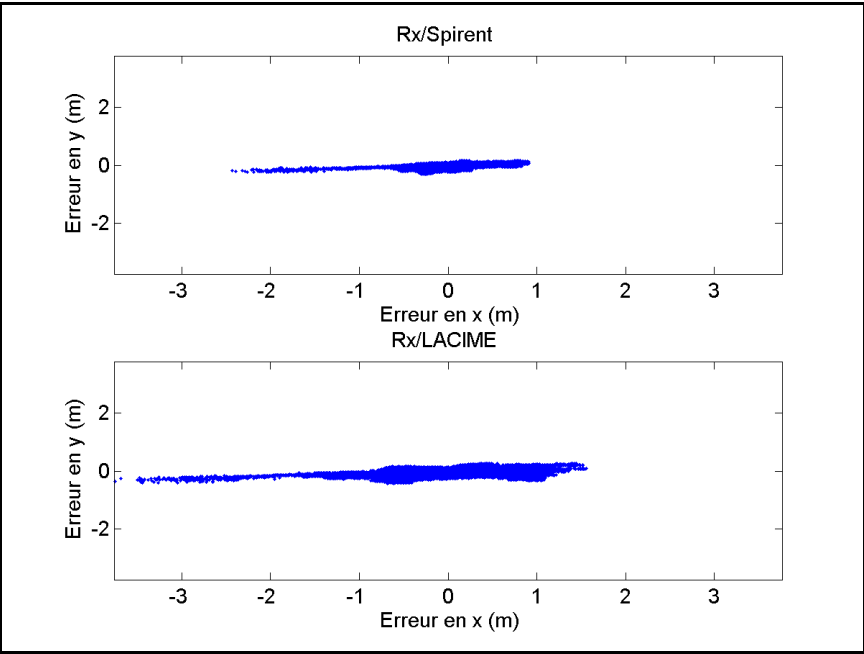


Figure-A X-3 Nuage de points pour une simulation au Honduras (Rx/Spirent versus Rx/LACIME)

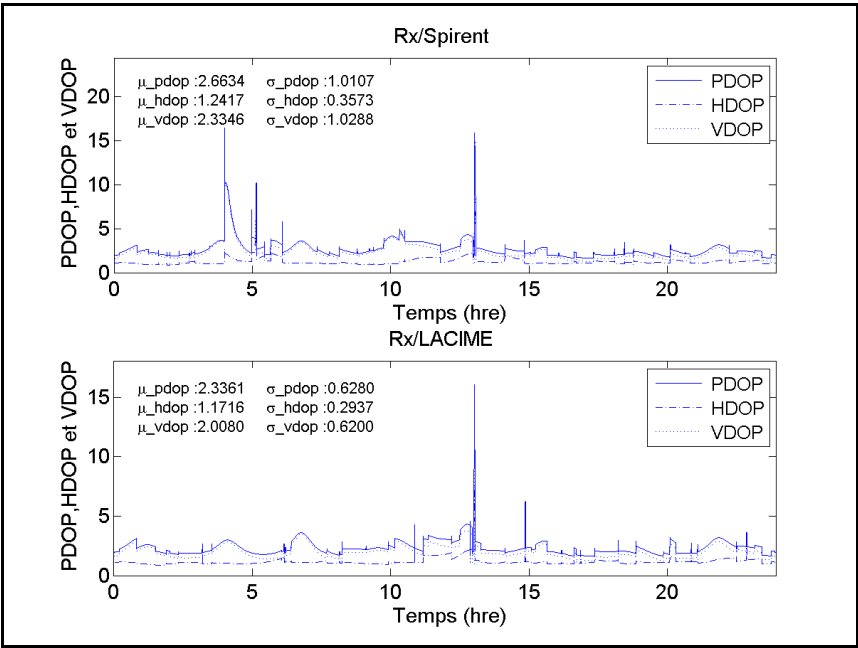


Figure-A X-4 PDOP, HDOP et VDOP pour une simulation au Honduras (Rx/Spirent versus Rx/LACIME)

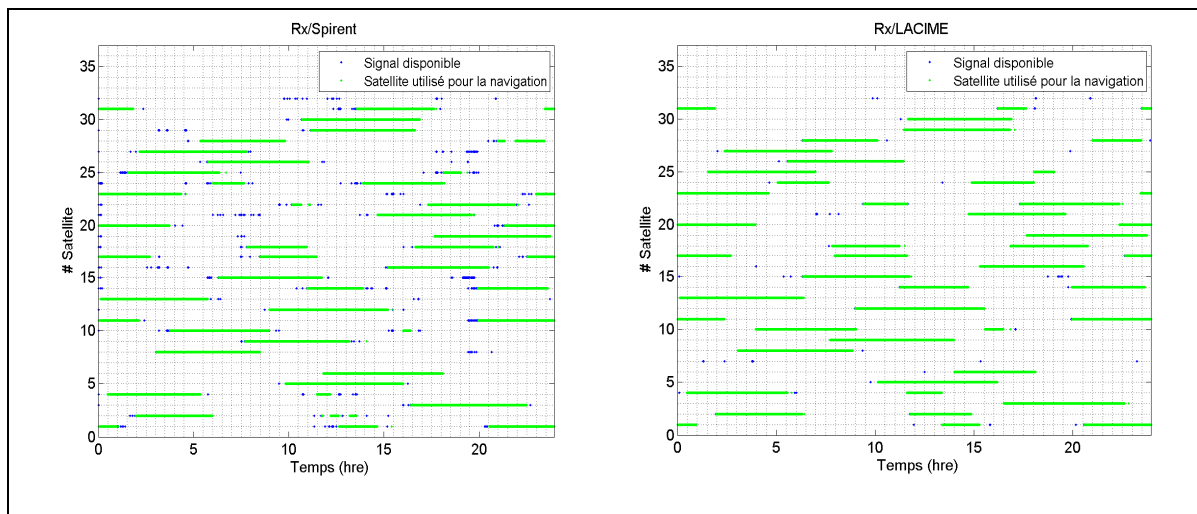


Figure-A X-5 Visibilités des satellites pour une simulation à Marseille (Rx/Spirent versus Rx/LACIME)

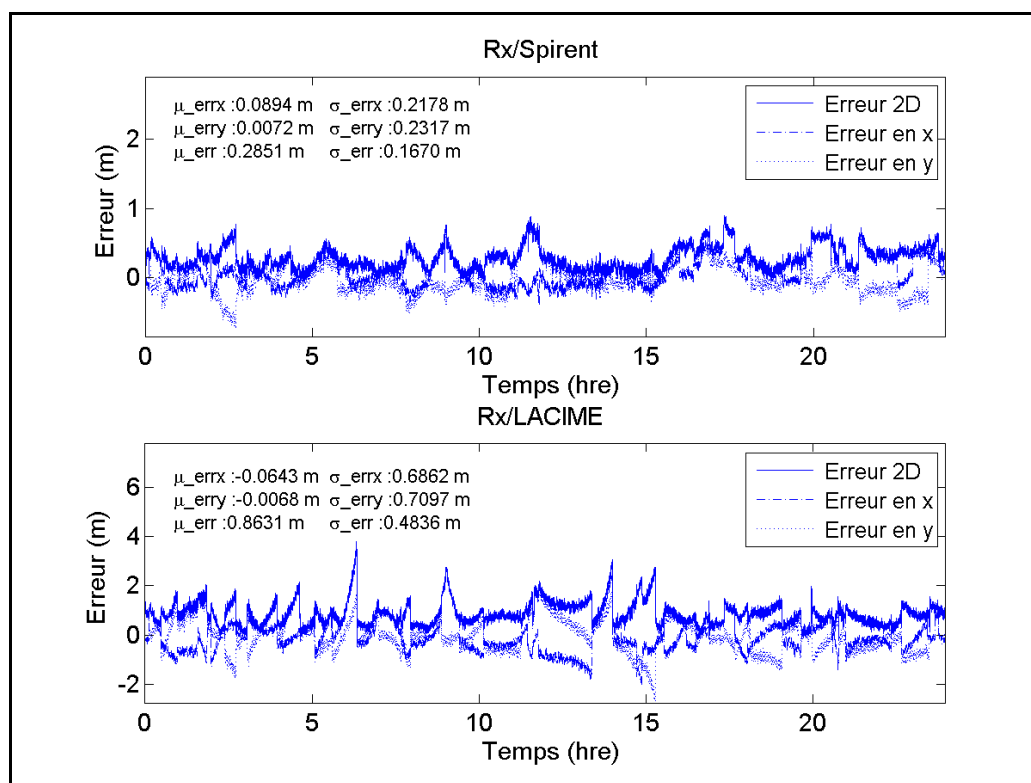


Figure-A X-6 L'erreur de position pour une simulation à Marseille (Rx/Spirent versus Rx/LACIME)

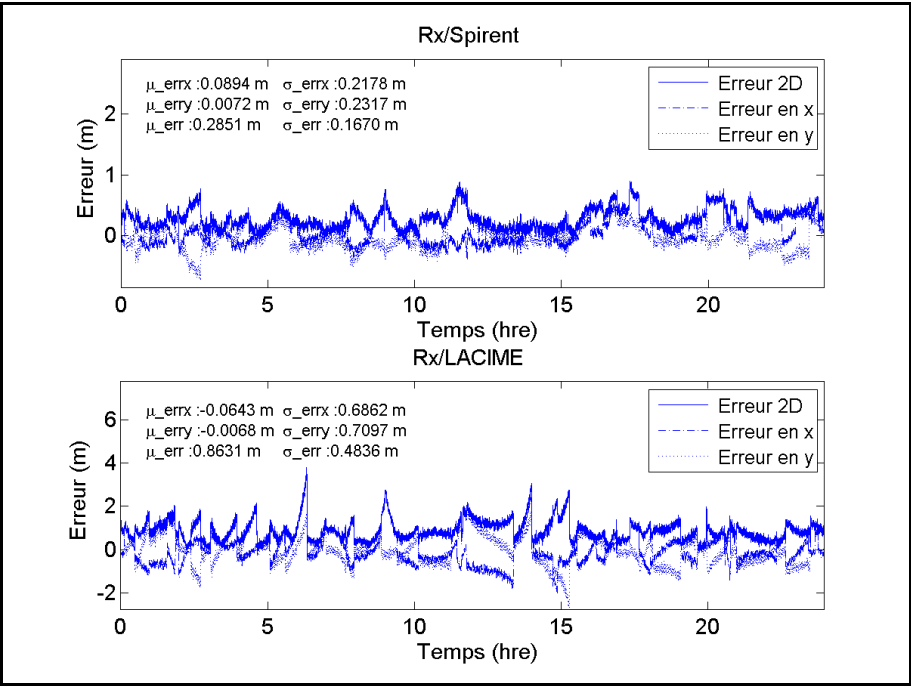


Figure-A X-7 L'erreur de position pour une simulation à Marseille (Rx/Spirent versus Rx/LACIME)

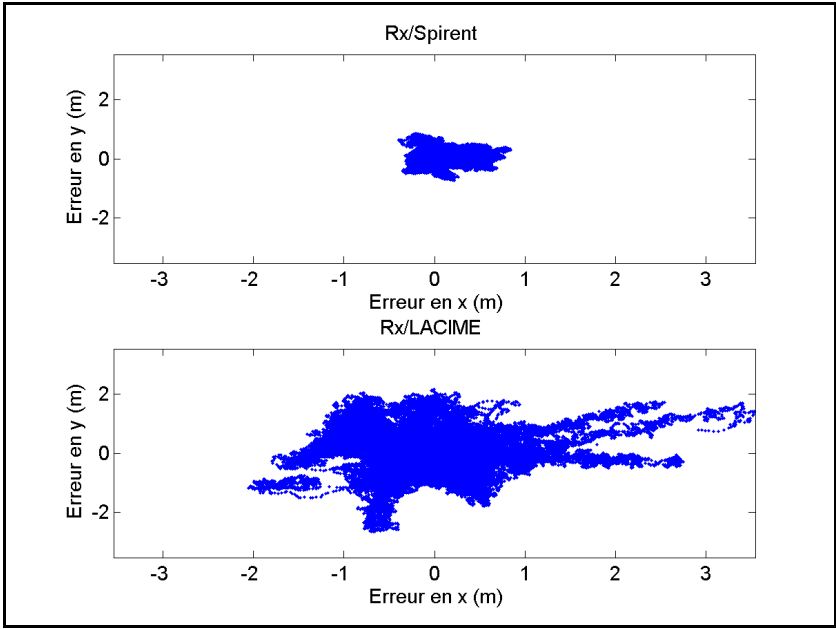


Figure-A X-8 Nuage de points pour une simulation à Marseille (Rx/Spirent versus Rx/LACIME)

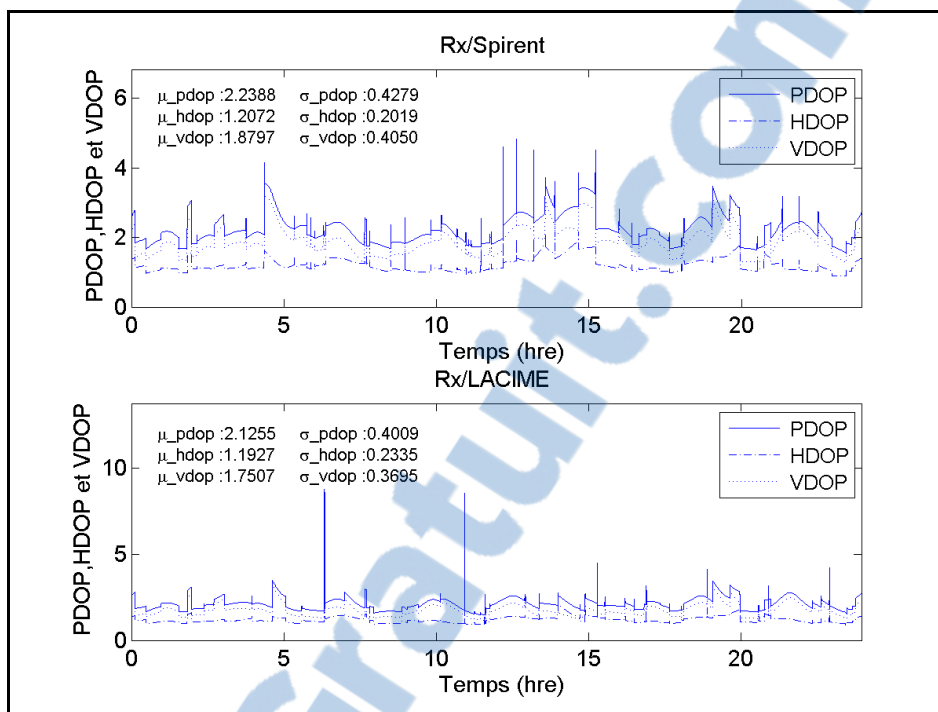


Figure-A X-9 PDOP, HDOP et VDOP pour une simulation à Marseille (Rx/Spirent versus Rx/LACIME)

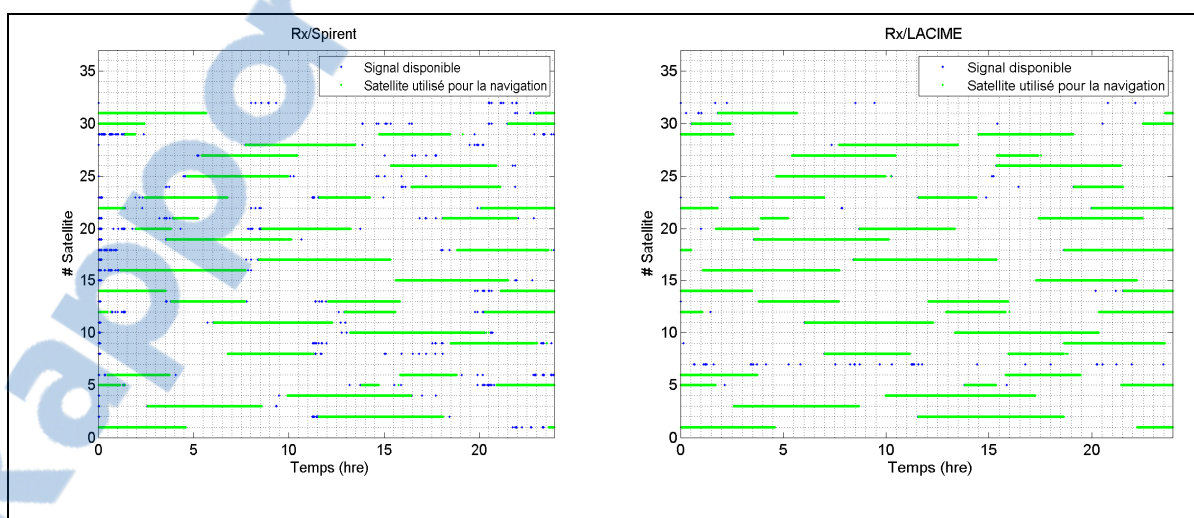


Figure-A X-10 Visibilités des satellites pour une simulation en Chine (Rx/Spirent versus Rx/LACIME)

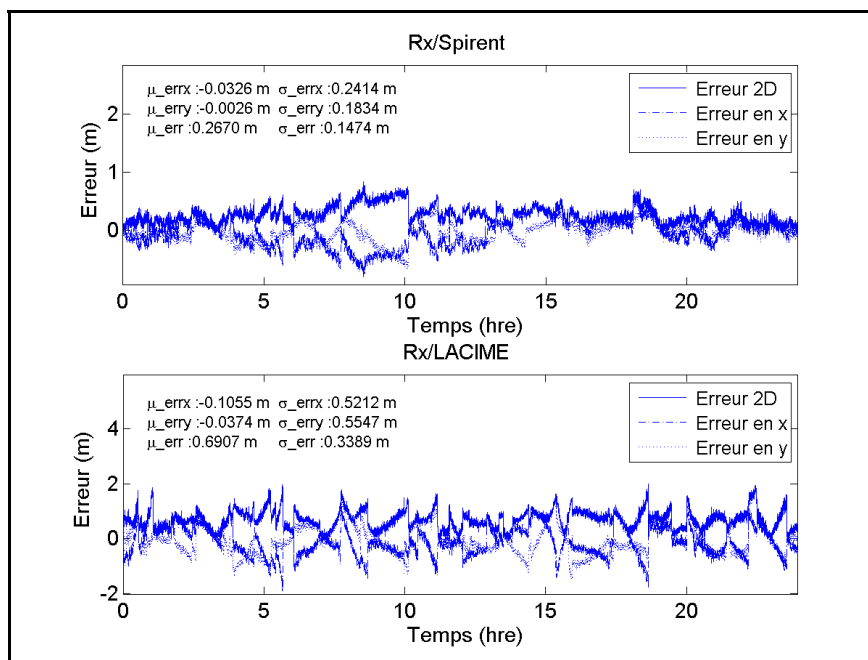


Figure-A X-11 L'erreur de position pour une simulation en Chine (Rx/Spirent versus Rx/LACIME)

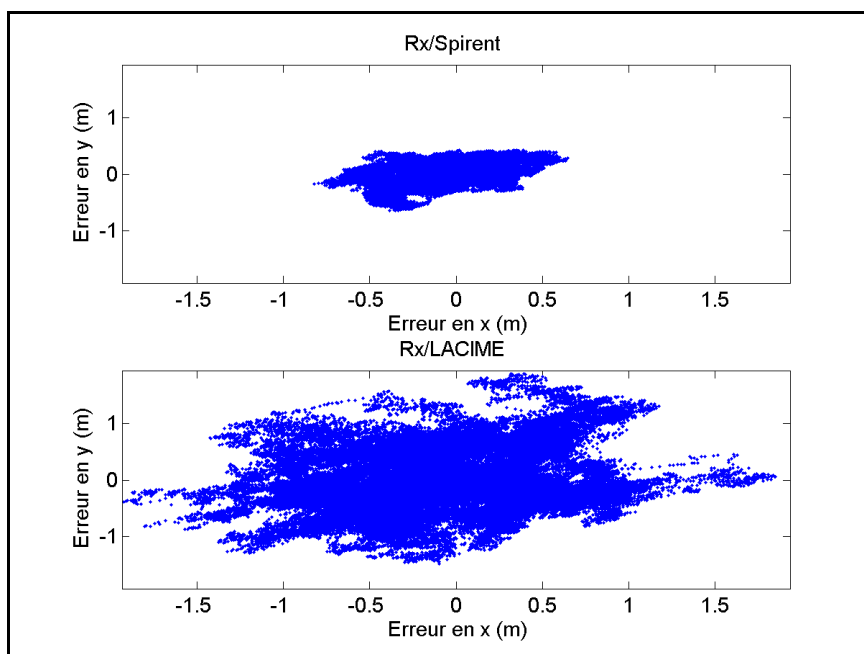


Figure-A X-12 Nuage de points pour une simulation en Chine (Rx/Spirent versus Rx/LACIME)

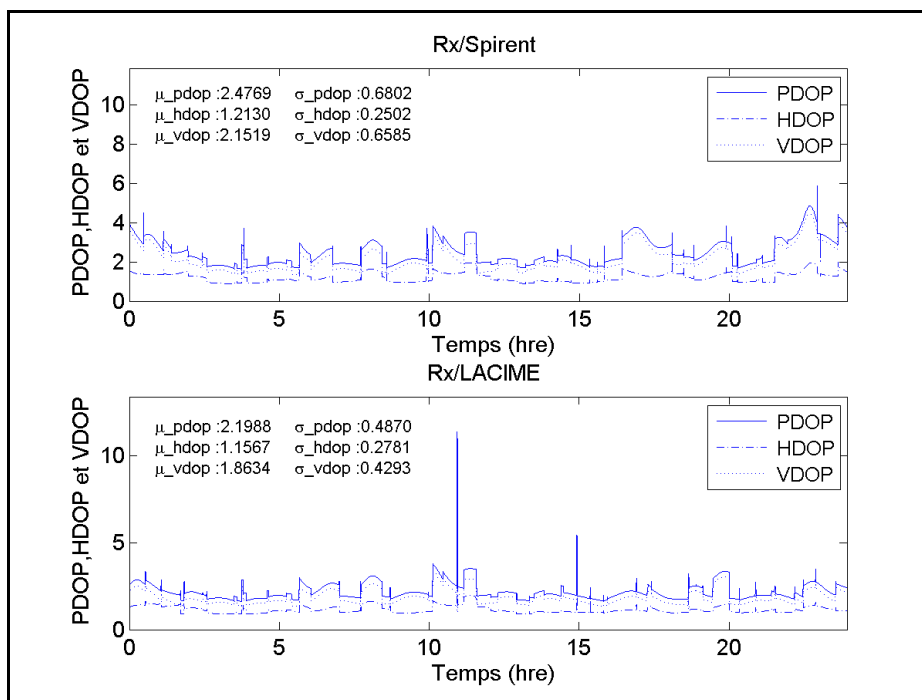


Figure-A X-13 PDOP, HDOP et VDOP pour une simulation en Chine (Rx/Spirent versus Rx/LACIME)

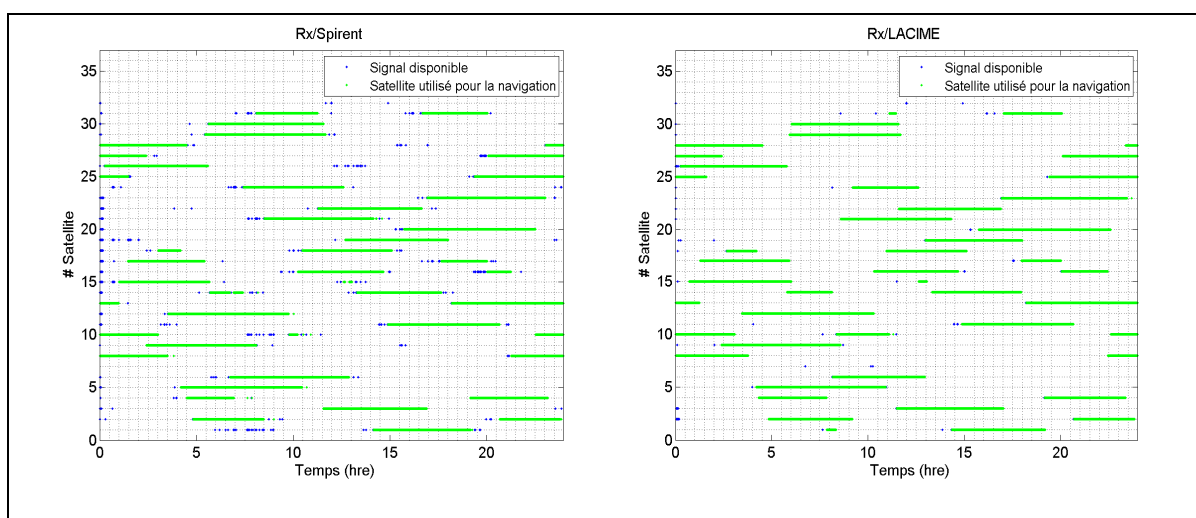


Figure-A X-14 Visibilités des satellites pour une simulation en Argentine (Rx/Spirent versus Rx/LACIME)

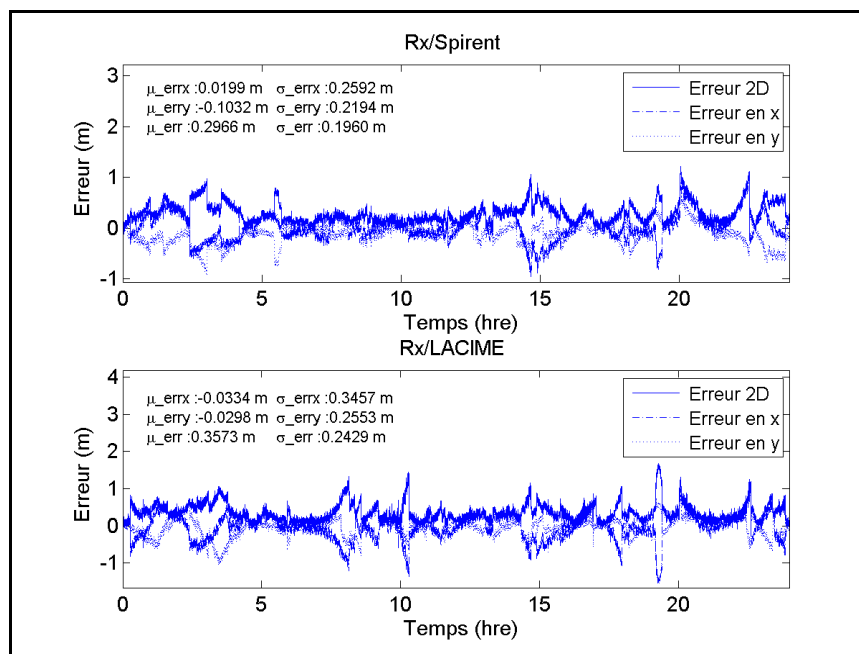


Figure-A X-15 L'erreur de position pour une simulation en Argentine (Rx/Spirent versus Rx/LACIME)

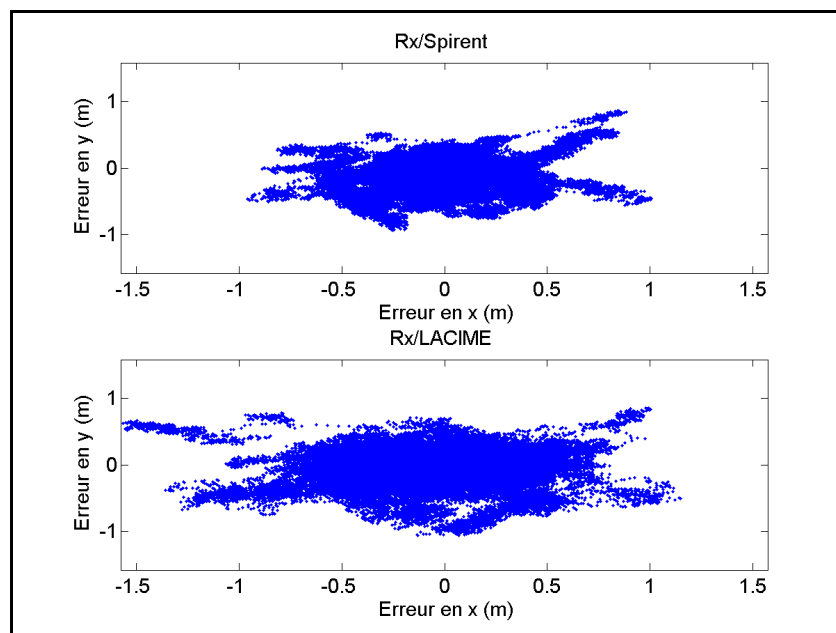


Figure-A X-16 Nuage de points pour une simulation en Argentine (Rx/Spirent versus Rx/LACIME)

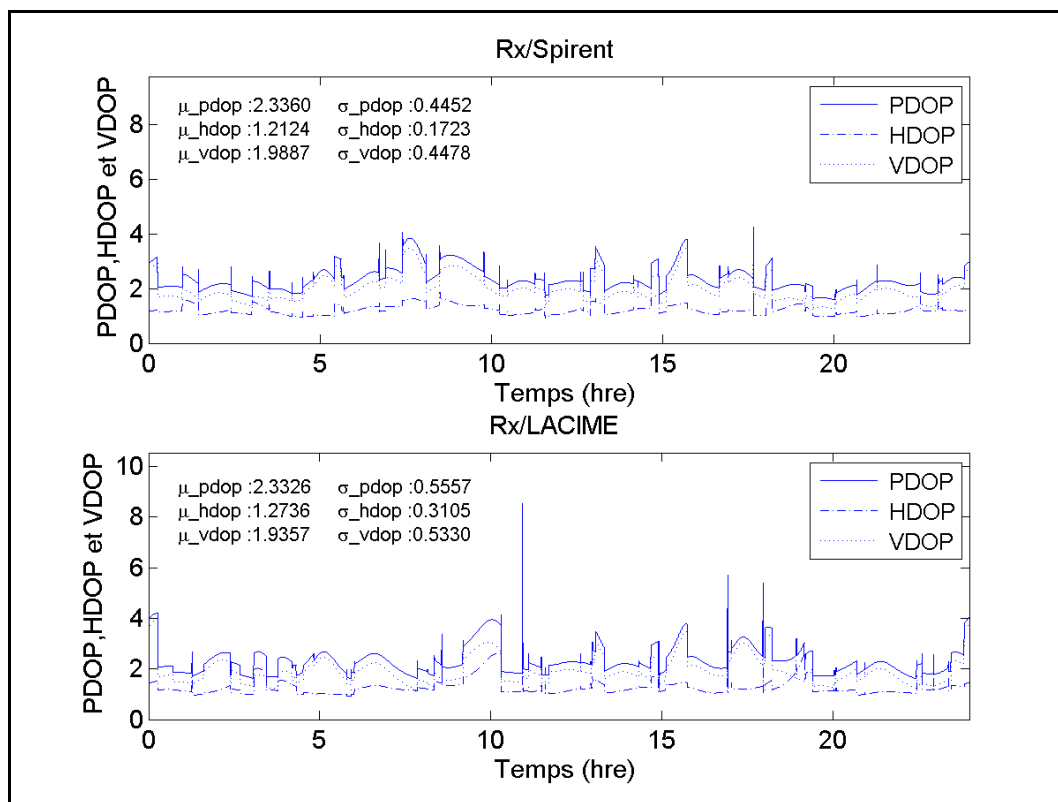


Figure-A X-17 PDOP, HDOP et VDOP pour une simulation en Argentine (Rx/Spirent versus Rx/LACIME)

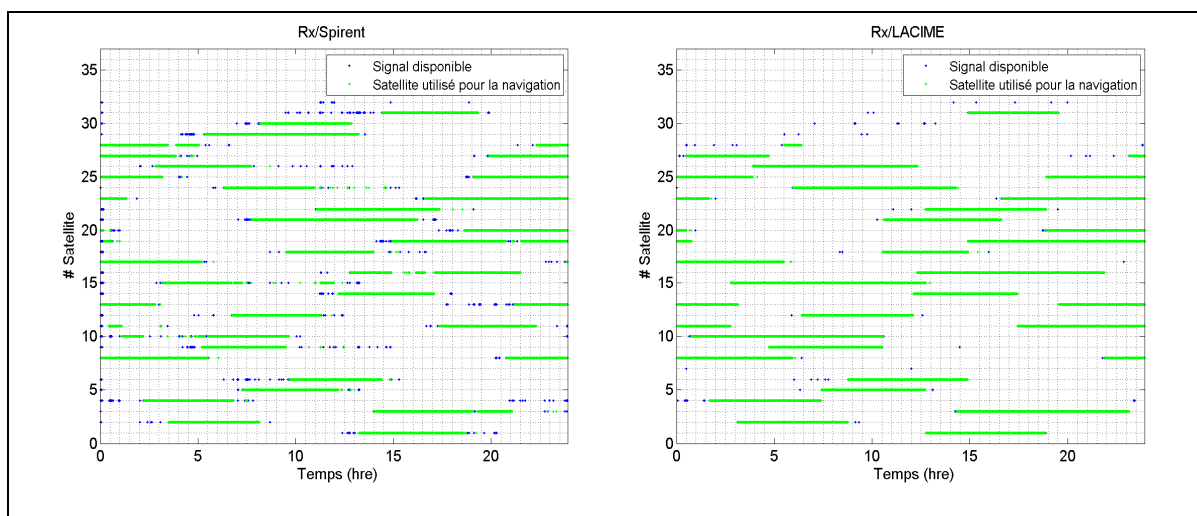


Figure-A X-18 Visibilités des satellites pour une simulation à Natal (Rx/Spirent versus Rx/LACIME)

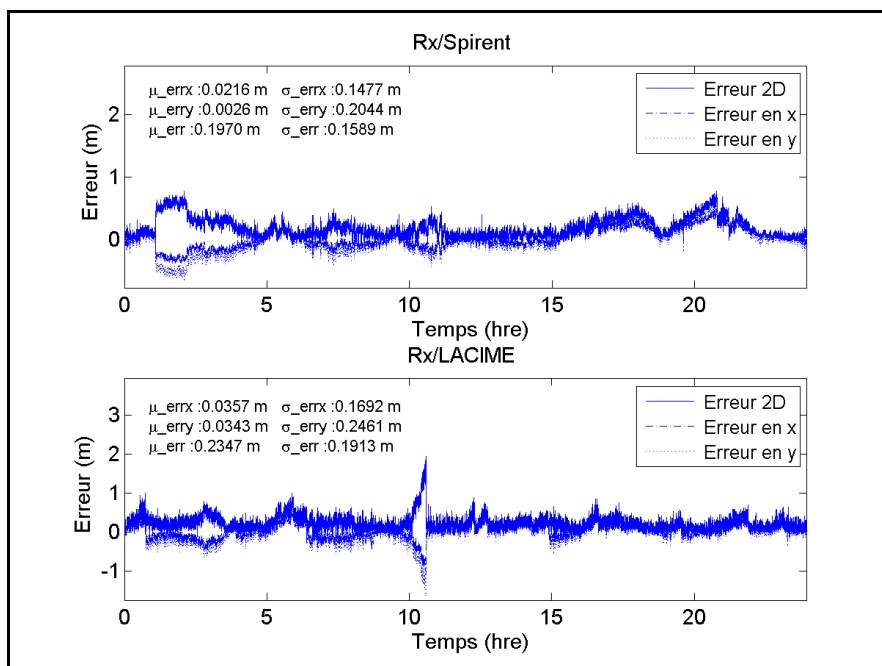


Figure-A X-19 L'erreur de position pour une simulation à Natal (Rx/Spirent versus Rx/LACIME)

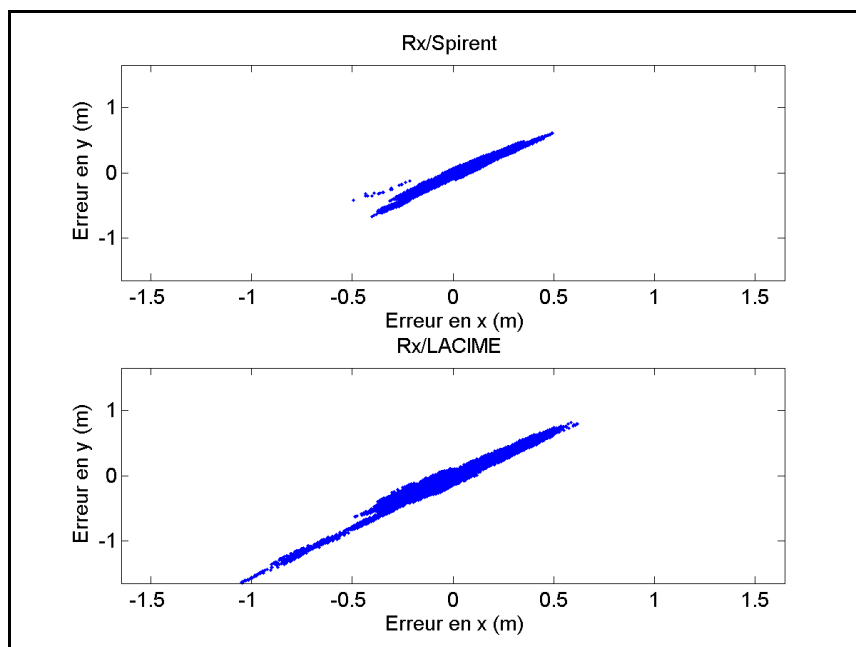


Figure-A X-20 Nuage de points pour une simulation à Natal (Rx/Spirent versus Rx/LACIME)

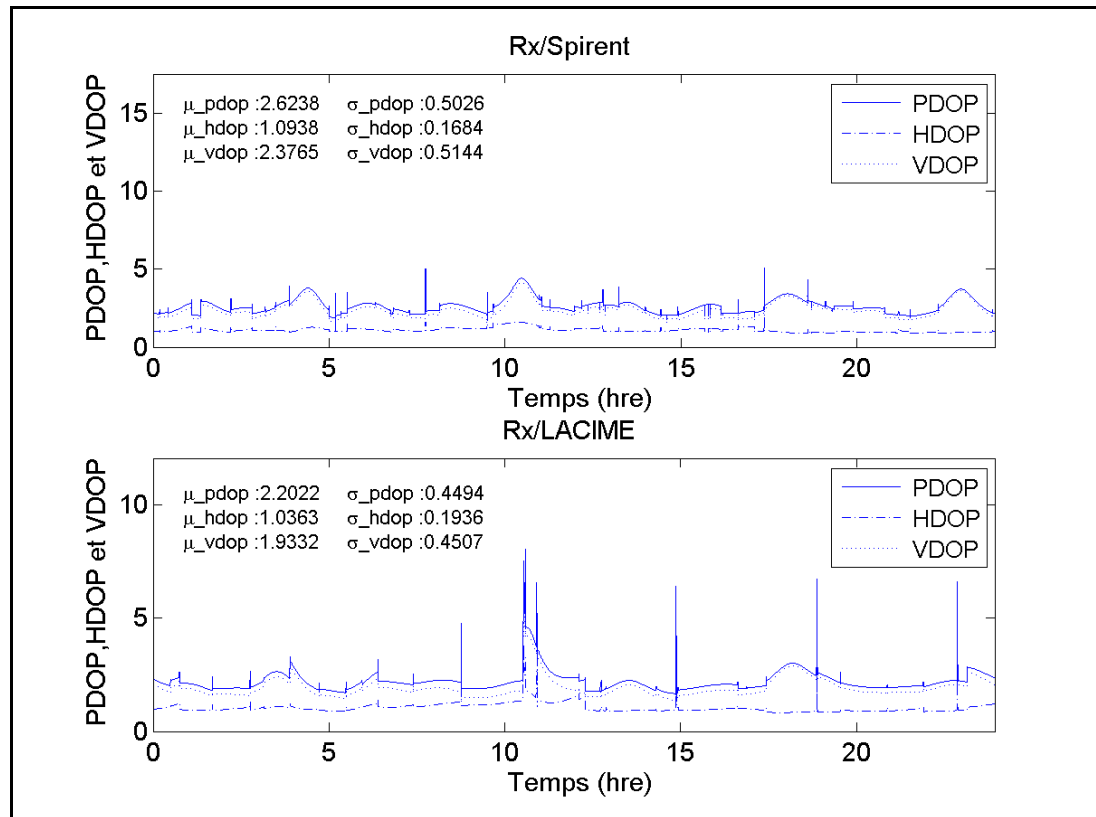


Figure-A X-21 PDOP, HDOP et VDOP pour une simulation à Natal (Rx/Spirent versus Rx/LACIME)

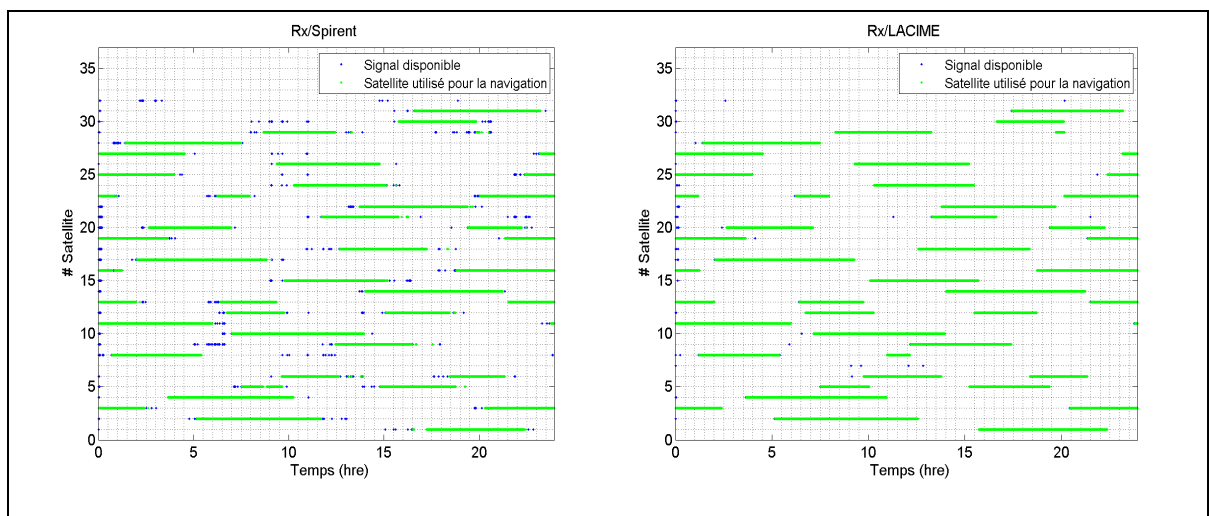


Figure-A X-22 Visibilités des satellites pour une simulation en Afrique du Sud (Rx/Spirent versus Rx/LACIME)

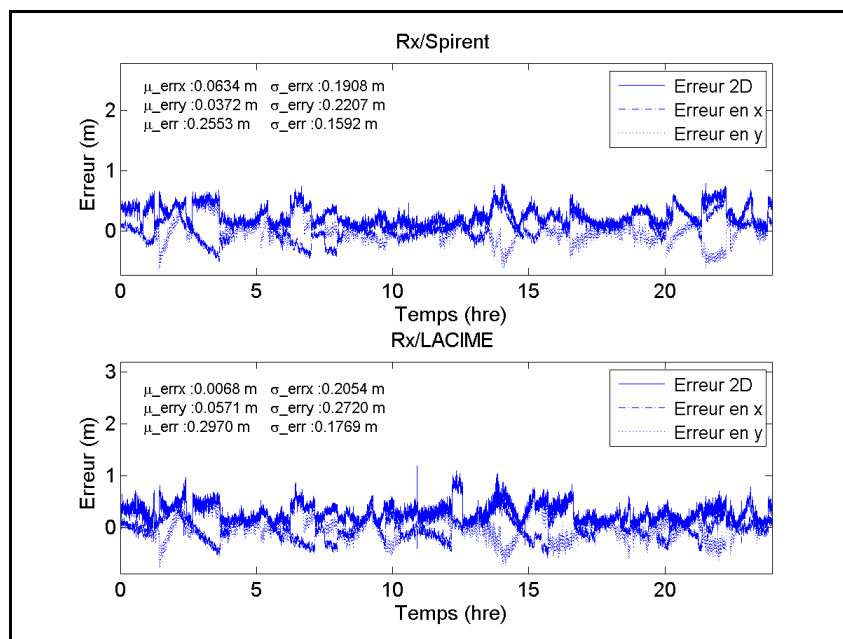


Figure-A X-23 L'erreur de position pour une simulation en Afrique du Sud (Rx/Spirent versus Rx/LACIME)

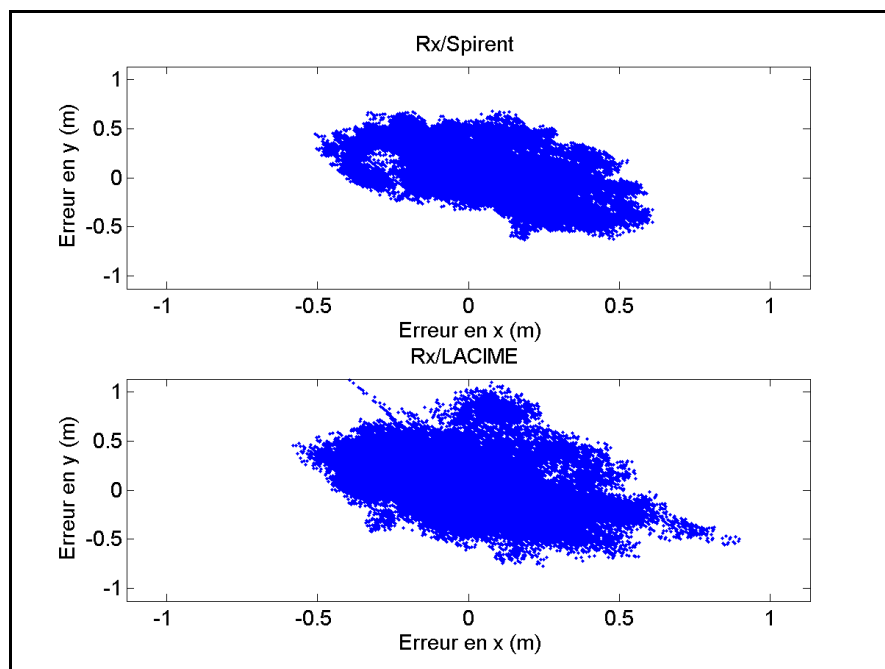


Figure-A X-24 Nuage de points pour une simulation en Afrique du Sud (Rx/Spirent versus Rx/LACIME)

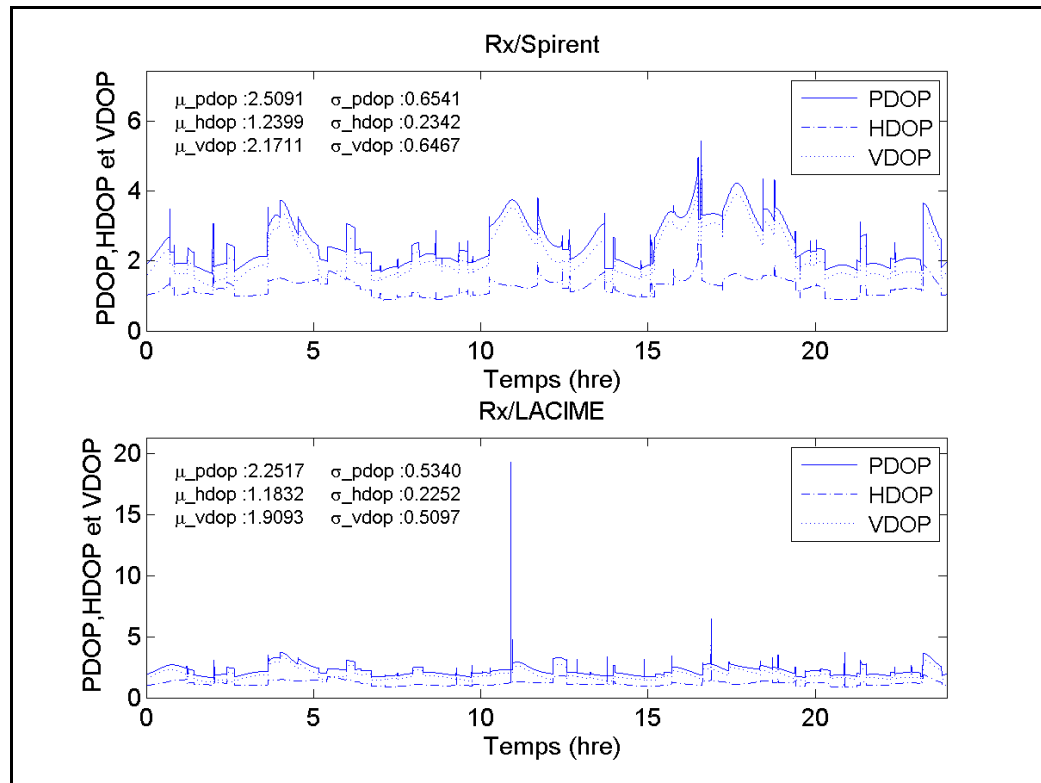


Figure-A X-25 PDOP, HDOP et VDOP pour une simulation en Afrique du Sud (Rx/Spirent versus Rx/LACIME)

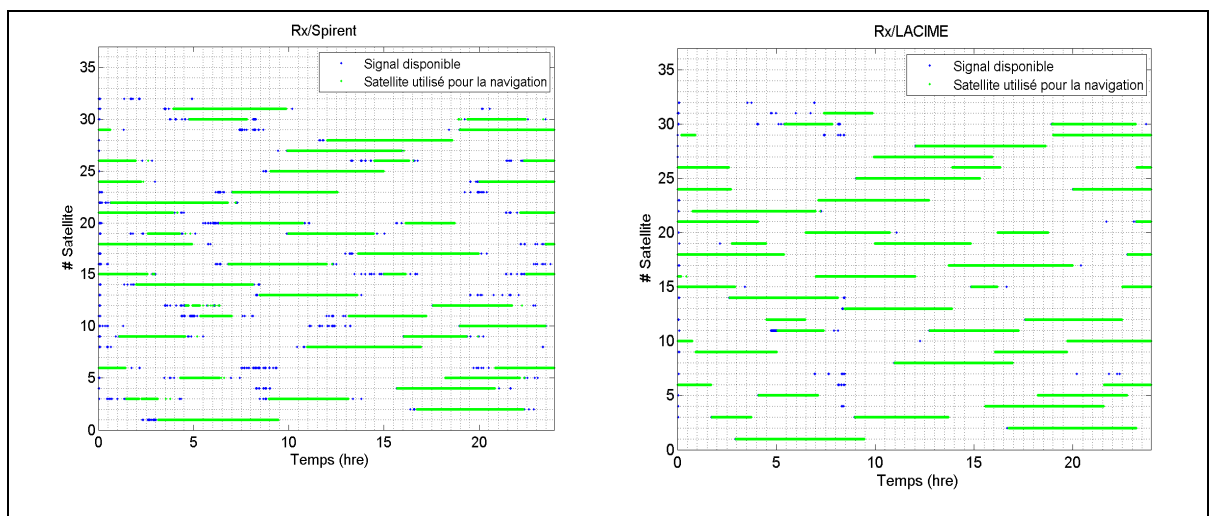


Figure-A X-26 Visibilités des satellites pour une simulation en Nouvelle-Zélande (Rx/Spirent versus Rx/LACIME)

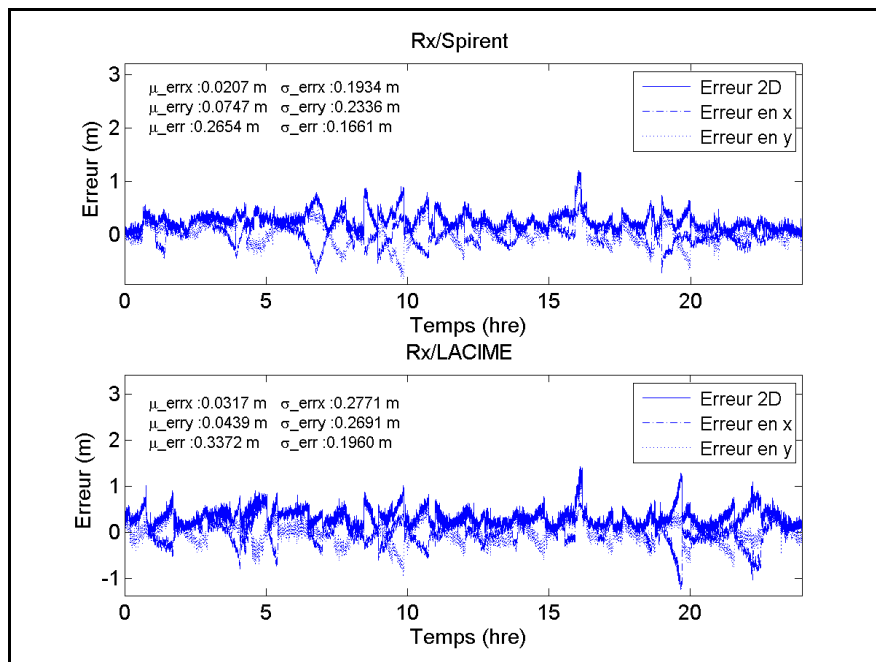


Figure-A X-27 L'erreur de position pour une simulation en Nouvelle-Zélande (Rx/Spirent versus Rx/LACIME)

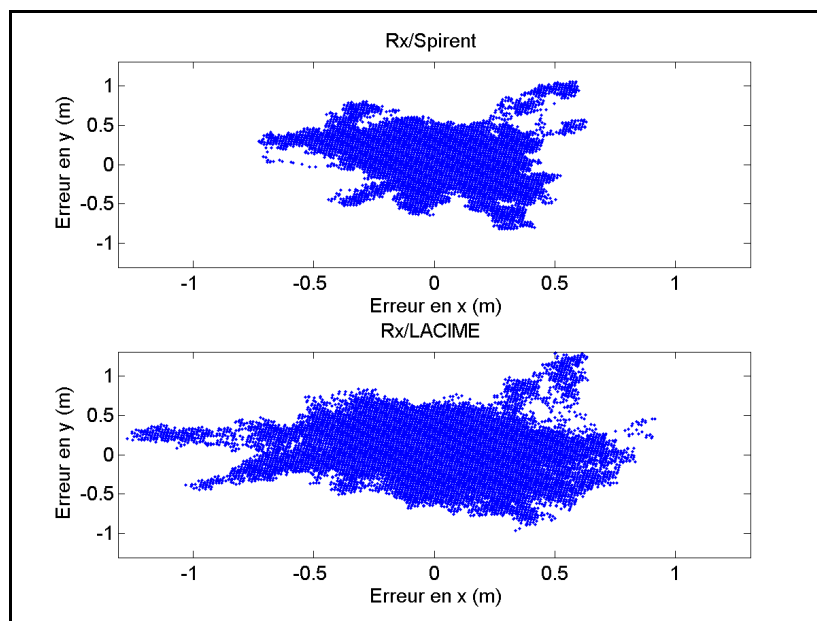


Figure-A X-28 Nuage de points pour une simulation en Nouvelle-Zélande (Rx/Spirent versus Rx/LACIME)

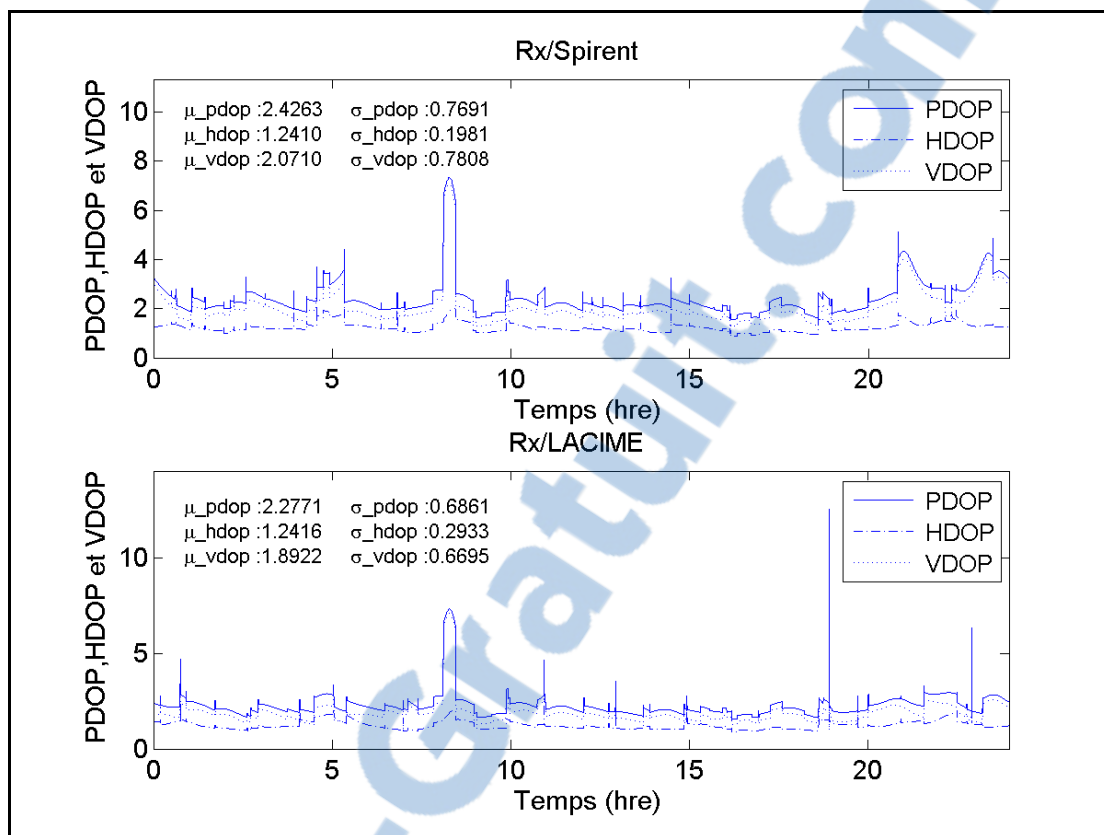


Figure-A X-29 PDOP, HDOP et VDOP pour une simulation en Nouvelle-Zélande (Rx/Spirent versus Rx/LACIME)

ANNEXE XI

RÉSULTATS FINAUX DES SIMULATIONS EN DYNAMIQUE

Voici les résultats de deux simulations en dynamiques avec une vitesse constante du véhicule de 14 m/s et de 28 m/s.

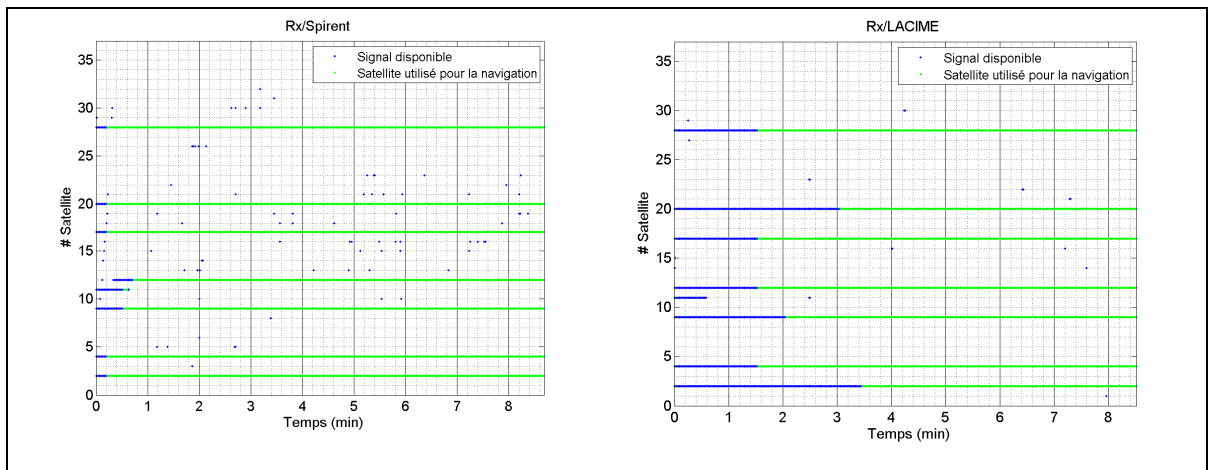


Figure-A XI-1 Visibilités des satellites pour une vitesse de 14 m/s (Rx/Spirent versus Rx/LACIME)

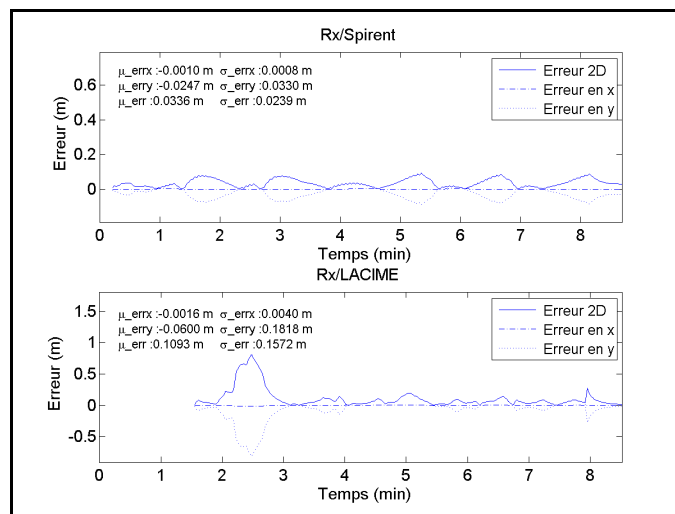


Figure-A XI-2 Erreur de la position pour une vitesse de 14 m/s (Rx/Spirent versus Rx/LACIME)

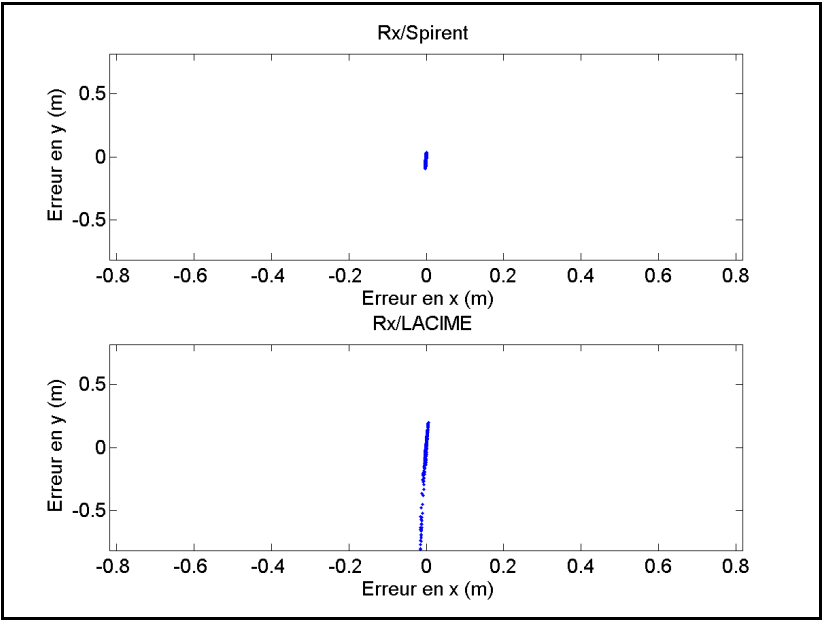


Figure-A XI-3 Nuage de points pour une vitesse de 14 m/s
(Rx/Spirent versus Rx/LACIME)

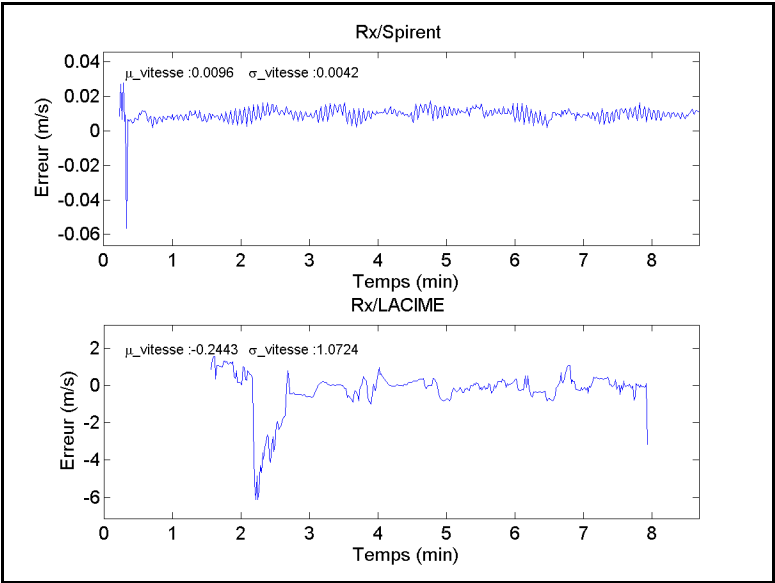


Figure-A XI-4 L'erreur de la vitesse pour une vitesse de 14 m/s
(Rx/Spirent versus Rx/LACIME)

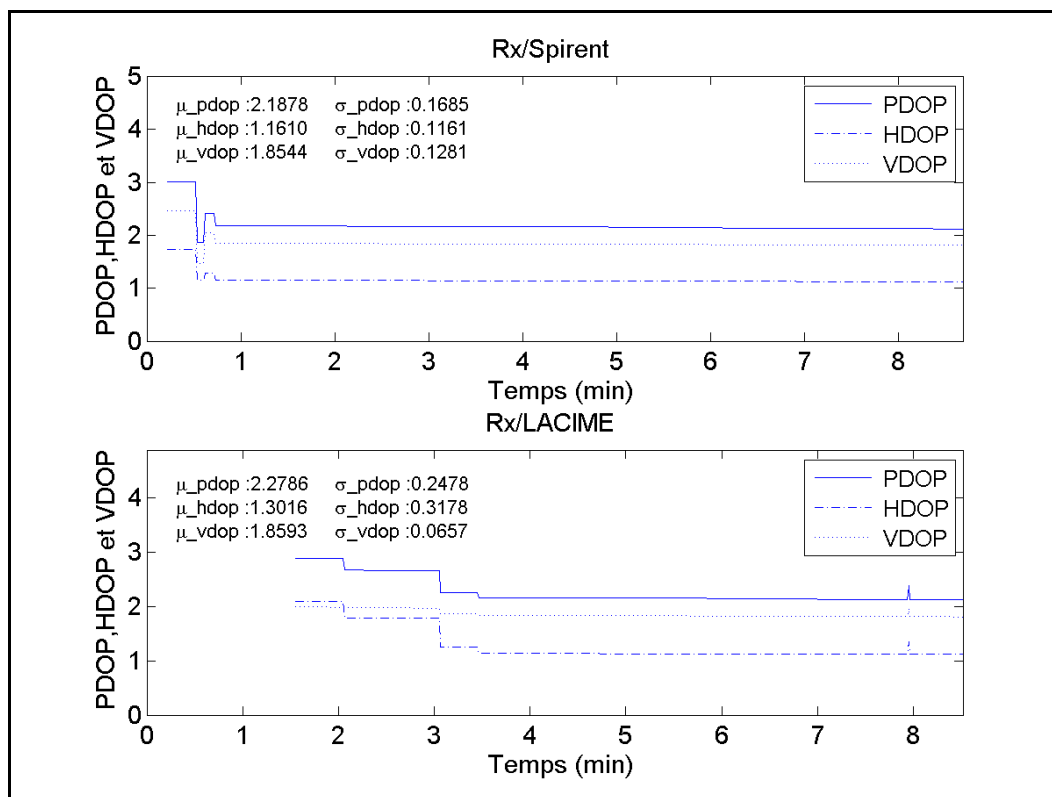


Figure-A XI-5 PDOP, HDOP et VDOP pour une vitesse de 14 m/s
(Rx/Spirent versus Rx/LACIME)

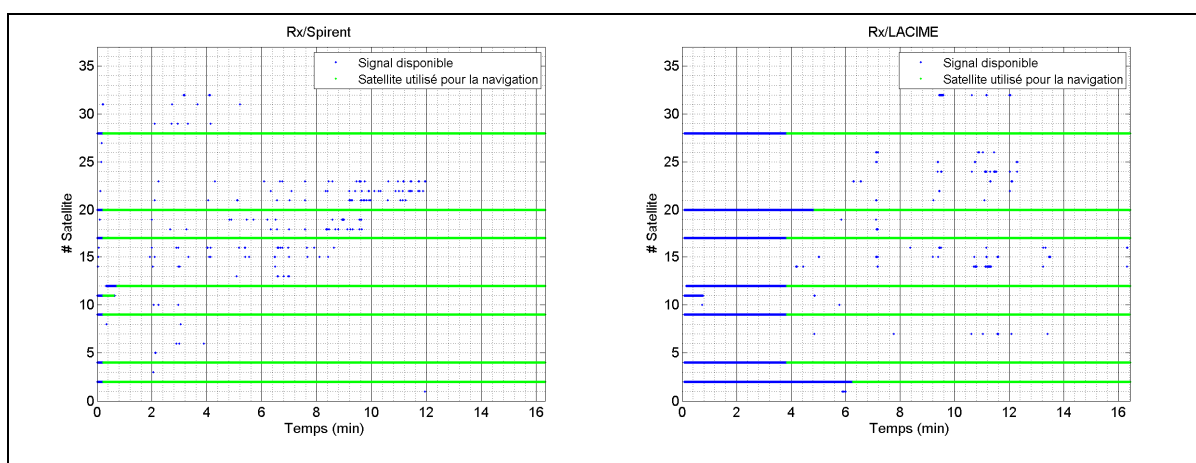


Figure-A XI-6 Visibilités des satellites pour une vitesse de 28 m/s
(Rx/Spirent versus Rx/LACIME)

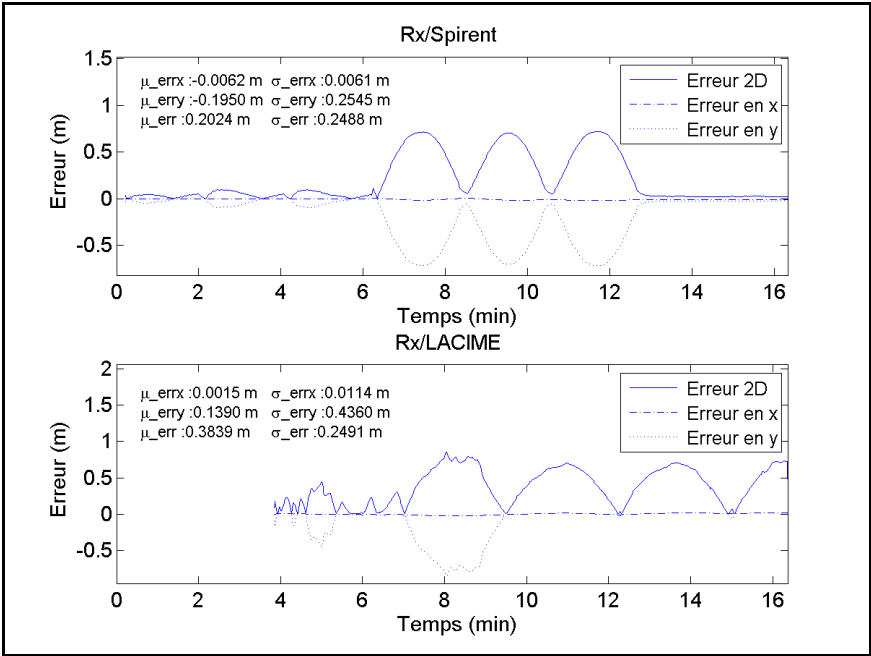


Figure-A XI-7 L'erreur de la position pour une vitesse de 28 m/s (Rx/Spirent versus Rx/LACIME)

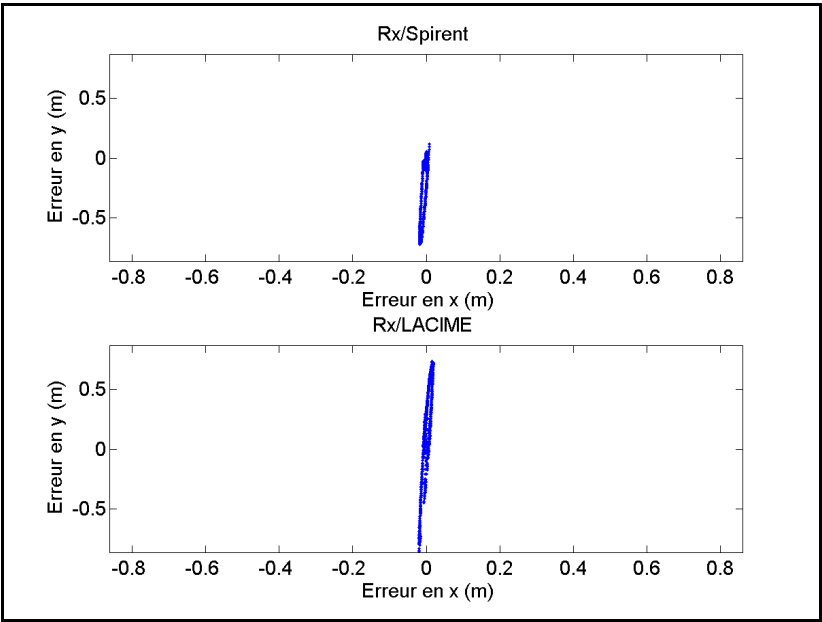


Figure-A XI-8 Nuage de points pour une vitesse de 28 m/s (Rx/Spirent versus Rx/LACIME)

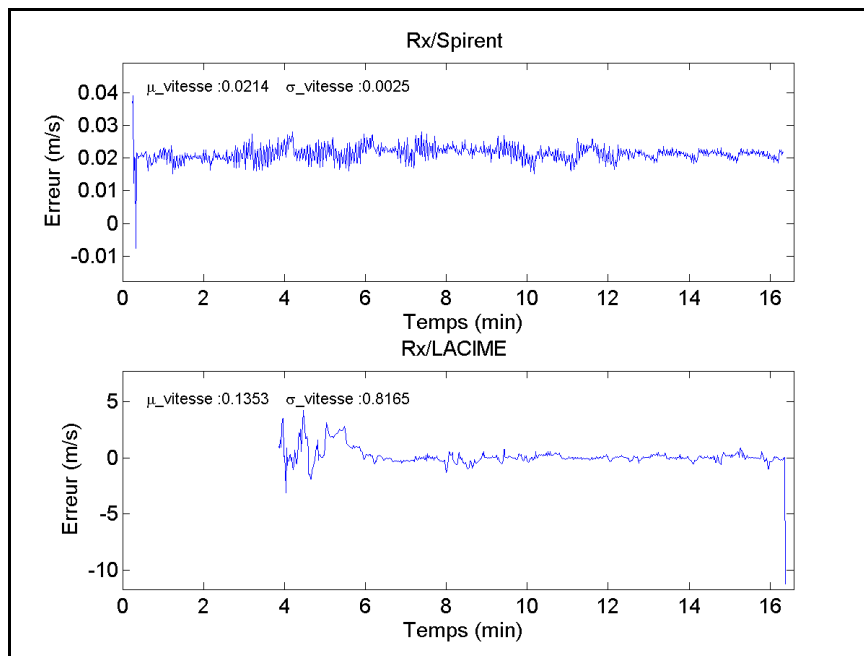


Figure-A XI-9 L'erreur de la vitesse pour une vitesse de 28 m/s (Rx/Spirent versus Rx/LACIME)

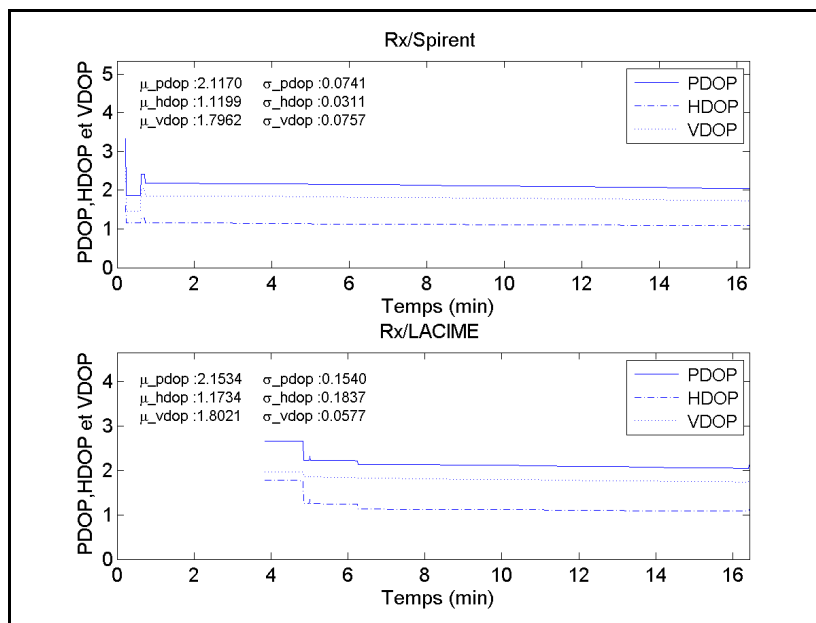


Figure-A XI-10 PDOP, HDOP et VDOP avec une vitesse de 28 m/s (Rx/Spirent versus Rx/LACIME)

BIBLIOGRAPHIE

- Adane, Yassine. 2007a. *Master Clock generation and distribution*. Montréal: LACIME, 10 p.
- Adane, Yassine. 2007b. *The RF Module of the GNSS simulator*. Montréal: LACIME, 6 p.
- Aeroflex. 2010. « GPSG-1000 - Aeroflex ». < http://www.aeroflex.com/ats/products/product/Avionics/GPS_Simulators/GPSG-1000~713.html >. Consulté le 15 novembre.
- Aeroflex. 2011. *GPSG-1000 GPS/Galileo Positionnal Simulator Operation Manual*. 158 p.
- Agilent Technologies. 2012a. « N7609B Signal Studio for Global Navigation Satellite Systems ». In *Agilent N7609B Online Documentation*. En ligne. < <http://wireless.agilent.com/wireless/helpfiles/n7609b/n7609b.htm> >. Consulté le 17 septembre 2012.
- Agilent Technologies. 2012b. « Signal Studio for Global Navigation Satellite Systems (GNSS) N7609B ». En ligne. 10 p. < <http://cp.literature.agilent.com/litweb/pdf/5990-9363EN.pdf> >. Consulté le 17 septembre 2012.
- Andrillon, Tanguy. 2011. « L'iphone 4S compatible avec le GPS russe ». En ligne. < <http://www.macworld.fr/2011/10/20/iphone/l-iphone-compatible-avec-gps-russe/520487/> >. Consulté le 5 février 2012.
- ARINC Engineering Services. 2004. « Navstar GPS Space Segment/Navigation User Interfaces ». < www.navcen.uscg.gov/pdf/IS-GPS-200D.pdf >.
- ARINC Engineering Services. 2010. « Navstar GPS Control Segment to User Support Community Interfaces ». < http://www.navcen.uscg.gov/pdf/gps/geninfo/ICD_GPS_240A_Final_12Jan10.pdf >.
- Belzile, Jean. 2006. *Notes on IF signal processing*. École de Technologie Supérieure, 34 p.
- Ben Salah, Ahmed. 2010. « Conception et mise en oeuvre d'une architecture logicielle pour l'amélioration des performances d'un simulateur GNSS et l'intégration de modèles multitrajets ». MR63836. Canada, École de Technologie Supérieure (Canada), 158 p.
- Canton Fair. 2012. « China to launch 3 satellites for Beidou ». En ligne. < <http://www.canfair.com/china-to-launch-3-satellites-for-beidou/> >. Consulté le 19 mai 2012.
- Cast Navigation. 2010. « Products - GPS Simulators, GPS/INS Simulation, EGI integration tools ». < http://castnav.com/products/cast_2000.html >. Consulté le 15 novembre.

- Chee, Yi-Min, Katrin Franke, Max Froumentin et Larry Yaeger. 2006. « Ink Markup Language (InkML) ». En ligne. < <http://www.w3.org/TR/2006/WD-InkML-20061023/diffmarked.html> >. Consulté le 10 juin 2012.
- Chicheportiche, Olivier. 2010. « GPS : encore du retard pour Galileo, les Français écartés du projet ». En ligne. < <http://www.zdnet.fr/actualites/gps-encore-du-retard-pour-galileo-les-francais-ecartes-du-projet-39712156.htm> >. Consulté le 19 mai 2012.
- European Commission. 2011. « Satellite.c File Reference ». In EGNOS Toolkit. En ligne. < <http://nacc.upc.es/egnos/doc/a00058.html> >. Consulté le 28 décembre 2012.
- European Space Agency. 2012. « Next Galileo satellites to launch after the summer ». En ligne. < http://www.esa.int/esaNA/SEMWOQWJ1H_index_0.html >. Consulté le 19 mai 2012.
- Federal Space Agency. 2012a. « GLONASS constellation status ». En ligne. < <http://www.glonass-center.ru/en/GLONASS/> >. Consulté le 19 mai 2012.
- Federal Space Agency. 2012b. « GPS constellation status ». En ligne. < <http://www.glonass-center.ru/en/GPS/> >. Consulté le 19 mai 2012.
- GPSoft. 2003. *Satellite Navigation TOOLBOX 3.0*. Athens: GPSoft LLC, 146 p.
- IFEN. 2010. « IFEN - NavX-NCS RF Constellation Generator ». < <http://www.ifen.com/navx-ncs-overview> >. Consulté le 15 novembre 2010.
- IFEN. 2012. *NavX-NCS Navigation Constellation Simulator User Manual*. 236 p.
- Kaplan, Elliott D. 1996. *Understanding GPS : principles and applications*. Boston: Artech House, 554 p.
- Kaplan, Elliott D., et Christopher J. Hegarty. 2006. *Understanding GPS : principles and applications*, 2nd. Coll. « Artech House mobile communications series ». Boston: Artech House, 707 p.
- Korvenoja, Paula, et Robert Piché. 2000. « Efficient Satellite Orbit approximation ». *ION GPS 2000*. p. 1930-1937. < http://math.tut.fi/posgroup/korvenoja_piche_ion2000a.pdf >.
- Labsat. 2010. « LabSat GNSS Simulator - Generates, records and replays GPS & GLONASS RF data ». < <http://www.labsat.co.uk/> >. Consulté le 15 novembre.
- Lacasse, D. 2008. « Émulation temps réel de signaux de navigation satellite ». MR47244. Canada, École de Technologie Supérieure (Canada), 134 p.

- Le nouvel Observateur. 2010. « Le GPS Galileo commencera à être opérationnel en 2014 ». En ligne. < <http://tempsreel.nouvelobs.com/monde/20100107.OBS2965/le-gps-galileo-commencera-a-etre-operationnel-en-2014.html> >. Consulté le 5 février 2012.
- Le Quotidien du Peuple en ligne. 2011a. « La Chine a prévu de lancer huit satellites de navigation Beidou d'ici 2012 ». En ligne. < <http://french.peopledaily.com.cn/Sci-Edu/7385286.html> >. Consulté le 19 mai 2012.
- Le Quotidien du Peuple en ligne. 2011b. « La Chine cartographie le monde ». En ligne. < <http://french.peopledaily.com.cn/Sci-Edu/7342124.html> >. Consulté le 19 mai 2012.
- Le Quotidien du Peuple en ligne. 2012. « La Chine lance le 11e satellite de son système de navigation Beidou ». En ligne. < <http://french.peopledaily.com.cn/Sci-Edu/7740145.html> >. Consulté le 19 mai 2012.
- Leblal, Serge. 2011. « Beidou, le système GPS chinois désormais opérationnel ». En ligne. < <http://www.lemondeinformatique.fr/actualites/lire-beidou-le-systeme-gps-chinois-desormais-operationnel-47190.html> >. Consulté le 5 février 2012.
- Loukil, Rami. 2010. « Conception et mise en oeuvre de module de contrôle de puissance et de génération de bruit pour conditionner les signaux RF d'un simulateur de constellation GPS et Galiléo ». M.Ing. Canada, École de Technologie Supérieure (Canada).
- Parkinson, Bradford W., et James J. Spilker. 1996. *Global positioning system : theory and applications*. Coll. « Progress in astronautics and aeronautics », 163. Washington, D.C.: American Institute of Aeronautics and Astronautics.
- Petovello, Mark, et Cillian O'Driscoll. 2010. « Carrier phase and its measurement for GNSS ». *InsideGNSS*. Juillet/Août 2010. < <http://www.insidegnss.com/node/2146> >. Consulté le 19 octobre 2011.
- RIA Novosti. 2011. « GLONASS, le "GPS russe", couvre l'ensemble du globe ». En ligne. < <http://fr.rian.ru/science/20111208/192341634.html> >. Consulté le 5 février 2012.
- Rizos, Chris. 1999. « GPS Hardware for Surveying ». In *GPS Instrumentation*. En ligne. < http://www.gmat.unsw.edu.au/snap/gps/gps_survey/chap4/423.htm#codeless >. Consulté le 14 octobre 2012.
- Rohde & Schwarz. 2012. « GNSS Simulator for the R&S SMBV100A Vector Signal Generator ». En ligne. < <http://www.rohde-schwarz.ca/product/GNSS.html> >. Consulté le 17 septembre 2012.
- Sauriol, Bruno. 2007. *XMEM Bus*. Montréal: LACIME, 9 p.

- Silicom. 2012. « RTG2 Real Time GNSS Generator ». En ligne. 4 p.
< <http://www.silicom.fr/silicom/en/images/stories/rtg2.pdf> >. Consulté le 17 septembre 2012.
- Spectracom. 2010. « GSG-62 L1+L2 Dual Frequency 32-channel Multi-GNSS Simulator ». In *GLONASS & GPS, L1 & L2 Simulator - 32 channel - Spectracom GSG-62*. En ligne.< <http://www.spectracomcorp.com/ProductsServices/TestandMeasurement/GPSSimulators/GSG62L1L232channelGNSSSimulator/tabid/1525/Default.aspx> >. Consulté le 17 septembre 2012.
- Spectracom. 2012. *GSG-5 / 6 Series GPS and GLONASS Simulator User Manual*. 93 p.
Consulté le 17 septembre 2012.
- Swarthmore College Computer Society. 2011. « Korean Air Lines Flight 007 ». En ligne. < http://www.sccs.swarthmore.edu/users/08/ajb/tmve/wiki100k/docs/Korean_Air_Lines_Flight_007.html >. Consulté le 19 mai 2012.
- Tsui, James Bao-yen. 2005. *Fundamentals of global positioning system receivers : a software approach*, 2nd. Coll. « Wiley series in microwave and optical engineering ». New Jersey: John Wiley & Sons, Inc., 352 p.