

ÉCOLE NATIONALE SUPÉRIEURE DE L'AÉRONAUTIQUE ET DE L'ESPACE

UNIVERSITÉ LIBRE DE BRUXELLES - FACULTÉ DES SCIENCES APPLIQUÉES



Professeur responsable : B. Escudier



Professeur responsable : B. Leduc

Manœuvres en attitude d'un satellite avec évitements de zones interdites

STAGE DE FIN D'ETUDES
RÉALISÉ AU CNES



Travail réalisé par : Christian Delacroix
Nicolas Descouvemont

Responsable de stage : N. Théret

2005-2006

REMERCIEMENTS

En préambule à ce travail, nous tenons à remercier le CNES et le Centre Spatial de Toulouse de nous avoir accueillis pendant cinq mois pour effectuer notre stage.

Plus particulièrement, nous remercions notre maître de stage, Monsieur Nicolas THÉRET, pour son aide et ses idées qui nous ont été précieuses pour la réalisation de notre projet, mais aussi pour la gentillesse de son accueil et son soutien tout au long des cinq mois.

En outre, nous remercions Monsieur Gérard LASSALE-BALIER, chef du service où le stage a été effectué, pour ses nombreux conseils et sa disponibilité.

Nos remerciements également à l'adresse de Madame Marie-Claire RAYMONDIS, secrétaire du service, pour son aide concernant les questions administratives du stage.

Enfin, nous n'oublions pas de remercier notre université d'origine, l'Université Libre de Bruxelles, pour nous avoir offert la possibilité d'étudier à Toulouse, ainsi que SupAero, et plus particulièrement Madame Bénédicte ESCUDIER, pour nous avoir permis de réaliser ce stage de fin d'études.

Table des matières

I	Introduction	1
II	Manœuvre en attitude : objectif et méthodologie du projet	3
1	Objectif	4
1.1	Pourquoi ce programme?	4
1.2	Définition théorique du programme	5
2	Hypothèses émises	6
3	Outils informatiques	8
3.1	Scilab	8
3.2	MANIAC	9
4	Concepts utilisés	10
4.1	Remarques préliminaires sur la représentation graphique	10
4.2	Zones interdites	12
4.2.1	Définition	12
4.2.2	Illustration des zones interdites	12
4.2.3	Coefficient de sécurité	14
4.3	Evitement	15
4.3.1	Méthode des tangentes	15
4.3.2	Plusieurs points intermédiaires (appels récursifs)	16
4.3.3	Plusieurs zones à éviter (appels récursifs)	17
III	Description technique du programme et résultats obtenus	18
5	Initialisation du problème	21
5.1	Introduction des données	21
5.1.1	Dates de début et de fin de manœuvre	21
5.1.2	Paramètres orbitaux du satellite	21
5.1.3	Orientation des instruments	22
5.1.4	Données pour MANIAC	22
5.2	Position des astres - zones interdites	23
5.2.1	Terre-satellite	23
5.2.2	Lune-satellite – Soleil-satellite	23
5.2.3	Rayon des zones interdites	24

6	Manœuvre simple (sans évitement)	25
6.1	Interfaçage de MANIAC	25
6.1.1	Deux fichiers d'interface : <i>intInit.c</i> et <i>intDfx.c</i>	25
6.1.2	Fonctionnement de l'interface	26
6.1.3	Utilisation de l'interface	27
6.2	Appel à MANIAC	28
6.2.1	Conditions aux limites	28
6.2.2	Manœuvre en durée minimum	29
6.3	Traitement des données	31
6.3.1	Vitesses et accélérations à chaque pas de temps	31
6.3.2	Quaternions à chaque pas de temps	31
7	Manœuvre avec évitement	33
7.1	Détection des zones interdites	33
7.1.1	Direction de visée des instruments	33
7.1.2	Entrée dans une zone interdite	34
7.2	Évitement d'une zone interdite	35
7.2.1	Calcul des coordonnées du point C	35
7.2.2	Calcul des coordonnées des points de tangence (A_t et B_t)	36
7.2.3	Choix et association des points de tangence (A_t et B_t)	40
7.2.4	Caractéristiques du point C	41
7.2.5	Cas particulier	42
7.3	Quaternions d'attitude aux points intermédiaires	45
7.3.1	Rotation selon une géodésique	45
7.3.2	Problème de l'orientation du satellite	46
7.3.3	Calcul du quaternion du point intermédiaire (évitement simple)	46
7.3.4	Calcul du quaternion de plusieurs points intermédiaires	51
7.4	Évitement de plusieurs zones	52
7.5	Choix de la trajectoire la plus rapide	55
8	Résultats obtenus	57
8.1	Évitement simple	57
8.2	Cas généraux	60
8.3	Cas particuliers	61
8.4	Validation du processus récursif de l'arbre	62
8.5	Prise en compte de la durée des manœuvres obtenues	64
IV	Conclusion	65
V	Annexes	68
9	Présentation de la société	69
10	Éléments de mécanique spatiale	71
10.1	Notions de mécanique spatiale képlérienne	71
10.1.1	Les lois de Kepler	71
10.1.2	Les paramètres orbitaux	73
10.1.3	Quelques formules de mouvement képlérien	77

10.2	Position du satellite - construction de l'orbite	78
10.2.1	Utilisation des anomalies	78
10.2.2	Orbite de départ dans le plan $Z = 0$	78
10.2.3	Orbite réelle	79
10.3	Les repères	80
10.4	Attitude d'un satellite	82
10.5	Le Système de Contrôle d'Attitude (SCA)	83
11	Aspects mathématiques	84
11.1	Triangles sphériques	84
11.1.1	Définition	84
11.1.2	Relations trigonométriques	85
11.1.3	Résolution des triangles sphériques	86
11.1.4	Remarque	86
11.2	Quaternions	87
11.2.1	Définition	87
11.2.2	Algèbre des quaternions	87
11.2.3	Rotation avec un quaternion	88
11.2.4	Quaternions – Angle d'Euler	90
12	Documentation des fonctions	91
12.1	Fichiers exécutables	93
12.2	Fonctions de dessin	94
12.3	Fonctions de calcul	96
12.3.1	Triangles sphériques - Géométrie	96
12.3.2	Quaternions	98
12.3.3	Rotations - Repères	99
12.3.4	Mécanique spatiale	100
12.3.5	Calcul de la trajectoire	101
12.3.6	Divers	105

Première partie

Introduction

Tout au long de leur vie, les satellites ont constamment besoin de modifier leur orientation dans l'espace. Jusqu'à présent, la faisabilité d'une manœuvre en attitude n'est vérifiée qu'après le calcul de celle-ci. Si elle s'avère impossible à réaliser, il faut la redéfinir manuellement. Le but de ce projet est de créer un programme permettant de résoudre les problèmes pouvant survenir lors d'une manœuvre, et d'apporter une solution pour le satellite sans intervention extérieure.

Le Projet de Fin d'Etudes constitue l'aboutissement d'une mission d'application de cinq mois (1^{er} avril au 1^{er} septembre 2006) dans le domaine de la mécanique spatiale, mené au Centre Spatial de Toulouse, appartenant au CNES¹, dans le service Mission et Programmation de la Sous-Direction Système Bord/Sol, situé au Centre Spatial de Toulouse (DCT/SB/MP)².

Les auteurs sont tous deux étudiants à SupAero³ dans le cadre d'un accord de double diplôme (TIME⁴) avec leur établissement d'origine : la Faculté de Polytechnique de l'ULB⁵. Ce projet représente ainsi notre travail de fin d'études dans chacun de ces établissements. C'est pourquoi, il est probable que certaines parties de ce rapport soient plus familières à l'une ou l'autre de ces institutions. Ceci explique pourquoi nous avons décidé d'être les plus exhaustifs possible dans ce travail.

Le rapport s'arrête dès lors, dans un premier temps (partie II), sur les étapes préalables à la conception d'un programme informatique : l'explication de l'objectif visé et de la méthodologie suivie. Il y sera notamment question de certaines hypothèses afférant aux manœuvres en attitude. Des précisions seront également apportées sur les outils informatiques utilisés ainsi que sur les principaux concepts étudiés ("zones interdites" et "évitement").

Ceci permettra alors d'envisager le programme dans ses aspects techniques et dans son fonctionnement (partie III). Il faudra à cet égard distinguer deux cas de figure : la manœuvre simple et la manœuvre avec évitement. Un chapitre spécifique sera en outre consacré aux principaux résultats obtenus par le programme, permettant de valider celui-ci.

¹Centre National d'Etudes Spatiales, en France.

²Une présentation de la société est disponible dans les annexes (chapitre 9).

³ENSAE, Ecole Nationale Supérieure de l'Aéronautique et de l'Espace.

⁴Top Industrial Managers for Europe.

⁵Université Libre de Bruxelles.

Deuxième partie

Manœuvre en attitude : objectif et méthodologie du projet

Chapitre 1

Objectif

1.1 Pourquoi ce programme ?

L'intitulé du stage est "manœuvres sous contraintes", ce qui peut signifier beaucoup de choses. Dans le cas présent, il s'agit de manœuvres en attitude¹ d'un satellite, avec des contraintes dynamiques sur les vitesses et accélérations que le satellite étudié peut fournir.

L'objectif est ainsi de développer un outil informatique permettant à un satellite d'éviter des "zones interdites" lors de ses changements d'orientation. Il n'est nullement question ici de modifier la trajectoire ou la vitesse orbitale du satellite, mais uniquement son attitude. Les zones interdites correspondent à l'ensemble des directions que le satellite doit éviter pour le bon fonctionnement de ses instruments.

Dans le cas présent, ces attitudes interdites sont les directions d'astres soit suffisamment gros (la Terre) pour gêner les senseurs stellaires permettant au satellite de connaître son orientation à tout moment en se repérant par rapport aux étoiles, soit trop lumineux (le Soleil, la Lune) et qui par conséquent menacent d'endommager des instruments sensibles² ou d'éblouir les senseurs stellaires. Il est donc nécessaire de s'assurer que lors d'un changement d'attitude, aucun de ces instruments ne passe par une des zones interdites, et si le cas se présente, de trouver la trajectoire optimale permettant de les éviter.

¹L'attitude d'un satellite est son orientation dans l'espace autour de son centre de gravité.

²Certaines caméras nécessaires par exemple à l'étude d'étoiles lointaines ne peuvent pas être directement exposées à des rayons lumineux trop importants.

1.2 Définition théorique du programme

Il vient ainsi une définition claire de la mission.

A une date donnée et pour un satellite connu désirant effectuer une manœuvre en attitude précise, le programme développé vérifie que les instruments ne pointent à aucun moment dans une direction interdite, c'est-à-dire la caméra vers le Soleil ou les senseurs stellaires vers le Soleil, la Terre ou la Lune. Si tel est le cas, le programme modifie la manœuvre de manière à éviter toute zone interdite.

La description globale du programme se trouve dans la partie du même nom, et permet au lecteur d'en comprendre le fonctionnement. Toutes les étapes sont expliquées par ordre chronologique d'intervention. De plus, une représentation graphique (décrite au chapitre 4.1) des changements d'attitude sur une sphère unité permet à l'utilisateur de situer les points pertinents et d'apprécier la manœuvre effectuée.

Chapitre 2

Hypothèses émises

Pour la réalisation du programme, une série de simplification peut être envisagée.

1. Afin de tester le programme, certains paramètres propres au satellite doivent être fixés. Dans un premier temps, un modèle de satellite est défini. Celui-ci comporte une caméra et trois senseurs stellaires. Afin d'effectuer tous les essais nécessaires, la disposition des instruments sur le satellite est souvent modifiée. Par la suite, pour valider totalement le modèle, les données introduites (nombre et position des différents instruments à bord, inertie, etc.) sont inspirées d'un satellite réel, *Picard*, qui étudie la variabilité de certains paramètres solaires (irradiance, intérieur, forme et diamètre du Soleil) en fonction de son activité.
2. Dans les calculs permettant de délimiter les zones interdites, les formes irrégulières de la Terre et de la Lune sont négligées. Ces astres sont considérés comme parfaitement sphériques. Les zones interdites sont donc toujours représentées par des cercles sur la sphère unité. Cette hypothèse est tout à fait légitime car les rayons de la Lune et de la Terre utilisés dans le projet sont équatoriaux, donc maximaux. Par conséquent, la zones interdites correspondantes sont maximales.
3. Un modèle d'orbite basse képlérienne simpliste est utilisé. Celui-ci est basé sur la future orbite de *Picard*, et présenté dans les annexes à la section 10.2. Le modèle ne tient pas compte des perturbations (aplatissement de la Terre, influence de Jupiter, vents solaires, etc.). Dans la pratique, les perturbations sont souvent corrigées en cours de mission pour maintenir le satellite sur l'orbite désirée. Néanmoins, il est possible d'adapter le programme à des orbites non képlériennes¹ sans modification majeure. En effet, c'est une fonction indépendante qui renvoie la position du satellite en fonction de la date. Ainsi, celle-ci peut sans problème être remplacée par une fonction adéquate. Il n'a pas semblé pertinent toutefois de s'attarder sur cette question étrangère à l'objet central du travail.

¹Les orbites non képlériennes sont utiles notamment lors de missions à constellation de satellites.

4. En pratique, les manœuvres en attitude peuvent durer jusqu'à quelques dizaines de minutes, et certains paramètres varient avec le temps. Ainsi, une fois que le satellite s'est déplacé sur son orbite, les astres n'ont plus la même direction dans le repère inertiel. En ce qui concerne la Lune et le Soleil, cet effet demeure négligeable, puisque pour une durée de 15 minutes, les erreurs sur la direction de la Lune et du Soleil sont respectivement de l'ordre de 1% et 0,01%. Par contre, la direction de la Terre est sensiblement modifiée. Il n'est donc pas possible de simplifier ici. Pour rester aussi rigoureux que possible, les directions des trois astres sont calculées à chaque pas de temps.

Chapitre 3

Outils informatiques

3.1 Scilab

Le logiciel de calcul utilisé lors du stage s'appelle Scilab. Cet outil mathématique a été développé au début des années 90 par des chercheurs de l'INRIA¹ et l'ENPC². Il est disponible gratuitement sur Internet depuis 1994, et destiné essentiellement à la recherche et l'enseignement.

Scilab est en quelque sorte un pseudo-clone libre de Matlab. Ses fonctionnalités sont pratiquement identiques et l'environnement est très ressemblant. De plus, il est équipé de l'outil Scicos permettant de créer des modèles de systèmes, similairement à l'outil Simulink de Matlab.

Différents aspects, positifs ou négatifs, ressortent de l'utilisation de ce logiciel pendant les cinq mois du stage. La ressemblance avec Matlab facilite nettement la prise en main pour des étudiants habitués à ce logiciel de calcul. Les quelques différences entre Scilab et Matlab ne posent pas de réel problème d'adaptation, mais présentent divers avantages et inconvénients.

Tout d'abord, l'interface graphique, quoique relativement complexe au premier abord et certainement moins pratique que celle de Matlab, permet de traiter les différents éléments d'une image comme des objets. Dès lors, il est possible de régler de nombreux paramètres pour chacun de ces objets. L'outil graphique de Scilab est donc très complet, même si il est moins intuitif que celui de Matlab au premier abord.

D'autre part, Scilab fournit un mécanisme simple permettant d'interfacer du code compilé. Il est notamment conçu pour interfacer le langage C, ce qui est indispensable au projet puisque le logiciel MANIAC utilisé pour simuler les manœuvres en attitude est écrit en C (cf. section suivante).

Enfin, en ce qui concerne la programmation et l'utilisation de fonctions en général, Scilab présente la même simplicité que Matlab, bien qu'un grand nombre des fonctions disponibles n'aient pas exactement le même nom dans les deux logiciels. Malheureusement, le menu d'aide aux utilisateur n'est pas vraiment pratique. Certaines fonctions sont décrites en anglais, d'autres en français, ce qui atteste que Scilab est en évolution permanente et manque parfois de cohérence et de clareté.

¹Institut National de Recherche en Informatique et en Automatique.

²Ecole Nationale des Ponts et Chaussées.

3.2 MANIAC

Le CNES possède une bibliothèque informatique de guidage en rendez-vous, développée en langage C, nommée MANIAC (MANœuvre Imposée en Attitude sous Contraintes). Elle permet de calculer la trajectoire entre deux attitudes définies par des conditions statiques (quaternions d'attitude) et cinématiques (conditions sur les vitesses, accélérations et jerks angulaires) aux extrémités de la manœuvre. Celle-ci est soumise à des contraintes dynamiques sur le moment et le couple maximum.

MANIAC est utilisé en "boîte noire" dans le projet pour calculer une manœuvre entre deux attitudes précises. Celles-ci sont fournies par le programme principal (dans Scilab) qui gère la plupart des calculs concernant l'évitement des zones interdites. Le programme ne traite donc pas les questions de cinématiques (domaine de MANIAC), mais résoud un problème géométrique (cf. chapitre sur l'évitement).

La version de MANIAC disponible depuis le début du stage est encore provisoire. Il existe d'ailleurs une version plus récente apparue à la mi-période du stage. Celle-ci est plus générale, plus robuste et plus complète. De plus, elle comporte de nombreuses options supplémentaires, comme notamment un choix plus conséquent des conditions aux limites. Enfin, elle offre la possibilité de calculer des manœuvres en durée minimum, en plus de la fonction de calcul en durée fixe disponible dans la version utilisée pour le projet. Cette notion de durée est exposée à la section 6.2.2 du rapport.

En ce qui concerne l'utilisation, le programme MANIAC est conçu de façon à tourner sur des machines Sun, avec l'OS³ Solaris. Le projet a été réalisé en se connectant depuis des postes personnels sur les deux stations de travail disponibles au service où le stage a été effectué.

³Operating System, système d'exploitation.

Chapitre 4

Concepts utilisés

4.1 Remarques préliminaires sur la représentation graphique

Bien que la représentation graphique ne constitue que la dernière étape du programme, il semble utile de l'expliquer avant le reste. En effet, de nombreux dessins issus de cette fonction illustrent les propos et les descriptions qui suivent.

La représentation graphique a pour but de dessiner les manœuvres en attitude effectuées. Pour cela, les directions de visée des instruments embarqués portées sur une sphère unité, appelée "sphère d'attitude". Chaque point de cette sphère correspond à une direction normée dans le repère inertiel. Un "trajet" d'attitude sur la sphère est ainsi obtenu pour chacun de ces axes.

Les différents axes pouvant être représentés (et leur couleur) sont :

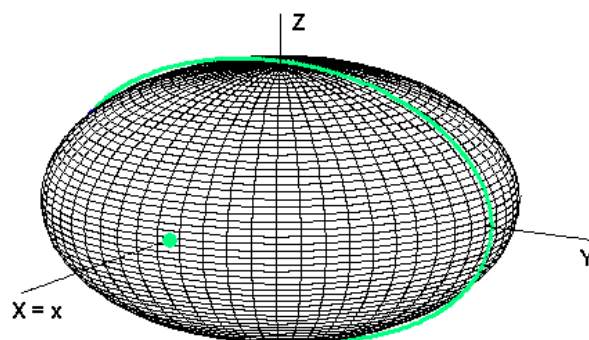
- l'axe de visée de la caméra (en bleu),
- l'axe de visée de chaque senseur stellaire (en rouge),
- les axes (x, y, z) du repère satellite (en vert), qui permettent notamment de mieux visualiser le changement d'orientation du satellite.

Pour représenter l'attitude du satellite, deux directions différentes doivent être connues. Cette condition peut être illustrée par un exemple simple (voir figures 4.1 et 4.2). Soit le repère satellite (x, y, z) initialement aligné sur le repère inertiel (X, Y, Z) , et soit l'axe x l'unique direction représentée sur la sphère unité. Suite à une rotation d'angle quelconque autour de l'axe X , la direction de l'axe x reste inchangée.

$$\vec{x} = \vec{X} = (1, 0, 0)$$

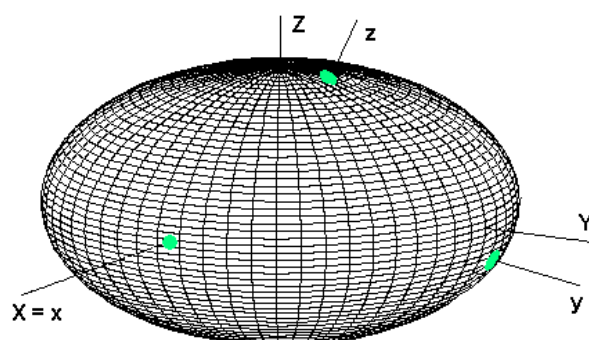
En effet, la direction d'un seul vecteur ne suffit pas à définir l'orientation du repère satellite, c'est-à-dire l'attitude de celui-ci. La rotation autour de X n'étant pas définie, il y a une infinité d'orientations possibles pour les axes y et z . Autrement dit, il reste un degré de liberté au quaternion¹ permettant de représenter l'attitude du satellite. Par conséquent, celle-ci reste indéterminée.

¹L'utilisation des quaternions est exposée plus en détails à la section 11.2 des annexes

FIG. 4.1 – Un seul axe (x) positionné sur la sphère unité : orientation indéfinie

Pour que l'orientation du repère satellite soit entièrement déterminée, il est nécessaire de connaître également l'angle de la rotation. Pour ce faire, la direction d'un second vecteur est requise, comme par exemple un des deux autres axes (y ou z). Avec deux points minimum sur la sphère unité, le quaternion est ainsi complètement défini, et l'attitude peut être représentée.

Sur la figure suivante, les trois axes du satellites sont représentés par une rotation donnée autour de X . L'attitude est alors bien définie avec trois points.

FIG. 4.2 – Plusieurs axes (x, y, z) positionnés : orientation bien définie

Enfin, la représentation graphique permet de dessiner les zones interdites sur la sphère unité (cf. chapitre suivant), et le mouvement² de celles-ci dans le temps.

²Comme expliqué au chapitre 2 concernant les hypothèses, le mouvement des zones interdites dues au Soleil ou à la Lune est négligeable, mais il en est tout de même tenu compte lors du contrôle de l'évitement par mesure de sécurité.

4.2 Zones interdites

4.2.1 Définition

Il existe différents types de zones interdites qui dépendent des instruments considérés. Deux catégories sont à distinguer dans le problème traité. La caméra d'une part, prévue par exemple pour scruter les étoiles lointaines, est très sensible à la lumière et peut être endommagée si de trop fortes expositions surviennent (par exemple en pointant vers le Soleil). Les senseurs stellaires d'autre part, permettent de déterminer l'attitude du satellite par comparaison à une bibliothèque d'images du ciel. Par conséquent, un senseur stellaire doit pointer vers les étoiles à tout moment. S'il venait à s'orienter vers la Terre, son information serait indisponible et l'attitude du satellite deviendrait impossible à déterminer. De plus, si un senseur s'orientait vers le Soleil ou la Lune, l'éblouissement serait trop fort pour permettre une bonne reconnaissance des étoiles aux alentours.

Ces zones interdites sont définies par la portion de surface sphérique (calotte sphérique) interceptée par un cône (en rouge) partant du centre de la sphère (où se trouve l'instrument), et dont l'angle au sommet dépend à la fois de la distance et du rayon de l'astre considéré, mais aussi de l'angle d'ouverture (moitié du champ de vision) de l'instrument.

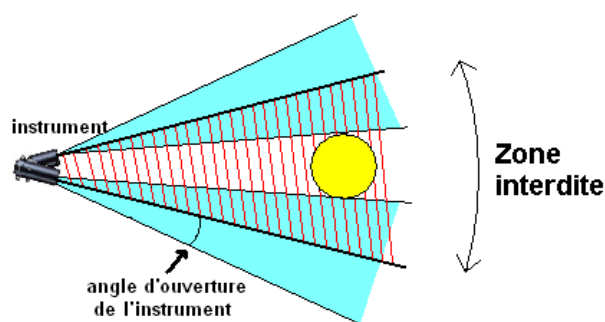


FIG. 4.3 – Définition d'une zone interdite

La figure précédente permet de constater que la zone interdite se réduit lorsque l'astre considéré est plus loin, ou que son rayon diminue, ou encore que l'angle d'ouverture de l'instrument utilisé est moindre.

4.2.2 Illustration des zones interdites

Voici, sur les figures 4.4 et 4.5, deux exemples de zones interdites représentées sur des sphères d'attitudes.

Le premier exemple représente une zone interdite due à la Lune (gauche) et une autre due à la Terre (droite), pour un senseur stellaire d'angle d'ouverture de 15° . La zone interdite due à la Terre est nettement plus grande que celle due à la Lune. En effet, le rayon de la Terre est presque quatre fois plus important, et le satellite en est bien plus proche que de la Lune. La distance Terre – Lune vaut environ 380000 km en moyenne, tandis que la distance Terre – Satellite vaut environ 7200 km (demi-grand axe).

D'autre part, cette figure permet de constater le déplacement non négligeable de la zone de la Terre durant la manoeuvre. Deux grandes zones sont visibles : celle dont le contour est vert représente la position initiale de la zone, tandis que l'autre représente sa position en fin de manoeuvre.

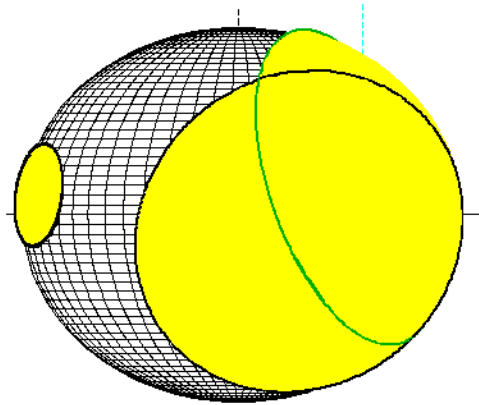


FIG. 4.4 – Comparaison Terre – Lune

Le deuxième exemple compare les angles d'ouverture des différents instruments (caméra et senseurs stellaires). En effet, le cas particulier du Soleil présente un danger aussi bien pour l'un que pour l'autre. Cependant, l'angle d'ouverture de la caméra (5°) est plus petit que celui des senseurs (15°). La caméra doit pouvoir observer des objets lointains. C'est la raison pour laquelle son ouverture est plus faible que celle des senseurs stellaires qui nécessitent moins de précision. Au contraire, ceux-ci doivent pouvoir photographier un maximum d'étoiles pour permettre au satellite de bien s'orienter.

La zone interdite à la caméra est représentée en bleu, tandis que celle interdite aux senseurs est représentée en jaune. Les deux cercles sont concentriques puisqu'ils représentent tous les deux le Soleil.

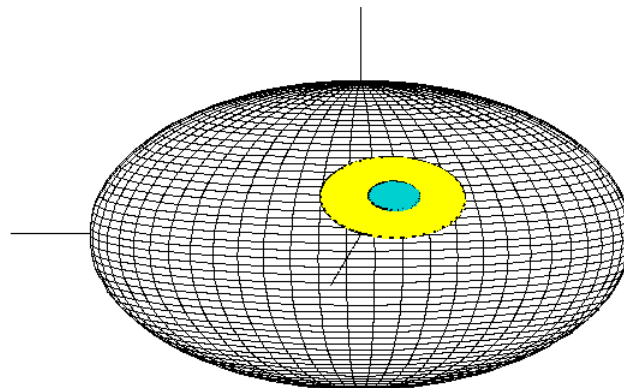


FIG. 4.5 – Comparaison caméra – senseurs stellaires

4.2.3 Coefficient de sécurité

Pour terminer, voici la représentation d'une sphère d'attitude complète, en tenant compte de toutes les zones interdites (en jaune pour les senseurs, en bleu pour la caméra). De plus, un cercle concentrique à chaque zone est également dessiné. Celui-ci représente un coefficient de sécurité supplémentaire³ pour éviter la zone.

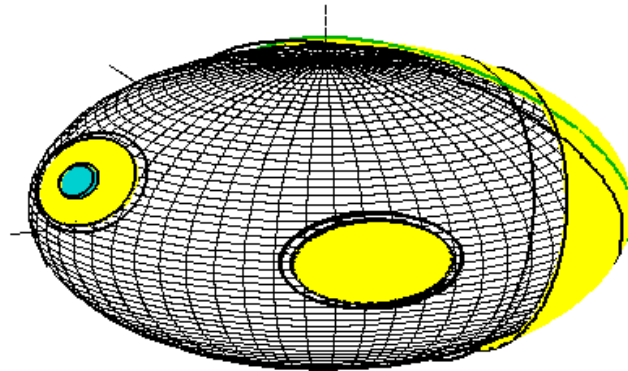


FIG. 4.6 – Toutes les zones interdites

³Le coefficient de sécurité est fixé à 1,15 dans le programme.

4.3 Evitement

Lors d'une manœuvre en attitude du satellite, il faut s'assurer, pour chacune des zones interdites, qu'aucun des instruments concernés ne la traverse. Cette vérification est opérée à chaque pas de temps. Si le programme détecte un passage dans une zone interdite, le processus d'évitement est enclenché automatiquement.

L'évitement d'une zone interdite consiste à trouver un point intermédiaire (point C) sur la sphère unité, tel que le passage par ce point intermédiaire contourne la zone interdite. Le trajet initial (A – B) est alors remplacé par deux trajets consécutifs : A – C et C – B. L'objectif de ce chapitre est d'exposer le raisonnement appliqué pour déterminer le point C et ainsi éviter une zone interdite. Cependant, son implémentation dans le programme est décrite avec précision dans la section 7.2.

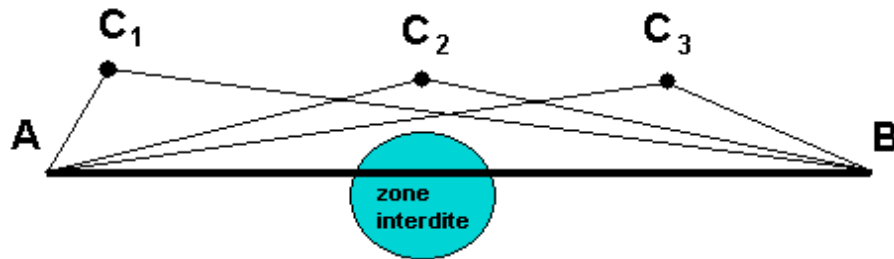


FIG. 4.7 – Différents points intermédiaires possibles (représentation bidimensionnelle)

La figure précédente n'est qu'une représentation bidimensionnelle de l'évitement. En réalité, ce ne sont pas des lignes droites sur la sphère unité. Néanmoins, il apparaît assez clairement que le chemin le plus court pour éviter un obstacle est de prendre la trajectoire tangente à celui-ci.

4.3.1 Méthode des tangentes

La résolution du problème consiste dans un premier temps à déterminer le meilleur point intermédiaire à atteindre, c'est-à-dire celui qui s'éloigne le moins de la trajectoire initiale, tout en permettant d'éviter la zone à problème. C'est donc un problème de géométrie sphérique. Pour le résoudre de manière optimale, il est nécessaire d'utiliser les formules concernant les triangles sphériques, développées à la section 11.1 des annexes.

D'autre part, le coefficient de sécurité représenté à la figure 4.8 est pris en compte. La zone à éviter est alors légèrement plus grande que la zone interdite. La méthode d'évitement utilisée consiste à tracer des tangentes à cette zone, à partir des points de départ et d'arrivée. L'intersection entre une tangente provenant du point de départ et une autre provenant du point d'arrivée donne le point intermédiaire.

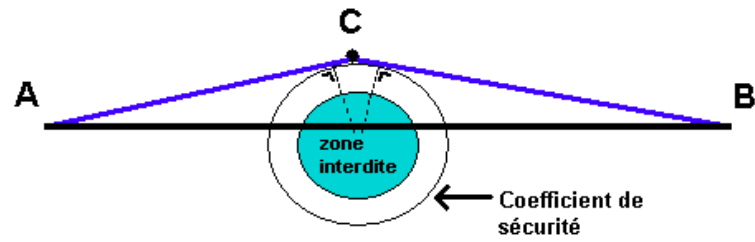


FIG. 4.8 – Point intermédiaire obtenu par la méthode des tangentes

4.3.2 Plusieurs points intermédiaires (appels récursifs)

Dans certains cas, il peut arriver que les points de départ et/ou d'arrivée soient très proches de la zone à éviter. Dans ce cas, la méthode des tangentes donne un résultat trop éloigné. Pour éviter que la trajectoire ne fasse un détour démesuré, la méthode est répétée plusieurs fois pour définir des points intermédiaires supplémentaires. De cette manière, la trajectoire contourne la zone de plus près. Cette opération est répétée jusqu'à obtenir une trajectoire suffisamment proche de la zone à éviter.

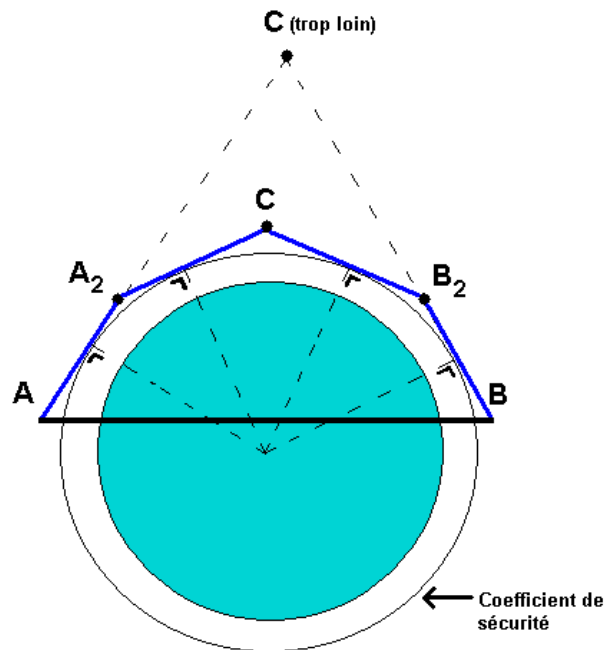


FIG. 4.9 – Contournement proche

4.3.3 Plusieurs zones à éviter (appels récursifs)

Une fois l'évitement calculé, le programme simule à nouveau la manœuvre. Il peut arriver que sur la nouvelle trajectoire d'évitement, un deuxième instrument soit orienté dangereusement. L'opération d'évitement est alors lancée autant de fois que nécessaire, de manière récursive, jusqu'à obtenir une trajectoire autorisée (qui ne touche pas la zone bleue).

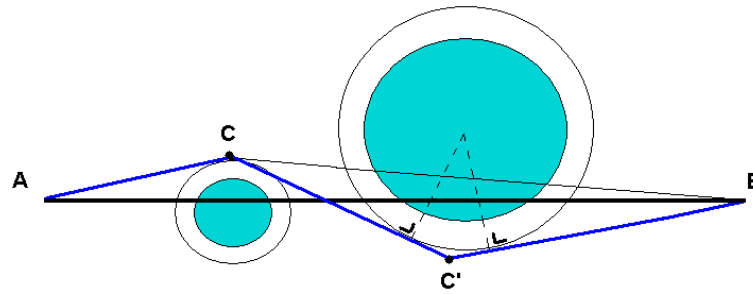


FIG. 4.10 – Plusieurs zones à éviter

Troisième partie

Description technique du programme et résultats obtenus

Avant de détailler le programme, le diagramme suivant permet de présenter celui-ci dans son ensemble. Les différents chapitres exposés dans cette partie représentent chacune des étapes importantes dans la structure du programme.

1. La première phase consiste à initialiser le problème (cf. chapitre 5).
 - Initialiser MANIAC : il s’agit d’entrer les données concernant la structure du satellite ainsi que les conditions aux limites de la manœuvre.
 - Calculer la configuration des instruments à bord.
 - Définir la date.

A partir de ces données, il est possible de calculer la position dans l’Espace du satellite ainsi que celles des astres. Le programme connaît alors toutes les variables du problème et est prêt à les traiter.

2. Il s’agit alors, dans un second temps, d’appeler MANIAC. Cette bibliothèque de rendez-vous d’attitude calcule des données brutes concernant la trajectoire telles que le nombre de segments, les quaternions et les coefficients des vitesses aux bords de chaque segment. Le traitement de ces données dans Scilab permet de retrouver la cinématique et les quaternions de l’ensemble de la manœuvre. Une fois ces quaternions déterminés, l’évolution des axes de visée des instruments sur la sphère d’attitude peut être déterminée (cf. chapitre 6). Il est alors possible de vérifier si la trajectoire passe par des zones à éviter (test sur les zones interdites).
3. Si tel est le cas, le processus d’évitement est enclenché. Selon l’instrument et l’astre concerné et en fonction du coefficient de sécurité imposé, le programme détermine le(s) point(s) intermédiaire(s) qui permette(nt) d’éviter la zone. De plus, les quaternions amenant à ce(s) point(s) sont également calculés. A partir de ces nouvelles données, il convient de rappeler MANIAC pour calculer la nouvelle trajectoire, puis de vérifier si celle-ci ne comporte pas, à son tour, de zones à éviter. Il s’agit donc d’un système récursif qui est activé jusqu’à trouver une solution satisfaisante (cf. chapitre 7).
4. Enfin, la représentation graphique, décrite à la section 4.1, permet de visualiser le changement d’attitude final proposé par le programme.

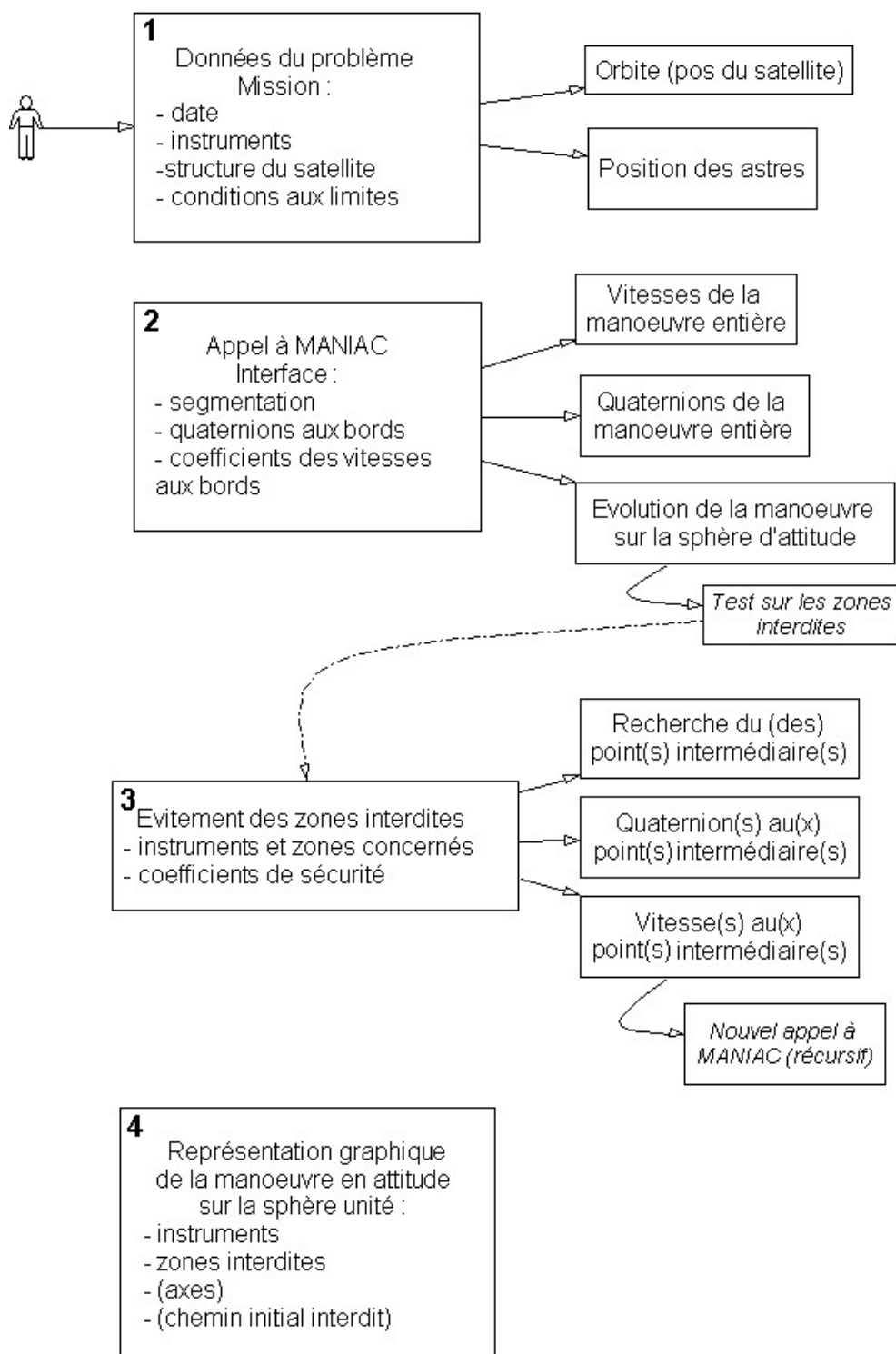


FIG. 4.11 – Diagramme du programme développé

Chapitre 5

Initialisation du problème

5.1 Introduction des données

La première étape du programme consiste en l'introduction de toutes les données permettant d'effectuer correctement la manœuvre en attitude. Ces données sont regroupées par catégorie en fonction de leur utilité.

5.1.1 Dates de début et de fin de manœuvre

Les dates sont nécessaires pour définir la position des astres et du satellite dans le repère inertiel. Les différents repères utilisés sont présentés à la section 10.3 des annexes. La différence de temps entre la date de début et la date de fin de manœuvre, calculée en secondes, détermine la durée maximale pendant laquelle la manœuvre doit se dérouler. Le programme calcule la manœuvre optimale (du point de vue de la durée) dans cette fenêtre temporelle.

5.1.2 Paramètres orbitaux du satellite

L'orbite est supposée képlérienne (cf. hypothèses, chapitre 2). Par conséquent, les cinq paramètres orbitaux (cf. annexe 10.1.2) suivants sont constants :

- a : demi-grand axe
- e : excentricité
- i : inclinaison
- Ω : ascension droite du nœud ascendant
- ω : argument du périhélie.

Le sixième paramètre, l'anomalie moyenne M , est calculé à chaque pas de temps¹ durant la durée totale du changement d'attitude.

¹Dans ce projet, le pas de temps vaut 1 seconde.

5.1.3 Orientation des instruments

La position des instruments à bord du satellite n'est pas connue précisément, mais cette information n'est pas nécessaire au programme. En effet, seule la direction de visée de ceux-ci est requise. Cette orientation est définie en coordonnées sphériques dans le repère satellite, ce qui fournit directement la direction de visée sur la sphère unité après passage au repère inertiel.

5.1.4 Données pour MANIAC

Voici toutes les données nécessaires à MANIAC pour réaliser une manœuvre entre deux points :

- capacités du satellite (moments d'inertie et couples maximum sur les trois axes)
- inertie du satellite
- conditions initiales et finales (attitude, vitesse, accélération et variation de l'accélération²)
- données mathématiques (degré des polynômes des consignes des manœuvres, choix de la discrétisation des durées calculées).

²La variation de l'accélération est appelée "jerk" en anglais. Cette appellation est également utilisée dans certains ouvrages en français.

5.2 Position des astres - zones interdites

A ce stade-ci, avec la connaissance des six paramètres orbitaux, la position du satellite dans le repère inertiel est connue en fonction de la date. Ce calcul est détaillé dans les annexes à la section 10.2. Il reste à calculer les positions, dans ce repère, des différents astres intervenant dans le problème afin d'en déduire les directions par rapport au satellite. Bien entendu, ces positions dépendent également de la date (sauf pour la Terre supposée fixée au centre du repère inertiel).

5.2.1 Terre-satellite

En ce qui concerne la Terre, le calcul est immédiat. En effet, les coordonnées $(X_{sat}, Y_{sat}, Z_{sat})$ du satellite dans le repère inertiel (centré sur la Terre) ayant été déterminées à la section précédente, la distance Terre-satellite s'obtient comme suit.

$$d_{Terre-sat} = \sqrt{X_{sat}^2 + Y_{sat}^2 + Z_{sat}^2}$$

La direction \vec{T} de la Terre par rapport au satellite s'en déduit aisément, puisqu'il suffit de prendre l'opposé des coordonnées du satellite sur son orbite, divisées par la distance Terre-satellite (pour normer).

$$\vec{T} = (T_X, T_Y, T_Z) = -\frac{1}{d_{Terre-sat}}(X_{sat}, Y_{sat}, Z_{sat})$$

5.2.2 Lune-satellite – Soleil-satellite

Les directions et distances de la Lune et du Soleil par rapport à la Terre à une date donnée sont calculées à l'aide d'une routine fournie par le CNES ("poluso" pour position de la Lune et du Soleil). Il ne reste plus qu'à les corriger pour que celles-ci soient définies par rapport au satellite et non plus par rapport au centre de la Terre.

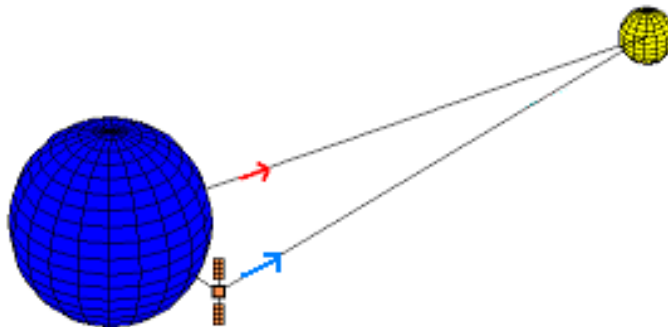


FIG. 5.1 – Position et direction de la Lune par rapport au satellite

Un simple calcul permet de déterminer ces nouvelles distances et directions. En voici les équations dans le cas de la Lune.

$$\begin{cases} dX = X_{Lune} - X_{sat} \\ dY = Y_{Lune} - Y_{sat} \\ dZ = Z_{Lune} - Z_{sat} \end{cases} \Rightarrow d_{Lune-sat} = \sqrt{dX^2 + dY^2 + dZ^2}$$

En découle la direction \vec{L} de la Lune par rapport au satellite.

$$\vec{L} = (L_x, L_y, L_z) = \frac{1}{d_{Lune-sat}}(dX, dY, dZ)$$

Les manipulations concernant le Soleil sont tout à fait identiques, et permettent de déterminer sa distance $d_{Soleil-sat}$ et sa direction \vec{S} par rapport au satellite.

5.2.3 Rayon des zones interdites

Les calculs précédents permettent de positionner les zones interdites sur la sphère unité. En effet, les vecteurs direction des différents astres (\vec{T} , \vec{L} , \vec{S}) correspondent aux coordonnées des centres des zones interdites causées par chaque astre. Ces zones sont délimitées par un cercle sur la sphère, formant ainsi une calotte sphérique.

Le dernier paramètre à calculer est le rayon angulaire³ de la zone interdite. Ce rayon (comme expliqué au chapitre 4.2) est la somme du rayon angulaire apparent de l'astre⁴ et de l'ouverture (moitié du champ de vision) l'instrument considéré. Exemple du senseur stellaire 1 avec la Lune :

$$\alpha_{Lune-st1} = \alpha_{st1} + \alpha_{Lune}$$

où α_{st1} est l'angle d'ouverture du senseur stellaire 1,

et $\alpha_{Lune} = \arctan \frac{R_{Lune}}{d_{Lune-sat}}$, avec $R_{Lune} = 1738,2 \text{ km}$.

Pour les zones interdites correspondant à la Lune et au Soleil, l'ouverture des instruments (entre 5° et 15° généralement) est bien supérieure au rayon apparent ($R_{appSol} \approx R_{appLun} \approx 0,3^\circ$). Ceci a pour conséquence que ces deux astres représentent une zone interdite de taille quasi identique. Par contre, la Terre étant beaucoup plus proche du satellite, le rayon apparent n'est pas du tout négligeable, comme on peut le voir sur le dessin suivant.

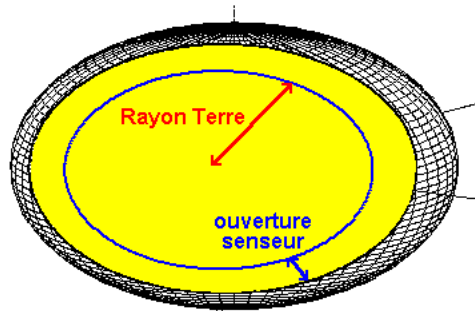


FIG. 5.2 – Décomposition du rayon apparent de la zone interdite due à la Terre

³Le rayon angulaire de la zone interdite correspond à l'angle au sommet du cône interceptant celle-ci sur la sphère d'attitude.

⁴Le rayon angulaire apparent d'un astre est l'angle au sommet du cône interceptant celui-ci.

Chapitre 6

Manœuvre simple (sans évitement)

Avant d'exposer précisément la méthode d'évitement, ce chapitre décrit comment effectuer une manœuvre simple. L'appel à la maquette informatique MANIAC depuis Scilab, ainsi que le traitement des données calculées sont exposées ci-dessous.

6.1 Interfaçage de MANIAC

Comme cela a été expliqué dans le chapitre 3, la manœuvre à proprement parler est calculée par MANIAC, la bibliothèque de guidage en rendez-vous développée au CNES. Ce programme est écrit en langage C. Il a dès lors fallu développer une interface pour l'utiliser via Scilab. Cette interface a elle-même été écrite en langage C. Le fait que MANIAC soit écrit dans ce langage de programmation présente l'avantage d'être très rapide et facilement interfaçable avec Scilab. L'interface réalise l'appel à MANIAC à partir du programme Scilab en lui passant toutes les entrées en paramètres, et récupère les sorties de MANIAC pour les attribuer à des variables de Scilab.

6.1.1 Deux fichiers d'interface : *intInit.c* et *intDfix.c*

De nombreuses options sont disponibles dans MANIAC, mais elles ne sont pas toutes utiles à la réalisation du projet. Deux fonctions seulement suffisent à réaliser une manœuvre simple.

- *Init* : permet comme son nom l'indique d'initialiser MANIAC.
- *Dfix* : calcule réellement le changement d'attitude pour une "durée fixe".

Par conséquent, deux fichiers d'interface sont nécessaires, un pour chaque fonction.

6.1.2 Fonctionnement de l'interface

Les deux fichiers d'interface sont semblables. La seule différence est qu'ils font appel à deux fonctions de MANIAC distinctes mais leur fonctionnement est toutefois identique. Par conséquent, celui-ci est décrit de façon tout à fait générale dans la suite de ce paragraphe. Un schéma explicatif (figure 6.1) est représenté à la fin de la description ci-dessous.

Chacun des deux fichiers consiste en une série de commandes écrites en C, mais que Scilab peut comprendre. Ces lignes de code sont en fait regroupées en deux fonctions différentes en C écrites au sein du même fichier. Le choix de découpler chaque fichier d'interface en deux fonctions permet de scinder les deux parties importantes de l'interfaçage et de cette manière obtenir plus de clarté.

La première fonction est celle appelée par Scilab. Il faut lui passer en paramètre les entrées requises par la fonction MANIAC à interfacier. La fonction vérifie alors si le nombre de paramètres d'entrée et de sortie est correct lors de l'appel dans Scilab. Ensuite, elle vérifie si les paramètres d'entrée sont bien définis (type et taille des matrices), et affiche un message d'erreur si ce n'est pas le cas. Enfin, la fonction crée les variables de retour avec le bon type et la bonne taille.

Après la création de ces variables, la deuxième fonction est appelée. Le rôle de celle-ci est de transtyper les variables d'entrée, car MANIAC redéfinit tous les types de variables. Cette fonction contient donc une suite de lignes de code presque identiques dont le but est de changer le type de chaque variable d'entrée. Une fois le transtypage réalisé, les données sont prêtes à être traitées par MANIAC et l'appel à celui-ci peut être effectué. Quand la manœuvre est calculée, il reste à procéder au transtypage inverse des sorties fournies par l'appel à MANIAC, afin que celles-ci soient comprises par Scilab. La deuxième fonction est alors achevée, le programme retourne dans la première fonction et termine en envoyant les variables de sortie vers Scilab.

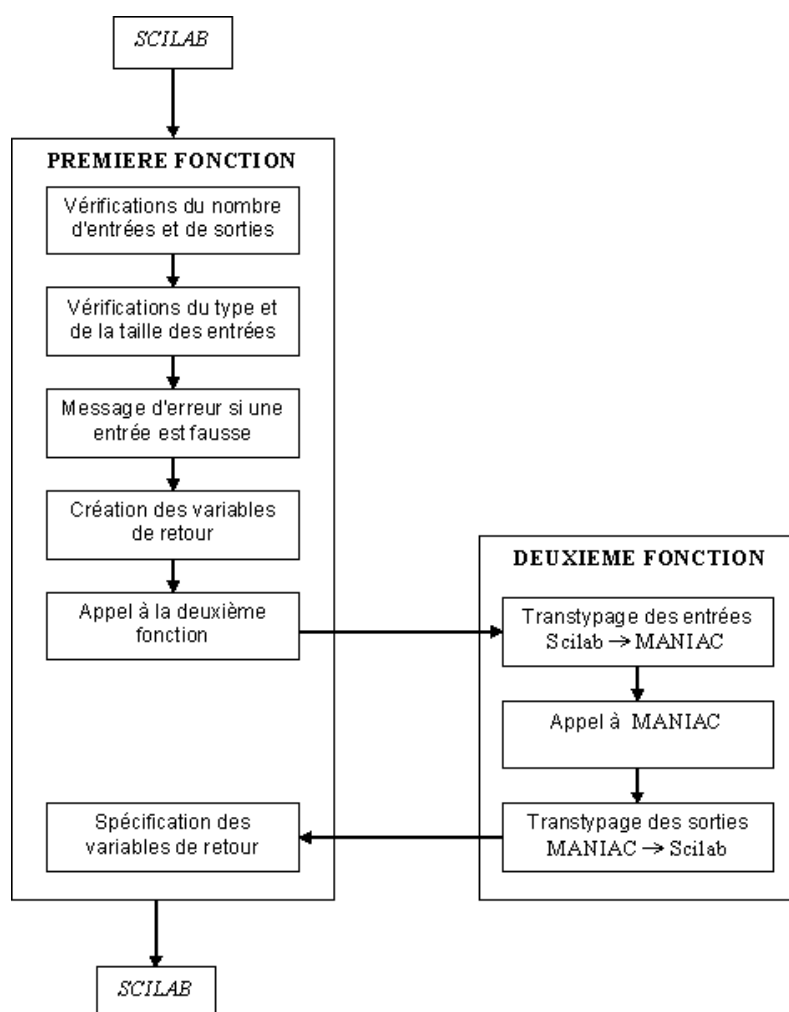


FIG. 6.1 – Schéma de l'interface

6.1.3 Utilisation de l'interface

Pour utiliser ces fonctions d'interface avec Scilab, quelques opérations supplémentaires sont toutefois nécessaires. Tout d'abord, il faut écrire un fichier *builder.sce* (dans Scilab) qui a pour but d'associer les appels de Scilab aux fonctions d'interface en C (*intInit.c* et *intDfix.c*). L'appel à ces fichiers se fera ainsi de la même manière que pour une fonction écrite avec Scilab. La fonction *builder* construit de plus les bibliothèques dynamiques permettant d'utiliser MANIAC via Scilab, ainsi qu'un fichier *loader.sce* capable de charger ces bibliothèques.

L'exécution de la fonction *builder* ne doit être effectuée qu'une seule fois, à moins bien sûr de modifier le code d'un des fichiers d'interface. La fonction *loader* quant à elle doit être exécutée à chaque nouvelle ouverture d'une session de Scilab.

6.2 Appel à MANIAC

Comme décrit dans la section précédente, deux fonctions de MANIAC ont été interfacées. La première (*intInit.c*), l'initialisation, reçoit des données en rapport avec le satellite (couple maximum, moment maximum, inerties) ainsi que des données mathématiques liées au calcul (degré des polynômes des consignes des manœuvres, discrétisation des durées calculées). Cette fonction d'initialisation ne possède aucune sortie.

La deuxième fonction (*intDfix.c*) a besoin de connaître d'autres paramètres comme la durée, ainsi que toutes les conditions aux limites. Cette fonction calcule réellement la manœuvre et, par conséquent, renvoie de nombreux résultats. La suite de cette section a pour objectif de décrire le fonctionnement du calcul de la manœuvre avec la fonction *Dfix* (en durée fixe).

6.2.1 Conditions aux limites

L'utilisateur peut choisir le type de conditions aux limites qu'il impose, en modifiant la variable *methode* :

- méthode *B00* : seules les vitesses (autour des 3 axes - tangage, roulis, lacet) aux bords sont définies,
- méthode *B01* : fixe également les accélérations en fin de manœuvre,
- méthode *B12* : fixe également les accélérations aux bords, et les jerks en fin de manœuvre.

La manœuvre est dès lors découplée sur plusieurs segments, chacun étant caractérisé par une vitesse et une accélération propres¹. La manœuvre la plus simple et la plus utilisée dans ce projet est celle de type *B00* avec des vitesses nulles. Dans ce cas, la trajectoire est scindée en trois segments : le premier à accélération constante jusqu'à la vitesse de manœuvre, le deuxième à vitesse constante, qui dure la majeure partie du temps de la manœuvre, et le troisième à décélération constante jusqu'à atteindre les conditions finales homogènes. Dans les cas de conditions aux limites plus complexes, les trajectoires peuvent contenir jusqu'à 21 segments.

La figure suivante permet de comparer le cas *B00* homogène (figure de gauche) à un cas plus complexe, *B00* avec conditions limites sur la vitesse non nulles (figure de droite). Ce dernier est constitué de 5 segments.

¹Le jerk n'est pas calculé, mais sert uniquement de condition aux limites pour la méthode *B12*

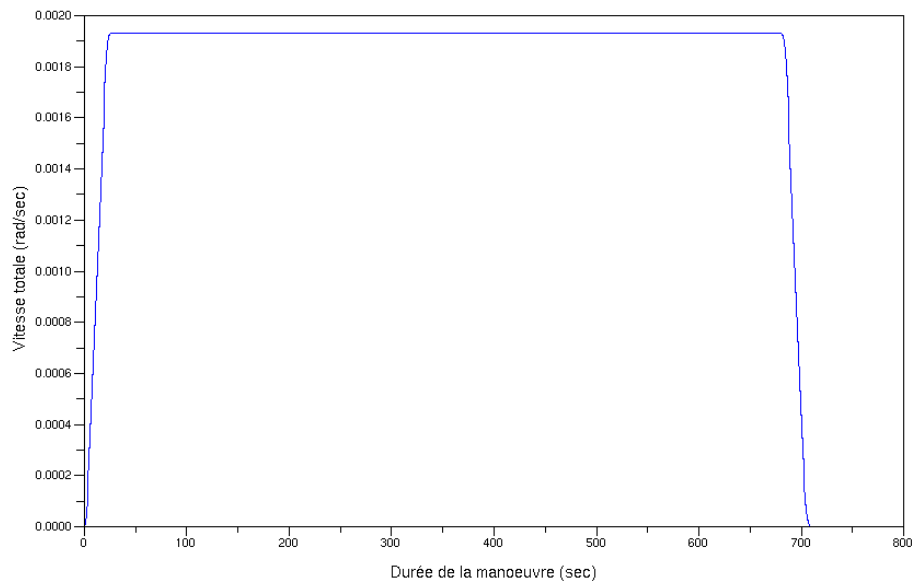


FIG. 6.2 – Profil de la vitesse lors d'une manœuvre simple

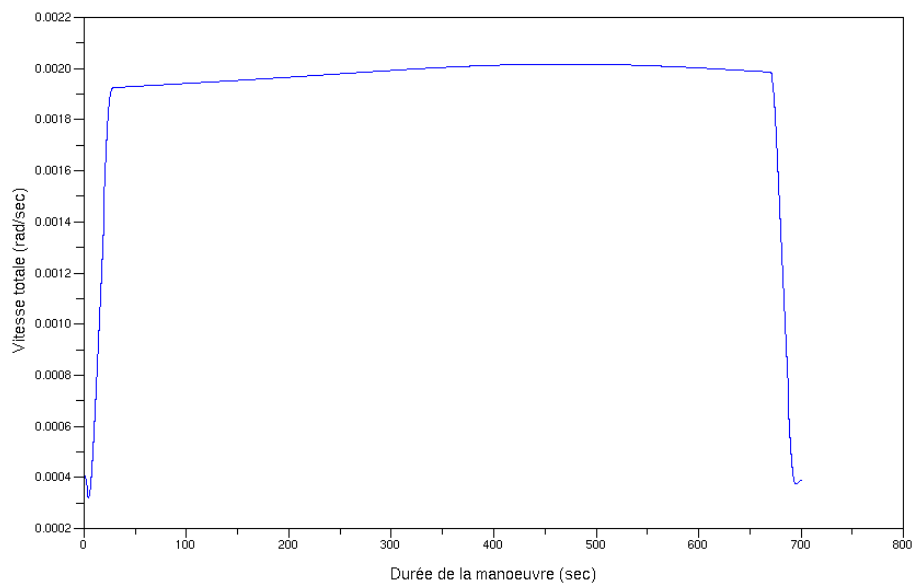


FIG. 6.3 – Profil de la vitesse lors d'une manœuvre complexe

6.2.2 Manœuvre en durée minimum

Le paramètre de sortie *ratio* correspond au rapport entre la durée minimale estimée de la manœuvre et la durée imposée (durée fixe) de celle-ci. S'il est supérieur à 1, la manœuvre est impossible car il n'y a pas assez de temps pour la réaliser. S'il est inférieur à 1, elle pourrait être raccourcie.

Afin d'optimiser la durée de la trajectoire, le calcul du changement d'attitude par MANIAC est appelé récursivement en modifiant la durée voulue de la manœuvre, jusqu'à ce que le ratio converge vers 0,99 avec une précision de 10^{-2} (marge de sécurité).

$$durée_n = ratio \cdot durée_{n-1}$$

Cette opération converge très rapidement, comme on peut le voir sur l'exemple ci-dessous (moins de cinq appels à MANIAC).

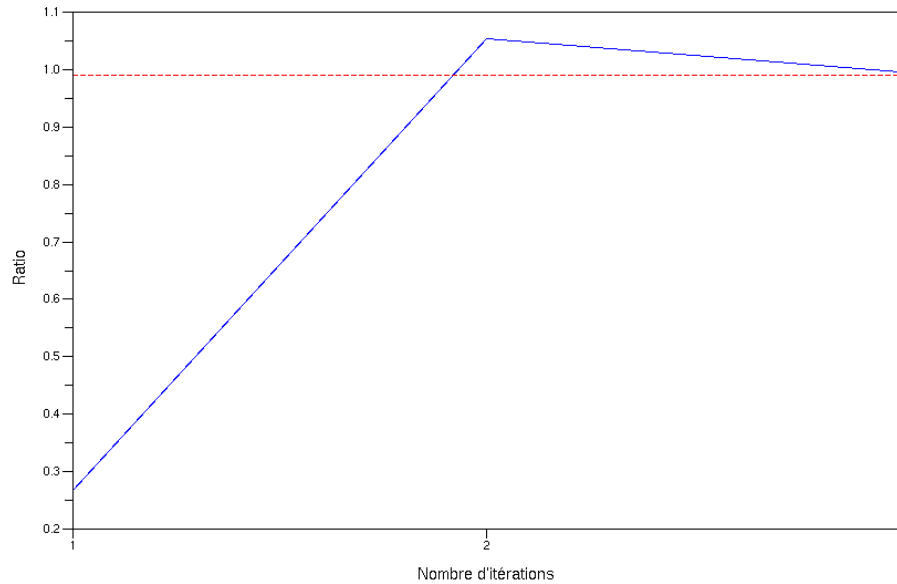


FIG. 6.4 – Convergence du ratio vers 0,99 (en fonction du nombre d'itérations)

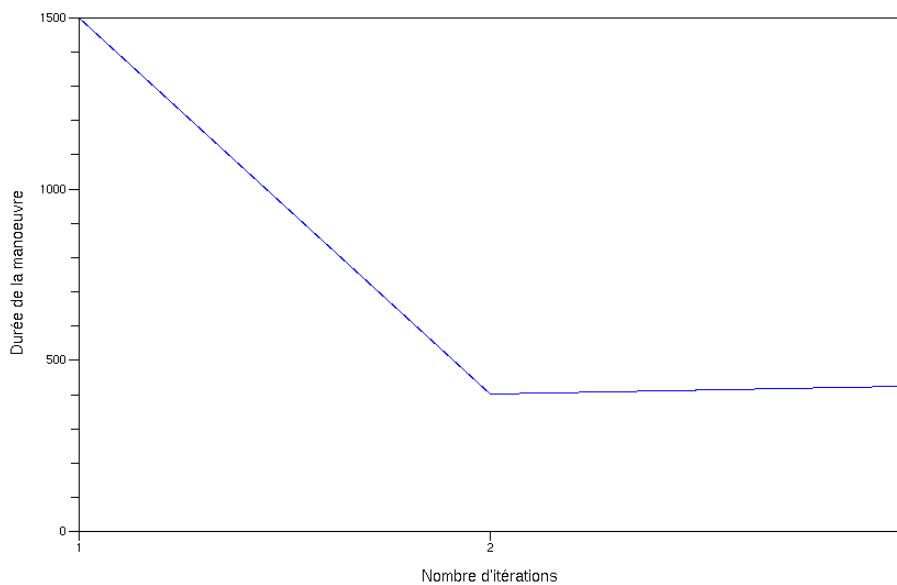


FIG. 6.5 – Durée de la manœuvre (en fonction du nombre d'itérations)

Il est à noter que la nouvelle version de MANIAC (cf. section 3.2), contient une fonction qui calcule directement une manœuvre en un temps minimal. Cette opération avec le ratio n'est donc plus nécessaire si la nouvelle version est utilisée.

6.3 Traitement des données

Avant de traiter les données obtenues, une mise en forme préalable est nécessaire, pour ne pas stocker de zéros inutiles, ou pour obtenir des matrices plus faciles à utiliser par la suite. En effet, la mémoire informatique disponible sur un satellite n'est pas énorme et il faut donc veiller à ne pas l'utiliser vainement. Les variables retournées par MANIAC sont généralement des tableaux à 21 lignes (nombre maximum de segments), alors que dans la plupart des cas (méthode B00), le nombre de segments est largement inférieur.

6.3.1 Vitesses et accélérations à chaque pas de temps

Les vitesses calculées par MANIAC sont retournées sous forme de matrices à six colonnes, représentant les six coefficients des polynômes des vitesses de degré 5, autour des trois axes inertiels (X , Y , et Z). Ces polynômes donnent l'évolution des vitesses en fonction du temps.

Pour la suite du projet, il est nécessaire de connaître les vitesses et les accélérations instantanées, afin de pouvoir spécifier des conditions aux limites lors d'appels postérieurs durant l'évitement. En utilisant les coefficients des polynômes obtenus, le calcul des vitesses ainsi que des accélérations à chaque pas de temps de la manœuvre est immédiat. Les vitesses et accélération autour de l'axe X (par exemple) valent

$$\omega_X(t_n) = \alpha + \beta t_n + \gamma t_n^2 + \delta t_n^3 + \epsilon t_n^4 + \zeta t_n^5$$

$$\dot{\omega}_X(t_n) = \beta + 2 \gamma t_n + 3 \delta t_n^2 + 4 \epsilon t_n^3 + 5 \zeta t_n^4$$

avec

$$t_n = \frac{t}{duree} \Rightarrow 0 \leq t_n \leq 1 .$$

Les vitesse et accélération autour de Y ($\omega_Y(t_n)$ et $\dot{\omega}_Y(t_n)$) et Z ($\omega_Z(t_n)$ et $\dot{\omega}_Z(t_n)$) s'obtiennent de manière similaire.

6.3.2 Quaternions à chaque pas de temps

Les quaternions d'attitude sont calculés par MANIAC aux bords de chaque segment, et retournés sous forme de tableau à 4 colonnes. Pour une manœuvre en quatre segments par exemple, il y aura cinq lignes à la matrice des quaternions.

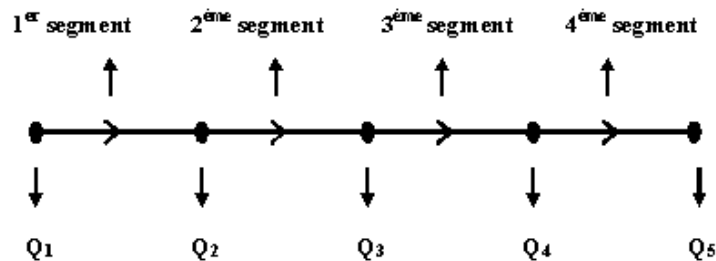


FIG. 6.6 – Quaternions intermédiaires d'une manœuvre simple

Les quaternions aux bords des segments sont des vecteurs ligne : $Q_i = (q_{i1}, q_{i2}, q_{i3}, q_{i4})$. La matrice des quaternions de la manœuvre entière à n segments est de taille $(n + 1) \times 4$:

$$Q = \begin{pmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ \vdots & \vdots & \vdots & \vdots \\ q_{i1} & q_{i2} & q_{i3} & q_{i4} \\ \vdots & \vdots & \vdots & \vdots \\ q_{(n+1)1} & q_{(n+1)2} & q_{(n+1)3} & q_{(n+1)4} \end{pmatrix}.$$

De même qu'au point précédent, la suite du projet nécessite de connaître les quaternions à chaque pas de temps. Pour cela, la méthode utilisée consiste à calculer le quaternion au temps t_{n+1} à partir du quaternion au temps t_n , en multipliant ce dernier par un quaternion de rotation élémentaire (dQ). Ce quaternion élémentaire s'obtient en chaque pas de temps (dt) en manipulant les angles de rotation élémentaires comme le montrent les équations² ci-dessous.

$$\begin{cases} d\theta_X = \omega_X dt \\ d\theta_Y = \omega_Y dt \\ d\theta_Z = \omega_Z dt \end{cases}$$

$$\Rightarrow d\theta = d\theta_X^2 + d\theta_Y^2 + d\theta_Z^2$$

$$\begin{cases} dQ_1 = 1 - \frac{d\theta}{8} \\ dQ_2 = \frac{24-d\theta}{48} d\theta_X \\ dQ_3 = \frac{24-d\theta}{48} d\theta_Y \\ dQ_4 = \frac{24-d\theta}{48} d\theta_Z \end{cases}$$

$$\Rightarrow dQ = (dQ_1, dQ_2, dQ_3, dQ_4)$$

Le quaternion au temps t_{n+1} s'obtient en appliquant le produit de quaternions suivant :

$$Q(t_{n+1}) = Q(t_n) dQ$$

Finalement, il ne reste plus qu'à normer le quaternion obtenu :

$$Q(t_{n+1}) = \frac{Q(t_{n+1})}{\|Q(t_{n+1})\|}$$

²Cet algorithme particulier est appelé "méthode d'Edwards du second ordre" et provient du code de MANIAC.

Chapitre 7

Manœuvre avec évitement

L'objectif de ce chapitre est d'exposer comment le satellite évite les zones interdites. Les points intermédiaires nécessaires à l'évitement sont calculés, ainsi que les quaternions permettant d'orienter le satellite vers ces points. La trajectoire finale sera ainsi composée de plusieurs trajectoires simples, évitant au satellite de s'orienter vers des directions dangereuses.

Plusieurs exemples ont été réalisés pour valider le bon fonctionnement du programme. Les résultats de ces exemples se trouvent au chapitre suivant.

7.1 Détection des zones interdites

7.1.1 Direction de visée des instruments

Les zones interdites sont représentées sur la sphère unité à chaque pas de temps, de même que la direction de visée des différents instruments. En effet, pour détecter si un instrument (caméra, senseur stellaire) pointe vers une de ces zones, il est nécessaire de pouvoir également se représenter la trajectoire de cet instrument sur la sphère unité. Pour rappel, la représentation graphique de la sphère unité est décrite au chapitre 4.1.

Grâce aux quaternions d'attitude calculés à chaque pas de temps, il est possible de déterminer l'évolution de l'attitude du satellite durant sa manœuvre et, par conséquent, la direction de visée (\vec{V}) de chaque instrument – et éventuellement des axes de référence du satellite (x, y, z) – sur la sphère unité. En effet, un quaternion décrit l'attitude du satellite à un instant donné, c'est-à-dire la rotation de son repère par rapport au repère inertiel. En appliquant ce quaternion aux directions de visée (vecteurs) des instruments dans le repère satellite (\vec{V}_{sat}), il est possible de les exprimer dans le repère inertiel (\vec{V}_{in}). Ce résultat permet de représenter la trajectoire des instruments sur la sphère unité. Cette méthode de rotation par un quaternion est exposée à la section 11.2.3 des annexes.

La rotation qui amène le repère satellite sur le repère inertiel se traduit par la relation suivante (cf. annexes, 11.2.3) :

$$\vec{V}_{in} = Q_{sat/in} \vec{V}_{sat} \bar{Q}_{sat/in} .$$

7.1.2 Entrée dans une zone interdite

A chaque pas de temps, le programme vérifie qu'aucun instrument n'entre dans une zone qui lui est interdite. Cette vérification s'effectue directement après le calcul exposé dans la section ci-dessus. La position de chaque obstacle est calculée à chaque instant, même si, comme décrit dans le chapitre 2 concernant les hypothèses, les directions du Soleil et de la Lune varient extrêmement peu.

Un test est effectué pour vérifier si aucun des instruments à bord n'est orienté vers une zone dangereuse. Ce test consiste à calculer¹, sur la sphère, l'écart angulaire entre le point courant de chaque instrument et le centre des zones interdites qui le concernent. Si cet écart entre les deux points est inférieur ou égal au rayon de la zone interdite, alors cela signifie que le point courant se trouve à l'intérieur de celle-ci. Un message d'avertissement apparaît alors à l'écran, et le processus d'évitement est aussitôt enclenché.

Le calcul de détection de zone interdite défini ci-dessus est l'étape la plus longue lors de l'exécution du programme. En effet, le modèle de satellite étudié comporte une caméra et trois senseurs stellaires. Il faut donc calculer quatre² rotations avec des quaternions à chaque pas de temps. De plus, le stockage de la trajectoire des instruments peut créer des matrices de taille importante. Pour une manœuvre de 20 minutes par exemple, la taille d'une telle matrice est 1200×3 . Par ailleurs, la caméra ne doit éviter que le Soleil, tandis que chacun des trois senseurs doit éviter les trois astres (Terre, Lune Soleil). **Dix vérifications doivent donc être effectuées à chaque pas de temps.**

Dès qu'une zone interdite est détectée sur la trajectoire d'un instrument, le processus d'évitement est nécessairement enclenché. Le programme est conçu de manière à éviter la zone interdite rencontrée en premier lieu. Par conséquent, il n'est d'aucune utilité de poursuivre la routine de détection. Celle-ci est interrompue afin d'éviter des calculs inutiles et d'obtenir un gain de temps non négligeable, d'autant plus que cette fonction de détection est appelée plusieurs fois lors de la manœuvre.

Toutefois, il est possible de spécifier au programme de calculer la position des instruments jusqu'au bout malgré une entrée dans une zone, ceci par exemple afin de représenter la trajectoire de la manœuvre sans évitement.

¹La formule permettant de calculer un écart angulaire entre deux points sur une sphère est exposée à la section 11.1.2 des annexes.

²Par défaut, l'orientation des axes de référence n'est pas calculée, et n'est par conséquent pas représentée sur la sphère unité.

7.2 Evitement d'une zone interdite

Si le programme détecte qu'une zone interdite est franchie, alors la manœuvre en attitude devient plus complexe car elle doit contourner la zone. Comme cela a été expliqué au chapitre 4.3, la méthode utilisée pour éviter une zone interdite en s'écartant le moins possible de la trajectoire initiale consiste à déterminer un point intermédiaire à l'aide de tangentes. Ce point intermédiaire est appelé point C , les points de départ et d'arrivée de la manœuvre complète étant les points A et B .

Le but de cette section est de déterminer les coordonnées sur la sphère des différents points intermédiaires (points C) lors d'une manœuvre avec évitement. Il est à noter que dans certaines configurations, plusieurs points intermédiaires sont nécessaires. Ces cas particuliers sont traités à la fin de cette section, au point 7.2.5.

7.2.1 Calcul des coordonnées du point C

Le chemin le plus court entre deux points sur une sphère est l'arc de grand cercle, c'est-à-dire la géodésique³. Le problème consiste maintenant à déterminer les géodésiques issues du départ (point A) et de l'arrivée (point B), et tangentes à la zone à éviter (en A_t et B_t). Le point C est à l'intersection de ces deux courbes. Afin de trouver les coordonnées du point C , il semble nécessaire de déterminer les équations des deux géodésiques. Dans la pratique, il est cependant plus simple de calculer les équations des plans (π_A et π_B) contenant ces deux courbes.

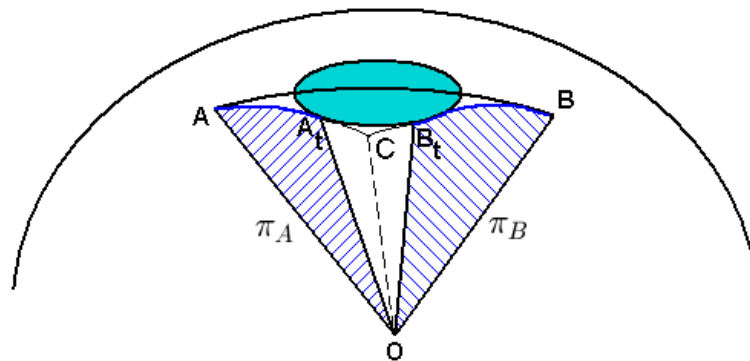


FIG. 7.1 – Représentation des deux plans

Pour définir l'équation d'un plan, il suffit d'en connaître trois points. Dans le cas présent, deux points sont déjà connus :

- le centre de la sphère $O = (X_O, Y_O, Z_O) = (0, 0, 0)$
- le point de départ $A = (X_A, Y_A, Z_A)$ ou d'arrivée $B = (X_B, Y_B, Z_B)$.

Le troisième point à déterminer est le point de tangence $A_t = (X_{A_t}, Y_{A_t}, Z_{A_t})$ en partant du départ, ou $B_t = (X_{B_t}, Y_{B_t}, Z_{B_t})$ en partant de l'arrivée. Le calcul des coordonnées des points de tangence est exposé au point suivant (7.2.2).

³Pour rappel, les grands cercles sont les géodésiques d'une sphère, c'est-à-dire les chemins possédant la plus petite courbure centrés en $(0, 0, 0)$. La longueur d'une géodésique est par ailleurs appelée distance du grand cercle.

Une fois les trois points connus, l'équation du plan s'obtient facilement en calculant un déterminant.

$$\pi_A \equiv \begin{vmatrix} X - X_O & Y - Y_O & Z - Z_O \\ X_A & Y_A & Z_A \\ X_{A_t} & Y_{A_t} & Z_{A_t} \end{vmatrix} = \begin{vmatrix} X & Y & Z \\ X_A & Y_A & Z_A \\ X_{A_t} & Y_{A_t} & Z_{A_t} \end{vmatrix} = 0$$

$$\Leftrightarrow (Y_A Z_{A_t} - Z_A Y_{A_t}) X + (Z_A X_{A_t} - X_A Z_{A_t}) Y + (X_A Y_{A_t} - Y_A X_{A_t}) Z = 0$$

$$\Leftrightarrow \alpha_A X + \beta_A Y + \gamma_A Z = 0$$

$$\pi_B \equiv \begin{vmatrix} X - X_O & Y - Y_O & Z - Z_O \\ X_B & Y_B & Z_B \\ X_{B_t} & Y_{B_t} & Z_{B_t} \end{vmatrix} = \begin{vmatrix} X & Y & Z \\ X_B & Y_B & Z_B \\ X_{B_t} & Y_{B_t} & Z_{B_t} \end{vmatrix} = 0$$

$$\Leftrightarrow (Y_B Z_{B_t} - Z_B Y_{B_t}) X + (Z_B X_{B_t} - X_B Z_{B_t}) Y + (X_B Y_{B_t} - Y_B X_{B_t}) Z = 0$$

$$\Leftrightarrow \alpha_B X + \beta_B Y + \gamma_B Z = 0$$

L'intersection des deux plans donne obligatoirement une droite, car les plans sont non parallèles. Cette droite passe par le centre de la sphère (d'équation $X^2 + Y^2 + Z^2 = 1$) et compte ainsi deux intersections avec celle-ci. Pour trouver ces intersections, il faut résoudre un système de trois équations à trois inconnues.

$$\begin{cases} \alpha_A X + \beta_A Y + \gamma_A Z = 0 \\ \alpha_B X + \beta_B Y + \gamma_B Z = 0 \\ X^2 + Y^2 + Z^2 = 1 \end{cases}$$

Ce système donne deux solutions, puisque les géodésiques tangentes à une zone interdite se croisent en deux points de la sphère. La solution la plus proche du centre de la zone étudiée est le point $C = (X_C, Y_C, Z_C)$. L'autre point (C^*) est à l'opposé de la sphère par rapport à la zone, mais seul le point C est intéressant pour la suite du projet.

7.2.2 Calcul des coordonnées des points de tangence (A_t et B_t)

Comme expliqué au point précédent, les points de tangence A_t et B_t doivent être connus pour pouvoir déterminer le point C . Le calcul menant aux coordonnées des points de tangence fait appel à la géométrie sphérique.

Soit le triangle sphérique rectangle AA_tM où A est le point de départ du chemin, A_t est le point de tangence entre le chemin et la zone à éviter, et M est le centre de la zone. Le contour bleu foncé du schéma suivant représente le coefficient de sécurité.

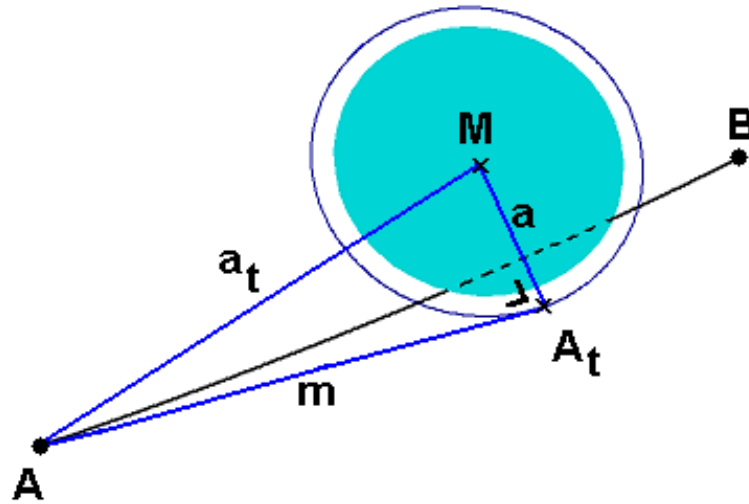


FIG. 7.2 – Recherche du point de tangence A_t (triangle sphérique AA_tM)

Le calcul des coordonnées du point A_t se base sur les équations fournies en annexe (section 11.1). Il se divise en trois étapes.

1. Déterminer les six paramètres (angles) du triangle sphérique, c'est-à-dire trois côtés et trois sommets.
2. Calculer les coordonnées du point tangent dans une configuration précise.
3. Ramener le triangle dans la configuration désirée, en lui appliquant des rotations d'Euler.

Dans le cas présent, deux côtés et un sommet – autre que celui situé entre les deux côtés – sont connus. La première étape consiste donc à déterminer les trois angles restants.

Données :

- a = angle d'ouverture de la zone interdite (multiplié par le coefficient de sécurité),
- a_t = distance angulaire entre le point de départ et le milieu de la zone,
- $A_t = 90^\circ$ (angle au point de tangence).

Inconnues (cf. annexes, section 11.1.3) : A , m , M .

$$\left\{ \begin{array}{l} \sin A = \frac{\sin a \sin A_t}{\sin a_t} \\ \tan \frac{m}{2} = \tan \frac{a_t + a}{2} \frac{\cos \frac{A_t + A}{2}}{\cos \frac{A_t - A}{2}} \\ \tan \frac{M}{2} = \cot \frac{A_t + A}{2} \frac{\cos \frac{a_t - a}{2}}{\cos \frac{a_t + a}{2}} \end{array} \right.$$

Tous les paramètres du triangle sont à présent connus, ce qui permet de passer à la deuxième étape : le calcul des coordonnées du point tangent. Ce calcul est immédiat, à condition de considérer le triangle sphérique dans une configuration particulière. Les points A et M doivent être situés dans le plan OXZ, avec M sur l'axe OZ et A dans le demi-plan $X < 0$ (cf. annexes, section 11.1.2).

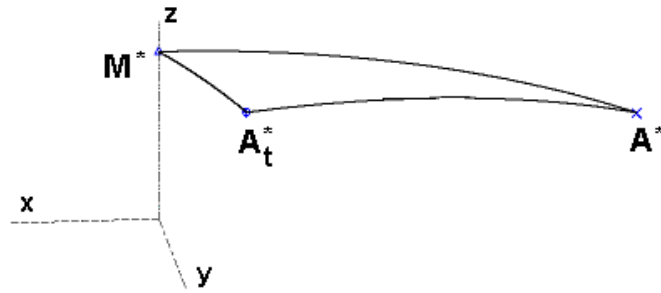


FIG. 7.3 – Triangle sphérique initial

Dans cette configuration (notée *), les coordonnées des points sont :

- $M^* = (0, 0, 1)$
- $A^* = (-\sin a_t, 0, \cos a_t)$
- $A_t^* = (-\sin a \cos M, \sin a \sin M, \cos a)$ ou $A_t^* = (-\sin a \cos M, -\sin a \sin M, \cos a)$

Deux solutions sont obtenues pour le point A_t^* . Ce résultat était attendu, puisque la zone interdite peut être évitée des deux côtés. La figure suivante montre les deux points tangents (A_t) obtenus en partant du point A (en bleu), ainsi que les deux points tangents (B_t) obtenus en partant du point B (en rouge). Le choix et l'association de ces points de tangence sont expliqués au point suivant (7.2.3).

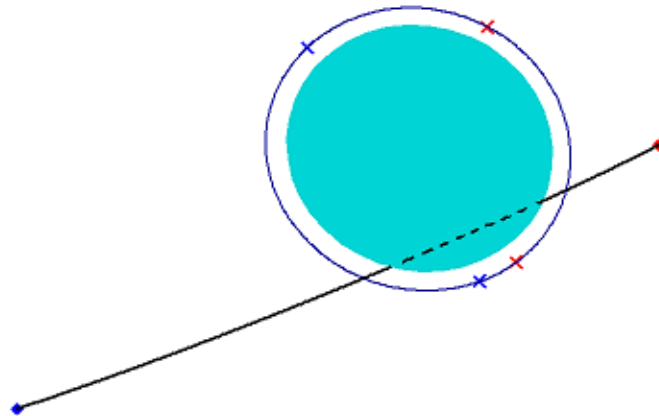


FIG. 7.4 – Points tangents

Maintenant que les coordonnées du point tangent sont connues dans la configuration choisie, la troisième étape consiste à leur appliquer des rotations d'Euler afin de déterminer les coordonnées réelles. Les deux premiers angles de rotation (ψ et θ) s'obtiennent immédiatement à partir des angles de latitude et de longitude du point M , milieu de la zone.

$$\begin{cases} \psi = \arctan \frac{Y_M}{X_M} + \frac{\pi}{2} \\ \theta = \arccos Z_M \end{cases}$$

Après ces deux premières rotations, le point milieu est correctement situé sur la sphère unité, mais l'orientation autour de ce point n'est pas correcte. Cette différence d'orientation

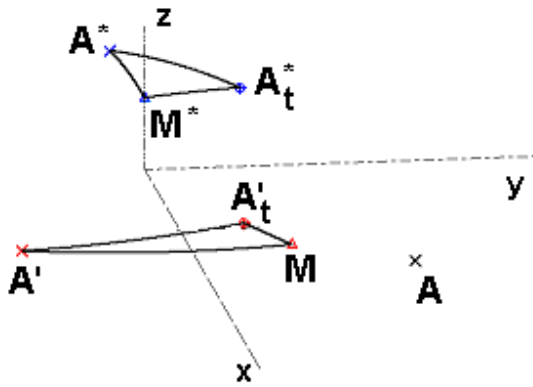


FIG. 7.5 – Triangle $A'A_tM'$ après les deux premières rotations (ψ et θ)

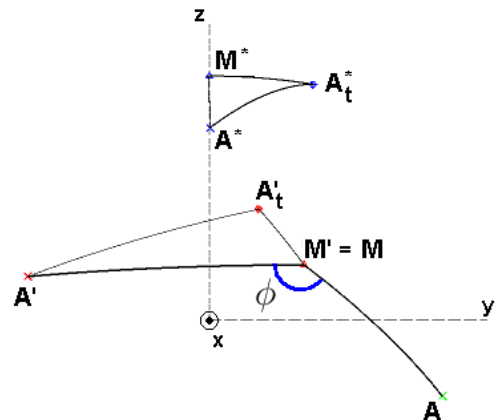


FIG. 7.6 – Figure 7.5 vue de face

correspond au troisième angle (ϕ) de rotation d'Euler. C'est l'angle entre le point de départ (A) réel, et le point intermédiaire (A') trouvé après les deux premières rotations.

Finalement, les coordonnées du point tangent A_t réel s'obtiennent en appliquant les trois rotations d'Euler (ψ , θ et ϕ) au point A_t^* de la configuration initiale. Les deux figures suivantes permettent de visualiser, sur la sphère unité, le triangle dans la configuration initiale $A^*A_t^*M^*$ (en bleu), le triangle intermédiaire $A'A_t'M$ (en rouge) et le triangle sphérique recherché AA_tM (en vert).

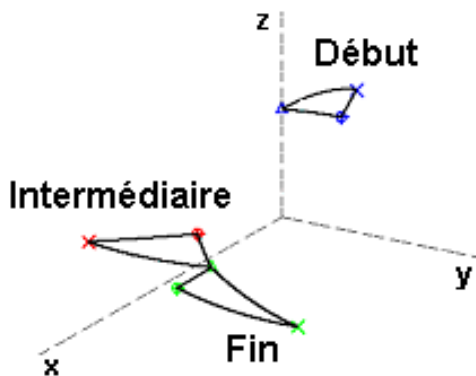


FIG. 7.7 – Triangle sphérique final

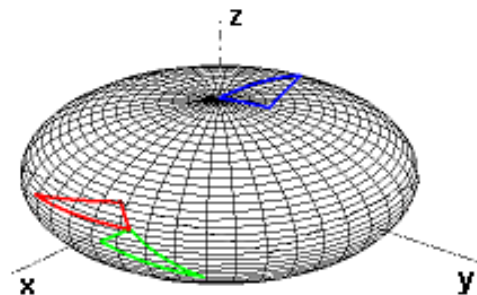


FIG. 7.8 – Vue sur la sphère

Les coordonnées du point de tangence B_t issu du point d'arrivée B se calculent en procédant de la même manière.

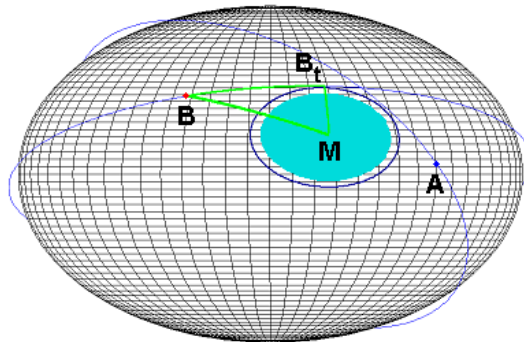


FIG. 7.9 – Triangle sphérique avec le point B

7.2.3 Choix et association des points de tangence (A_t et B_t)

Le calcul du point C, comme expliqué précédemment, dépend entièrement de la position des points de tangence A_t et B_t . Par ailleurs, l'existence de deux points A_t et deux points B_t a également été signalée au point précédent. Par conséquent, il est nécessaire d'associer correctement les points A_t aux points B_t . Dans le cas contraire, l'intersection des plans π_A et π_B donne un point C incohérent.

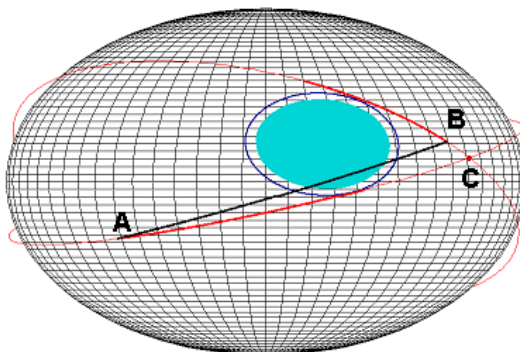


FIG. 7.10 – Mauvaise association des points de tangence

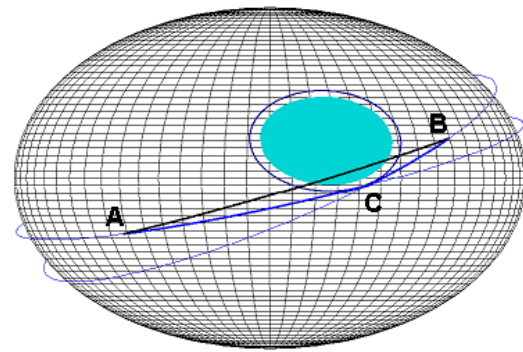


FIG. 7.11 – Bonne association des points de tangence

Le dessin de gauche représente une mauvaise association. En effet, l'intersection des deux géodésiques ne donne pas une solution plausible. Il faut donc changer un des points tangents, ce qui est fait sur le dessin de droite. Cette association est faite en calculant l'écart angulaire entre une extrémité et les deux points tangents issus de l'autre (A avec les B_t et B avec les A_t). Les plus petits écarts sont associés entre eux, et par conséquent, les deux autres points sont également associés pour former l'autre chemin.

Une fois que les points tangents sont bien associés, l'intersection des géodésiques donne deux points C , donc deux chemins possibles pour l'évitement.

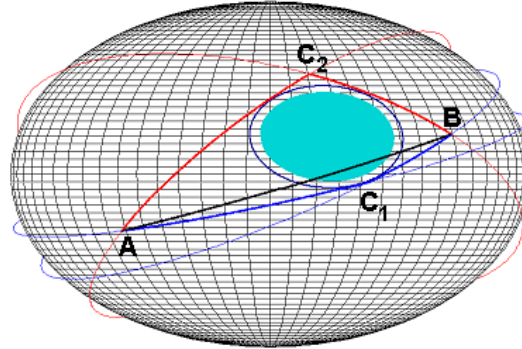


FIG. 7.12 – Les deux chemins possibles pour contourner la zone

Il ne reste plus qu'à comparer les deux chemins d'évitement possibles. Pour cela, le programme fait appel à MANIAC en spécifiant le chemin choisi, et MANIAC renvoie la durée de la manœuvre. Dans le cas général, le programme choisit toujours le chemin le plus rapide. Cependant, dans certains cas plus complexes, plusieurs zones doivent être évitées. Le programme adopte alors une vue plus globale, et il se peut qu'il choisisse d'éviter une des zones en prenant le chemin à priori le plus long. L'évitement de plusieurs zones est exposé à la section 7.4.

7.2.4 Caractéristiques du point C

La méthode de la tangente développée dans ce projet fournit de très bons résultats. Les points intermédiaires (points C) obtenus présentent plusieurs avantages :

1. nombre fini (deux) de solutions,
2. calculés rapidement,
3. toujours en dehors de la zone à éviter,
4. généralement très proche de la zone.

En effet, le point C se situe le plus souvent à une distance négligeable du contour de la zone, en particulier lorsque celle-ci est petite. Par contre, pour une zone plus grande comme celle de la Terre par exemple, ce n'est plus le cas.

7.2.5 Cas particulier

Généralement, le point C est très proche de la zone. Néanmoins, lorsque la zone est grande (la Terre) ou quand les points de départ (A) et d'arrivée (B) sont proches de la zone, le point C risque d'être fort éloigné de celle-ci.

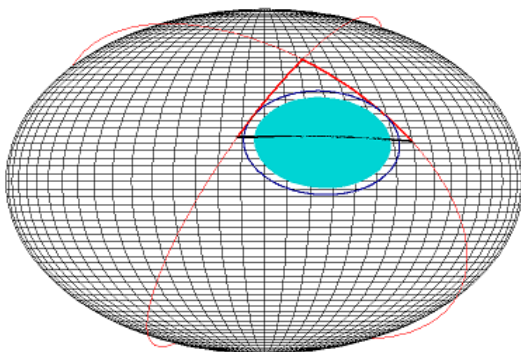


FIG. 7.13 – Point C trop éloigné

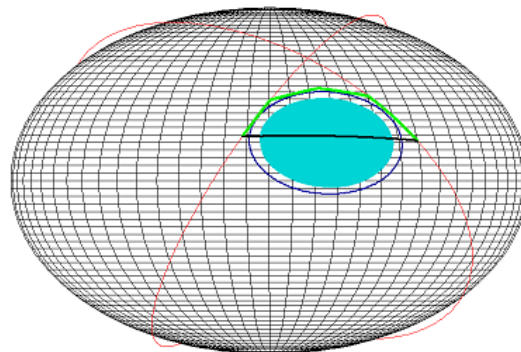


FIG. 7.14 – Contournement rapproché

Si le point C est trop éloigné, le chemin est d'autant plus long et la manœuvre dure trop longtemps. Pour cette raison, le programme va comparer la distance $C - M$ au rayon de la zone. Si le rapport de ces deux distances est supérieur à un coefficient maximum⁴, alors deux nouveaux points intermédiaires sont définis : l'un sur le chemin $A - C$ (point A_2), l'autre sur le chemin $B - C$ (point B_2).

Les points A_2 et B_2 sont illustrés à la figure suivante. A partir de ces deux nouveaux points, le programme détermine le point C_2 qui permet de passer de A_2 à B_2 en évitant la zone interdite.

⁴Le coefficient maximum est fixé à 1,5. Pour rappel, le coefficient minimum (de sécurité) est fixé à 1,15 (cf. section 4.2.3).

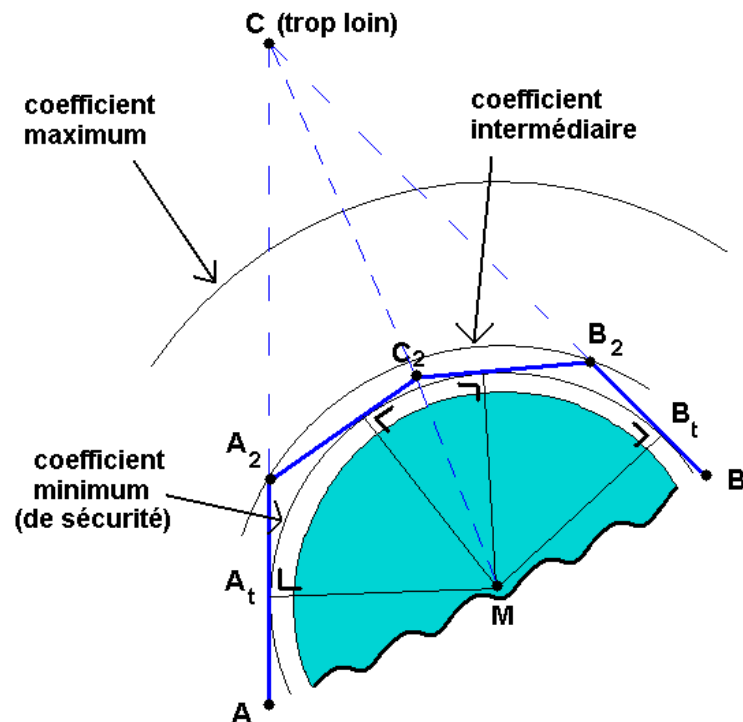


FIG. 7.15 – Cas particulier : plusieurs points intermédiaires

Les points A_2 et B_2 sont situés à une distance égale du centre de la zone. Cette distance correspond au rayon de la zone multiplié par un coefficient appelé coefficient intermédiaire. Celui-ci est calculé à partir des coefficients maximum et minimum. En effet, le coefficient intermédiaire doit être tel que la trajectoire reste toujours convexe, car une trajectoire concave représenterait un détour.

Ce critère permet de majorer le coefficient en se basant sur le cas le plus contraignant, c'est-à-dire lorsque le point C se trouve à la limite de la zone définie par le coefficient maximum. Dans ce cas, le point C_2 est aligné avec A_2 et B_2 , comme l'illustre la figure suivante.

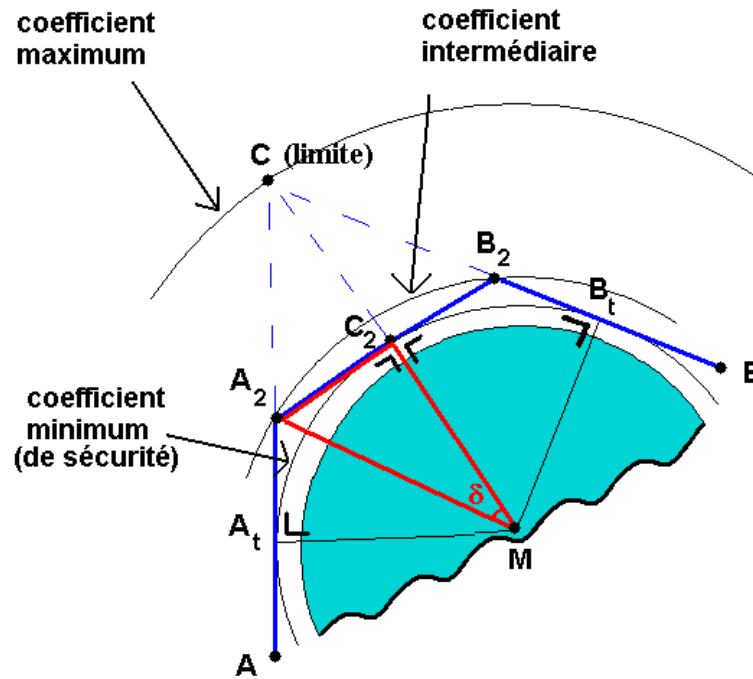


FIG. 7.16 – Cas limite

En prenant, pour simplifier, une zone interdite de rayon unitaire, les différents coefficients sont respectivement égaux aux rayons des cercles auxquels ils correspondent. De cette manière, le coefficient intermédiaire maximum est égal à l'hypoténuse du triangle rectangle dessiné en rouge sur la figure précédente, d'où

$$(coef\ int)_{max} = MA_2 = \frac{coef\ min}{\cos \delta}$$

où δ est la moitié de l'angle $A_t\widehat{M}C$

$$\delta = \frac{\arccos \frac{coef\ min}{coef\ max}}{2} .$$

Pour éviter d'avoir trop de points intermédiaires, ce qui pourrait poser problème à MANIAC⁵, le coefficient intermédiaire est pris le plus grand possible.

$$coef\ int = (coef\ int)_{max} = MA_2 = \frac{coef\ min}{\cos \frac{\arccos \frac{coef\ min}{coef\ max}}{2}}$$

⁵MANIAC a besoin d'une durée minimum de quelques dizaines de secondes pour effectuer une manœuvre.

7.3 Quaternions d'attitude aux points intermédiaires

Les coordonnées des points intermédiaires sur la sphère unité sont maintenant connues. Pour réaliser la manœuvre entière (avec évitement), il va falloir faire appel à MANIAC pour chaque partie de l'évitement, et relier de cette manière les points intermédiaires obtenus à la section précédente.

Cependant, pour réaliser un appel à MANIAC, la connaissance de la trajectoire d'un instrument sur la sphère unité n'est pas suffisante. En effet, MANIAC ne tient pas compte des instruments embarqués. Son unique rôle est de calculer l'évolution cinématique du satellite lors d'un changement d'attitude de celui-ci. Par conséquent, les seules données qu'il est capable de comprendre et de traiter ensuite sont les quaternions d'attitude de départ et d'arrivée d'une manœuvre.

L'objectif de cette section est de définir les quaternions d'attitude aux points intermédiaires nécessaires à MANIAC pour calculer la manœuvre.

7.3.1 Rotation selon une géodésique

Comme cela a été décrit dans la section précédente, l'évitement d'une zone interdite consiste à rejoindre un point intermédiaire (point C) situé à l'intersection des tangentes (géodésiques) issues des points de départ (point A) et d'arrivée (point B). Les quaternions d'attitude du satellite aux points A et B font partie des données à introduire en début de calcul et sont ainsi connus. Le but est donc de déterminer le quaternion d'attitude du point C .

Une première approche est de suivre la trajectoire géodésique pour relier le point C . Le quaternion de rotation le long de la géodésique reliant A à C se calcule facilement. En effet, l'axe de la rotation est la normale au plan AOC (défini à la section 7.2) et l'angle de la rotation est la distance angulaire AC .

Soit $\pi_A \equiv \alpha_A X + \beta_A Y + \gamma_A Z = 0$. Le vecteur normal au plan est défini par ses coefficients, c'est-à-dire $\vec{n}_A = (\alpha_A, \beta_A, \gamma_A)$. L'axe de rotation du quaternion est égal au vecteur \vec{n}_A normé. Le signe de ce vecteur est automatiquement correct de par sa construction (cf. point 7.2.1), puisque la rotation se fait de A vers C .

$$\vec{u}_{AC} = \frac{\vec{n}_A}{\|\vec{n}_A\|}$$

Soit $A = (X_A, Y_A, Z_A)$ et $C = (X_C, Y_C, Z_C)$. L'angle de la rotation est égal à la distance angulaire entre A et C (cf. annexes, section 11.1.2).

$$\theta_{AC} = \arccos(X_A X_C + Y_A Y_C + Z_A Z_C)$$

Le quaternion de la rotation suivant la géodésique vaut :

$$Q_{AC}^T = \begin{pmatrix} \cos \frac{\theta_{AC}}{2} \\ u_{AC} \sin \frac{\theta_{AC}}{2} \\ v_{AC} \sin \frac{\theta_{AC}}{2} \\ w_{AC} \sin \frac{\theta_{AC}}{2} \end{pmatrix}.$$

Finalement, le quaternion d'attitude du satellite lorsque l'instrument est dirigé vers le point C s'obtient par composition de rotation, c'est-à-dire à l'aide d'un produit de

quaternion (cf. annexes, section 11.2.2).

$$Q_C = Q_{AC} Q_A$$

Ce quaternion est appelé par abus de langage "le quaternion d'attitude au point C".

7.3.2 Problème de l'orientation du satellite

La méthode décrite au paragraphe précédent permet de calculer Q_C , le quaternion d'attitude au point C . Cette méthode a pour seul but d'amener l'instrument considéré à contourner la zone interdite. Le satellite passe d'une attitude de départ (définie par l'utilisateur) à une attitude intermédiaire qui minimise la rotation nécessaire à l'évitement.

De cette attitude intermédiaire, le satellite subit ensuite une rotation pour atteindre l'attitude d'arrivée. En procédant de la même manière que pour la première partie du trajet, l'angle de la rotation est minimal en longeant la géodésique. Le quaternion de la rotation suivant la géodésique vaut :

$$Q_{CB}^T = \begin{pmatrix} \cos \frac{\theta_{CB}}{2} \\ u_{CB} \sin \frac{\theta_{CB}}{2} \\ v_{CB} \sin \frac{\theta_{CB}}{2} \\ w_{CB} \sin \frac{\theta_{CB}}{2} \end{pmatrix}.$$

La connaissance de Q_C et de Q_{CB} permet, en principe, de calculer le quaternion au point B .

$$Q_B = Q_{CB} Q_C$$

Cependant, cette méthode fournit un résultat incorrect. En effet, l'attitude d'arrivée du satellite (Q_B) calculée ci-dessus est différente (sauf hasard) de l'attitude de consigne définie par l'utilisateur. L'instrument a bien évité la zone interdite, et pointe vers la bonne direction à l'issue de la manœuvre, mais mis à part cet axe unique, le reste du satellite est mal orienté. Ce résultat est absolument normal. **L'attitude du satellite**, c'est-à-dire son orientation complète dans l'espace, **ne peut pas être déterminée à l'aide d'un seul point de la sphère unité** (cf. chapitre 4.1).

Par conséquent, la rotation ne peut pas se faire suivant une géodésique. Et la méthode pour trouver le quaternion intermédiaire (Q_C) n'est pas applicable.

7.3.3 Calcul du quaternion du point intermédiaire (évitement simple)

Le point précédent a permis de constater qu'en effectuant des rotations suivant la géodésique, le quaternion d'attitude final du satellite ne correspond pas à l'orientation finale désirée. Une autre méthode doit être mise en place, mais en fixant à présent les quaternions aux bords (Q_A et Q_B). L'objectif est de calculer le quaternion du point intermédiaire (Q_C) tel que la trajectoire épouse au mieux le contour de la zone à éviter.

Un premier essai consiste à garder le quaternion Q_C calculé à la section précédente, c'est-à-dire le quaternion d'attitude issu de la rotation suivant la géodésique entre A et C . De cette manière, la première partie de l'évitement ($A - C$) est optimale puisqu'elle suit la géodésique. Par contre, la seconde partie de l'évitement ($C - B$) subit une déviation importante à cause de l'attitude finale requise (Q_B).

En modifiant le quaternion au point intermédiaire (Q_C), il est possible d'améliorer la deuxième partie de l'évitement, au détriment de la première. En effet, modifier Q_C équivaut à modifier le quaternion de rotation (Q_{AC}) qui amène le point A au point C , c'est-à-dire modifier l'orientation de l'axe de rotation dans le plan perpendiculaire à la droite AC . De cette manière, la première partie de l'évitement ne suit plus une géodésique.

Sur la figure suivante sont représentées différentes trajectoires entre A et C , correspondant à différentes inclinaisons de l'axe de rotation. Pour simplifier le dessin, les trajectoires sont projetées sur le plan contenant la droite AC et parallèle à l'axe de la rotation géodésique (en noir). De cette manière, la projection de la géodésique AC est la droite AC .

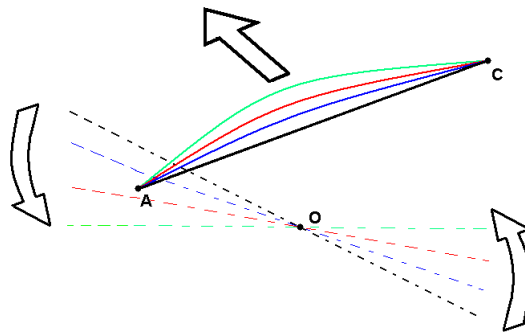


FIG. 7.17 – Différentes trajectoires A – C (projetées sur un plan)

Ce dessin permet de constater qu'il existe une infinité de rotations qui amènent le point A au point C . Une seule de ces rotations (en noir) suit la géodésique. Elle correspond à la trajectoire optimale. Par conséquent, toutes les autres rotations donnent des trajectoires plus longues. Toutes ces rotations sont en fait les compositions de deux rotations : d'une part la rotation suivant la géodésique, et d'autre part une rotation autour de l'axe OC , appelée "rotation propre". La figure suivante illustre cette composition des deux rotations.

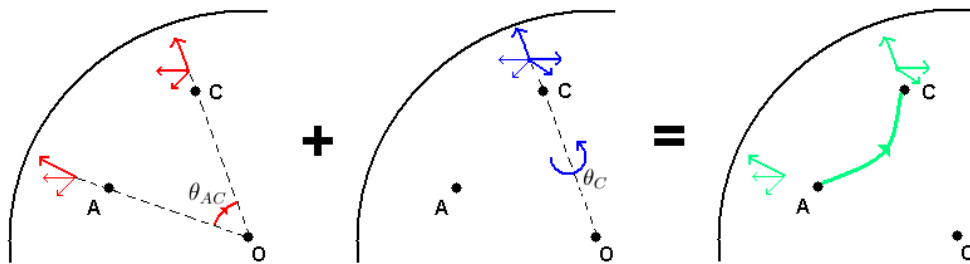


FIG. 7.18 – Composition de quaternions

Finalement, pour définir l'attitude du satellite au point C , il suffit de déterminer la rotation propre qu'il faut lui faire subir, ou plus précisément, l'angle de cette rotation puisque l'axe OC est connu. Il vient alors l'idée de calculer le quaternion du point C tel que la seconde partie de l'évitement ($C - B$) suive la géodésique CB et soit, par conséquent, optimale du point de vue géométrique. Le résultat ainsi obtenu (Q_C^I) peut alors être comparé au résultat obtenu précédemment en optimisant la première partie de l'évitement (Q_C^L).

Pour rappel, Q_C^I se calcule en effectuant le produit de quaternion suivant⁶

$$Q_C^I = Q_{AC} Q_A$$

tel que

$$Q_{AC} = \begin{pmatrix} \cos \frac{\theta_{AC}}{2} \\ u_{AC} \sin \frac{\theta_{AC}}{2} \\ v_{AC} \sin \frac{\theta_{AC}}{2} \\ w_{AC} \sin \frac{\theta_{AC}}{2} \end{pmatrix}$$

avec $\theta_{AC} = \arccos(X_A X_C + Y_A Y_C + Z_A Z_C)$.

Le calcul de Q_C^{II} est similaire, mais en partant de l'orientation finale (Q_B) et en suivant la géodésique BC .

$$Q_C^{II} = Q_{BC} Q_B$$

tel que

$$Q_{BC} = \begin{pmatrix} \cos \frac{\theta_{BC}}{2} \\ u_{AC} \sin \frac{\theta_{BC}}{2} \\ v_{AC} \sin \frac{\theta_{BC}}{2} \\ w_{AC} \sin \frac{\theta_{BC}}{2} \end{pmatrix}$$

avec $\theta_{BC} = \arccos(X_B X_C + Y_B Y_C + Z_B Z_C)$.

La rotation propre entre Q_C^I et Q_C^{II} donne le quaternion de rotation

$$Q_{rot}^{I-II} = Q_C^{II} \overline{Q_C^I}$$

dont les composantes sont $(q_{1\ rot}, q_{2\ rot}, q_{3\ rot}, q_{4\ rot})$.

L'angle de cette rotation s'en extrait aisément :

$$\theta = 2 \arccos q_{1\ rot} .$$

L'axe de cette rotation vaut :

$$\vec{u}_{rot} = \frac{1}{\sin \frac{\theta}{2}} (q_{2\ rot}, q_{3\ rot}, q_{4\ rot}) .$$

Cet axe, une fois normé, est égal soit au vecteur \vec{u}_{OC} , soit à ce même vecteur changé de signe ($-\vec{u}_{OC}$). En effet, la rotation s'effectue autour de l'axe OC , dans un sens ou dans l'autre. La validation du choix du signe de la rotation est exposée dans les résultats, au chapitre 8.

La figure suivante permet de visualiser l'angle θ de la rotation propre. Les repères représentent l'orientation du satellite. Le trajet en rouge correspond au calcul de Q_C^I , le trajet en bleu à celui de Q_C^{II} .

⁶voir annexes section 11.2

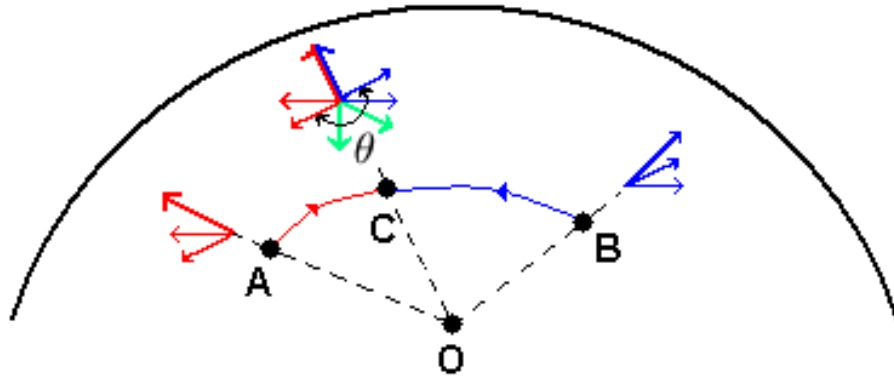


FIG. 7.19 – Angle θ de la rotation propre

Il reste à trouver un compromis entre les deux orientations, une sorte d'orientation intermédiaire. Celle-ci est représentée en vert sur le dessin précédent. Cette attitude du satellite correspond au quaternion du point C dont le calcul est l'objet de ce chapitre. Déterminer ce quaternion revient finalement à déterminer l'angle θ^I tel que $\theta^I + \theta^{II} = \theta$. Voici un zoom de la figure précédente sur lequel les angles θ^I et θ^{II} sont représentés.

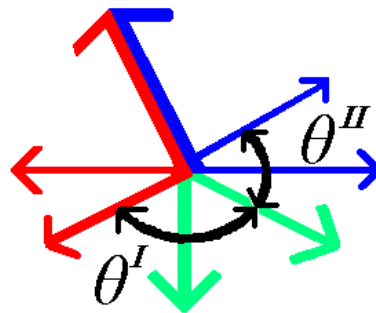


FIG. 7.20 – Zoom sur la rotation propre

L'angle θ^I est une fraction de l'angle θ de la rotation propre

$$\theta^I = \theta - \theta^{II} = \eta \cdot \theta$$

où η est le facteur de pondération. Ce facteur dépend de la position du point C par rapport aux points A et B .

$$\eta = \frac{\theta_{AC}}{\theta_{AC} + \theta_{BC}}$$

Le quaternion de la rotation propre à appliquer au point C , d'angle θ^I et d'axe OC ou CO (suivant le résultat obtenu précédemment), vaut par conséquent :

$$Q_{rot C} = \begin{pmatrix} \cos \frac{\theta^I}{2} \\ \pm u_{OC} \sin \frac{\theta^I}{2} \\ \pm v_{OC} \sin \frac{\theta^I}{2} \\ \pm w_{OC} \sin \frac{\theta^I}{2} \end{pmatrix} = \begin{pmatrix} \cos \frac{\theta^I}{2} \\ \pm X_C \sin \frac{\theta^I}{2} \\ \pm Y_C \sin \frac{\theta^I}{2} \\ \pm Z_C \sin \frac{\theta^I}{2} \end{pmatrix}.$$

Finalement, le quaternion d'attitude au point intermédiaire (Q_C) s'obtient, à partir du quaternion initial de la manœuvre (Q_A), en combinant deux rotations : la première (Q_{AC}) le long de la géodésique AC et la seconde ($Q_{rot C}$) autour de l'axe OC .

$$Q_C = Q_{rot C} Q_{AC} Q_A$$

Un exemple d'évitement est représenté sur les figures suivantes, et comparé à la solution obtenue sans rotation propre autour de OC , c'est-à-dire en suivant la géodésique AC et en rejoignant ensuite l'attitude finale de la manœuvre (Q_B). Cet exemple permet de constater que la déviation dans la seconde partie de l'évitement est corrigée au détriment de la première partie.

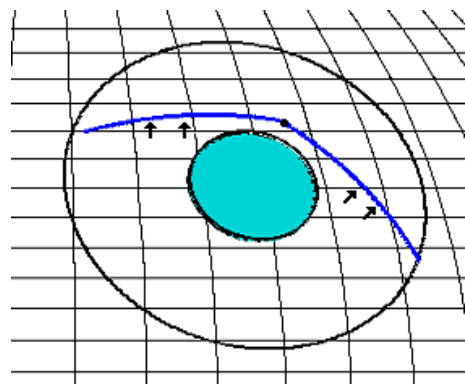
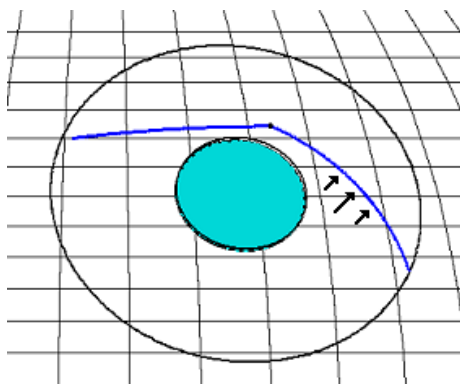


FIG. 7.21 – Evitement sans rotation propre FIG. 7.22 – Evitement avec rotation propre

De plus, les autres instruments (ici les senseurs en rouge) suivent une trajectoire plus optimale avec la correction.

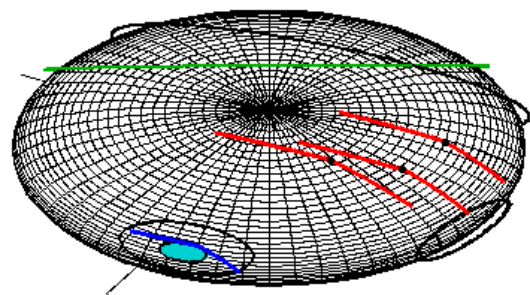
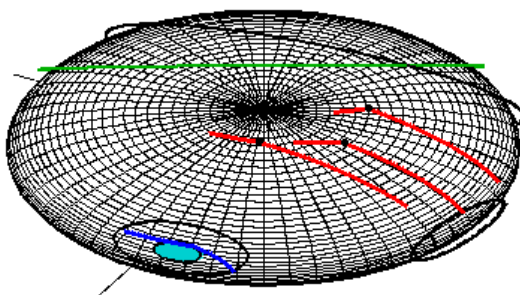


FIG. 7.23 – Trajectoire des senseurs (sans cor- FIG. 7.24 – Trajectoire des senseurs (avec cor-
rection) rection)

7.3.4 Calcul du quaternion de plusieurs points intermédiaires

Dans certains cas particuliers (point 7.2.5), le point C est trop éloigné de la zone à éviter. Dans ce cas, deux points intermédiaires (A_2 et B_2) sont calculés. A partir de ces deux points, un troisième point intermédiaire (C_2) est obtenu par la méthode des tangentes. Si ce dernier est toujours trop loin, la procédure est répétée et deux nouveaux points intermédiaires sont trouvés. Ce processus est récursif jusqu'à obtenir un point C_n suffisamment proche de la zone.

Le quaternion (Q_{C_n}) du point C_n se calcule comme celui du point C , c'est-à-dire en ajoutant une rotation propre (cf. point précédent). Cette rotation se calcule en comparant les quaternions issus des rotations géodésiques $A_n C_n$ et $B_n C_n$. A chaque itération n , il est donc nécessaire de définir en premier lieu les quaternions Q_{A_n} et Q_{B_n} .

Pour calculer Q_{A_n} , la méthode utilisée est toujours la même. Tout d'abord, il faut définir $Q_{A_{n-1}A_n}$ et $Q_{B_{n-1}A_n}$, et calculer $Q_{A_n}^I$ et $Q_{A_n}^{II}$.

$$\begin{cases} Q_{A_n}^I = Q_{A_{n-1}A_n} Q_{A_{n-1}} \\ Q_{A_n}^{II} = Q_{B_{n-1}A_n} Q_{B_{n-1}} \end{cases}$$

Ensuite, le quaternion de la rotation propre totale est calculé comme précédemment

$$Q_{rot}^{I-II} = Q_{A_n}^{II} \overline{Q_{A_n}^I}$$

ce qui permet d'en extraire l'angle $\theta = 2 \arccos q_1 \text{ rot}$.

L'angle de la rotation propre ($Q_{rot A_n}$) à appliquer au point A_n vaut

$$\theta^I = \eta \cdot \theta$$

$$\text{avec } \eta = \frac{\theta_{A_{n-1}A_n}}{\theta_{A_{n-1}A_n} + \theta_{B_{n-1}A_n}} .$$

Finalement, le quaternion intermédiaire (Q_{A_n}) du point A_n s'obtient en combinant deux rotations : la première ($Q_{A_{n-1}A_n}$) le long de la géodésique $A_{n-1}A_n$ et la seconde ($Q_{rot A_n}$) autour de l'axe OA_n .

$$Q_{A_n} = Q_{rot A_n} Q_{A_{n-1}A_n} Q_{A_{n-1}}$$

Le quaternion Q_{B_n} se calcule similairement par symétrie.

Des exemples d'évitement à plusieurs points intermédiaires sont illustrés au chapitre 8.

7.4 Evitement de plusieurs zones

Maintenant que l'évitement d'une seule zone a été détaillé, il peut être répété de manière récursive pour en éviter plusieurs. En effet, il est possible que lors d'une manœuvre d'évitement, une nouvelle entrée dans une zone interdite se produise. Le processus d'évitement de zone interdite doit alors être relancé.

Le programme gère ces évitements successifs sous forme d'un "arbre". En effet, si le premier évitement ne suffit pas, deux possibilités sont envisageables : soit recommencer l'évitement en contournant la zone interdite par l'autre côté, soit persévérer sur le même chemin, et éviter la nouvelle zone rencontrée. C'est ce dernier choix qui est effectué en priorité, et ce par souci de facilité de programmation et de possibilité de comparaison future avec les autres chemins (cf. section suivante).

Pour rappel, la trajectoire de l'instrument suite à un évitement est composée d'au moins deux segments. Lorsqu'une nouvelle zone interdite est traversée, il est nécessaire de détecter dans quel segment la trajectoire de l'instrument la rencontre.

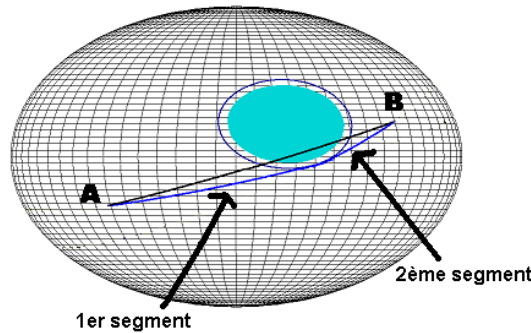


FIG. 7.25 – Segments de la trajectoire

Si la détection survient lors du premier segment, alors les conditions aux limites ne changent pas, et la manœuvre part de nouveau du même point (dessin de gauche – manœuvre complète en vert), en évitant la nouvelle zone. Par contre, si la détection a lieu dans le deuxième segment, l'attitude initiale de l'évitement sera celle du point de départ du segment (dessin de droite – manœuvre complète en noir puis en vert). L'attitude finale, quant à elle, reste la même dans tous les cas.

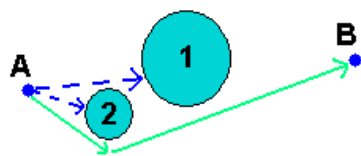


FIG. 7.26 – Premier segment

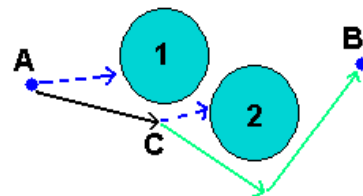


FIG. 7.27 – Deuxième segment

Il y a un nombre maximum de zones pouvant être évitées. Pour la plupart des essais, ce maximum est fixé à trois ce qui est un bon compromis entre la rapidité de calcul, et la qualité du chemin trouvé. L'utilisateur peut toutefois modifier ce nombre s'il le désire.

Si le maximum d'évitements successifs est atteint et qu'il reste néanmoins une zone à éviter, le programme revient sur la dernière zone évitée, et tente de la contourner par

l'autre côté, c'est-à-dire en prenant cette fois-ci le chemin le plus long. Il peut également arriver que lors d'un évitement ultérieur, un instrument retourne dans une zone interdite qu'il a déjà évitée. Dans ce cas également, l'évitement en cours est relancé en contournant la zone par l'autre côté.

Exemple

Dans un premier temps, la caméra évite le Soleil et lors de cet évitement, un senseur entre dans le champ de la Terre. Dans un second temps, le senseur évite à son tour la zone de la Terre, mais cela provoque le retour de la caméra dans la zone du Soleil. Pour éviter ce problème, le senseur doit éviter la Terre par l'autre côté, c'est-à-dire le plus long.

Les deux schémas suivants illustrent le concept de l'arbre. Ils ne représentent pas la trajectoire réelle d'un seul instrument, mais bien une image de l'évitement global, composé de plusieurs instruments, et donc plusieurs trajectoires sur la sphère unité. Chaque évitement ne concerne pas nécessairement le même instrument. Par ailleurs, une même zone interdite peut intervenir plusieurs fois.

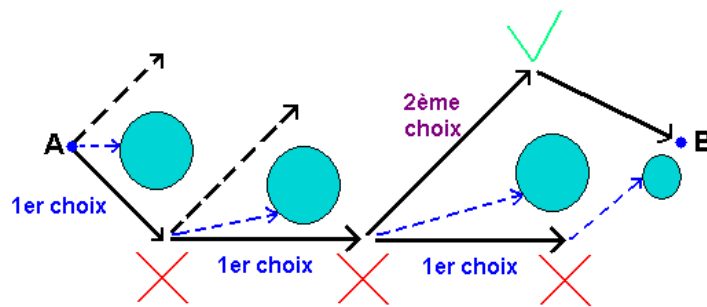


FIG. 7.28 – Arbre des trajectoires

Le deuxième choix peut également créer des problèmes d'évitement. Il y a alors deux cas de figure à distinguer.

1. Si le nombre d'étages maximum est atteint (après le chemin 4 sur le schéma), le programme "remonte" l'arbre d'un étage, et opte pour le deuxième choix à l'évitement précédent (chemin 5).
2. Si le programme est déjà "remonté" d'un ou plusieurs étages, il "redescend" l'arbre comme initialement (après le chemin 5) jusqu'à trouver une solution (chemin 7).

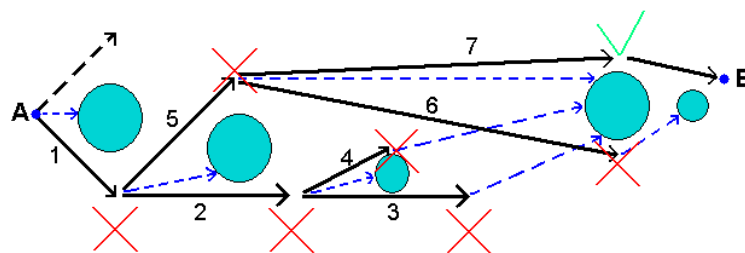


FIG. 7.29 – Arbre complexe

Cette récursivité est répétée jusqu'à ce que le programme trouve une solution, ou bien arrive au bout de l'arbre (quatorze chemins) sans solution acceptable. Si c'est le cas, la manœuvre est alors impossible à effectuer par le programme.

Pour conclure, un exemple réel est présenté dans les résultats (chapitre 8).

7.5 Choix de la trajectoire la plus rapide

Comme cela a été expliqué à la section précédente, l'arbre peut contenir jusqu'à quatorze chemins différents. Plusieurs d'entre eux peuvent être des solutions acceptables. Par ailleurs, la première solution trouvée par le programme n'est peut-être pas la plus rapide. En effet, si plusieurs évitements successifs ont déjà été effectués, la manœuvre devient complexe et la durée de changement d'attitude croît significativement.

Le schéma de la figure 7.29, à la section précédente, permet de visualiser ce phénomène. Sur ce schéma, les six premiers chemins rencontrés ne sont pas acceptables. Seul le septième chemin mène à une solution. Cependant, l'autre moitié de l'arbre (non représentée) peut également mener à des solutions, pouvant même s'avérer plus courtes.

A chaque étape de la manœuvre, le choix de contourner une zone par un côté plutôt qu'un autre dépend uniquement de la durée de la manœuvre à l'étape considérée. Le programme opte simplement pour le chemin le plus rapide, sans se demander si la trajectoire obtenue traverse ou non une nouvelle zone interdite. La durée du chemin "alternatif" (le plus long) est néanmoins stockée.

Le schéma suivant illustre le début de l'exemple exposé précédemment. A ce stade, les zones grisées n'ont pas encore été détectées, et c'est en toute logique que le programme choisit la trajectoire la plus rapide (celle du bas). La durée de celle-ci est appelée "temps 1", tandis que celle du chemin alternatif est appelée "temps 2".

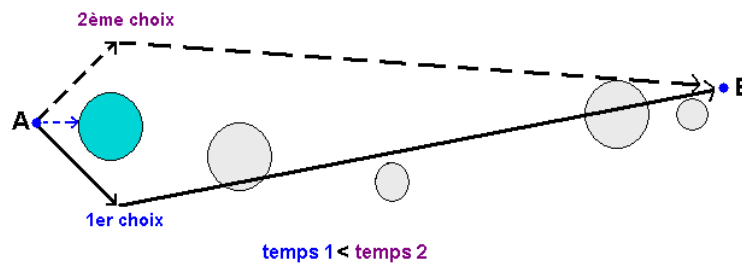


FIG. 7.30 – Premier évitement

Cependant, la solution obtenue en empruntant le premier choix est nettement plus complexe (cf. figure suivante). Par conséquent, la durée de la solution finale peut très bien être plus longue que la durée du chemin alternatif (temps 2). Dans ce cas, le programme relance la fonction de détection des zones interdites en optant pour le deuxième choix.

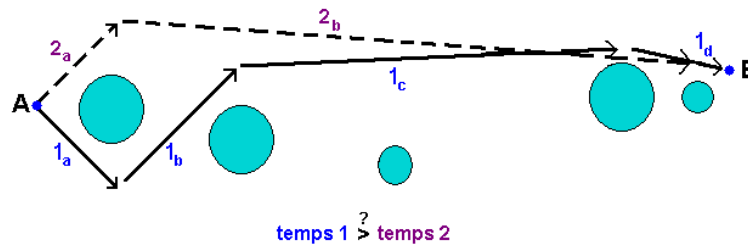


FIG. 7.31 – Comparaison temporelle

Si le deuxième choix mène à une solution acceptable, alors la solution trouvée précédemment est remplacée par ce chemin alternatif puisqu'il est plus rapide. Par contre, si une zone est détectée (comme sur le schéma suivant), alors le processus d'évitement s'enclenche à nouveau, menant à une nouvelle solution. Deux possibilités sont alors envisageables. Si la solution obtenue avec le nouveau chemin devient plus longue que la solution obtenue avec le précédent, alors celle-ci est abandonnée et la première solution est validée. Dans le cas contraire, c'est le chemin alternatif qui est gardé.

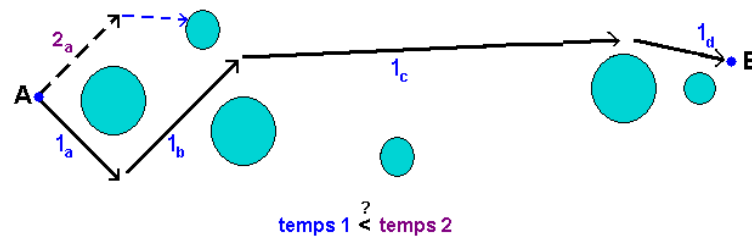


FIG. 7.32 – Nouvelle détection

Une illustration de ce concept est présentée dans les résultats (chapitre 8).

Chapitre 8

Résultats obtenus

Cette dernière section a pour objectif de valider le programme à partir des différents résultats obtenus. Il s'agit de vérifier, au travers d'exemples spécifiques puis généraux, le bon fonctionnement des différentes fonctions informatiques développées, et ce à chaque étape du processus mis en place.

Afin d'effectuer ces tests de validation, il convient de simuler diverses situations pouvant survenir lors d'une manœuvre en attitude : position des astres, type de manœuvre à effectuer, etc. Certaines d'entre elles sont plutôt courantes et donc faciles à reproduire virtuellement. Mais d'autres cas plus complexes sont quant à eux difficile à définir. C'est pourquoi, il est nécessaire de forcer ces dernières en modifiant notamment les dates du calcul mais aussi les positions des instruments sur le satellite.

8.1 Evitement simple

La première étape qu'il est nécessaire de valider concerne l'évitement par la méthode des tangentes. Il faut ainsi tester les quaternions obtenus pour passer par les points intermédiaires, et vérifier que les trajectoires proposées ne dévient pas trop de celle qui aurait été empruntée sans évitement.

Les tests effectués ont permis de mettre en avant des situations inacceptables. Dans le premier exemple, les trajectoires s'effectuaient sans corriger la rotation propre. La première partie de l'évitement se faisait le long d'une géodésique, tandis que la deuxième modifiait l'entièreté du changement d'orientation imposé pour la manœuvre.

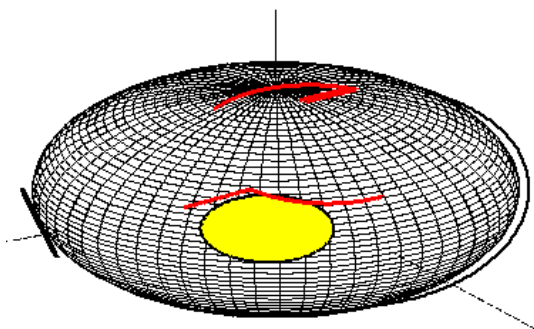


FIG. 8.1 – Trajectoire sans correction de la rotation propre

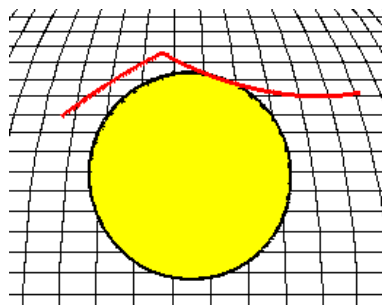


FIG. 8.2 – Zoom sur l'évitement

En effectuant la manœuvre dans le sens inverse, les différences se distinguaient aisément : alors que les points de départ et d'arrivée étaient identiques, les changements d'orientation du satellite lors de chacune des manœuvres d'évitement n'étaient pas équivalents. Ces différences illustrent l'importance de corriger progressivement l'orientation du satellite.

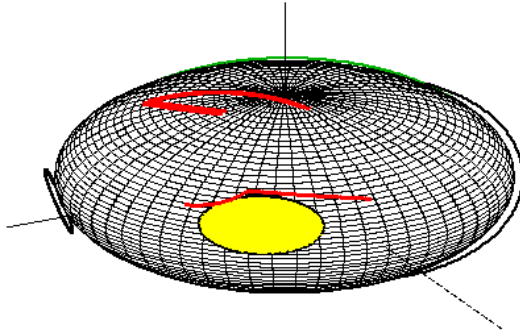


FIG. 8.3 – Trajectoire retour sans correction de la rotation propre

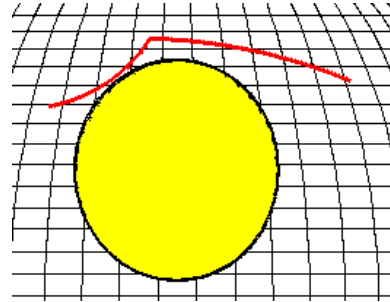


FIG. 8.4 – Zoom sur l'évitement

Dans l'exemple suivant, la rotation propre est répartie sur les deux segments de l'évitement, comme exposé à la section 7.3.3. La manœuvre correctement effectuée est présentée ci-dessous. Les trajectoires des senseurs (en rouge), celui qui évite la Lune ainsi que les deux autres (presque confondus) situé au sommet du satellite, présentent des similitudes au niveau de la courbure. La trajectoire résultante est bien plus courte que celles fournies précédemment et peut être considérée comme optimale pour le cas étudié.

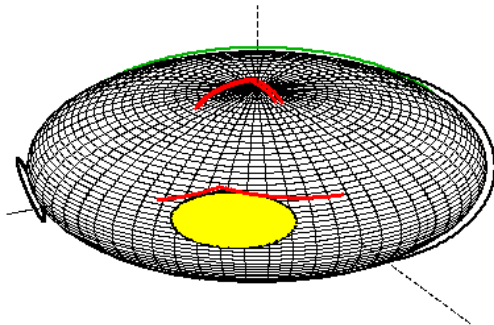


FIG. 8.5 – Trajectoire correcte avec correction de la rotation propre

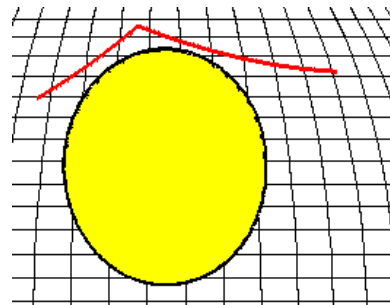


FIG. 8.6 – Zoom sur l'évitement

Le dernier test concernant l'évitement simple consiste à vérifier si le sens de la rotation lors des trajectoires d'évitement est correct. Ces deux images montrent l'évitement obtenu lorsque il y a une erreur de sens. La manœuvre, dans ce cas, n'est pas optimale puisque le capteur se rapproche et même traverse à nouveau la zone interdite qu'il est censé éviter. Le programme a donc été conçu de façon à éviter cette situation inacceptable (cf. 7.3.3).

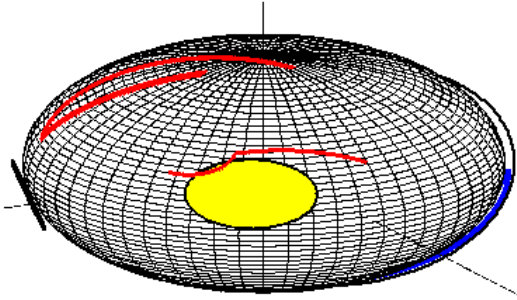


FIG. 8.7 – Sens de la rotation inversé

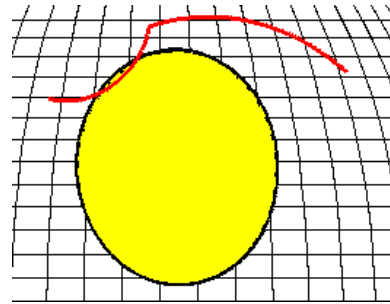


FIG. 8.8 – Zoom sur l'évitement

8.2 Cas généraux

Ces premières validations ont été effectuées avec un seul instrument (un senseur stellaire) devant éviter un même astre (la Lune). L'étape suivante est donc de vérifier cet évitement avec des cas plus généraux. Il s'agit de s'assurer que le processus d'évitement, défini plus haut pour un senseur stellaire et la Lune, est également valable pour les autres instruments et les autres astres. Pour ce faire, il s'agit de définir différentes configurations simples de telle sorte que lors d'une manœuvre en attitude, les instruments soient exposés aux zones interdites les concernant. Chacune de ces situations est étudiée séparément et permet de valider le programme. Les résultats obtenus pour ces cas généraux sont ceux escomptés et ne révèlent aucun problème à cette étape dans le code.

Ci-dessous deux exemples, concernant l'évitement du Soleil. L'image de gauche représente un évitement par la caméra, tandis que l'image de droite illustre un évitement par un senseur stellaire.

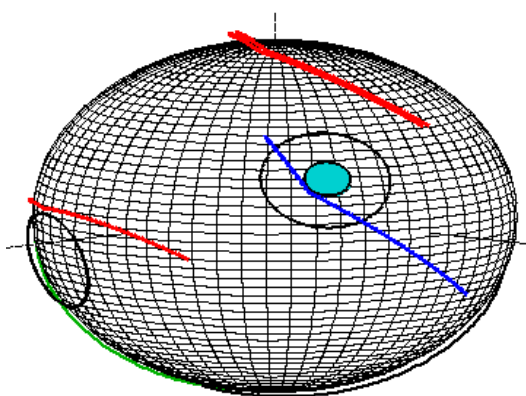


FIG. 8.9 – Evitement caméra – Soleil

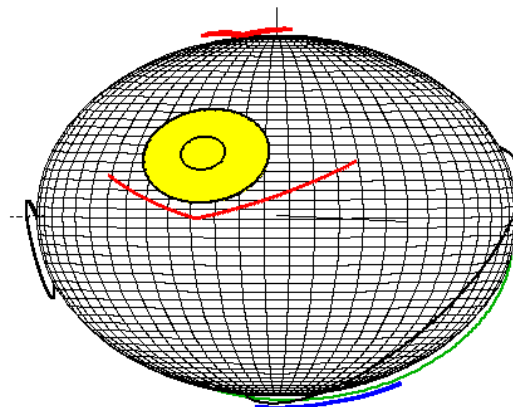


FIG. 8.10 – Evitement senseur – Soleil

Egalement, un exemple simple d'évitement par un senseur stellaire de la Lune (à gauche) et de la Terre (à droite).

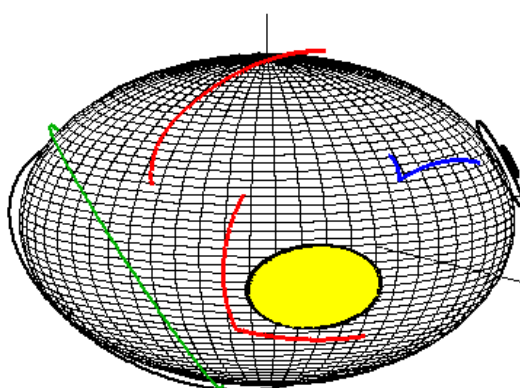


FIG. 8.11 – Evitement senseur – Lune

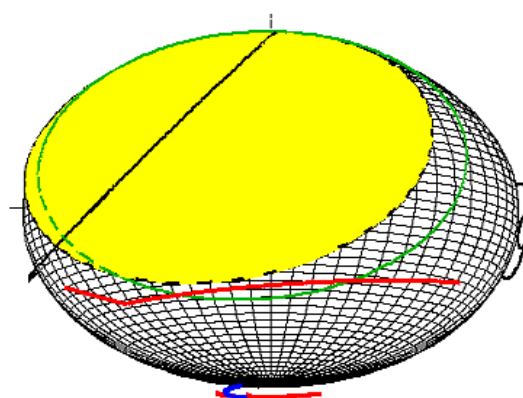


FIG. 8.12 – Evitement senseur – la Terre

Les évitements ainsi obtenus sont basiques, les trajectoires ne sont composées que de deux segments avant d'arriver à l'attitude finale.

8.3 Cas particuliers

Si une majorité des situations rencontrées relève des cas généraux, il est cependant à noter que certaines configurations sont plus particulières à traiter. En effet, comme décrit à la section 7.2.5, il arrive que la trajectoire d'évitement optimale soit composée de plus de deux segments. Les exemples ci-dessous démontrent que le programme gère cette difficulté en proposant une solution acceptable. Il est en effet possible d'observer que le changement d'attitude reste cohérent malgré la multiplication des points intermédiaires.

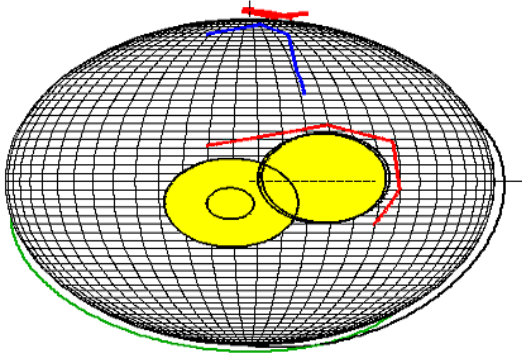


FIG. 8.13 – Contournement de deux zones en 4 segments

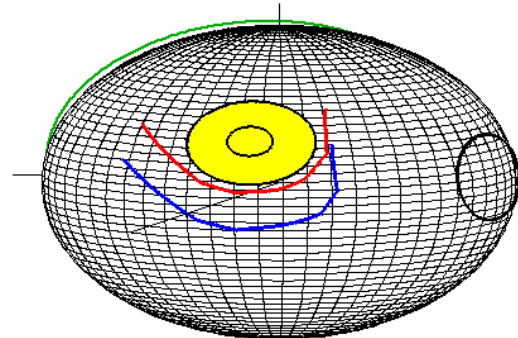


FIG. 8.14 – Contournement en 6 segments

8.4 Validation du processus récursif de l'arbre

Pour vérifier le bon fonctionnement du programme en ce qui concerne l'évitement multiple (cf. section 7.4), il faut s'assurer de la cohérence des résultats proposés au cours du processus récursif de l'arbre. L'enchaînement ci-dessous détaille les étapes intermédiaires de la construction de celui-ci. Etapes qui sont autant de résultats permettant de valider le processus étudié.

Dans un premier temps, la caméra (trait bleu) traverse la zone du Soleil, qui se colore dès lors en bleu clair. De plus, un des senseurs (trait rouge) traverse la zone de la Lune. Celle-ci ne se colore pas car elle n'a pas encore été détectée. En effet, l'intrusion de la caméra dans la zone du soleil est la première à avoir lieu. C'est donc cette zone que le programme évite en priorité. (voir figure 8.15)

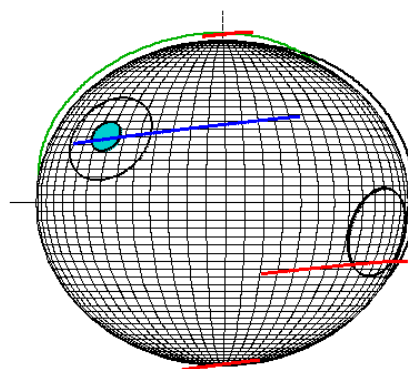


FIG. 8.15 – Camera – Soleil

Le premier évitement s'enclenche, et le satellite effectue sa manœuvre de telle sorte que la caméra ne pointe pas vers le Soleil. Cependant, comme le schéma précédent le laissait déjà supposer, un capteur entre dans la zone de la Lune. Un deuxième évitement est alors nécessaire, pour passer à côté de la zone de la Lune. Comme le premier segment de l'évitement précédent ne présente aucun problème, le deuxième évitement commence au point intermédiaire. (voir figure 8.16)

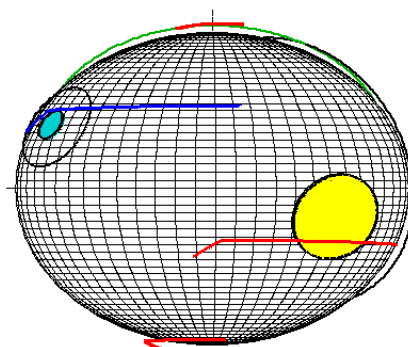


FIG. 8.16 – Senseur – Lune

Lors de ce deuxième évitement, la caméra entre à nouveau dans la zone du Soleil, quel que soit le chemin emprunté pour éviter la Lune. L'évitement par le haut est presque correct, mais la caméra traverse toutefois légèrement la zone interdite. Il faut donc trouver une autre solution pour atteindre l'attitude finale désirée. (voir figures 8.17 et 8.18)

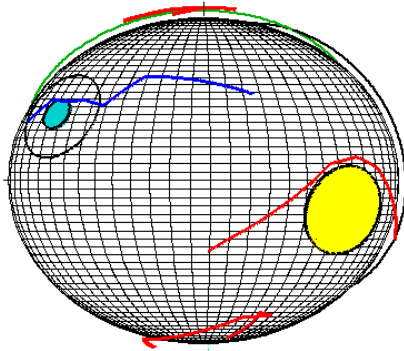


FIG. 8.17 – Evitement par le haut

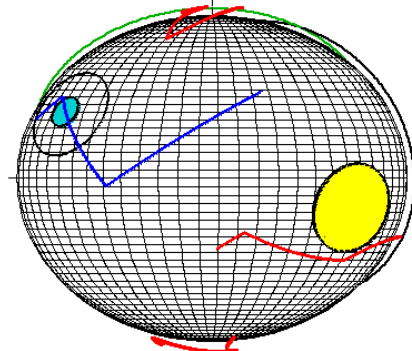


FIG. 8.18 – Evitement par le bas

Dans ce cas, le programme remonte d'un étage et tente de contourner la première zone (du Soleil) par l'autre côté. Mais une fois encore, le senseur rencontre la zone de la Lune. Il faut à nouveau l'éviter. (voir figure 8.19)

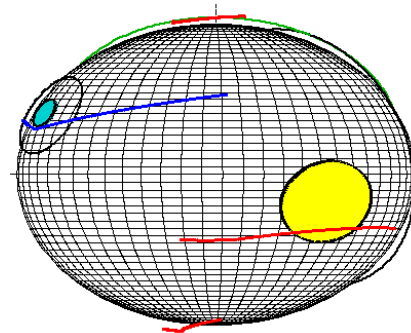


FIG. 8.19 – Camera – Soleil par l'autre côté

Cette fois-ci, l'évitement par le bas donne une solution correcte. Le satellite peut effectuer la manœuvre demandée. (voir figure 8.20)

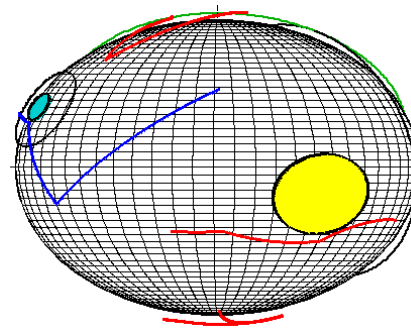


FIG. 8.20 – Solution finale

Ce petit exemple permet d'observer que les résultats proposés par le programme répondent aux objectifs assignés. Il est possible d'obtenir une solution de manœuvre en attitude qui tienne compte des évitements multiples à effectuer pour privilégier une trajectoire optimale.

8.5 Prise en compte de la durée des manœuvres obtenues

Comme expliqué à la section 7.5, le programme a été conçu de manière à privilégier la manœuvre la plus rapide à effectuer. Il convient donc de vérifier au travers des exemples ci-dessous que le programme remplit correctement cette dernière tâche.

Dans un premier temps, le satellite s'est orienté de façon à ce que le senseur, en situation dangereuse, évite la zone interdite (zone jaune, Soleil pour les senseurs). Cependant, lors de cet évitement, la caméra passe devant une zone interdite (zone bleue, Soleil pour la caméra), ce qui provoque un deuxième évitement avant d'arriver à destination. La solution est donc obtenue après deux évitements. (voir figure 8.21)

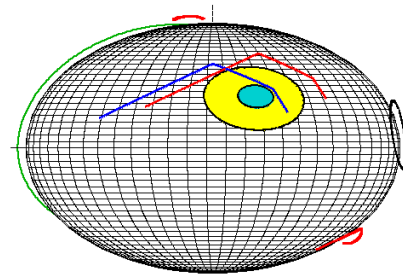


FIG. 8.21 – Première solution

Par contre, en faisant éviter au senseur la zone du Soleil par l'autre côté, la caméra ne traverse plus la petite zone du Soleil qui, par conséquent, ne se colorie pas en bleu. La manœuvre totale ne comporte qu'un seul évitement et est donc plus rapide à effectuer. Le programme décide alors d'opter pour cette trajectoire. Ce résultat permet de valider les calculs effectués pour optimiser la durée du changement d'attitude. (voir figure 8.22)

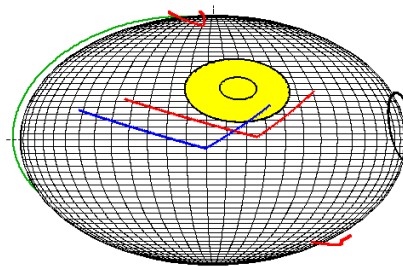


FIG. 8.22 – Deuxième solution, plus rapide

Cependant, il arrive que la comparaison de plusieurs trajectoires possibles démontre que la solution initialement déterminée est effectivement la plus rapide. Dans ce cas, le programme revient à ce premier évitement calculé. La figure suivante, qui illustre ce cas particulier, montre que le programme est conçu de telle manière à ce que le processus récursif de l'arbre soit relancé pour explorer de nouvelles solutions. (voir figure 8.23)

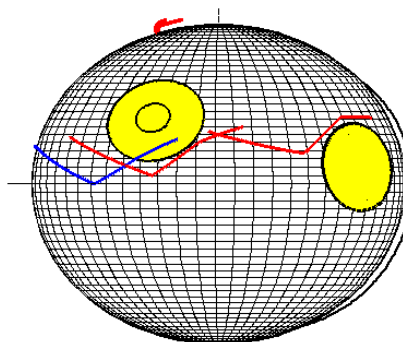


FIG. 8.23 – Deuxième solution, plus lente

Quatrième partie

Conclusion

Au terme de cette recherche, nous sommes parvenus à définir et mettre au point un programme informatique qui permet de calculer les manœuvres en attitude d'un satellite en tenant compte des zones spécifiques vers lesquelles certains instruments doivent éviter de pointer. Cet apport dans le domaine du contrôle d'attitude marque un pas supplémentaire vers l'autonomie des satellites. En effet, le programme défini vise à gérer les changements d'attitude sans intervention extérieure.

Nous avons vu que notre programme est conçu selon une structure qui permet de procéder par étapes successives. Divers calculs, que l'on peut classer en plusieurs phases, s'enchaînent pour évoluer vers une solution finale. Ainsi, en résumé, les premiers calculs servent à déterminer la manœuvre que doit effectuer le satellite sans prendre en considération les zones interdites. Le programme vérifie ensuite si la trajectoire définie ne traverse aucune d'entre elles. Si tel est le cas, il propose la solution sous forme graphique et communique les données de la manœuvre. Si au contraire, cette première trajectoire passe par une zone interdite, le processus d'évitement mis au point est enclenché.

Celui-ci se présente sous forme d'un système récursif. Il s'agit d'abord de déterminer le ou les points intermédiaires selon la méthode des tangentes, et ensuite de calculer la manœuvre qui passe par ces points. Comme précédemment, le programme vérifie que cette nouvelle trajectoire ne traverse aucune autre zone à éviter. Auquel cas, le processus est relancé jusqu'à obtenir une solution. Enfin, une dernière phase de calcul permet de s'assurer que le changement d'attitude trouvé est le plus rapide. Le programme compare la durée de la manœuvre définie avec celle des autres possibilités. Il retient la plus courte et enregistre les quaternions de la manœuvre choisie.

Si ce programme facilite les calculs de changement d'attitude, il présente cependant certaines limites. Nous avons déjà mentionné qu'il a été réalisé avec une version dépassée de MANIAC. A cela, il faut ajouter qu'il n'utilise pas à ce jour de modèle d'orbites plus complet, ni de configuration du satellite plus complexe (capable par exemple d'empêcher une face entière d'un satellite de pointer vers une certaine direction). En outre, des améliorations pourraient encore lui être apportées pour une optimisation plus poussée de l'évitement, ce qui permettrait notamment de détecter les rares cas où le programme ne parvient pas à réaliser l'évitement.

Certaines lacunes de notre travail s'expliquent par la rareté des sources disponibles. Le projet repose principalement sur les recherches que nous avons effectuées au sein du Centre de Documentation du CNES, et sur Internet. Nous nous sommes également tournés vers d'autres agences spatiales mais ces recherches s'y sont révélées infructueuses. L'absence quasi-totale de résultats disponibles s'explique à notre sens par le fait que le sujet que nous abordons ici fait l'objet de recherches tout à fait récentes dans le milieu de la mécanique spatiale.

Le domaine est vaste et de nombreux aspects de notre travail pourraient assurément encore être approfondis davantage que ce que nous avons fait. Ce constat n'enlève toutefois rien à notre satisfaction personnelle d'avoir mené à bien un projet si riche en enseignements théoriques, pratiques et humains au sein de cet environnement professionnel de grande qualité qu'est le CNES.

Bibliographie

- [1] C Programming Tutorial.
<http://cplus.about.com/od/beginnerctutorial/1/blctut.htm>.
- [2] Cours de mécanique spatiale - UNIMECA (Université d'Aix-Marseille).
<http://artemis.univ-mrs.fr/cybermeca/formcont/mecaspa/>.
- [3] Encyclopédie libre Wikipedia.
<http://fr.wikipedia.org>.
- [4] L'intégrale des maths.
<http://membres.lycos.fr/emauvais/idm/GeoAEDistSph.htm>.
- [5] Site du CNES.
<http://www.cnes.fr>.
- [6] Site officiel de scilab.
<http://www.scilab.org>.
- [7] Tutorial de Scilab.
http://cermics.enpc.fr/~jpc/mopsi/the_bouc/.
- [8] Tutorial de Scilab.
http://www.math-info.univ-paris5.fr/~Egk/L1_ECS/node1.html.
- [9] CNES. *Techniques et technologies des véhicules spatiaux*, volume 1 (Généralités et contraintes de développement).
CNES, 2002.
- [10] CNES. *Techniques et technologies des véhicules spatiaux*, volume 3 (Plateformes).
CNES, 2002.
- [11] André Danjon. *Astronomie Générale*.
Librairie Scientifique et Technique Albert Blanchard, 2^{ème} édition, 1986.
- [12] T. Duhamel and P. Marchal. Dynamique et stabilisation d'attitude des satellites.
Technical report,
Ecole Nationale Supérieure de l'Aéronautique et de l'Espace, 1996.
- [13] B. Escudier and J.-Y. Pouillard. Mécanique spatiale. Technical report,
Ecole Nationale Supérieure de l'Aéronautique et de l'Espace, 1997.
- [14] Jack B. Kuipers. *Quaternions and rotation sequences*, chapter 5 (Quaternion Algebra).
Princeton University Press, 1999.
- [15] Shingo Shimoda, Takashi Kubota, and Ichiro Nakatani. Attitude control of two-wheel-satellite using riemannian geometry. In *AAIA Guidance, Navigation and Control Conference and Exhibit*, Août 2004.
- [16] James R. Wertz, editor. *Spacecraft Attitude Determination and Control*.
Kluwer Academic Publishers, 1997.

Cinquième partie

Annexes

Chapitre 9

Présentation de la société

Le CNES, le Centre National (français) d'Etudes Spatiales, est un établissement public à caractère industriel et commercial (EPIC). Acteur majeur de l'Europe spatiale, son rôle est de proposer au gouvernement puis de mettre en œuvre la politique spatiale de la France au sein de l'Europe pour le compte des ministères de la Recherche et de la Défense. Le CNES a été créé en 1961. Aujourd'hui, le budget annuel total de la société s'élève à 1698,6 millions d'euros, incluant la contribution de la France à l'Agence Spatiale Européenne (ESA) à hauteur de 685 millions d'euros.

Le CNES est composé de quatre centres : le siège à Paris, le Centre Spatial de Toulouse (lieu du stage), la Direction des Lanceurs (Evry) et le Centre Spatial Guyanais (Kourou). La société emploie actuellement 2524 salariés, composés en majorité d'ingénieurs et cadres, dont 1771 personnes à Toulouse. Les rôles des différents établissements sont répartis comme suit.

- L'établissement de Paris définit les orientations stratégiques et les programmes prioritaires en s'appuyant sur les autres centres.
- Le Centre spatial d'Evry conduit les développements du lanceur Ariane, pour le compte de l'Agence Spatiale Européenne. Il mène également un travail de prospective sur les nouveaux concepts de lanceurs et les systèmes avancés de propulsion.
- Le Centre Spatial de Toulouse (CST) conduit les projets de systèmes orbitaux. Il développe des systèmes spatiaux complets depuis leur conception jusqu'à leur mise en service opérationnelle. Il travaille en partenariat avec l'industrie et les laboratoires de recherche scientifique. Le CST réalise les opérations de mise et maintien à poste des satellites dont le CNES a la responsabilité.
- Le Centre Spatial Guyanais (CSG) est le port spatial de l'Europe dédié au lanceur Ariane et bientôt aux lanceurs Soyouz et Véga. Il dirige les opérations de préparation finale des charges utiles (de lancement), et coordonne les moyens nécessaires à leur contrôle : poursuite radar, réception et traitement des informations transmises par le lanceur.

Le CNES est l'auteur de différentes missions scientifiques spatiales, notamment les missions *Spot*, *Pléiades* et *Corot*. De plus, il contribue à la plupart des missions ESA¹ comme *Mars Express*, *Venus Express*, *Rosetta* et *GMES*. Il est à noter que le CNES participe également à des missions d'agences étrangères comme *Odin* (Suède) ou à des partenariats comme *Calipso* et *Jason* (NASA²).

Le Centre Spatial de Toulouse est divisé en plusieurs Sous-Directions. Celles-ci sont réparties en deux catégories : les Sous-Directions techniques et celles de Gestion.

Les Sous-Directions techniques comprennent plusieurs services. C'est ainsi que le service Mission et Programmation dans lequel le stage a été effectué dépend de la Sous-Direction Système Bord/Sol (entourée en rouge sur l'organigramme ci-dessous). Ce service s'occupe principalement du domaine du guidage de satellite, mais aussi de quelques aspects missions ou encore de la génération de télécommande (algorithme de guidage qui est envoyé sur le satellite).

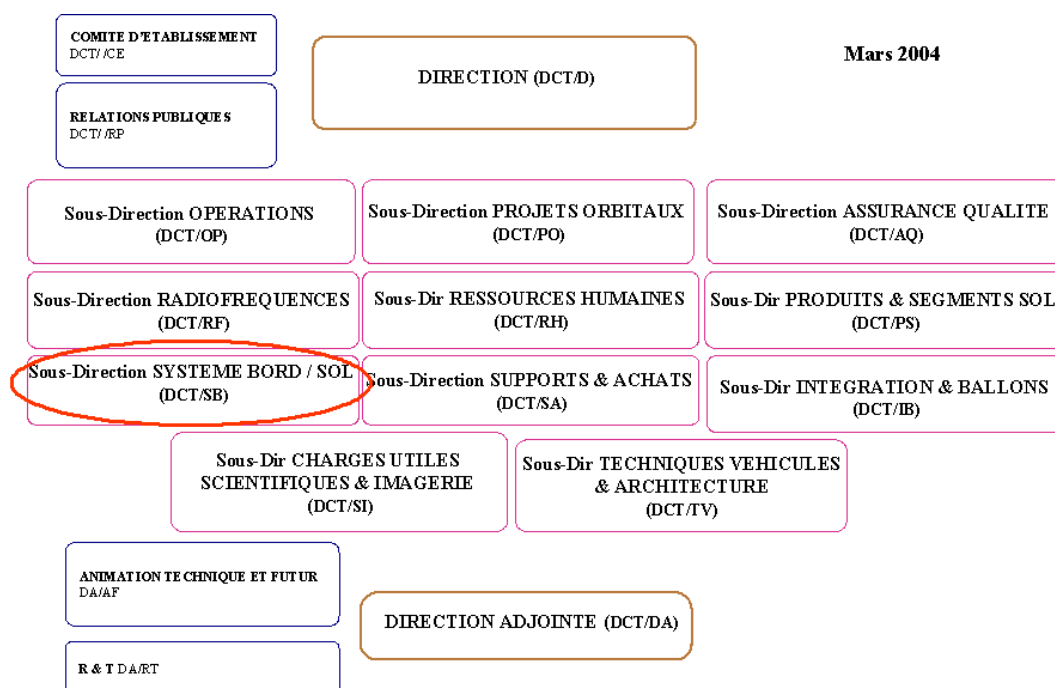


FIG. 9.1 – Organigramme du CST – Source Interagency Operations Advisory Group (IOAG) [5]

¹European Space Agency, agence spatiale européenne.

²National Aeronautics and Space Administration, aux Etats-Unis d'Amérique.

Chapitre 10

Elements de mécanique spatiale

10.1 Notions de mécanique spatiale képlérienne

Cette partie d'introduction a pour but de présenter quelques formules de base de la mécanique spatiale, notamment les concepts utilisés à un moment ou un autre lors du projet.

10.1.1 Les lois de Kepler

Pour exposer les trois lois de Kepler, il faut partir de la relation fondamentale de Newton $\vec{F} = m\vec{a}$, et considérer en première approximation qu'au voisinage de la Terre (comme c'est le cas pour les satellites), l'unique force à prendre en compte est l'attraction terrestre (masse ponctuelle M).

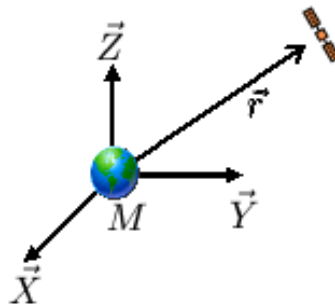


FIG. 10.1 – Force centrale de gravitation

La loi de Newton s'écrit

$$\vec{F} = -\mu \frac{m}{\|\vec{r}\|^3} \vec{r}$$

avec μ la constante de gravitation, qui vaut $\mu = 398600.641 \text{ km}^3 \text{ s}^{-2}$.

Cette force est une force centrale, c'est-à-dire alignée sur le rayon vecteur, d'où $\vec{r} \wedge \ddot{\vec{r}} = \vec{0}$.

Première loi

Elle décrit le mouvement d'un point attiré par une telle force.

Soit le moment cinétique

$$\vec{H} = m\vec{C} \text{ avec } \vec{C} = \vec{r} \wedge \dot{\vec{r}}$$

tel que sa dérivée temporelle est nulle, comme le montre l'équation suivante.

$$\dot{\vec{C}} = \dot{\vec{r}} \wedge \ddot{\vec{r}} + \dot{\vec{r}} \wedge \dot{\vec{r}} = \vec{0}$$

Ceci implique que le moment cinétique est constant dans le temps, et donc le mouvement se fait dans le plan orthogonal à \vec{C} et passant par le point M . Le lieu des points est ainsi une conique dont un des foyers est M , et dont l'équation en coordonnées polaires est :

$$\frac{1}{r} = \frac{1}{p}(1 + e \cos(\theta - \theta_0))$$

avec

- r le rayon de la position courante,
- $(\theta - \theta_0)$ l'angle entre la position courante et le périégée (P sur le dessin),
- $p = \frac{C^2}{\mu}$ le paramètre de la conique,
- e l'excentricité.

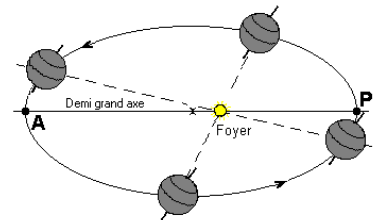


FIG. 10.2 – Orbite

Deuxième loi (loi des aires)

Le vecteur (centre Terre - satellite) balaye des aires égales durant des temps égaux. En effet la vitesse aréolaire (c'est-à-dire la surface balayée par le rayon vecteur \vec{r} par unité de temps) s'écrit (en coordonnées polaires) :

$$\frac{dA}{dt} = \frac{1}{2}r^2\dot{\theta} = \frac{1}{2}\|\vec{C}\| = \text{constante.}$$

La figure 10.3 illustre parfaitement que le satellite accélère jusqu'à vitesse maximale au périégée (gauche $C - D$), puis ralentit jusqu'à vitesse minimale à l'apogée (droite $A - B$).

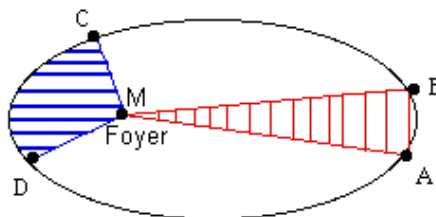


FIG. 10.3 – Loi des aires

Troisième loi

Cette dernière loi permet d’obtenir une relation entre la période (temps de parcours de l’ellipse entière) et son aire. En effet, la surface balayée pendant une période T vaut πab (où a est le demi-grand axe de l’ellipse, et b le demi-petit axe). En partant de la deuxième loi de Kepler, il vient

$$\frac{1}{2} \|\vec{C}\| T = \pi ab = \pi a \sqrt{pa} = \pi a \sqrt{\frac{C^2}{\mu}} a$$

ce qui mène immédiatement à la troisième loi de Kepler.

$$\Rightarrow T = 2\pi \sqrt{\frac{a^3}{\mu}}$$

10.1.2 Les paramètres orbitaux

Le domaine d’application de la mécanique spatiale ne permet pas d’utiliser facilement les équations habituelles pour décrire le mouvement d’un corps dans l’espace. En effet, en exprimant les équations en coordonnées cartésiennes $(x, y, z, \dot{x}, \dot{y}, \dot{z})$, celles-ci deviennent vite extrêmement complexes à résoudre. Pour cette raison, six paramètres orbitaux ont été introduits, d’une part pour simplifier et alléger les équations couramment utilisées, et d’autre part pour permettre une bien meilleure interprétation physique des divers phénomènes.

Forme de l’orbite

Pour décrire la forme de l’orbite, deux paramètres sont utilisés :

– le demi-grand axe a

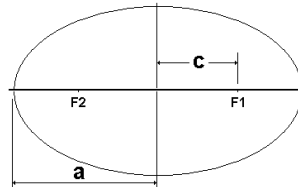


FIG. 10.4 – Demi-grand axe

– l’excentricité $e = \frac{c}{a}$ où c est la distance centre – foyer. Pour une ellipse, l’excentricité vaut $0 \leq e < 1$. Le cas particulier $e = 0$ représente un cercle, comme par exemple les orbites géostationnaires. $e = 1$ représente une parabole (trajectoire des lanceurs) et $e > 1$ une hyperbole (passage dans la zone d’influence d’un astre).

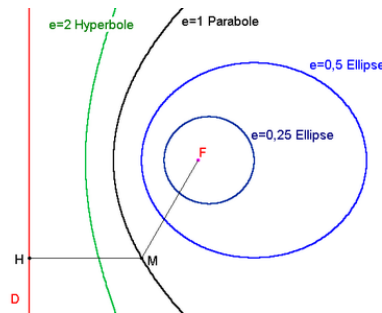


FIG. 10.5 – Excentricité des coniques – Image Wikipedia [3]

Position de l'orbite

Pour exprimer la position de l'orbite dans l'espace, trois nouveaux paramètres sont nécessaires :

- l'inclinaison par rapport à l'équateur i . Les intersections entre cette orbite inclinée et l'équateur s'appellent "nœud ascendant" et "nœud descendant". Le nœud ascendant est celui qui correspond au passage du satellite de l'hémisphère sud à l'hémisphère nord, tandis que le nœud descendant correspond au passage de l'hémisphère nord à l'hémisphère sud,
- l'ascension droite du nœud ascendant Ω . C'est l'angle entre l'axe comprenant le point vernal (voir section 10.3 – "Les repères") et le nœud ascendant,
- l'argument du périhélie ω . Il exprime l'angle entre le nœud ascendant et le périhélie de la trajectoire (point le plus proche du foyer d'où est issu la force centrale, la Terre pour un satellite).

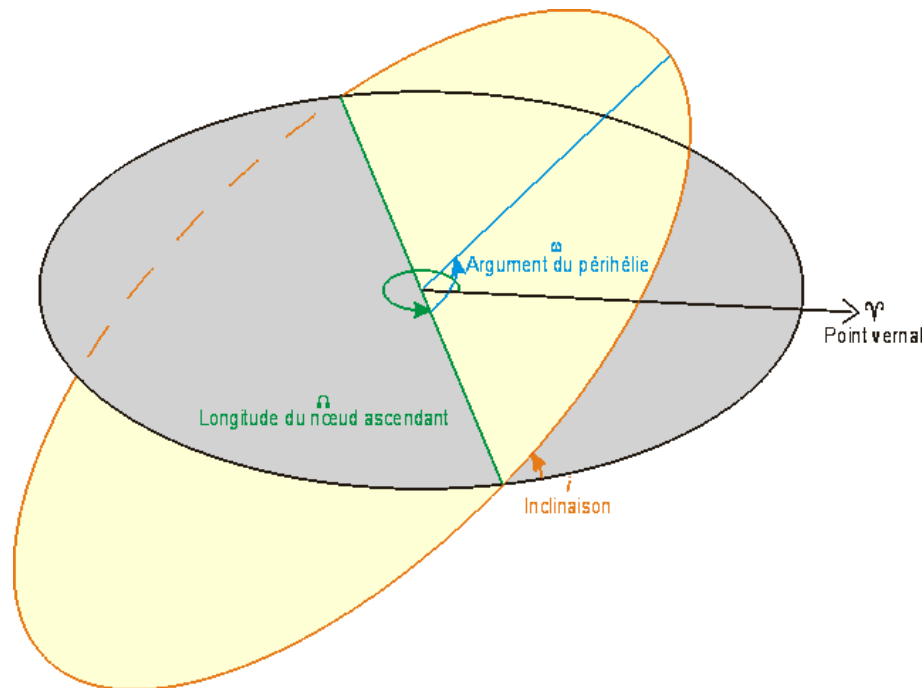


FIG. 10.6 – Paramètres de position de l'orbite – Image Wikipedia [3]

Position du satellite sur l'orbite

Il reste un ultime paramètre à déterminer, qui permet de situer le satellite sur l'orbite. Ce paramètre s'appelle l'anomalie. Il en existe trois expressions différentes : l'anomalie vraie v , l'anomalie excentrique E et l'anomalie moyenne M . Chacune possède des propriétés bien spécifiques. Le choix de l'une d'entre elles dépend du problème considéré et des équations y intervenant.

Anomalie vraie v

L'anomalie vraie correspond aux coordonnées polaires du point sur l'ellipse. C'est l'angle entre la direction du périégée et la position courante du point, mesuré au foyer de l'ellipse. Ce paramètre est surtout utilisé pour le contrôle orbital, mais pas dans les commandes d'attitude. Il n'apparaît donc en aucun endroit dans les calculs.

Anomalie moyenne M

Dans ce projet, il est plus intéressant d'utiliser l'anomalie moyenne car c'est la seule pour laquelle il existe une relation simple avec le temps (et par conséquent la date).

$$M = n(t_c - t_p)$$

avec

- n mouvement moyen, tel que $n = \sqrt{\frac{\mu}{a^3}} = \text{cste}$ (troisième loi de Kepler),
- t_p temps au point de référence (dans le programme, ce point est le périégée),
- t_c temps courant.

Il ne reste plus qu'à ramener M à une valeur comprise entre 0 et 2π en calculant le modulo suivant : $M = \text{modulo}(M, 2\pi)$.

Anomalie excentrique E

Pour calculer les coordonnées cartésiennes de la position du satellite, il est préférable de faire appel à l'anomalie excentrique. Les équations suivantes expriment ces coordonnées en fonction de E pour un satellite situé sur l'orbite de départ (plan $Z = 0$).

$$\begin{cases} X = a(\cos E - e) \\ Y = b \sin E = a\sqrt{1 - e^2} \sin E \\ Z = 0 \end{cases}$$

La figure ci-dessous permet de comparer les trois anomalies, sur une représentation cartésienne de l'orbite. Les anomalies vraie et excentrique ont les mêmes coordonnées en X . La différence est que l'anomaie vraie représente la réelle évolution sur une orbite elliptique, tandis que l'anomalie excentrique représente cette évolution portée sur un cercle de rayon $r = a$. L'anomalie moyenne n'a aucune représentation physique, l'angle parcouru dépend seulement du temps écoulé.

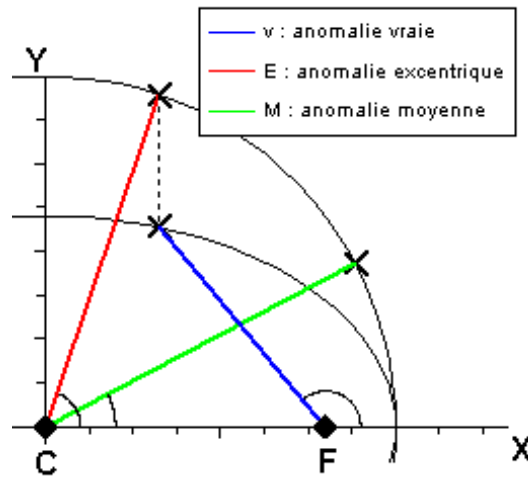


FIG. 10.7 – Comparaison des anomalies

Sur la figure suivante, on constate tout de suite l'intérêt de l'anomalie moyenne M . En effet, elle est définie de sorte à évoluer linéairement dans le temps, au contraire des deux autres qui sont définies géométriquement, et dont l'évolution temporelle n'est pas linéaire.

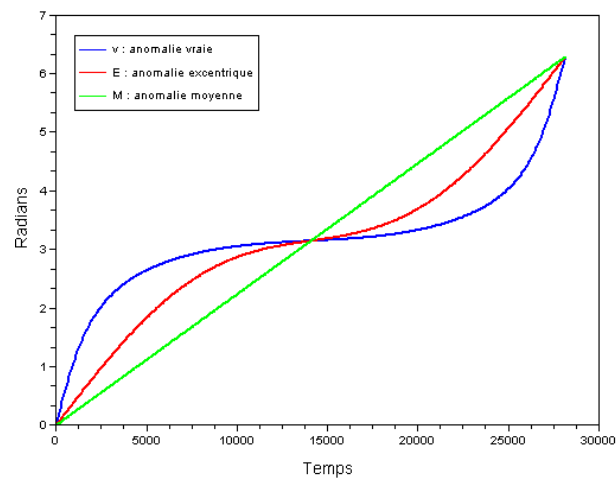


FIG. 10.8 – Evolution temporelle des anomalies sur une période

10.1.3 Quelques formules de mouvement képlérien

Les formules de mécanique spatiale décrites ci-dessous ont également été utilisées dans ce projet. Elles ne sont évidemment valables que pour des coniques de type elliptique ($0 \leq e < 1$).

- rayon vecteur :

$$r = a(1 - e \cos E)$$

- coordonnées cartésiennes :

$$\begin{cases} X = a(\cos E - e) \\ Y = b \sin E = a\sqrt{1 - e^2} \sin E \end{cases}$$

- module de la vitesse :

$$V = \sqrt{\mu\left(\frac{2}{r} - \frac{1}{a}\right)} = \sqrt{\frac{\mu}{a} \frac{1 + e \cos E}{1 - e \cos E}}$$

- période :

$$T = 2\pi\sqrt{\frac{a^3}{\mu}}$$

- relation anomalie excentrique - anomalie moyenne :

$$M = E - e \sin E$$

10.2 Position du satellite - construction de l'orbite

Afin de connaître la position du satellite dans le repère inertiel de référence, centré sur la Terre, il est nécessaire de calculer un modèle d'orbite. Ce travail est basé sur l'hypothèse d'un modèle képlérien (voir chapitre 2), qui ne tient pas compte des différentes perturbations (orbite sans dérive). Par la suite, il sera possible d'intégrer un modèle plus complet si nécessaire.

10.2.1 Utilisation des anomalies

Comme expliqué à la section 10.1.2, plusieurs définitions de l'anomalie existent. Pour situer un point sur une orbite en fonction du temps, deux d'entre elles sont utilisées. Tout d'abord, l'anomalie moyenne M est déterminée, en fonction de la date ($M = n(t_c - t_p)$). Ensuite, afin de connaître les coordonnées cartésiennes du point sur l'orbite, il faut avoir recours à l'anomalie excentrique E . Cependant, la relation qui existe entre M et E n'est pas linéaire : $E = M + e \sin E$.

Il faut donc déterminer E par itération.

$$\begin{cases} E_0 = M \\ E_{n+1} = M + e \sin E_n \end{cases}$$

Ce petit algorithme simple converge très rapidement. Il faut moins de quinze itérations pour atteindre une précision d'ordre 10^{-7} et moins de vingt itérations pour une précision d'ordre 10^{-10} , précision généralement requise pour les calculs de mécanique spatiale.

10.2.2 Orbite de départ dans le plan $Z = 0$

La première étape du calcul d'orbite consiste à déterminer, à chaque pas de temps, la position du point courant sur une orbite elliptique de même excentricité, mais située au niveau de l'équateur (c'est-à-dire dans le plan $Z = 0$) et dont le périhélie est orienté vers l'axe X du repère inertiel. Le calcul du point courant sur cette orbite requiert l'utilisation de trois des six paramètres nécessaires à décrire une orbite : l'excentricité e , le demi-grand axe a et l'anomalie excentrique E calculée à chaque pas de temps via l'anomalie moyenne M . La figure ci-dessous représente cette orbite de départ. Pour ne pas alourdir le dessin, M est considéré comme nul. Le point courant représenté est donc le périhélie de l'orbite étudiée.

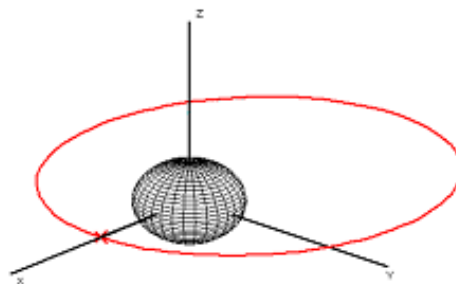


FIG. 10.9 – Orbite dans le plan $Z = 0$

10.2.3 Orbite réelle

L'orbite réelle s'obtient ensuite en appliquant une rotation d'Euler (ψ, θ, ϕ) à l'ensemble des points trouvés, en définissant les trois angles de rotation avec les trois derniers paramètres d'Euler (l'ascension droite du nœud ascendant Ω , l'inclinaison i et l'argument du périégée ω).

$$\begin{cases} \psi = \Omega \\ \theta = i \\ \phi = \omega \end{cases}$$

Les schémas ci-dessous reprennent les rotations une à une. Celui en bas à droite (la vue de face) illustre mieux la rotation θ : les rotations ψ et ϕ se font dans le plan de l'orbite.

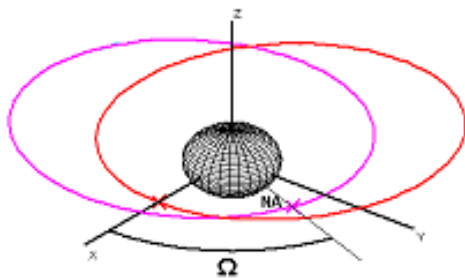


FIG. 10.10 – Rotation ψ vers le nœud ascendant : angle Ω

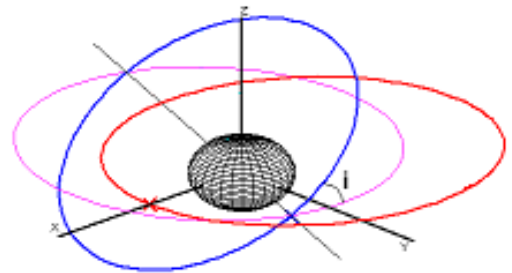


FIG. 10.11 – Rotation θ , changement d'inclinaison de l'orbite : angle i

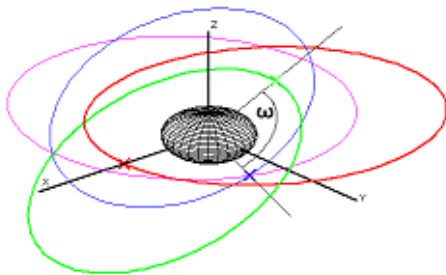


FIG. 10.12 – Rotation ϕ de l'argument du périégée : angle ω

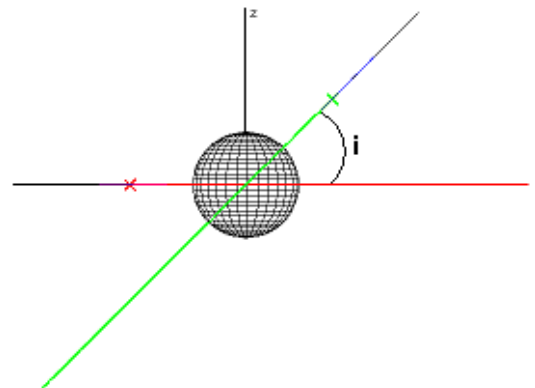
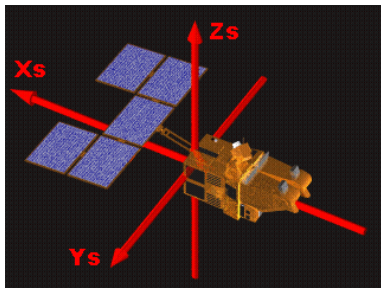


FIG. 10.13 – Vue de face

De cette manière, les coordonnées des points de l'orbite sont connus à tout moment (puisque M dépend de la date) dans le repère inertiel centré sur la Terre.

10.3 Les repères

- **Repère inertiel** : X, Y, Z . Ce repère de référence est fixé au centre d'inertie de la Terre. De plus, il est immobile par rapport aux étoiles. Le méridien "zéro" de ce repère est défini grâce au point vernal γ . Ce point est la position sur la sphère céleste (sphère de rayon quelconque centrée sur la Terre, qui permet de représenter les astres vus depuis la Terre) du Soleil au moment de l'équinoxe de printemps (21 mars). C'est une direction fixe dans le système solaire qui fait office de référence pour le calcul des coordonnées astronomiques. Par conséquent, le seul mouvement de la Terre dans ce repère est sa rotation autour d'elle-même. Le Soleil quant à lui tourne autour de la Terre (centre du repère) à une fréquence d'un tour par an.
- **Repères mobiles** : Il y a deux repères mobiles différents à considérer dans ce projet.



Repère satellite : x, y, z . Il est lié à la structure même du satellite. Les positions des axes sont décidées arbitrairement, généralement afin de faciliter les calculs. Les axes sont alignés sur la caméra principale, sur les panneaux solaires, etc.

FIG. 10.14 – Repère satellite – Image CNES [5]

Repère orbital local^a : x_{ol}, y_{ol}, z_{ol} . Il peut être défini en chaque point de l'orbite notamment par les trois vecteurs unitaires $(\vec{T}, \vec{R}, \vec{L})$, construits^b à partir du vecteur position et du vecteur vitesse du satellite. Le vecteur \vec{L} est parallèle au vecteur position \vec{P} (sur l'axe centre Terre - satellite), il définit l'axe de lacet. Le vecteur \vec{T} est perpendiculaire au plan de l'orbite (\vec{L}, \vec{V}) et définit l'axe de tangage. Le dernier vecteur \vec{R} complète le trièdre. Il appartient au plan de l'orbite et définit l'axe de roulis. Il ne coïncide pas exactement avec le vecteur vitesse à cause de l'excentricité de l'orbite^c.

^aLe repère orbital local ne présente pas un grand intérêt dans ce travail. En effet, ce type de repère est surtout utilisé pour les observations de la Terre ou les manœuvres de calcul d'orbites et non d'attitude.

^bPour définir le repère orbital local, il existe d'autres conventions dans la littérature.

^cDans le cas d'une orbite circulaire, l'excentricité étant nulle, les vecteurs \vec{V} et \vec{R} sont alignés.

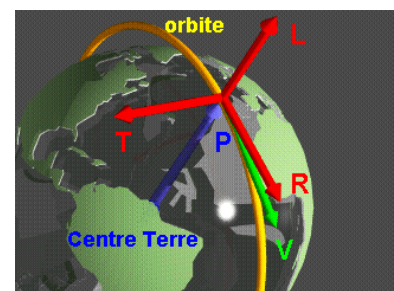


FIG. 10.15 – Repère orbital local – Image CNES [5]

Les positions des différents astres (le Soleil, la Terre, la Lune), ainsi que celle du satellite, sont calculées et exprimées dans le repère inertiel. La direction de chaque instrument à bord du satellite est quant à elle exprimée dans le repère de ce dernier, c'est-à-dire un repère mobile. Il faut dès lors utiliser une matrice de changement de repère pour les ramener dans le repère inertiel.

10.4 Attitude d'un satellite

L'attitude d'un satellite, c'est son orientation dans l'espace autour de son centre de gravité. Elle est définie par l'orientation du repère satellite exprimée dans un repère donné, généralement inertiel ou orbital local. Dans le problème traité, l'attitude est toujours représentée dans le repère inertiel pour faciliter les calculs, puisque les positions et directions des différents astres sont elles aussi exprimées dans ce repère.

L'attitude peut être définie de différentes manières.

- *Angles de tangage, roulis, lacet* : *TRL*. Ce sont les angles de rotation respectivement autour des axes y , x et z du repère satellite. Ces angles sont largement utilisés pour définir une orientation dans le domaine de l'aéronautique et de l'espace.

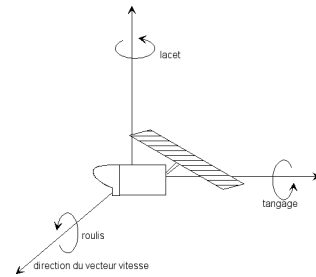


FIG. 10.16 – Anlges TRL

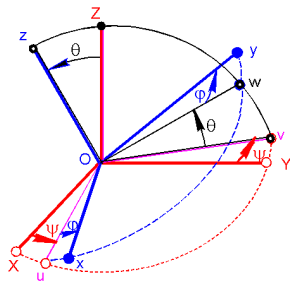


FIG. 10.17 – Anlges d'Euler

- *Angles d'Euler* : ψ (précession), θ (nutation), ϕ (rotation propre). Ces angles représentent trois rotations successives. Généralement, on utilise la séquence 3-1-3 : rotation autour de z , puis autour de x' , et enfin autour de z'' .

Cette séquence (3-1-3) est la plus répandue dans la littérature. Il existe cependant bien d'autres représentations des angles d'Euler, notamment la séquence 3-2-1 aussi connue sous le nom d'angles de Cardan.

- *Quaternion* : Cet outil mathématique permet de n'utiliser qu'un seul angle de rotation. En effet, toute composition de plusieurs rotations d'angles quelconques autour d'axes quelconques, peut être ramenée à une seule rotation d'un angle θ autour d'un axe u . Les quaternions sont développés dans la section 11.2 du même nom.

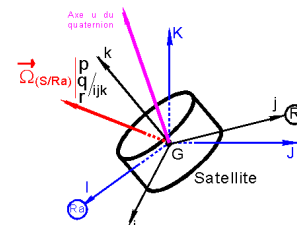


FIG. 10.18 – Rotation avec un quaternion

10.5 Le Système de Contrôle d'Attitude (SCA)

Un satellite artificiel est généralement doté d'un système de contrôle d'attitude et d'orbite (SCAO). Le système de contrôle d'orbite consiste à commander les positions, vitesses et accélérations du centre de gravité du satellite dans l'espace. Le système de contrôle d'attitude (SCA) quant à lui remplit deux fonctions essentielles : mesurer et modifier l'orientation du satellite par rapport à un repère fixé. Le SCA étant abordé dans ce projet de fin d'études, il semble intéressant d'en exposer les domaines d'application.

- **Observation** : un satellite doit être capable de pointer sa charge utile (antenne, télescope, ...) de manière très précise vers des zones géographiques données. A titre d'exemple, une erreur de pointage de $0,1^\circ$ à une altitude de 36000 km (orbite géostationnaire) correspond à une erreur au sol de 63 km.
- **Alimentation** : il doit pouvoir orienter ses panneaux solaires vers le Soleil afin de recharger ses batteries avant que celles-ci ne se déchargent trop et ne deviennent inutilisables, ce qui rendrait le satellite incapable d'assurer sa mission.
- **Préparation aux manœuvres d'orbite** : le satellite doit également pouvoir orienter convenablement le système propulsif (tuyères) lors de manœuvres de contrôle d'orbite.
- **Détection de zones interdites** : le satellite doit veiller à ce qu'aucun de ses instruments ne pointe vers une direction (comme le Soleil) qui pourrait le détruire du fait de sa forte sensibilité. Cela comprend les antennes, la charge utile (caméra), les instruments du SCA lui-même (senseurs). En l'occurrence, c'est ce domaine d'application qui fait l'objet du projet de fin d'études.

Les satellites sont classés en plusieurs grandes familles en fonction de leur SCA.

1. **Les satellites stabilisés passivement par gradient de gravité.** Ce dernier a pour origine l'attraction différentielle qui s'exerce sur des masses situées à des distances différentes du centre de la Terre.
2. **Les satellites spinnés.** Ces satellites sont lancés en rotation autour d'un de leurs axes (axe stabilisé par phénomène de raideur gyroscopique). Ils sont utilisés pour scanner rapidement une bande au sol (satellite météo) ou l'espace environnant (satellite scientifique analysant le plasma de l'ionosphère).
3. **Les satellites stabilisés trois axes.** Pour ces satellites le SCA permet de rejoindre une attitude et de s'y stabiliser. La plupart des satellites d'observation, utilisent ce type de plate-forme. De plus, MANIAC a été conçu pour être implémenter dans ce type de satellite. C'est donc pour ceux-ci que notre programme pourra s'adresser.

Chapitre 11

Aspects mathématiques

Dans cette section sont présentées quelques formules mathématiques issues de différents ouvrages¹. Des formules de géométrie sphérique notamment sont largement utilisées dans ce travail. En effet, les techniques d'évitement développées nécessitent des calculs concernant les triangles sphériques. Ces techniques d'évitement sont utilisées à la section 7.2.

11.1 Triangles sphériques

11.1.1 Définition

Par définition, un triangle sphérique est une surface limitée, sur une sphère, par trois arcs de grand cercle reliant trois points de la sphère, appelés sommets. Ses côtés sont définis par des "lignes géodésiques", c'est-à-dire la généralisation de la notion de "lignes droites" aux espaces courbes. Ces géodésiques représentent la distance la plus courte entre deux points situés sur une sphère, appelée la "distance de grand cercle". De plus, pour se limiter aux "triangles simples", les côtés du triangle doivent tous être plus petits qu'une demi-circonférence.

Comme pour tout triangle, il y a six paramètres (angles) qui sont les trois sommets et les trois côtés. Les cotés sont représentés par des minuscules et les sommets par des majuscules.

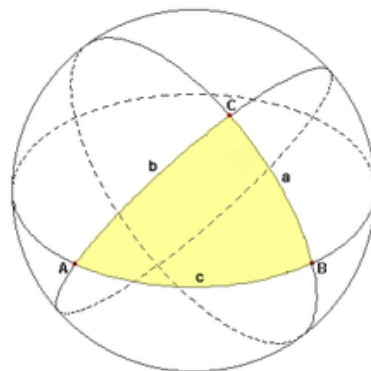


FIG. 11.1 – Triangle sphérique – Image Wikipedia [3]

¹Notamment André DANJON [11]

11.1.2 Relations trigonométriques

Coordonnées cartésiennes d'un sommet

Un premier calcul utile à effectuer est de retrouver les coordonnées d'un sommet d'un triangle sphérique sur la sphère unité, connaissant tous ses paramètres (trois angles et trois côtés) ainsi que les coordonnées des deux autres sommets.

Dans le cas présent, les deux autres sommets sont placés à des endroits bien particuliers de la sphère. Un des sommets doit se situer sur l'axe des Z , donc en $(0, 0, 1)$ (sommet B sur le schéma) tandis que l'autre se situe dans le plan OXZ , avec $X < 0$ (sommet A). Dans ce cas, les coordonnées du dernier sommet se trouvent comme suit.

$$\begin{cases} X_c = -\sin a \cos B \\ Y_c = \sin a \sin B \\ Z_c = \cos a \end{cases}$$

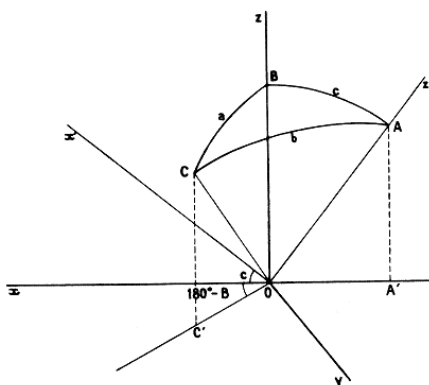


FIG. 11.2 – Coordonnées du sommet C – Image Astronomie Générale

Distance angulaire

Une autre formule souvent utilisée est celle qui permet de calculer l'écart angulaire θ entre deux points (A et B) situés sur une sphère, comme par exemple la distance entre deux sommets d'un triangle sphérique. En géométrie sphérique, une distance entre deux points est exprimée non pas en unité de longueur, mais bien en distance angulaire (radians).

$$\cos \theta = \frac{X_A X_B + Y_A Y_B + Z_A Z_B}{\sqrt{X_A^2 + Y_A^2 + Z_A^2} \sqrt{X_B^2 + Y_B^2 + Z_B^2}}$$

Comme tous les calculs du programme s'effectuent sur une sphère unité, le dénominateur de cette expression vaut évidemment 1, et la formule se simplifie comme suit.

$$\cos \theta = X_A X_B + Y_A Y_B + Z_A Z_B$$

11.1.3 Résolution des triangles sphériques

Comme pour les triangles dans l'espace à deux dimensions, la connaissance de trois des six paramètres est nécessaire pour déterminer les trois autres. Par exemple, si les trois angles des sommets sont connus, il est possible de calculer les trois côtés.

Dans le problème traité (évitement), deux côtés ainsi qu'un angle sont connus (entourés en rouge sur le schéma). Cet angle n'est cependant pas celui situé entre les deux côtés.

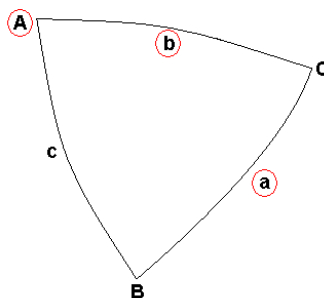


FIG. 11.3 – Triangle sphérique étudié

Pour trouver les trois derniers paramètres, il faut utiliser les formules suivantes :

$$\begin{cases} \sin B = \frac{\sin b \sin A}{\sin a} \\ \tan \frac{c}{2} = \tan \frac{a+b}{2} \frac{\cos \frac{A+B}{2}}{\cos \frac{A-B}{2}} \\ \tan \frac{C}{2} = \cot \frac{A+B}{2} \frac{\cos \frac{a-b}{2}}{\cos \frac{a+b}{2}} \end{cases}$$

où a , b et A sont connus.

11.1.4 Remarque

Bien entendu, la connaissance de trois paramètres différents sur les six demande l'utilisation d'autres formules. Toutefois, puisque le cas explicité ci-dessus est le seul utilisé dans le projet, les formules liées aux autres cas ne sont pas développées.

11.2 Quaternions

Comme cela a été expliqué à la section 10.4, il existe de nombreux moyens pour représenter l'attitude d'un satellite. Cependant, les angles d'Euler (comme ceux de Cardan) présentent des singularités en certains points. En effet, dans le cas où l'angle de nutation est nul par exemple, $\sin \theta = 0$ (rotation autour de l'axe x'), il est impossible de distinguer la précession de la nutation propre (les axes z et z'' sont confondus). Il est donc nécessaire d'introduire un nouvel outil afin de lever ces singularités : le quaternion.

11.2.1 Définition

Le quaternion est un outil mathématique qui sert à définir une rotation dans un espace à trois dimensions. Par conséquent, il peut définir l'attitude d'un corps (un satellite par exemple) en définissant la rotation de son repère local (repère satellite) par rapport au repère global (repère inertiel). Le quaternion est un vecteur à quatre composantes matérialisant l'angle de la rotation ainsi que son axe :

$$Q = \left(\cos \frac{\theta}{2}, \vec{u} \sin \frac{\theta}{2} \right)$$

où θ est l'angle de rotation et $\vec{u} = (u, v, w)$ l'axe de rotation.

Une autre approche consiste à définir le quaternion comme un nombre complexe d'ordre quatre (ou "hypercomplexe"). C'est une combinaison linéaire de quatre symboles : l'unité scalaire "1" et trois symboles imaginaires i, j, k .

$$Q = q_1 + q_2i + q_3j + q_4k$$

Les opérations d'addition et de multiplications se définissent en faisant l'analogie avec les nombres complexes : $i^2 = j^2 = k^2 = -1$ et $ij = k$. Dans le cas particulier où q_1 est nul, le quaternion est appelé quaternion pur.

11.2.2 Algèbre des quaternions

Voici les propriétés des quaternions les plus couramment utilisées.

Conjugué :

$$\bar{Q} = q_1 - q_2i - q_3j - q_4k$$

Norme :

$$|Q| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} = \sqrt{\cos^2 \frac{\theta}{2} + \|\vec{u}\| \sin^2 \frac{\theta}{2}}$$

Cette norme vaut toujours 1 dans le projet, car l'axe de rotation (\vec{u}) du quaternion est normé. En effet, \vec{u} est toujours défini à partir d'un point de la sphère unité.

Produit de quaternions $R = QP$ (non commutatif) :

$$R = \begin{bmatrix} q_1 & -q_2 & -q_3 & -q_4 \\ q_2 & q_1 & -q_4 & q_3 \\ q_3 & q_4 & q_1 & -q_2 \\ q_4 & -q_3 & q_2 & q_1 \end{bmatrix} \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix}$$

Produit scalaire :

$$\vec{Q} \cdot \vec{P} = \text{Re}(Q \bar{P})$$

Inverse :

$$\text{si } Q \neq 0 \quad Q^{-1} = \frac{\bar{Q}}{\|Q\|^2}$$

11.2.3 Rotation avec un quaternion

Comme expliqué au début de cette section, les quaternions sont intéressants pour exprimer une attitude, car ils évitent les singularités pouvant survenir avec les angles d'Euler. Mais ils sont également très pratiques pour appliquer des rotations à des vecteurs, puisque un quaternion représente avant tout une rotation.

Il existe deux conventions pour les quaternions de rotation entre deux repères.

1. Le changement de repère de 1 vers 2 :

$$\vec{V}_1 = \bar{Q}_{1 \rightarrow 2} \vec{V}_2 Q_{1 \rightarrow 2}$$

Composition : $Q_{1 \rightarrow 3} = Q_{1 \rightarrow 2} Q_{2 \rightarrow 3}$

2. La rotation qui amène 1 sur 2

$$\vec{V}_2 = Q_{1/2} \vec{V}_1 \bar{Q}_{1/2}$$

Composition : $Q_{1/3} = Q_{2/3} Q_{1/2}$

Dans ce projet, c'est la deuxième convention qui est utilisée par commodité, car toutes les données sont traitées dans le repère inertiel.

Exemple

Soit un instrument du satellite, par exemple la caméra. Dans le repère satellite, celle-ci pointe dans la direction \vec{V}_{sat} . Dès lors, le quaternion d'attitude du satellite dans le repère inertiel représente la rotation qu'il faut appliquer à \vec{V}_{sat} pour l'amener sur \vec{V}_{in} de la caméra dans le repère inertiel.

La relation qui lie \vec{V}_{in} à \vec{V}_{sat} fait intervenir les produits de quaternions définis au point précédent. Cependant, le vecteur \vec{V}_{sat} est de dimension 3, alors que le quaternion est de dimension 4. Afin que les dimensions concordent, il faut transformer \vec{V}_{sat} en quaternion $\vec{V}_{q sat}$ en ajoutant l'unité scalaire qui vaut 0.

$$\begin{aligned} \vec{V}_{sat} &= (V_{x sat}, V_{y sat}, V_{z sat}) \\ \Rightarrow \vec{V}_{q sat} &= (0, V_{x sat}, V_{y sat}, V_{z sat}) \end{aligned}$$

Si la rotation (sat/in) est définie par trois angles d'Euler (ψ, θ, ϕ) , alors un quaternion est associé à chacun de ces angles.

$$Q_1(\psi), Q_2(\theta), Q_3(\phi)$$

Le quaternion $Q_{sat/in}$ résultant des trois rotations est le produit de ces trois quaternions, dans l'ordre inverse. C'est le quaternion de la rotation qui amène le repère satellite sur le repère inertiel.

$$Q_{sat/in} = Q_3 Q_2 Q_1$$

Il est à présent possible d'effectuer la rotation désirée comme suit :

$$\begin{cases} \vec{V}_{1q} = Q_1 \vec{V}_{q\ sat} \bar{Q}_1 \\ \vec{V}_{2q} = Q_2 \vec{V}_{1q} \bar{Q}_2 \\ \vec{V}_{3q} = Q_3 \vec{V}_{2q} \bar{Q}_3 \end{cases}$$

$$\Rightarrow \vec{V}_{q\ in} = \vec{V}_{3q} = Q_3 Q_2 Q_1 \vec{V}_{q\ sat} \bar{Q}_1 \bar{Q}_2 \bar{Q}_3$$

$$\Rightarrow \vec{V}_{q\ in} = Q_{sat/in} \vec{V}_{q\ sat} \bar{Q}_{sat/in}$$

Pour rappel, le vecteur $\vec{V}_{q\ in}$ est de dimension 4. Il correspond au quaternion dont l'axe de rotation est \vec{V}_{in} , la direction de la caméra après rotation.

$$\vec{V}_{q\ in} = \left(\cos \frac{\theta}{2}, \vec{V}_{in} \sin \frac{\theta}{2} \right) = \left(\cos \frac{\theta}{2}, V_{x\ in} \sin \frac{\theta}{2}, V_{y\ in} \sin \frac{\theta}{2}, V_{z\ in} \sin \frac{\theta}{2} \right)$$

Pour récupérer le vecteur \vec{V}_{in} (et non le quaternion $\vec{V}_{q\ in}$), un dernier calcul est nécessaire :

$$\begin{cases} V_{x\ in} \\ V_{y\ in} \\ V_{z\ in} \end{cases} = \frac{1}{\sin \frac{\theta}{2}} \begin{cases} V_{q2\ in} \\ V_{q3\ in} \\ V_{q4\ in} \end{cases}$$

où $\frac{\theta}{2} = \arccos V_{q1\ in}$.

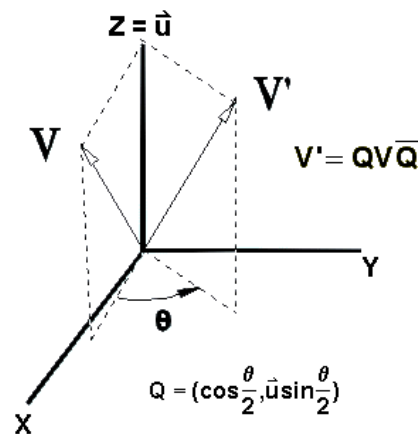


FIG. 11.4 – Rotation d'axe \vec{u} avec un quaternion

11.2.4 Quaternions – Angle d’Euler

Parfois, il peut être utile d’utiliser les angles d’Euler. Cependant, il est nécessaire de les ramener à des quaternions pour la suite des calculs.

$$Q^T = \begin{pmatrix} \cos \frac{\theta}{2} \cos \frac{\psi+\phi}{2} \\ \sin \frac{\theta}{2} \cos \frac{\psi-\phi}{2} \\ \sin \frac{\theta}{2} \sin \frac{\psi-\phi}{2} \\ \cos \frac{\theta}{2} \sin \frac{\psi+\phi}{2} \end{pmatrix}$$

Les avantages du quaternion par rapport aux angles TRL ou d’Euler, réside dans sa facilité d’utilisation puisque il ne fait intervenir que quatre paramètres au lieu de neuf. L’attitude du satellite est définie en un seul calcul, contrairement aux cas précédents où trois rotations étaient nécessaires. Les opérations avec les quaternions requièrent ainsi moins de puissance de calcul et moins de mémoire informatique. C’est pour ces raisons que l’utilisation des quaternions est plus que répandue dans le domaine du contrôle d’attitude.

L’inconvénient du quaternion est qu’il ne permet pas de visualiser rapidement l’attitude du satellite. Il faut alors repasser aux angles d’Euler pour se la représenter.

Chapitre 12

Documentation des fonctions

Ce chapitre reprend toutes les fonctions du programme. Le descriptif de chacune est repris des commentaires présents dans le code. Elles sont présentées dans chaque section par ordre alphabétique et non par apparition au sein du calcul.

A la page suivante est présenté un organigramme contenant toutes les fonctions du programme. Celles encadrées en rouge représentent les fichiers exécutable du programme, décrit à la prochaine section. Les fonctions encadrées en bleu sont les fonctions essentielles du programme. Elles se chargent des calculs principaux, tandis que les fonctions encadrées en noir sont plus des outils de calculs secondaires.

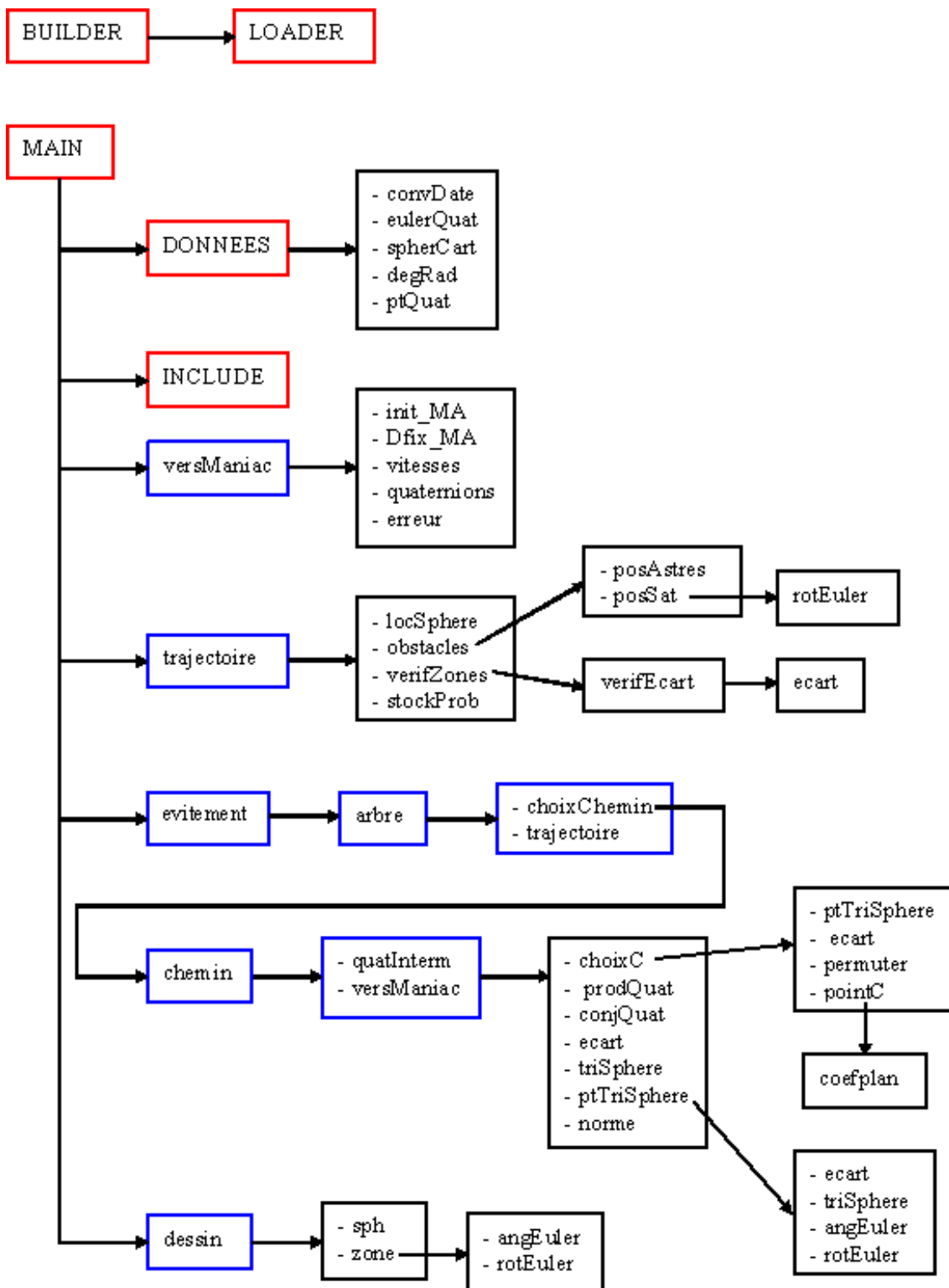


FIG. 12.1 – Organigramme du programme

12.1 Fichiers exécutable

Les fichiers suivants représentent le corps du programme. Ils n'ont pas la même extension (*.sce*) que les fonctions : ce sont de petites applications, qui n'ont pas besoin d'entrées ni de sorties. Toutes les variables créées et traitées dans ces fichiers sont connues partout dans le programme.

builder

Ce fichier a pour but d'associer les appels de Scilab aux fonctions d'interface écrites en C. Construit de plus les bibliothèques dynamiques, ainsi qu'un fichier *loader.sce* qui permet de les charger.

donnees

Contient toutes les données que l'utilisateur doit entrer pour faire fonctionner le programme.

include

Permet de charger toutes les fonctions (*.sci*) du programme en mémoire avant l'exécution de celui-ci.

loader

Permet de charger les bibliothèques dynamiques. Fichier créé par le fichier *builder.sce*.

main

Fichier principal, c'est celui qu'il faut exécuter pour lancer le programme.

12.2 Fonctions de dessin

Comme indiqué dans le titre de la section, ces trois fonctions ne servent qu'à la représentation graphique. Elles n'ont aucun impact sur les calculs du programme et peuvent être désactivées si on ne cherche pas à se représenter le résultat, comme lors d'un calcul à bord du satellite par exemple.

dessin

Réalise la représentation graphique : sphère unité, zones interdites et direction de visée des instruments tout au long de la manœuvre.

Entrées :

Cam coordonnées complètes de la caméra
 St1 coordonnées complètes du senseur stellaire 1
 St2 coordonnées complètes du senseur stellaire 2
 St3 coordonnées complètes du senseur stellaire 3
 Astres coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement

sph.sci

Calcule les points d'une sphère centrée en 0 afin de la représenter.

Entrées :

r rayon de la sphère
 n maillage
 alpha latitude jusqu'à laquelle il faut dessiner si $alpha = \pi$ alors c'est une sphère, si $alpha < \pi$, c'est une calotte sphérique (utilisé pour représenter les zones interdites)

Sorties :

sx coordonnées en x des points à représenter
 sy coordonnées en y des points à représenter
 sz coordonnées en z des points à représenter

zone.sci

Calcule les points d'une zone interdite sur la sphère ainsi que de son contour et du contour de sécurité, ceci dans le but de les représenter.

Entrées :

M coordonnées du centre de la zone interdite considérée
 alpha rayon de la zone interdite considérée
 coef coefficient de sécurité
 r rayon sur la sphère (permet de superposer des zones)

Sorties :

Sx coordonnées en x de la zone sur la sphère
Sy coordonnées en y de la zone sur la sphère
Sz coordonnées en z de la zone sur la sphère
Cont1 coordonnées du contour de la zone
Cont2 coordonnées du contour de la zone multiplié par le coefficient de sécurité

12.3 Fonctions de calcul

Ces multiples fonctions, souvent très simples, servent à effectuer des petits calculs. Elles sont généralement appelée de nombreuses fois lors de l'exécution du programme.

12.3.1 Triangles sphériques - Géométrie

coefplan.sci

Calcule les coefficients d'un plan passant par (0,0,0) et 2 autres points.

Entrées :

- P l'un des deux points par lequel passe le plan
- Q l'autre point

Sorties :

- A coefficient en x du plan
- B coefficient en y du plan
- C coefficient en z du plan

ecart.sci

Calcule l'écart angulaire entre deux directions.

Entrées :

- A coordonnées d'une des directions
- B coordonnées de la deuxième direction

Sortie :

- e écart angulaire entre A et B

pointC.sci

Calcule les coordonnées cartésiennes du point C pour une configuration donnée.

Entrées :

- M coordonnées du centre de la zone interdite considérée
- dep coordonnées du point de départ
- arr coordonnées du point d'arrivée
- At coordonnées du point tangent issu de A
- Bt coordonnées du point tangent issu de B

Sorties :

- ptC coordonnées du point C
- coefC coefficients normés des plans O-C-At et O-C-Bt

ptTriSphere.sci

Calcule les coordonnées sur la sphère unité d'un sommet d'un triangle sphérique connaissant les coordonnées des deux autres sommets.

Entrées :

ptM coordonnées du centre de la zone interdite considérée
 ptA coordonnées du deuxième sommet
 angA angle pris au sommet ptA du triangle sphérique
 b longueur du côté du triangle sphérique compris entre ptM et le point recherché
 signe permet de choisir l'orientation du triangle

Sortie :

tg coordonnées du sommet recherché

spherCart.sci

Calcule les coordonnées cartésiennes d'un point en fonction de ses coordonnées sphériques ($r = 1$).

Entrées :

theta angle de longitude
 phi angle de latitude

Sortie :

point vecteur des coordonnées cartésiennes du point

triSphere.sci

Calcule deux angles et un côté d'un triangle sphérique pour un angle connu qui n'est pas celui situé entre les deux côtés connus.

Entrées :

a premier côté
 b deuxième côté
 A angle en face du premier côté

Sorties :

B angle en face du deuxième côté
 C troisième angle
 c troisième côté

verifEcart.sci

Verifie si une direction pointe vers une zone interdite donnée.

Entrées :

cas renseigne l'instrument et l'astre concerné
 M coordonnées du centre de la zone interdite considérée
 a coordonnées de la direction courante
 alpha rayon de la zone interdite considérée

Sortie :

test variable de controle, test = cas si le point courant est dans la zone

12.3.2 Quaternions

conjQuat.sci

Conjugué un quaternion.

Entrée :

Q le quaternion à conjuguer

Sortie :

Qbar le quaternion conjugué

eulerQuat.sci

Calcule un quaternion à partir des angles d'Euler.

Entrées :

psi premier angle d'Euler

theta deuxième angle d'Euler

phi troisième angle d'Euler

Sortie :

Q quaternion résultant

prodQuat.sci

Calcule le produit de deux quaternions.

Entrées :

q premier quaternion

p deuxième quaternion

Sortie :

R résultat de la multiplication $q \cdot p$

ptQuat.sci

Crée un quaternion associé à une direction $[x, y, z] \Rightarrow [0, x, y, z]$.

Entrée :

pt coordonnées du point

Sortie :

pt_q quaternion obtenu

quaternions.sci

Calcule grâce à la méthode d'Edwards du second ordre un quaternion à un instant donné en fonction du quaternion au pas précédent et de la vitesse de rotation durant ce laps de temps.

Entrées :

om vecteur des vitesses de rotation autour des trois axes principaux : X Y et Z
 pas valeur du pas de temps
 q quaternion au pas de temps précédent

Sortie :

Q quaternion résultat

quatInterm.sci

Calcule les quaternions aux points intermédiaires par lequel le satellite passe s'il doit éviter une zone interdite.

Entrées :

Inst.b coordonnées initiales et finales de l'instrument considéré
 ptM coordonnées du centre de la zone interdite considérée
 alpha rayon de la zone interdite considérée
 trajet précise si le chemin choisi est le premier ou le deuxième calculé
 qb.A quaternion d'attitude du point de départ
 qb.B quaternion d'attitude du point d'arrivée

Sortie :

q.int matrice des quaternions aux points intermédiaires

12.3.3 Rotations - Repères**angEuler.sci**

Calcule deux des trois angles d'Euler pour arriver à une direction donnée.

Entrée :

P vecteur des coordonnées de la direction de destination

Sorties :

psi premier angle d'Euler
 theta deuxième angle d'Euler
 phi troisième angle d'Euler, toujours = 0.

rotEuler.sci

Applique une rotation d'Euler à une matrice de coordonnées donnée.

Entrées :

x coordonnées en x
 y coordonnées en y
 z coordonnées en z
 psi premier angle d'Euler
 theta deuxième angle d'Euler
 phi troisième angle d'Euler

Sorties :

X coordonnées en x après rotation
 Y coordonnées en y après rotation
 Z coordonnées en z après rotation

12.3.4 Mécanique spatiale**obstacles**

Fait appel au calcul, dans le repère inertiel, de la position des astres et du satellite à un temps donné. Corrige ensuite les directions et les distances pour les centrer sur le satellite.

Entrée :

temps date à laquelle les positions sont calculées

Sortie :

Ml coordonnées du centre de la Lune
 alphal rayon de la zone interdite créée par la Lune
 Ms coordonnées du centre du Soleil
 alphasc rayon de la zone interdite créée par le Soleil pour la caméra
 alphasst rayon de la zone interdite créée par le Soleil pour les senseurs
 Mt coordonnées du centre de la Terre
 alphot rayon de la zone interdite créée par la Terre

posAstres.sci

Calcule, à une date donnée, les coordonnées de la Lune et du Soleil dans le repère GAMMA50 à partir d'une version simplifiée des théories de BROWN pour la Lune et NEWCOMB pour le Soleil.

Entrée :

jour nombre de jours écoulés depuis le 01/01/1950

Sortie :

POS matrice 2x4 contenant sur la première ligne la position de la Lune et son rayon, et sur la deuxième ligne les mêmes informations sur le Soleil

posSat.sci

Calcule la position du satellite en fonction du temps sur une orbite képlérienne simple.

Entrée :

t nombre de jours écoulés depuis le 01/01/1950

Sortie :

Sat coordonnées du satellite dans le repère inertiel

12.3.5 Calcul de la trajectoire**arbre**

Crée et parcourt l'arbre des trajectoires d'évitement jusqu'à trouver une solution acceptable.

Entrées :

FIN	indicateur pour sortir de la boucle lorsqu'une solution est trouvée
Cam	coordonnées complètes de la caméra
St1	coordonnées complètes du senseur stellaire 1
St2	coordonnées complètes du senseur stellaire 2
St3	coordonnées complètes du senseur stellaire 3
Cam_b	coordonnées initiales et finales de la caméra
St1_b	coordonnées initiales et finales du senseur stellaire 1
St2_b	coordonnées initiales et finales du senseur stellaire 2
St3_b	coordonnées initiales et finales du senseur stellaire 3
temps1	matrice des temps à chaque étage de tous les différents segments de la manœuvre
temps2	vecteur des temps à chaque étage du chemin non emprunté pour permettre une comparaison future
etage	indique la progression dans l'arbre des trajectoires
zone_prob	renseigne l'instrument et l'astre concerné
zone_prob_old	vecteur de toutes les zones évitées et des instruments concernés
seg_prob_old	vecteur de tous les segments des trajectoires où ont eu lieu les détections de zones interdites
Q	matrice des quaternions de la manœuvre
Q_old	matrice des quaternions de la manœuvre jusqu'à l'évitement concerné
Astres	coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement
choix	précise si on désire le chemin le plus rapide ou le plus lent
qb_A	quaternion initial
wb_A	vitesses initiales
qb_B	quaternion final
wb_B	vitesses finales

Sorties :

FIN	indicateur pour sortir de la boucle lorsqu'une solution est trouvée
Cam	coordonnées complètes de la caméra
St1	coordonnées complètes du senseur stellaire 1
St2	coordonnées complètes du senseur stellaire 2
St3	coordonnées complètes du senseur stellaire 3
temps1	matrice des temps à chaque étage de tous les différents segments de la manœuvre
temps2	vecteur des temps à chaque étage du chemin non emprunté pour permettre une comparaison future
etage	indique la progression dans l'arbre des trajectoires
choix_total	vecteur des choix effectués sur les chemins durant le parcourt de l'arbre
zone_prob_old	vecteur de toutes les zones évitées et des instruments concernés
seg_prob_old	vecteur de tous les segments des trajectoires où ont eu lieu les détections de zones interdites
Q	matrice des quaternions de la manœuvre
Astres	coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement
choix	précise si on désire le chemin le plus rapide ou le plus lent

choixC.sci

Fait appel au calcul des 4 points tangents à la zone interdite et les associe.

Fait également appel au calcul du point C avec les points tangents désirés et renvoie ses coordonnées.

Entrées :

M	coordonnées du centre de la zone interdite considérée
dep	coordonnées du point de départ
arr	coordonnées du point d'arrivée
angle	rayon de la zone interdite considérée
choix	précise si on désire le chemin le plus rapide ou le plus lent

Sorties :

ptC	coordonnées du point C
coefC	coefficients normés des plans O-C-point tangent issu de A et O-C-point tangent issu de B

chemin.sci

Fait appel à MANIAC et au calcul des quaternions aux points intermédiaires pour construire les deux chemins possibles pour contourner une zone interdite.

Compare également la durée de ces deux-ci et renvoie la matrice des quaternions du chemin le plus rapide (si choix = 1) ou du plus lent (si choix = 2).

Entrées :

qb_A	quaternion d'attitude du point de départ
qb_B	quaternion d'attitude du point d'arrivée
wb_A	vecteur des vitesses au point de départ
wb_B	vecteur des vitesses au point d'arrivée
Inst.b	coordonnées initiales et finales de l'instrument considéré
M	coordonnées du centre de la zone interdite considérée
alpha	rayon de la zone interdite considérée
choix	précise si on désire le chemin le plus rapide ou le plus lent

Sorties :

Q	matrice des quaternions du chemin choisi
trajet	précise si le chemin choisi est le premier ou le deuxième calculé
duree_seg	matrice contenant la durée de chaque partie de la trajectoire
q_int	matrice des quaternions aux points intermédiaires

choixChemin

Choisit les paramètres d'entrée de la fonction qui calcule le chemin de l'évitement en fonction de l'instrument en danger et de la zone à éviter.

Entrées :

qb_A	quaternion d'attitude du point de départ
qb_B	quaternion d'attitude du point d'arrivée
wb_A	vecteur des vitesses au point de départ
wb_B	vecteur des vitesses au point d'arrivée
Cam_b	coordonnées initiales et finales de la caméra
St1_b	coordonnées initiales et finales du senseur stellaire 1
St2_b	coordonnées initiales et finales du senseur stellaire 2
St3_b	coordonnées initiales et finales du senseur stellaire 3
Astres	coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement
choix	précise si on désire le chemin le plus rapide ou le plus lent
zone_prob	renseigne l'instrument et l'astre concerné

Sorties :

Q	matrice des quaternions du chemin choisi
trajet	précise si le chemin choisi est le premier ou le deuxième calculé
duree_seg	matrice contenant la durée de chaque partie de la trajectoire
q_int	matrice des quaternions aux points intermédiaires

evitement

Calcule l'évitement entier. Fait appel à la construction de l'arbre puis compare, du point de vue de la durée, les différentes solutions trouvées.

Entrées :

Cam coordonnées complètes de la caméra
 St1 coordonnées complètes du senseur stellaire 1
 St2 coordonnées complètes du senseur stellaire 2
 St3 coordonnées complètes du senseur stellaire 3
 Astres coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement

Sorties :

Cam coordonnées complètes de la caméra
 St1 coordonnées complètes du senseur stellaire 1
 St2 coordonnées complètes du senseur stellaire 2
 St3 coordonnées complètes du senseur stellaire 3
 Astres coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement

locSphere.sci

Calcule les coordonnées d'une direction en fonction de la direction de départ et du quaternion de rotation.

Entrées :

V coordonnées de la direction de départ
 Q quaternion de rotation

Sortie :

pos coordonnées de la direction d'arrivée de la rotation

trajectoire

Calcule la trajectoire des différents axes (de référence et de visée des instruments)
 A chaque pas de temps, appelle le calcul de la position des astres et lance la vérification sur une entrée dans une zone interdite.

Entrées :

debut renseigne la position du début de l'écriture dans les matrices
 etage indique la progression dans l'arbre des trajectoires
 Q matrice des quaternions de la manœuvre
 Astres coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement

Sorties :

Cam	coordonnées complètes de la caméra
St1	coordonnées complètes du senseur stellaire 1
St2	coordonnées complètes du senseur stellaire 2
St3	coordonnées complètes du senseur stellaire 3
Astres	coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement
Ml	coordonnées du centre de la Lune
alphal	rayon de la zone interdite créée par la Lune
Ms	coordonnées du centre du Soleil
alphasc	rayon de la zone interdite créée par le Soleil pour la caméra
alphasst	rayon de la zone interdite créée par le Soleil pour les senseurs
Mt	coordonnées du centre de la Terre
alphat	rayon de la zone interdite créée par la Terre
zone_prob	renseigne l'instrument et l'astre concerné
i_prob	renseigne le temps auquel a eu lieu la détection

12.3.6 Divers

convDate.sci

Convertit une date exprimée en jour-mois-année en un nombre de jours juliens (fractionnaires) écoulés depuis le 01/01/1950.

Entrée :

dat vecteur contenant année - mois - jour

Sorties :

jour nombre de jours écoulés depuis le 01/01/1950
err variable de contrôle si la date entrée est invalide

degRad.sci

Convertit des degrés en radians.

Entrée :

deg angle en degrés

Sortie :

rad angle en radians

erreur.sci

Affiche un message d'erreur renvoyé par MANIAC.

Entrée :

cr code du message d'erreur

norme.sci

Calcule la norme d'un vecteur.

Entrée :

P vecteur à normer

Sortie :

n norme du vecteur

stockProb

Stocke dans une matrice la position et le rayon d'un astre quand un instrument entre dans celui-ci.

Entrées :

etage indique la progression dans l'arbre des trajectoires

Astres coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement

zone_prob renseigne l'instrument et l'astre concerné

Ms coordonnées du centre du Soleil

Ml coordonnées du centre de la Lune

Mt coordonnées du centre de la Terre

alphasc rayon de la zone interdite créée par le Soleil pour la caméra

alphast rayon de la zone interdite créée par le Soleil pour les senseurs

alphal rayon de la zone interdite créée par la Lune

alphat rayon de la zone interdite créée par la Terre

Sortie :

Astres coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement

permuter.sci

Permute deux variables.

Entrées :

A première variable

B deuxième variable

Sorties :

A ancien B

B ancien A

verifZones

Vérifie à un temps donné que chaque instrument est en dehors des zones interdites le concernant.

Entrée :

zone_prob renseigne l'instrument et l'astre concerné

Sorties :

Cam	coordonnées complètes de la caméra
St1	coordonnées complètes du senseur stellaire 1
St2	coordonnées complètes du senseur stellaire 2
St3	coordonnées complètes du senseur stellaire 3
Astres	coordonnées des astres au début et à la fin de la manœuvre ainsi qu'au moment d'un évitement
Ms	coordonnées du centre du Soleil
Ml	coordonnées du centre de la Lune
Mt	coordonnées du centre de la Terre
alphasc	rayon de la zone interdite créée par le Soleil pour la caméra
alphasst	rayon de la zone interdite créée par le Soleil pour les senseurs
alphal	rayon de la zone interdite créée par la Lune
alphat	rayon de la zone interdite créée par la Terre

versManiac

Réalise les appels à MANIAC, puis met en forme toutes les données retournées. Calcule la matrice des quaternions de la manœuvre à chaque pas de temps.

Entrées :

q_A	quaternion initial
om_A	vitesses initiales
q_B	quaternion final
om_B	vitesses finales
oldtaille	ancienne taille de la matrice Q, permet d'écrire à la suite
Q	matrice des quaternions de la manœuvre complète

Sorties :

Q	matrice des quaternions de la manœuvre complète
tf	temps total de la manœuvre

vitesses.sci

Calcule les vitesses et accélérations instantanées de rotation autour d'un axe.

Entrées :

tc	temps courant en secondes
duree	duree totale de la manœuvre
dp	degré du polynome des vitesses
c_om	coefficient du polynome des vitesses

Sorties :

om	vitesse instantanée
omp	accélération instantané