

# Projet Filière Cycle Ingénieur

## Commande & Systèmes

### Modélisation d'une Fusée Expérimentale

CHALIGNE Charles  
GAUTHERON Vincent  
SERIN Pierre

# Sommaire

Remerciements.....	5
Résumé.....	6
Abstract.....	6
Introduction.....	7
I °) Présentation du vol.....	8
1 °) Définition.....	9
2 °) Conditions de stabilité.....	9
3 °) Cas extrêmes : Surstabilité et Indifférence.....	10
3.1 °) La surstabilité.....	10
3.2 °) L'indifférence.....	12
4 °) La stabilité dynamique.....	12
5 °) Critères simplifiés de stabilité statique.....	13
6 °) Conclusion sur la stabilité.....	13
III °) Hypothèses d'études :.....	14
IV °) Angles et repères mis en jeu.....	15
1 °) Le repère de la fusée.....	15
2 °) Le repère Terrestre.....	16
3 °) Le repère aérodynamique.....	16
4 °) Le vent.....	17
V °) Mise en équation du système.....	17
1 °) La vitesse de la fusée.....	17
1.1 °) Le vent.....	18
1.2 °) La vitesse de la fusée.....	19
1.3 °) Expression de la vitesse relative.....	19
2 °) Bilan des forces.....	19
3 °) Densité de l'air.....	21
4 °) Coefficients aérodynamiques.....	21
4.1 °) Détermination du $C_z$ .....	21
4.2 °) Détermination du $C_y$ .....	23
4.3 °) Détermination du $C_x$ .....	24
5 °) Détermination des surfaces mises en jeu.....	24
6 °) Détermination des matrices d'inertie.....	25
7 °) Vitesses de rotation et angles d'Euler.....	25
8 °) Mise en équation.....	26
V °) Paramètres de stabilité d'une fusée.....	27
1 °) Détermination du Centre de Portance aérodynamique CPA.....	27
1.1 °) CPA de l'ogive.....	28
1.2 °) CPA de l'empennage.....	29
1.3 °) CPA d'un rétreint.....	29
1.4 °) CPA d'une jupe.....	29
1.5 °) CPA global de la fusée.....	30
2 °) Détermination du centre de gravité.....	30
3 °) Marge statique.....	31
VI °) Représentation d'état du système.....	32
1 °) Choix du vecteur d'état.....	32
2 °) Choix du vecteur d'entrée.....	32
3 °) Représentation d'état.....	32
VII °) Trajectographie de la fusée.....	34
VIII °) Modèle Simulink.....	35
1 °) Choix du modèle.....	35

2 °) Le bloc « paramètres (Z_R0) ».....	36
3 °) Le bloc « Perturbation ».....	37
4 °) Le bloc « Paramètres (t) » .....	38
5 °) Le bloc « Calculs des F et M ».....	39
5.1 °) Incidence, dérapage et bras de levier.....	39
5.2 °) Les effort aérodynamiques.....	41
5.3 °) Le poids.....	42
6 °) Réaction de la rampe.....	43
6.1 °) Masquage du vent.....	43
6.2 °) Réaction fond de rampe.....	43
6.3 °) Réaction des patins de la rampe .....	43
7 °) Le bloc « FUSEE » .....	44
8 °) Le bloc « trajectographie ».....	46
IX °) Validation : Comparaison de trajectoires issues de Trajec et de notre modèle.....	47
1 °) Fusée Expérimentale « Chamois ».....	48
1.1 °) Caractéristiques.....	48
1.2 °) Résultats synthétiques de Trajec .....	48
1.3 °) Résultats synthétiques du modèle.....	48
1.4 °) Résultats graphiques du modèle.....	49
1.5 °) Comparaison énergétique.....	50
2 °) Mini Fusée « Cariacou » .....	51
2.1 °) Caractéristiques.....	51
2.2 °) Résultats synthétiques de Trajec .....	51
2.3 °) Résultats synthétiques du modèle.....	51
2.4 °) Résultats graphiques du modèle.....	52
2.5 °) Comparaison énergétique.....	53
3 °) Fusée Expérimentale « Isard » .....	54
3.1 °) Caractéristiques.....	54
3.2 °) Résultats synthétiques de Trajec .....	54
3.3 °) Résultats synthétiques du modèle.....	54
3.4 °) Résultats graphiques du modèle.....	55
3.5 °) Comparaison énergétique.....	56
4 °) Interprétation des résultats .....	57
4.1 °) Comparatif Trajec/Modèle.....	57
4.2 °) Comportement en sortie de rampe.....	59
5 °) Etude de la stabilité.....	60
5.1 °) Cas avec vent face à la rampe .....	60
5.2 °) Cas avec vent oblique.....	60
5.3 °) Cas avec vent oblique.....	61
6.1 °) Vent avec $\varepsilon=180^\circ$ .....	62
6.2 °) Vent avec $\varepsilon=0^\circ$ .....	63
7 °) Etude de trajectoire d'une fusée lancée à partir d'un drone.....	65
8 °) Etude de trajectoire d'une fusée bi étage Chamois Chamois .....	67
X°) Réalisation de l'interface graphique.....	69
1 °) Choix de l'architecture.....	69
2 °) Mode de création des fenêtres.....	70
3 °) Caractéristiques de l'interface.....	75
3.1 °) Accès aux différents paramètres.....	75
3.2 °) Gestion des messages d'erreur .....	76
3.3 °) Affichage des données dans les listbox.....	77
3.4 °) Temps de calcul.....	77
3.5 °) Gestion de la taille de l'écran .....	78
XI °) Utilisation du logiciel.....	81
1 °) Lancement du logiciel.....	81
2 °) Caractéristiques de la fusée.....	83
2.1 °) Paramètres principaux.....	84
2.2 °) Changements de section.....	85
2.3 °) Menu Avancé .....	87
2.4 °) Charger/Sauvegarder la configuration.....	88
3 °) Conditions initiales.....	89

3.1 °) La rampe.....	89
3.2 °) La vitesse initiale et la position initiale de la fusée .....	89
3.3 °) Le vent.....	90
3.4 °) Charger/Sauvegarder vos conditions initiales .....	90
4 °) Charger / Sauvegarder lancement.....	91
4.1 °) Charger un lancement.....	91
4.2 °) Sauvegarder un lancement.....	92
5 °) Lancement de la simulation.....	93
5.1 °) Résultats primaires .....	93
5.2 °) Affichages complémentaires .....	94
5.3 °) Superposition de résultats.....	95
5.4 °) Trajectoire 3D .....	96
5.5 °) Listes de variables affichables dans la simulation.....	97
Conclusion .....	98

## Remerciements

Nous tenons tout d'abord à remercier, notre tuteur, Emmanuel Grunn pour son aide précieuse à la conduite de ce projet.

Nous remercions également Anh Tuan Phâm pour sa contribution à la suite du jury du projet 4<sup>ème</sup> année.

Nous remercions Emmanuel Mangin pour son aide concernant l'interface graphique.

Nous remercions l'ESTACA pour les moyens informatiques mis à notre disposition.

Nous remercions tous les membres de l'association Estaca Space Odyssey pour leur soutien, leurs conseils et leur bonne humeur.

## Résumé

Depuis 14 ans, l'association Estaca Space Odyssey conçoit et réalise des mini fusées et des fusées expérimentales au sein de l'ESTACA. Ces fusées sont des projets soutenus par le CNES qui les met en œuvre chaque année dans un objectif pédagogique.

A travers ce projet, commencé il y a tout juste un an, nous avons pu mettre en place un logiciel aidant à la conception de fusée expérimentale. Ce projet se décomposa en plusieurs phases : La mise en place des équations régissant le vol, la modélisation de la fusée sous MATLAB SIMULINK, la validation du modèle obtenu et la mise en place d'une interface graphique.

## Abstract

For 14 years, association Estaca Space Odyssey has designed and produced mini rockets and experimental rockets within the ESTACA. These rockets are projects supported by the CNES which implement them each year in a teaching objective.

Through this project, started just a year ago, we could set up software helping at the design of experimental rocket. This project broke up into several phases: The implement of the equations of the flight, the modeling of the rocket under MATLAB SIMULINK, the validation of the model obtained and the installation of a graphic interface.

## Introduction

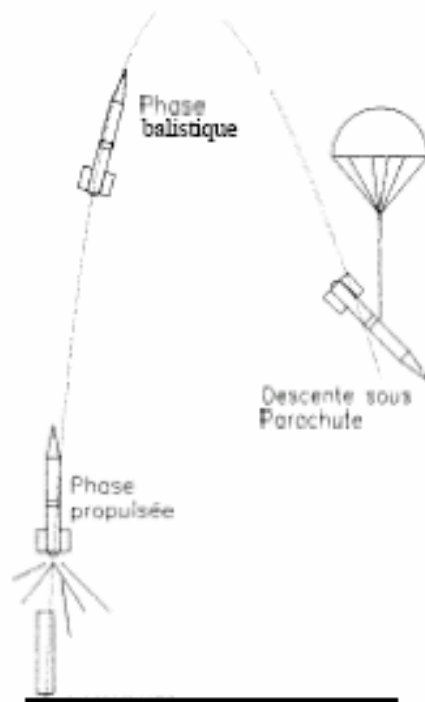
Depuis 14 ans, l'association Estaca Space Odyssey conçoit et réalise des mini fusées et des fusées expérimentales au sein de l'ESTACA. Ces fusées sont des projets soutenus par le CNES qui les met en œuvre chaque année dans un objectif pédagogique.

L'étude des principaux paramètres de ces fusées s'effectue habituellement grâce au logiciel TRAJEC développé par le CNES. Il a pour but de valider les critères de stabilité et d'analyser les phases de vols des fusées.

Cependant, lorsque l'association Estaca Space Odyssey décide de travailler sur des projets évolués, nécessitant soit des lois de commandes soit des études avancées de comportement en vol de ces fusées, l'association est confrontée aux limites de fonctionnalités du logiciel TRAJEC. C'est pour remédier à ce problème que l'association a décidé de se munir d'un modèle fiable et d'une interface d'utilisation de ce modèle.

A terme ce logiciel, s'il s'avère performant, pourrait être diffusé à l'ensemble des clubs aérospatiaux français.

## I °) Présentation du vol



Le vol d'une fusée se décompose en plusieurs phases :

- La phase propulsée.
- La phase balistique.
- La descente sous parachute.

Ces phases s'articulent autour d'évènements majeurs :

Tout d'abord, la période s'écoulant de l'instant de la mise à feu à la fin de combustion du propulseur, et qui s'appelle **la phase propulsée**. Elle comprend une partie où la fusée est guidée par la rampe de lancement et une partie où la fusée est livrée à elle-même.

Après l'extinction du propulseur commence **la phase balistique** pendant laquelle la fusée, uniquement soumise à son poids et à la résistance de l'air, exploite la vitesse acquise pendant la propulsion pour atteindre son altitude maximale.

Une fois la culmination atteinte, lorsque l'engin commence à retomber, la phase balistique se poursuit jusqu'à l'ouverture du parachute, ce qui lui permet de revenir au sol tout en demeurant la plus intact possible.

Bien sûr, on peut rencontrer des phases balistiques avortées lorsque le parachute s'ouvre avant la culmination, ou des vols balistiques complets sans ouverture de parachute.



## II °) La stabilité

Lorsqu'une fusée est lancée depuis une rampe, elle subit des perturbations qui influencent sa trajectoire. Cependant, il est inimaginable de lancer un tel projectile et à une telle vitesse sans pouvoir présumer un minimum de la trajectoire qu'elle va suivre.

Afin de contrôler ce facteur de perturbation, une notion de stabilité a été développée, afin de garantir aux clubs aérospatiaux que leurs réalisations ne subissent que modérément ces agents extérieurs.

### 1 °) Définition

**Une fusée est dite stable si elle retrouve naturellement sa position d'origine lorsque, pour une raison quelconque, elle se met en incidence.**

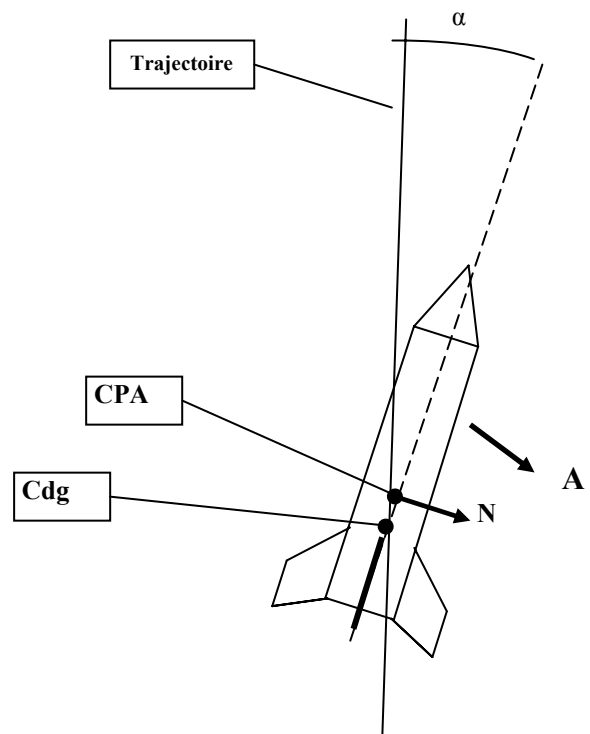
Expliquons cette définition

Supposons que la fusée est soumise à une perturbation, comme le vent par exemple. Elle va alors tout naturellement s'incliner par rapport à la position qu'elle aurait dans le cas où il n'y aurait pas cet agent extérieur. On dit alors qu'elle se met en incidence.

Du fait que la fusée se déplace alors en « crabe », il se développe essentiellement sur son ogive et son empennage des efforts aérodynamiques.

Nommons **A** la résultante de ces efforts aérodynamiques et **N** la composante normale à l'axe de l'engin de **A**.

Nommons enfin **CPA** (Centre de Portance Aérodynamique) le point où cette composante s'applique et **Cdg** le centre de gravité de la fusée.



L'objectif ici est donc de « dimensionner » la distance entre le CPA et le centre de gravité de la fusée, ainsi que les efforts aérodynamiques qui s'exercent sur la fusée, afin que celle-ci revienne comme l'indique la définition, dans sa position d'incidence nulle.

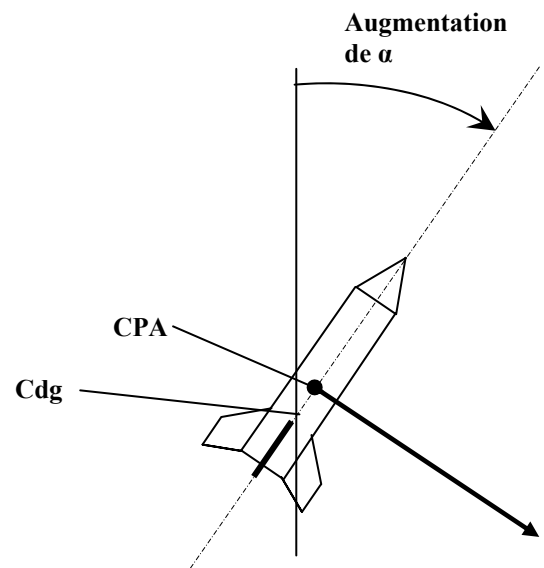
### 2 °) Conditions de stabilité

Comme nous venons de le montrer, l'un des paramètres essentiel à la stabilité de la fusée est la position de son centre de portance aérodynamique. Il est donc nécessaire de placer celui-ci de telle sorte que la marge statique, c'est-à-dire la distance entre le centre de gravité de la fusée et le centre de portance aérodynamique de celle-ci, assure la stabilité de la fusée.

Deux cas sont alors envisageables :

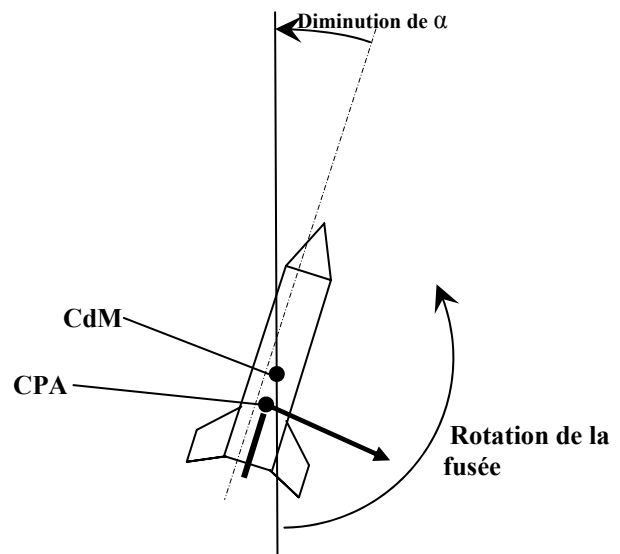
### 1<sup>er</sup> cas : le CPA est en avant du Cdg

En absence de contact avec le sol, toute rotation s'effectue par rapport au centre de gravité de la fusée. Il est aisé de voir, en observant le dessin ici présent que si le **CPA** se situe en avant du **Cdg**, cette rotation ne fera qu'augmenter l'angle d'embarée de la fusée. Celle-ci se retrouvera alors encore plus en travers sur sa trajectoire, ce qui ne fera qu'augmenter la rotation qui en découle et ainsi de suite : on dit alors que la fusée est instable



### 2<sup>ème</sup> cas : le CPA est en arrière du Cdg

Au contraire, si le **CPA** est en arrière du **Cdg**, la rotation induite des efforts aérodynamiques fera diminuer l'angle de la fusée sur sa trajectoire. L'embarée de la fusée est donc en bonne voie de réduction. On peut donc assurer que celle-ci est stable.



**Par conséquent, pour qu'une fusée soit stable, il faut que son centre de portance aérodynamique soit situé derrière son centre de gravité.**

## 3 °) Cas extrêmes : Surstabilité et Indifférence

### 3.1 °) La surstabilité

Bien qu'il faille s'assurer que le centre de portance aérodynamique se situe derrière le centre de gravité de la fusée, il faut néanmoins veiller que la marge statique ou la force de portance elle-même, ne soient pas trop importante. En effet, si tel est le cas, la force de rappel de la

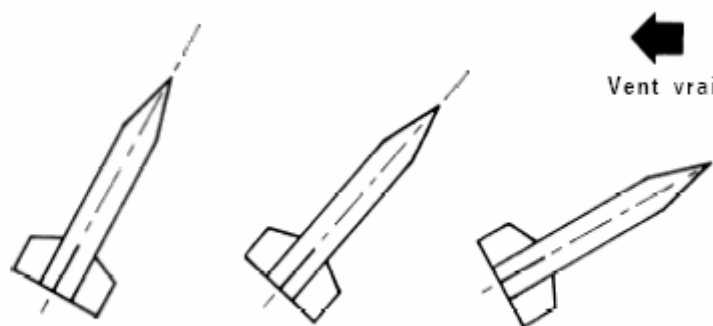
fusée vers son axe d'incidence nulle sera très grande, et enverra la fusée de l'autre côté du vent relatif. La fusée se mettra alors à osciller continuellement sans jamais trouver de position d'équilibre. On dit alors que l'on se trouve dans un cas de surstabilité, ce qui peut devenir critique, notamment si la résistance des matériaux n'est pas suffisante pour supporter ces contraintes.



Evolution de l'incidence d'une fusée surstable durant sa montée.

Un autre inconvénient de cette surstabilité est l'effet du vent météo. Ainsi, lorsque la fusée est stable, elle aligne son axe longitudinal avec le vent relatif donc en partie avec le vent vrai. Cela signifie que la fusée a tendance à remonter le vent. Le problème est que cette situation n'est pas statique : le vent propre cherche à s'aligner sur le vent relatif, qui lui-même se rapproche du vent vrai et la fusée se couche lentement dans le vent vrai.

Dans le cas d'une fusée surstable, la fusée se couchera presque immédiatement dans le vent vrai et partira donc quasiment horizontalement, ce qui n'est pas le but recherché . . .



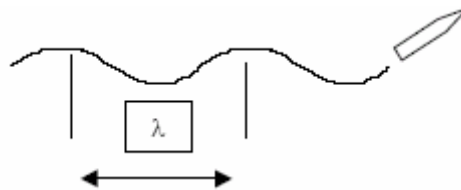
Evolution de l'attitude d'une fusée surstable durant sa montée.

### 3.2 °) L'indifférence

L'autre cas extrême est le cas où le centre de gravité et le centre de poussée sont confondus. Dans ce cas, la fusée n'est plus soumise à aucun couple de rotation : elle va errer dans une position quelconque, qui constitue alors une phase transitoire alternative entre stabilité et instabilité, difficilement évaluable avant le vol.

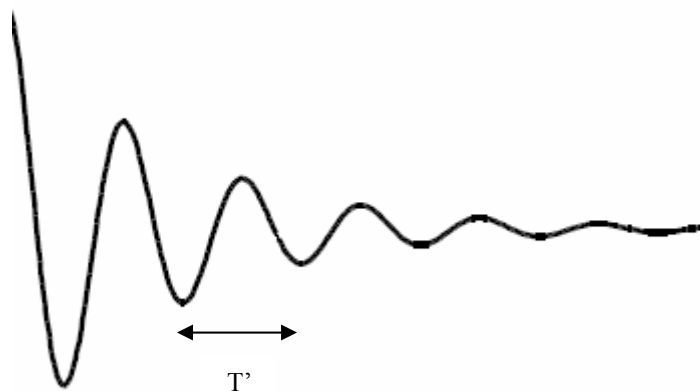
### 4 °) La stabilité dynamique

Comme nous l'avons expliqué auparavant, la fusée, soumise à des perturbations extérieures, va se mettre à osciller autour de sa position d'incidence nulle.



$\lambda$  représente ici la longueur d'onde de la fusée.

Dans le cas d'une fusée stable, ce qui est le but recherché, ces oscillations vont diminuer jusqu'à ce que l'incidence devienne nulle. Si on trace l'évolution de l'incidence d'une telle fusée, on obtient le graphique suivant :



Evolution de l'incidence d'une fusée stable durant sa montée.

On peut alors définir la pseudo-période de cette incidence, donnée par la relation :

$T' = \frac{T}{\sqrt{1 - \zeta^2}}$ , où  $\zeta$  est le coefficient d'amortissement des oscillations et où  $T$  représente la période des oscillations dans le cas où il n'y aurait pas d'amortissement.

Mathématiquement, on démontre que les cas où  $\zeta < 1$  sont préférables, et que dans le cas des fusées, l'idéal est de se situer dans la fourchette de 0,05 à 0,2 pour le coefficient d'amortissement  $\zeta$ .

Le tableau suivant présente les différents cas :

$\zeta$	$T'/T$	Rapport enter maxima successifs
0	1	1
0.02	1.002	0.88
0.05	1.001	0.73
0.1	1.005	0.53
0.2	1.02	0.27
0.4	1.09	0.06
0.6	1.25	0.009

Ainsi, dans le cas d'une étude de stabilité dynamique, il faudra évaluer la courbe d'incidence de la fusée au cours de sa montée, et en déduire les valeurs des paramètres  $T$ ,  $T'$  et  $\zeta$ , afin de s'assurer que ce dernier soit inscrit dans la fourchette précisée précédemment.

### 5 °) Critères simplifiés de stabilité statique

Comme nous venons de le voir, la stabilité consiste en un amortissement des oscillations de la courbe d'incidence de la fusée au cours de son vol, ce qui correspond donc à un retour à sa position d'incidence nulle. La première méthode permettant de déterminer si une fusée est stable ou non est celle évoquée ci-dessus.

Cependant, il est possible, de manière expérimentale, de définir des conditions moyennes de stabilité des fusées. Si ces valeurs ne sont nullement limitatives car les fusées peuvent très bien voler en dehors de ces limites, elles ont néanmoins l'avantage de donner simplement un ordre de grandeur des paramètres à configurer afin d'assurer la stabilité de la fusée.

Ces conditions sont :

- Un Coefficient de portance  $C_z$  compris entre 15 et 30.
- Une marge statique (distance entre le centre de gravité et le centre de portance aérodynamique de la fusée) comprise entre 1,5 et 7.
- Un produit de ces deux précédents paramètres compris entre 30 et 100.

### 6 °) Conclusion sur la stabilité

Pour qu'une fusée soit stable, deux solutions sont envisageables :

- La première, et aussi la plus précise, nécessite la détermination du coefficient d'amortissement de son incidence. Celui-ci peut facilement être réglé sur une fusée en jouant sur la taille et la position de l'empennage, à condition toutefois de connaître le moment d'inertie  $I$  du projectile.
- La seconde, moins précise mais plus pratique, consiste à utiliser les critères simplifiés de stabilité statique, en réglant les valeurs du coefficient de portance de la fusée et sa marge statique.

Dans notre étude, l'objectif sera de vérifier chacun de ces deux paramètres, afin de pouvoir garantir la stabilité de notre fusée.

### III °) Hypothèses d'études :

Afin de rendre possible notre étude, nous avons émis un certain nombre d'hypothèses simplificatrices, couramment utilisées dans le domaine de l'aéronautique.

Ces hypothèses sont :

- La fusée est infiniment rigide.
- Les axes de la fusée sont les axes d'inertie d'où  $I = \begin{pmatrix} A & 0 & 0 \\ 0 & B & 0 \\ 0 & 0 & C \end{pmatrix}_R$ , où A, B, C peuvent être déterminés en fonction des caractéristiques géométriques de la fusée. Pour cela on considérera une répartition homogène de la masse.
- Dans le repère terrestre, le vent est horizontal.
- Le vol s'effectue dans une atmosphère standard, où la densité  $\rho$  est connue pour une altitude donnée.

#### IV °) Angles et repères mis en jeu

Les repères mis en jeu, ainsi que les angles qui y sont associés, sont ceux du monde aéronautique. On peut en dénombrer essentiellement 3, sans tenir compte du vent qui sera explicité par la suite. Ces référentiels sont :

- Le repère  $R_0 (x_0, y_0, z_0)$  qui correspond au référentiel terrestre (absolu)
- Le repère  $R (x, y, z)$  qui correspond au référentiel de la fusée
- Le repère  $R_a (x_a, y_a, z_a)$  qui correspond au référentiel vitesse

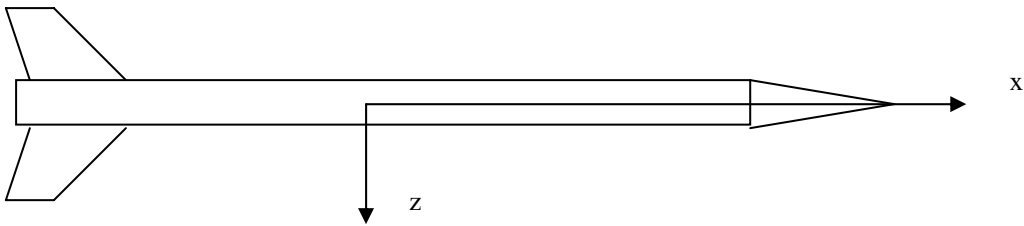
A noter que pour notre étude, le plus simple consiste à travailler dans le repère lié à notre fusée. De ce fait, l'ensemble des matrices de passage entre les différents, excepté pour le vent, seront explicité de manière à obtenir les valeurs des différents paramètres dans ce repère d'étude. Les matrices de passage seront donc données sous la forme :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = [Matrice] \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix}$$

Où  $M$  représente la matrice de passage entre les repères étudiés, et  $(x_1, y_1, z_1)$  le repère de départ.

##### 1 °) Le repère de la fusée

Ce repère a pour origine le centre de gravité de la fusée au moment du décollage. Les axes  $x$ ,  $y$  et  $z$  de ce repère sont porté par la fusée comme suit :



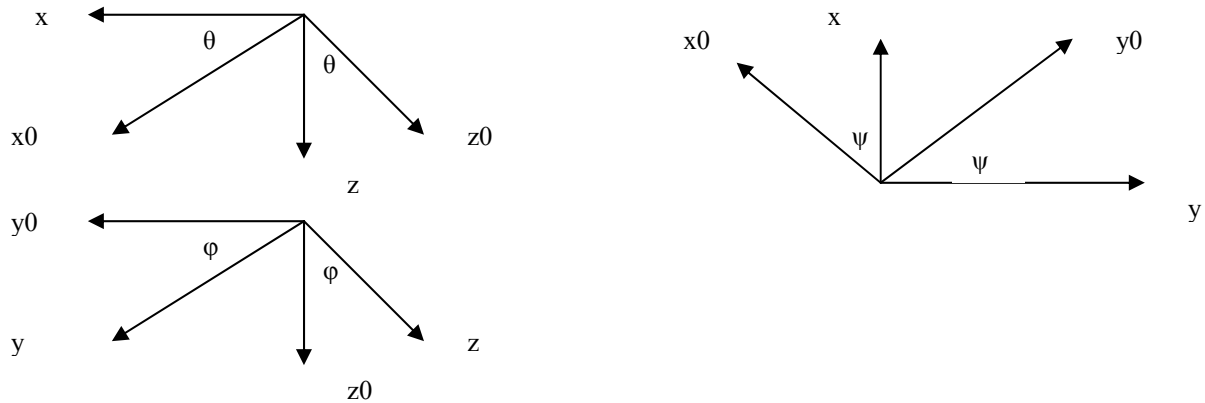
A noter ici que le vecteur  $z$  est donc orienté vers le sol !

Puisque ce repère est notre repère d'étude, c'est-à-dire celui dans lequel nous rapporterons par la suite l'ensemble des forces et moments qui s'exercent sur notre fusée, il est donc inutile d'écrire toute matrice de passage permettant d'exprimer les composantes de ce repère dans un autre.

## 2 °) Le repère Terrestre

C'est le repère absolu. C'est dans ce repère que nous réaliserons à la fin de notre étude, la trajectographie de notre fusée.

Les 3 angles existants entre le repère absolu et le repère de la fusée sont définis de la façon suivante :

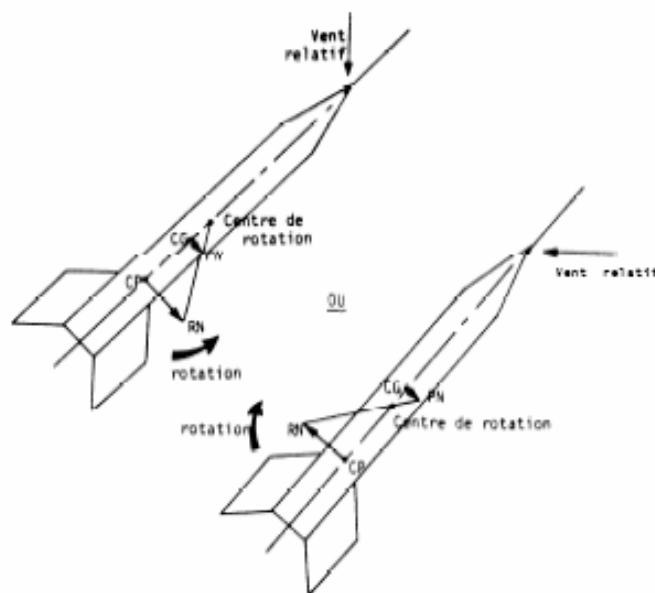


On en déduit donc la matrice de passage entre ce repère R0 et le repère fusée :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} \cos \psi \cdot \cos \theta & \sin \psi \cdot \cos \theta & -\sin \theta \\ -\sin \psi \cdot \cos \phi + \cos \psi \cdot \sin \theta \cdot \sin \phi & \cos \psi \cdot \cos \phi + \sin \psi \cdot \sin \theta \cdot \sin \phi & \cos \theta \cdot \sin \phi \\ \sin \psi \cdot \sin \phi + \cos \psi \cdot \sin \theta \cdot \cos \phi & -\cos \psi \cdot \sin \phi + \sin \psi \cdot \sin \theta \cdot \cos \phi & \cos \theta \cdot \cos \phi \end{bmatrix} \begin{pmatrix} x0 \\ y0 \\ z0 \end{pmatrix}$$

## 3 °) Le repère aérodynamique

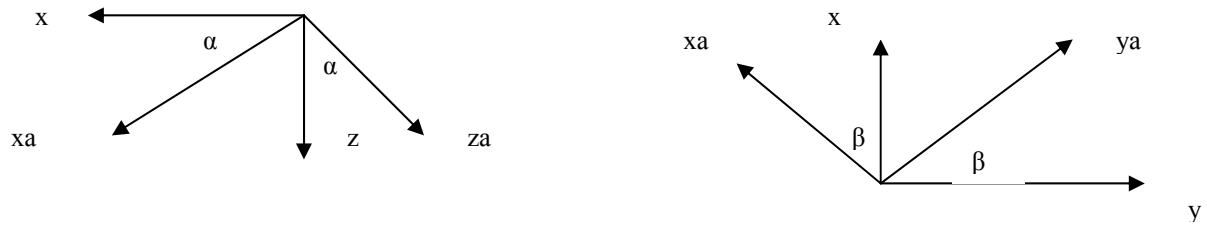
C'est le référentiel qui porte directement les coordonnées du vecteur vitesse (ou du vent relatif). On est donc dans le cas du dessin ci-dessous :



Rotation de la fusée sous l'effet d'une mise en incidence de la fusée



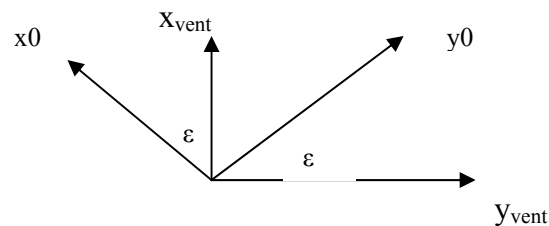
Les angles reliant ce repère lié à la vitesse au repère de la fusée sont définis comme suit :



A noter en particulier que l'angle  $\alpha$  est couramment appelé l'angle d'incidence, tandis que  $\beta$  est communément appelé l'angle de dérapage. Par ailleurs, la troisième rotation, autour de x, existant entre ces deux repères, n'est pas mentionnée ici puisque ne présentant aucun intérêt de part la symétrie de la fusée autour de cet axe.

#### 4 °) Le vent

Par hypothèse, on considère que le vent est toujours horizontal par rapport au repère terrestre. De plus, étant donné la nature de cette perturbation, et l'étendu de celle-ci sur la zone géographique considérée, on pourra considérer qu'il n'existe aucune rotation de ce vent autour de l'axe horizontal  $x_0$ . De ce fait, le passage du repère lié au vent et le repère absolu s'effectue simplement de la façon suivante :



La matrice de passage entre ces deux passages s'écrit donc directement :

$$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = \begin{bmatrix} \cos \varepsilon & -\sin \varepsilon & 0 \\ \sin \varepsilon & \cos \varepsilon & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_{vent} \\ y_{vent} \\ z_{vent} \end{pmatrix}$$

#### V °) Mise en équation du système

##### 1 °) La vitesse de la fusée

Comme nous l'avons précisé dans nos hypothèses, notre étude tient compte de l'effet du vent sur la fusée. De plus, il est également précisé que la fusée est toujours lancée face au vent

dominant du site de lancement. Ainsi on en déduit que la vitesse relative vue par la fusée pendant son vol peut être définie par dans le repère fusée par:

$$Vitesse\_relative_R = Vitesse\_absolue_R - Vitesse\_vent_R$$

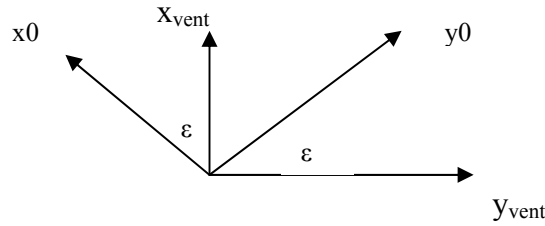
Exprimons donc cette vitesse dans le repère fusée noté R(x,y,z).

### 1.1 °) Le vent

On décide d'orienter le repère lié au vent  $R_{vent}(x_{vent}, y_{vent}, z_{vent})$  de telle sorte que le vent soit uniquement porté par la première composante de ce repère, à savoir  $x_{vent}$ .

Exprimons alors le vent dans le repère terrestre, soit  $V_{vent\_R0} = \begin{Bmatrix} V_{vent\_xo} \\ V_{vent\_yo} \\ V_{vent\_zo} \end{Bmatrix}_{R0}$

Suite aux hypothèses exprimées précédemment, le changement de repère entre  $R_{vent}$  et  $R0$  est le suivant :



En utilisant les matrices de passage explicité en IV, on trouve alors :

$$\begin{cases} V_{vent\_xo} = V_{vent} \cos \varepsilon \\ V_{vent\_yo} = V_{vent} \sin \varepsilon \\ V_{vent\_zo} = 0 \end{cases}$$

Exprimons désormais ce vecteur dans le repère fusée, soit  $V_{vent\_R} = \begin{Bmatrix} V_{vent\_x} \\ V_{vent\_y} \\ V_{vent\_z} \end{Bmatrix}_R$

En utilisant la matrice de passage entre  $R0$  et  $R$ , décrite en IV, on trouve finalement :

On trouve alors :

$$\begin{cases} V_{vent\_x} = V_{vent\_xo} \cos \psi \cdot \cos \theta - V_{vent\_yo} \sin \psi \\ V_{vent\_y} = V_{vent\_xo} (\sin \psi \cdot \cos \varphi + \sin \varphi \cdot \sin \theta) + V_{vent\_yo} \cos \psi \cdot \cos \varphi \\ V_{vent\_z} = V_{vent\_xo} (-\sin \psi \cdot \sin \varphi + \sin \theta \cdot \cos \varphi) - V_{vent\_yo} \cos \psi \cdot \sin \varphi \end{cases}$$

Ayant déterminé auparavant l'expression des termes  $V_{vent\_x}$  et  $V_{vent\_y}$  en fonction de la vitesse du vent et de l'angle  $\varepsilon$ , on peut établir les composantes du vent dans le repère fusée en fonction des différents angles mis en jeu et de sa vitesse.

Finalement, on obtient :

$$\begin{cases} V_{vent\_x} = V_{vent} \cdot \cos \varepsilon \cos \psi \cdot \cos \theta - V_{vent} \sin \varepsilon \cdot \sin \psi \\ V_{vent\_y} = V_{vent} \cos \varepsilon \cdot (\sin \psi \cdot \cos \varphi + \sin \varphi \cdot \sin \theta) + V_{vent} \sin \varepsilon \cdot \cos \psi \cdot \cos \varphi \\ V_{vent\_z} = V_{vent} \cos \varepsilon \cdot (-\sin \psi \cdot \sin \varphi + \sin \theta \cdot \cos \varphi) - V_{vent} \sin \varepsilon \cdot \cos \psi \cdot \sin \varphi \end{cases}$$

### 1.2 °) La vitesse de la fusée

On décide d'orienter le repère lié au vent  $R_a (x_a, y_a, z_a)$  de telle sorte que la vitesse de la fusée soit uniquement portée par la première composante de ce repère, à savoir  $x_a$ .

### 1.3 °) Expression de la vitesse relative

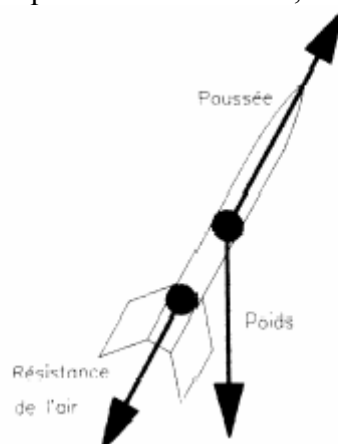
Comme on a exprimé auparavant que  $V_{itesse\_relative\_R} = -V_{itesse\_absolue\_R} + V_{itesse\_vent}$

Finalement il vient :

$$V_{relative} = \begin{cases} V_{r_x} = -u + V_{vent\_x} \\ V_{r_y} = -v + V_{vent\_y} \\ V_{r_z} = -w + V_{vent\_z} \end{cases}$$

### 2 °) Bilan des forces

Si on étudie les efforts qui s'appliquent sur notre fusée, on aboutit à :



### **Bilan des forces s'appliquant à la fusée**

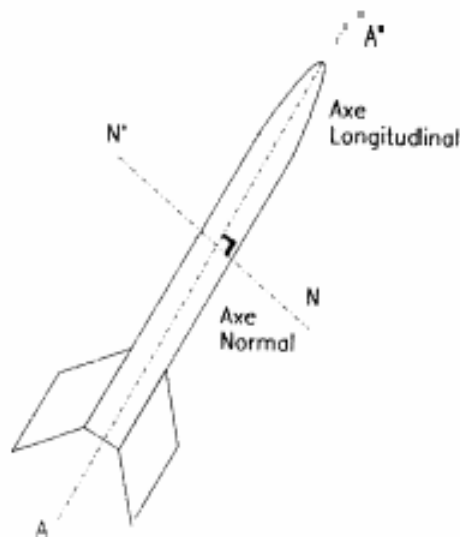
D'après la figure précédente, on dénombre à 3 les forces qui s'appliquent à la fusée.

- le poids
- les efforts aérodynamiques (traînée, portance)
- la poussée

L'évolution de ces trois forces et leur influence relative vont régir le comportement de la fusée et déterminer sa trajectoire et ses performances.

La méthode la plus naturelle consiste à travailler selon les deux axes suivants :

- **L'axe de la fusée.** En fonction de leur sens, les forces exercées dans cette direction vont favoriser ou freiner le déplacement de l'engin et influencer ses performances, en particulier l'altitude de culmination.
- **L'axe perpendiculaire à la fusée (ou axe normal).** Les forces exercées selon cette direction contribuent à faire tourner la fusée dans un sens ou dans l'autre autour de sa position d'équilibre. C'est la combinaison de ces effets antagonistes qui détermine la stabilité de la fusée.



#### Décomposition des forces en composante axiale et normale

En exprimant ces formes sous forme de torseurs afin d'expliciter chacune de leurs composantes, on obtient, dans le repère R lié à la fusée :

$$\text{Poids} = \begin{Bmatrix} -mg \cdot \sin \theta & 0 \\ mg \cdot \cos \theta \sin \varphi & 0 \\ mg \cos \theta \cos \varphi & 0 \end{Bmatrix}_R$$

$$\text{Poussée} = \begin{Bmatrix} X_f & L_f \\ Y_f & M_f \\ Z_f & N_f \end{Bmatrix}_R$$

$$\text{Forces aéro} = \begin{Bmatrix} Xa & La \\ Ya & Ma \\ Za & Na \end{Bmatrix}_R$$

où  $Xa$  représente la traînée, définie par  $Xa = -\frac{1}{2} * \rho * S_{\text{trainée}} * V_{Rx}^2 Cx$

où  $Ya$  représente la force latérale, définie par  $Ya = \frac{1}{2} * \rho * S_{\text{réf}} * V_{Rx}^2 Cy$

où  $Za$  représente la portance, définie par  $Za = -\frac{1}{2} * \rho * S_{\text{réf}} * V_{Rx}^2 Cz$

où  $La$  représente le moment de roulis, défini par  $La = \frac{1}{2} * \rho * S_{\text{trainée}} * V_{Rx}^2 Cx.lx$

où  $Ma$  représente le moment de tangage, défini par  $Ma = \frac{1}{2} * \rho * S_{\text{réf}} * V_{Rx}^2 Cz.lz$

où  $Na$  représente le moment de lacet, défini par  $Na = \frac{1}{2} * \rho * S_{\text{réf}} * V_{Rx}^2 Cy.ly$

### 3 °) Densité de l'air

Comme il a été précisé dans les hypothèses, le vol s'effectue dans une atmosphère standard, où la densité  $\rho$  est connue pour une altitude donnée. Du fait que les fusées étudiées ici ne dépassent jamais les 11 000m d'altitude, on considérera donc l'expression simplifiée ci-dessous de la densité de l'air  $\rho$  :

$$\rho = \rho_0 \cdot \frac{20000 - h}{20000 + h}$$

Avec :  $\rho_0 = 1.225 \text{ kg/m}^3$   
 $h$  l'altitude de la fusée à l'instant considérée (en m).

### 4 °) Coefficients aérodynamiques

Nous venons de constater que les efforts aérodynamiques faisaient intervenir 3 coefficients,  $Cx$ ,  $Cy$  et  $Cz$ , dont nous devons donc déterminer les valeurs pour la suite de notre étude.

#### 4.1 °) Détermination du $Cz$

Le  $Cz$  est défini par la formule :  $Cz = Cz_0 + Cz\alpha * \alpha$ . Mais du fait de la symétrie de la fusée,  $Cz_0$  est nul, d'où l'expression se réduit à  $Cz = Cz\alpha * \alpha$ .

Il nous faut donc déterminer les 2 paramètres  $Cz\alpha$  et  $\alpha$  afin de connaître le  $Cz$  de notre fusée à chaque instant du vol.

D'après les équations relatives à  $\alpha$  et  $\beta$ , on sort :

$$\frac{Vr_z}{V} = \sin \alpha \Leftrightarrow \alpha = \arcsin \frac{Vr_z}{V},$$

où V est la vitesse de la fusée, c'est-à-dire :  $V = \sqrt{Vr_x^2 + Vr_y^2 + Vr_z^2}$ .

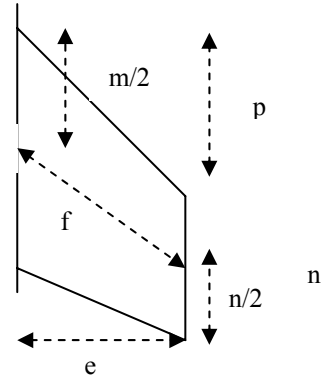
D'où on en déduit la valeur de l'incidence à chaque instant.

Il ne nous reste donc plus qu'à déterminer la valeur du  $Cz_\alpha$ , qui dépend directement de la géométrie de la fusée.

- Pour l'empennage, Le  $Cz_{\alpha\_empennage}$  est défini par la formule de Barrowman:

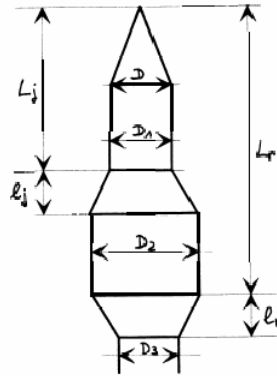
$$Cz_{\alpha\_empennage} = 4nb_{ailerons} \left( \frac{e}{D_{ogive}} \right)^2 \left( 1 + \frac{D_{Ailerons}}{2e + D_{Ailerons}} \right) \left( \frac{1}{1 + \sqrt{1 + \left( \frac{2f}{m+n} \right)^2}} \right)^m$$

Où  $f = \frac{\sqrt{p^2 + e^2} + \sqrt{(p+n-m)^2 + e^2}}{2}$



Et D le diamètre de la fusée.

- Pour les rétreints



On démontre que pour un rétreint, le coefficient de portance  $Cz_{\alpha\_retrait}$  est donné par la formule :

$$Cz_{\alpha\_retrait} = 2 \cdot \left[ \left( \frac{D3}{D} \right)^2 - \left( \frac{D2}{D} \right)^2 \right]$$

- Pour les jupes

En reprenant la même nomenclature que pour les retraints évoqués ci-dessus, on démontre que pour une jupe, le coefficient de portance  $Cz_{\alpha\_jupe}$  est donné par la formule

$$C_{Z_{\alpha\_jupe}} = 2 \cdot \left[ \left( \frac{D2}{D} \right)^2 - \left( \frac{D1}{D} \right)^2 \right]$$

- Pour le fuselage (ogive + tube), on a toujours, et ce quelle que soit la forme de celle-ci,  $C_{Z_{\alpha\_fuselage}} = 2$ .

Finalement, le coefficient de portance global de la fusée  $C_{Z_{\alpha\_global}}$  de la fusée en additionnant tout simplement les coefficients des différentes parties de la fusée :

On obtient au final :

$$C_{Z_{\alpha\_global}} = C_{Z_{\alpha\_ogive}} + C_{Z_{\alpha\_empennage}} + \sum_{i=1}^{i=n} C_{Z_{\alpha\_retreint(i)}} + \sum_{j=1}^{j=m} C_{Z_{\alpha\_jupe(j)}}$$

#### 4.2 °) Détermination du $C_y$

Par analogie avec le  $C_z$ , on a :  $C_y = C_{y0} + C_{y\beta} * \beta$ , expression qui peut se réduire du fait de la symétrie de la fusée par  $C_y = C_{y\beta} * \beta$

D'après les équations relatives à u et v, on sort :

$$\frac{V_{r_y}}{V} = \sin \beta \Leftrightarrow \beta = \arcsin\left(\frac{V_{r_y}}{V}\right)$$

où V est la vitesse de la fusée, c'est-à-dire :  $V = \sqrt{V_{r_x}^2 + V_{r_y}^2 + V_{r_z}^2}$ .

Enfin, du fait de la symétrie du corps étudié, on a toujours  $C_{y\beta} = C_{z\alpha}$  pour l'ogive comme pour l'empennage, et donc la relation suivante:

$$C_{y\beta\_global} = C_{z\alpha\_global}$$

#### 4.3 °) Détermination du $C_x$

Si le calcul des coefficients aérodynamiques est semblable dans les deux cas précédents, cela change pour le coefficient de traînée. En effet, celui-ci varie également en fonction de l'incidence, et est défini par :

$$C_x = C_{x_0} + \frac{1}{2} \cdot \alpha \cdot C_Z$$

D'après les calculs effectués précédemment, on connaît déjà l'incidence  $\alpha$  et le coefficient de portance  $C_z$ . Il ne nous reste donc plus qu'à connaître la valeur du  $C_{x_0}$ .

Cependant, si du fait de la symétrie de la fusée autour de son axe, les coefficients  $C_{z_0}$  et  $C_{y_0}$  étaient nuls, ce n'est pas le cas du  $C_{x_0}$ . Malheureusement, ce coefficient ne peut être déterminé qu'expérimentalement, soit en soufflerie, soit à l'aide d'un logiciel de calcul en « mécanique des fluides ».

Par cette raison, la valeur du  $C_{x_0}$  de la fusée devra donc être un paramètre rentré manuellement par l'utilisateur, qui l'aura, par la méthode de son choix, déterminé auparavant à l'aide de l'une de ces deux méthodes.

#### 5 °) Détermination des surfaces mises en jeu

Nous avons vu précédemment que les efforts aérodynamiques, portance, traînée et force latérale, font intervenir des surfaces de références.

- Le maître couple

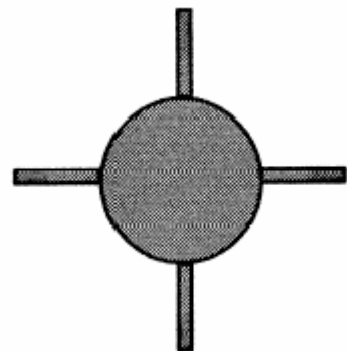
Cette surface correspond à la surface de la fusée vue de face comme ci-dessous : la surface hachurée est celle perpendiculaire à la direction de la vitesse dans le repère fusée, qui rappelons-le, est notre repère d'étude. On appelle cette surface le « maître couple ».

D'après la figure ci-contre, on en déduit directement que le maître couple vaut

$$\text{Maître\_couple} = \Pi \cdot D^2 / 4 + 4L \cdot e_{\text{aileron}}$$

Avec :

- $D$  le diamètre maximal de la fusée
- $L$  l'envergure des ailerons
- $e_{\text{aileron}}$  l'épaisseur des ailerons
  
- La surface de référence



Nous avons vu précédemment que les coefficients de portances déterminés pour les différentes parties de la fusée, à savoir pour l'ogive, l'empennage, les jupes et les rétreints, étaient issus des formules de Barrowman, qui furent calculées à partir en utilisant une surface



de référence. L'objectif de ce projet n'étant pas de re-démontrer ces formules, nous prendrons donc le résultat final des démonstrations.

Ainsi donc, la surface de référence considérée est :

$\text{Surface\_référence} = \Pi.D_{\text{ogive}}^2/4$
--

Avec  $D_{\text{ogive}}$  le diamètre de la fusée à la base de son ogive

#### 6 °) Détermination des matrices d'inertie

Afin d'obtenir une étude de la stabilité dynamique optimale, il sera laissé libre à l'utilisateur de saisir les valeurs des matrices d'inertie calculées à l'aide d'un logiciel de CAO. Cependant en première approximation, on calculera l'inertie de la fusée sur ses trois axes par approximation.

Ainsi, nous utiliserons les formules d'inertie du cylindre pour le tube, celles d'un cône pour l'ogive et enfin celles d'une plaque plane. L'inertie Prépondérante est l'inertie du tube énoncé par la formule ci dessous :

$$InertieTube = \begin{pmatrix} \frac{Diametre^2}{2} & 0 & 0 \\ 0 & \frac{LongueurTube^2}{12} & 0 \\ 0 & 0 & \frac{LongueurTube^2}{12} \end{pmatrix}$$

Les inerties sont ensuite pondérées de leurs masses en fonction du rapport de volume entre le tube, les ailerons et l'ogive.

Nous considérerons de plus, l'évolution des inerties fixes au cours du vol de la fusée, ceci se justifiant par le fait que l'inertie de la poudre est négligeable, et que de plus, la variation du centre de gravité bien que non négligeable à l'échelle de la fusée, n'engendre pas de modifications importantes sur les valeurs des trois moments d'inertie.

#### 7°) Vitesses de rotation et angles d'Euler

Nous appellerons pour la suite de notre étude, et ce afin d'alléger la lecture des équations, p, q et r les vitesses de rotations autour de chaque axe de la fusée. On définit ainsi le vecteur rotation  $\Omega_{R/R_0} = (p, q, r)^T$

La relation liant ces vitesses de rotation à leur angles respectifs est défini par la matrice d'Euler comme ci-dessous :

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{bmatrix} 0 & 1 & -\sin \theta \\ \cos \varphi & 0 & \cos \theta \sin \varphi \\ -\sin \varphi & 0 & \cos \theta \cos \varphi \end{bmatrix} \begin{pmatrix} \theta' \\ \varphi' \\ \psi' \end{pmatrix}$$

Par inversion de la matrice, on obtient finalement les relations suivantes :

$$\theta' = \cos \varphi . q - \sin \varphi . r$$

$$\varphi' = p + \tan \theta . (\cos \varphi . r + \sin \varphi . q)$$

$$\psi' = \frac{\cos \varphi}{\cos \theta} . r + \frac{\sin \varphi}{\cos \theta} . q$$

### 8 °) Mise en équation

D'après les équations de la mécanique du vol, on a dans le repère fusée :

Equations de forces :

$$m(u' + qv - rw) = X_a + X_f - mg \sin \theta$$

$$m(v' + rw - pu) = Y_a + Y_f + mg \cos \theta \sin \varphi$$

$$m(w' + pu - qv) = Z_a + Z_f + mg \cos \theta \cos \varphi$$

Equations des moments :

$$A p' + (C-B) qr = L_a + L_f$$

$$B q' + (A-C) pr = M_a + M_f$$

$$C r' + (B-A) pq = N_a + N_f$$

## V °) Paramètres de stabilité d'une fusée

Comme nous l'avons expliqué au II, une fusée doit être stable pour pouvoir être lancée. Par ailleurs, nous avons également montré que deux paramètres intervenaient dans l'étude de cette stabilité, à savoir :

- La valeur du coefficient  $C_z$  de la fusée.
- la valeur de la marge statique de la fusée.

Si nous avons déjà établi précédemment les formules permettant de déterminer son coefficient de portance, il nous reste encore à déterminer la marge statique de celle-ci.

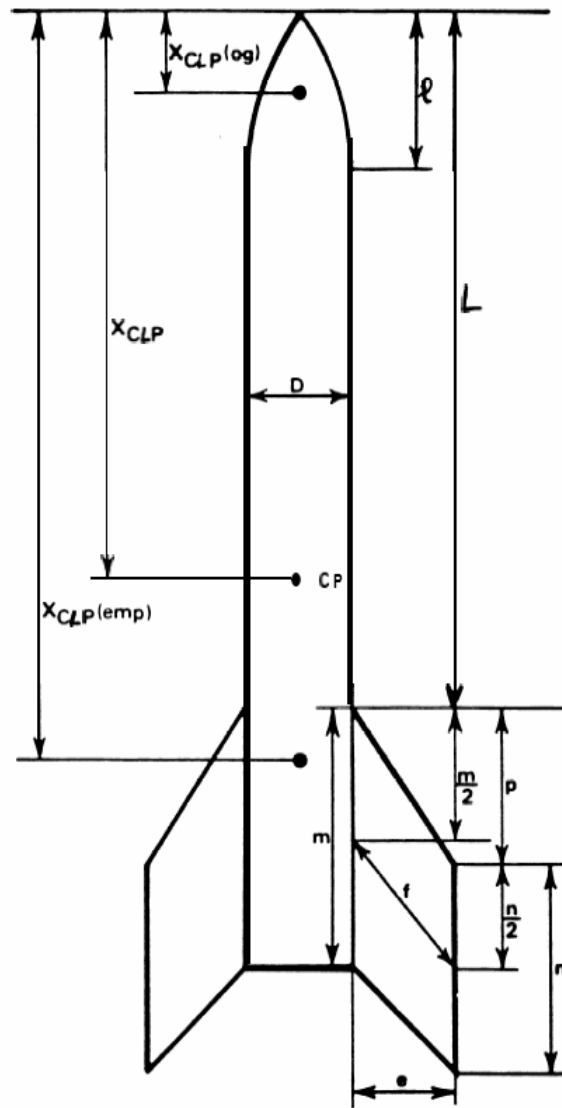
Rappel : la marge statique est la distance entre le centre de gravité de la fusée et son centre de portance aérodynamique

### 1 °) Détermination du Centre de Portance aérodynamique CPA

Le centre de portance aérodynamique est le point où s'appliquent les efforts aérodynamiques de la fusée, à savoir traînée, force latérale et portance.

Par ailleurs, on rappellera que, par hypothèse, la surface du corps de la fusée ne porte pas, ce qui se justifie par le fait que, bien que la fusée se mette en incidence au cours de son vol, cette incidence reste néanmoins assez faible, et donc seul l'ogive et l'empennage de la fusée la verront.

Les dimensions intervenant dans le calcul du centre de portance aérodynamique de la fusée sont référencées sur le dessin ci-après :



### 1.1 °) CPA de l'ogive

Le centre de portance aérodynamique est une grandeur physique qui dépend uniquement de la forme géométrique de l'objet considéré. Il est donc normal de trouver différentes valeurs en fonction des types d'ogives utilisés. Ainsi, on démontre que :

- Pour une ogive conique,  $X_{CPA} = \frac{2}{3}l$
- Pour une ogive parabolique,  $X_{CPA} = \frac{1}{2}l$

Où  $l$  représente la longueur totale de l'ogive, l'origine du repère étant fixée à la pointe de celle-ci.

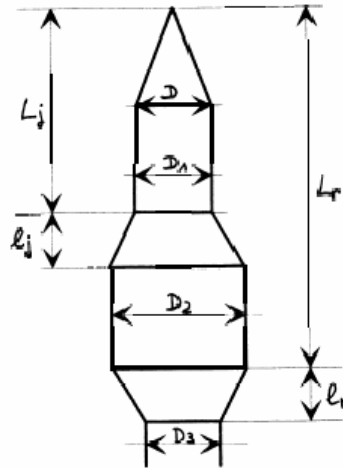
### 1.2 °) CPA de l'empennage

Par analogie avec l'ogive, le centre de portance aérodynamique de l'empennage dépend uniquement des dimensions des ailerons de la fusée. Ainsi, on montre, pour un empennage, le centre de portance est défini par :

$$X_{\text{CPA\_empennage}} = L + \frac{p(m+2n)}{3(m+n)} + \frac{1}{6} \left( m+n - \frac{m.n}{m+n} \right)$$

Où L, p, m et n sont des dimensions de la fusée, représentées sur le dessin de la page précédente.

### 1.3 °) CPA d'un rétreint



En utilisant les notations du dessin ci-dessus, on démontre que :

$$X_{\text{CPA\_rétreint}} = L_R + \left( 1 + \frac{1 - \frac{D_2}{D_3}}{1 - \left( \frac{D_2}{D_3} \right)^2} \right) \cdot \frac{L_R}{3}$$

### 1.4 °) CPA d'une jupe

En utilisant les mêmes notations que pour le rétreint évoqué ci-dessus, on démontre que :

$$X_{\text{CPA\_jupe}} = L_J + \left( 1 + \frac{1 - \frac{D_1}{D_2}}{1 - \left( \frac{D_1}{D_2} \right)^2} \right) \cdot \frac{L_J}{3}$$

### 1.5 °) CPA global de la fusée

Ayant déterminé la position du centre de portance aérodynamique de l'ogive et de l'empennage, et connaissant le coefficient de portance affecté à chacun de ces éléments, on peut facilement, à l'aide d'un calcul barycentrique, déterminer le centre de portance aérodynamique global de notre fusée.

Ainsi, on obtient :

$$X_{CPA\_global} = \frac{X_{CPA\_ogive} \cdot C_{z\alpha\_ogive} + X_{CPA\_empennage} \cdot C_{z\alpha\_empennage} + X_{CPA\_rétre\ int} \cdot C_{z\alpha\_rétre\ int} + X_{CPA\_jupe} \cdot C_{z\alpha\_jupe}}{C_{z\alpha\_ogive} + C_{z\alpha\_empennage} + C_{z\alpha\_rétre\ int} + C_{z\alpha\_jupe}}$$

D'après cette formule, on peut donc en déduire que le centre de portance aérodynamique de la fusée est indépendant du temps, et ne varie qu'en fonction de l'incidence  $\alpha$  et du dérapage  $\beta$  de la fusée au cours de son vol.

### 2 °) Détermination du centre de gravité

Au cours de la phase propulsive, de la masse est éjectée par propulseur ce qui induit un déplacement du centre de gravité. Ce déplacement de quelques dizaines de millimètres est négligeable à l'échelle du vol de la fusée, mais il ne l'est pas à l'échelle de la fusée. Ainsi, lors du calcul de la marge statique, on aura besoin de connaître la variation de la position du centre de gravité depuis le décollage.

Ainsi on est amené à calculer la position du centre de gravité au décollage dans un premier temps :

$$X_{cgInitial} = \frac{X_{cgSans\ Pr\ opulseur} \cdot M_{Sans\ Pr\ opulseur} + (X_{cg\ Pr\ opulseur\ Plein} + Emplacement_{Pr\ opulseur}) \cdot M_{Pr\ opulseur\ Initiale}}{M_{Sans\ Pr\ opulseur} + M_{Pr\ opulseur\ Initiale}}$$

Les différentes variables sont saisies par l'utilisateur dans le fichier de données de la fusée.

Puis, on calcule la variation du centre de gravité initial au cours du temps de la façon suivante :

$$Cd_g(t) = X_{cgInitial} - \frac{X_{cgSans\ Pr\ opulseur} \cdot M_{Sans\ Pr\ opulseur} + (X_{cg\ Pr\ opulseur} + Emplacement_{Pr\ opulseur}) \cdot M_{Pr\ opulseur}}{M_{Sans\ Pr\ opulseur} + M_{Pr\ opulseur}}$$

Les variations de masse et de centre de gravité du propulseur sont interpolés à partir de fichiers de points inclus dans les datasheet des propulseurs.

### 3 °) Marge statique

Par définition de la marge statique, il vient directement :

$$MS(t) = \frac{X_{CPA\_ogive} \cdot C_{z\alpha\_ogive} + X_{CPA\_empennage} \cdot C_{z\alpha\_empennage} + X_{CPA\_rétreint} \cdot C_{z\alpha\_rétreint} + X_{CPA\_jupe} \cdot C_{z\alpha\_jupe}}{C_{z\alpha\_ogive} + C_{z\alpha\_empennage} + C_{z\alpha\_rétreint} + C_{z\alpha\_jupe}} - C_{dg}(t)$$

Comme on pouvait s'en douter, la marge statique de la fusée dépend donc du temps.

## VI °) Représentation d'état du système

### 1 °) Choix du vecteur d'état

Notre fusée possède au totale 6 degrés de liberté autour de ces axes : 3 translations et 3 rotations. Comme, à chaque degré de liberté, on associe 2 composantes du vecteur d'état, alors on en déduit que notre vecteur d'état sera constitué de 12 composantes au total.

On choisit donc le vecteur d'état suivant :  $\text{Vect} = \begin{pmatrix} x \\ y \\ z \\ u \\ v \\ w \\ \varphi \\ \theta \\ \psi \\ p \\ q \\ r \end{pmatrix}_R$

### 2 °) Choix du vecteur d'entrée

Le vecteur d'entrée regroupe l'ensemble des données connues à chaque instant du vol.

Dans notre cas, le vecteur d'entrée serait donc :  $U = \begin{pmatrix} Xf \\ Yf \\ Zf \\ Lf \\ Mf \\ Nf \\ V_{vent} \\ \mathcal{E} \end{pmatrix}_R$ , où l'on y retrouve :

- les 6 composantes du torseur lié aux actions propulsives dans le repère fusée
- les 2 variables permettant, à l'aide des autres variables présentent dans le vecteur d'état, de déterminer les composantes du vent dans le repère fusée.

### 3 °) Représentation d'état

On est en présence d'un système non linéaire. On cherche donc à mettre les équations établies en IV.3 sous la forme :  $\text{Vect}' = f(\text{Vect}, U)$



On obtient alors :

$$\begin{aligned}
 u' &= \frac{1}{m} [ X_a + X_f - mg \sin \theta ] + rv - qw \\
 v' &= \frac{1}{m} [ Y_a + Y_f + mg \cos \theta \sin \varphi ] + pw - ru \\
 w' &= \frac{1}{m} [ Z_a + Z_f + mg \cos \theta \cos \varphi ] + qu - pv \\
 p' &= \frac{1}{A} [ L_a + L_f + (B-C) qr ] \\
 q' &= \frac{1}{B} [ M_a + M_f + (C-A) pr ] \\
 r' &= \frac{1}{C} [ N_a + N_f + (A-B) pq ]
 \end{aligned}$$

En remplaçant les termes  $X_a$ ,  $Y_a$ ,  $Z_a$ ,  $L_a$ ,  $M_a$  et  $N_a$  par leur expressions, cela donne :

$$\begin{aligned}
 u' &= \frac{1}{m} [ -\frac{1}{2} * \rho . S . V_{r_x}^2 C_x + X_f - mg \sin \theta ] + rv - qw \\
 v' &= \frac{1}{m} [ \frac{1}{2} * \rho . S . V_{r_y}^2 C_y + Y_f + mg \cos \theta \sin \varphi ] + pw - ru \\
 w' &= \frac{1}{m} [ -\frac{1}{2} * \rho . S . V_{r_z}^2 C_z + Z_f + mg \cos \theta \cos \varphi ] + qu - pv \\
 p' &= \frac{1}{A} [ \frac{1}{2} * \rho . S . V_{r_x}^2 C_x . l_x + L_f + (B-C) qr ] \\
 q' &= \frac{1}{B} [ \frac{1}{2} * \rho . S . V_{r_z}^2 C_z . l_z + M_f + (C-A) pr ] \\
 r' &= \frac{1}{C} [ \frac{1}{2} * \rho . S . V_{r_y}^2 C_y . l_y + N_f + (A-B) pq ]
 \end{aligned}$$

A noter que les deux variables  $V_{vent}$  et  $\varepsilon$  du vecteur d'entrée, bien que ne figurant pas directement dans les équations précédentes, sont nécessaires pour déterminer les composantes  $V_{r_x}$ ,  $V_{r_y}$  et  $V_{r_z}$ , comme nous l'avons démontré en V.1.

## VII °) Trajectographie de la fusée

Grâce à la démarche et aux calculs que nous venons d'expliciter, nous avons montré que nous étions capable de connaître à chaque instant les valeurs des différentes composantes de notre vecteur d'état dans le repère de la fusée. Néanmoins, l'objectif premier de notre modélisation étant de réaliser une trajectographie dans le repère terrestre, il nous reste donc à déterminer la méthode nous permettant d'aboutir au résultat souhaité.

On connaît à chaque instant les valeurs du vecteur d'état, Vect=

$$\begin{pmatrix} x \\ y \\ z \\ u \\ v \\ w \\ \varphi \\ \theta \\ \psi \\ p \\ q \\ r \end{pmatrix}_R$$

Or, on connaît l'expression de la matrice de transfert du repère fusée au repère terrestre suivante :

$$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = \begin{bmatrix} \cos \psi \cdot \cos \theta & -\sin \psi \cdot \cos \varphi + \cos \psi \cdot \sin \theta \cdot \sin \varphi & \sin \psi \cdot \sin \varphi + \cos \psi \cdot \sin \theta \cdot \cos \varphi \\ \sin \psi \cdot \cos \theta & \cos \psi \cdot \cos \varphi + \sin \psi \cdot \sin \theta \cdot \sin \varphi & -\cos \psi \cdot \sin \varphi + \sin \psi \cdot \sin \theta \cdot \cos \varphi \\ -\sin \theta & \cos \theta \cdot \sin \varphi & \cos \theta \cdot \cos \varphi \end{bmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Comme d'après notre représentation d'état, on connaît les valeurs des vitesses  $u$ ,  $v$  et  $w$  à chaque instant dans le repère  $R$  lié à la fusée, ainsi que les angles liant le repère fusée au repère terrestre, alors on peut facilement en déduire la valeur des vitesses dans ce dernier.

Mais, par définition, si  $\overrightarrow{OM}$  représente le vecteur position de notre fusée dans le repère terrestre  $R_0$ , alors on a :

$$V_{R_0} = \frac{d\overrightarrow{OM}}{dt}_{R_0}, \text{ où } V_{R_0} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}_{R_0}$$

Par conséquent, on en déduit qu'en intégrant les vitesses  $u$ ,  $v$  et  $w$  exprimées dans le repère  $R_0$ , alors on en déduira l'évolution du vecteur position de la fusée dans le repère terrestre, ce qui revient donc à réaliser la trajectographie de notre fusée.

## VIII °) Modèle Simulink

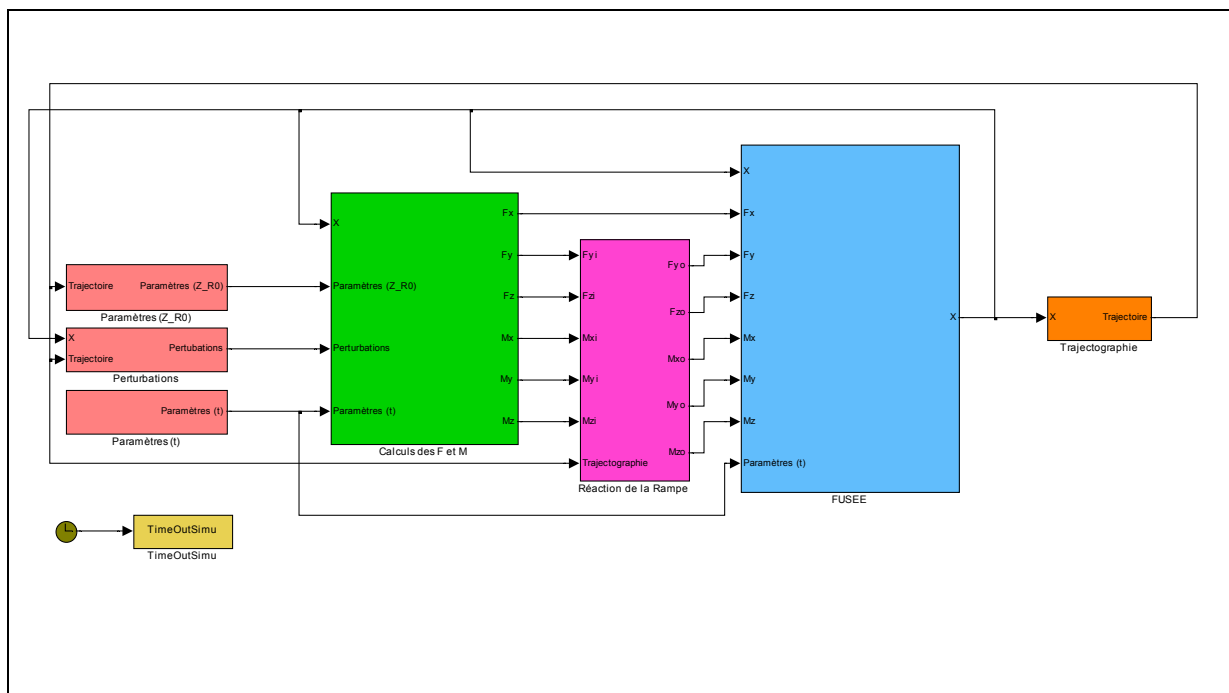
### 1 °) Choix du modèle

Comme nous venons de le constater, notre modèle fait intervenir un nombre relativement important de variables, paramètres et équations. De ce fait, il était fort à parier que le câblage de notre modèle allait être compliqué. De plus, notre modèle se voulant évolutif, dans le cas où nous souhaiterions affiner notre étude, il était nécessaire de le rendre le plus clair possible, afin de pouvoir modifier le plus simplement possible la structure du modèle.

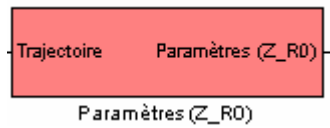
Nous avons donc décidé de concevoir notre modélisation autour de 4 « blocs », qui représentent les 4 grandes étapes de calculs de notre étude :

- les entrées (perturbation, paramètres dépendant du temps et de la position de la fusée)
- les calculs des forces et des moments s'appliquant à notre fusée
- la fusée, qui regroupe l'ensemble des équations de la représentation d'état de notre modèle.
- la trajectographie, qui nous permet d'obtenir l'évolution de la position de la fusée dans le repère absolu.

On aboutit donc finalement au modèle suivant :

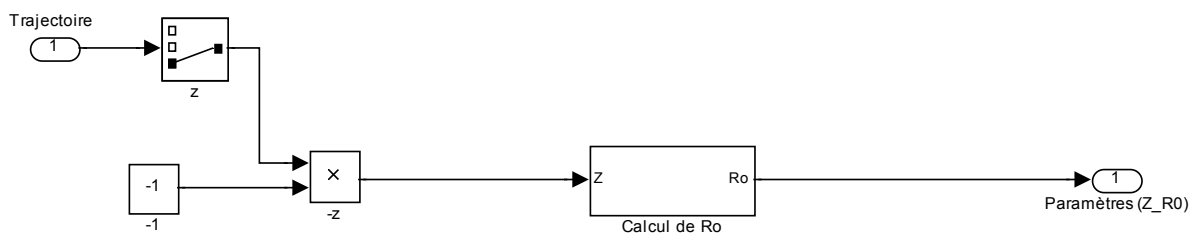


## 2 °) Le bloc « paramètres (Z\_R0) »

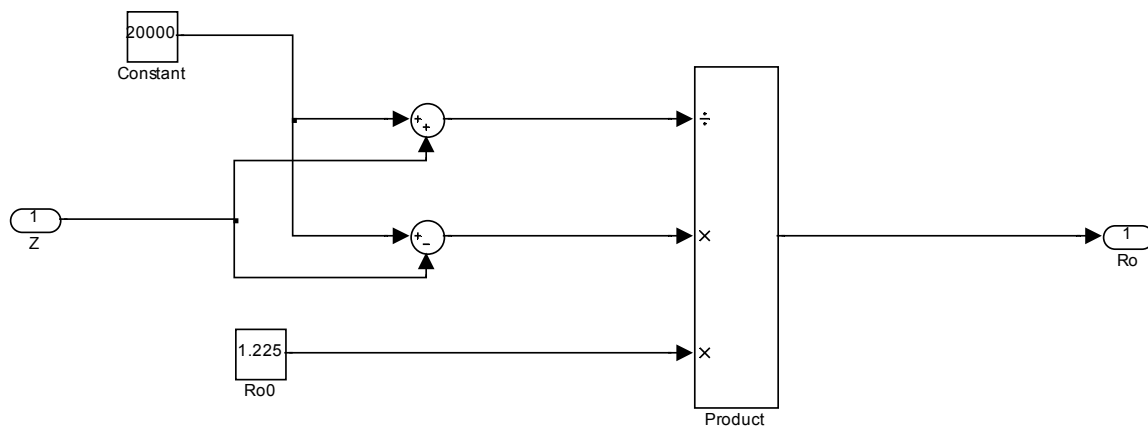


Ce bloc nous permet de déterminer les paramètres qui dépendent directement de la position de la fusée dans le repère absolu au cours du vol

Etudions de plus près la structure de celui-ci :



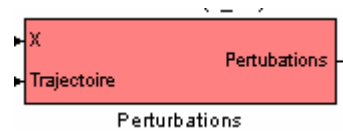
D'après les équations établies précédemment, on remarque que seule la densité de l'air ( fait intervenir l'altitude Z0 de la fusée. Il nous suffit donc de coder dans le bloc « Calcul de Ro » la formule établie V.3. On aboutit alors à :



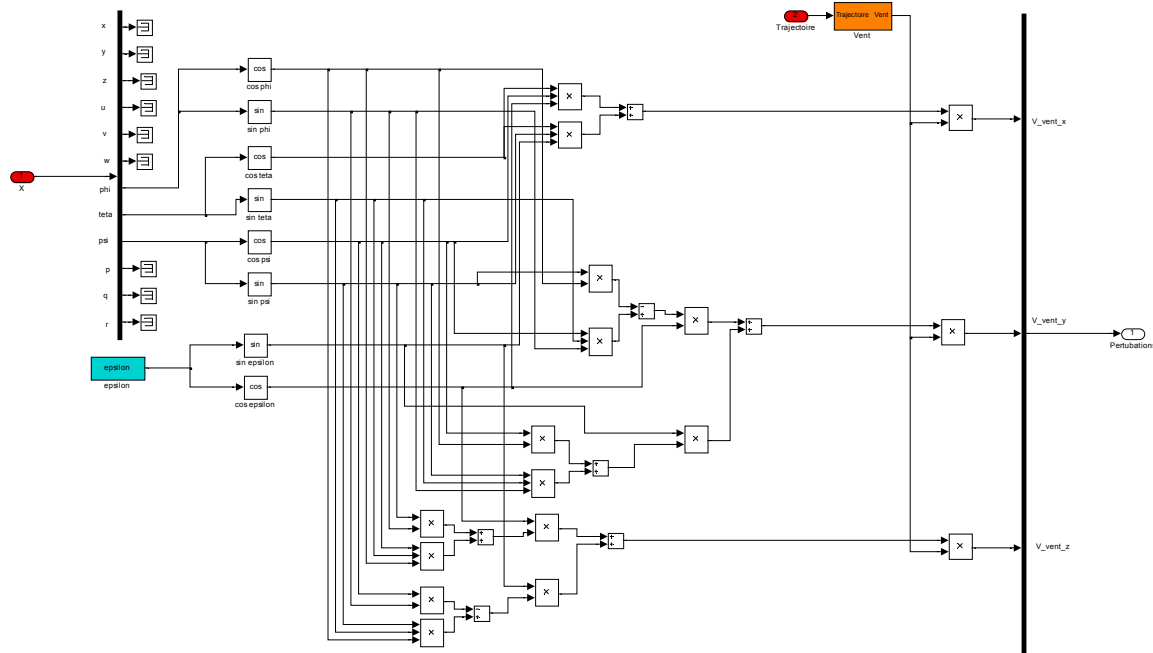
On retrouve ainsi bien la formule :

$$\rho = \rho_0 \cdot \frac{20000 - h}{20000 + h}$$

### 3 °) Le bloc « Perturbation »



Comme nous l'avons précisé dans nos hypothèses, notre étude tient compte du vent. L'architecture du bloc correspondant est la suivante :



On vérifie bien que l'on retrouve le vecteur d'état  $X$  et la trajectoire de la fusée en entrée de ce bloc, ce qui est normal puisque nous avons besoin des angles reliant le repère absolu  $R_0$  au repère  $R$  d'étude ainsi que de la position de la fusée par rapport à la rampe de lancement.

#### 4 °) Le bloc « Paramètres (t) »

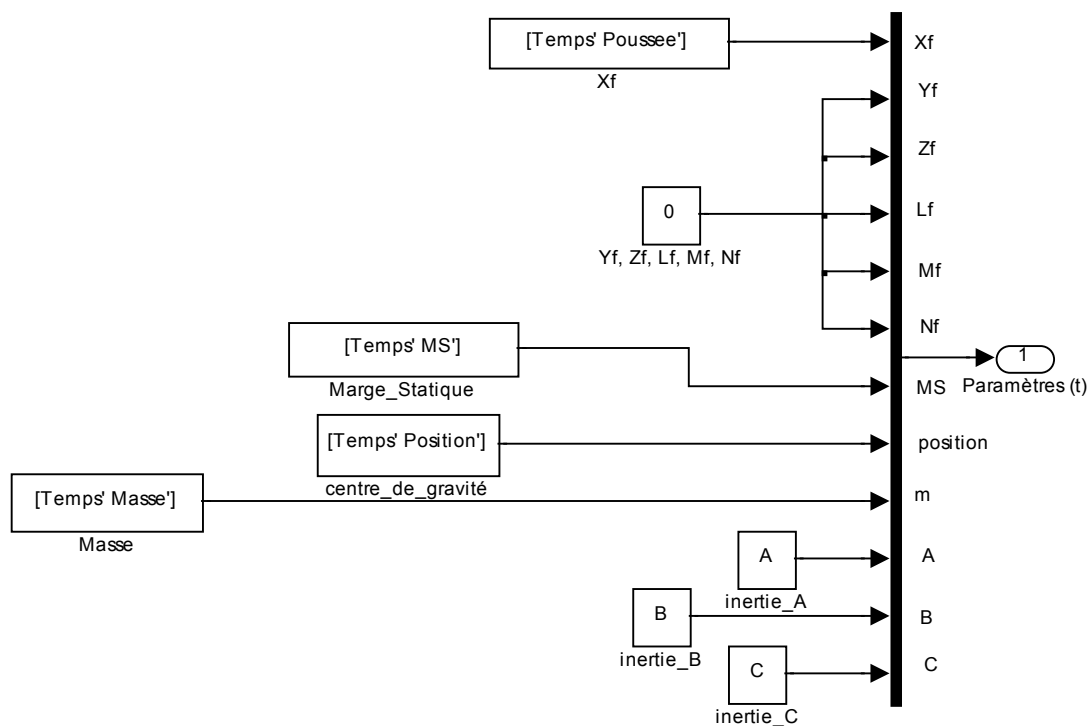


Ce bloc intègre l'ensemble des paramètres qui dépendent exclusivement du temps. De ce fait, il est donc normal que ce bloc ne possède aucune entrée liée aux autres paramètres de la fusée.

Ces paramètres sont :

- la propulsion
- la marge statique
- la masse de la fusée
- l'inertie de la fusée
- l'évolution de la position du centre de gravité de la fusée.

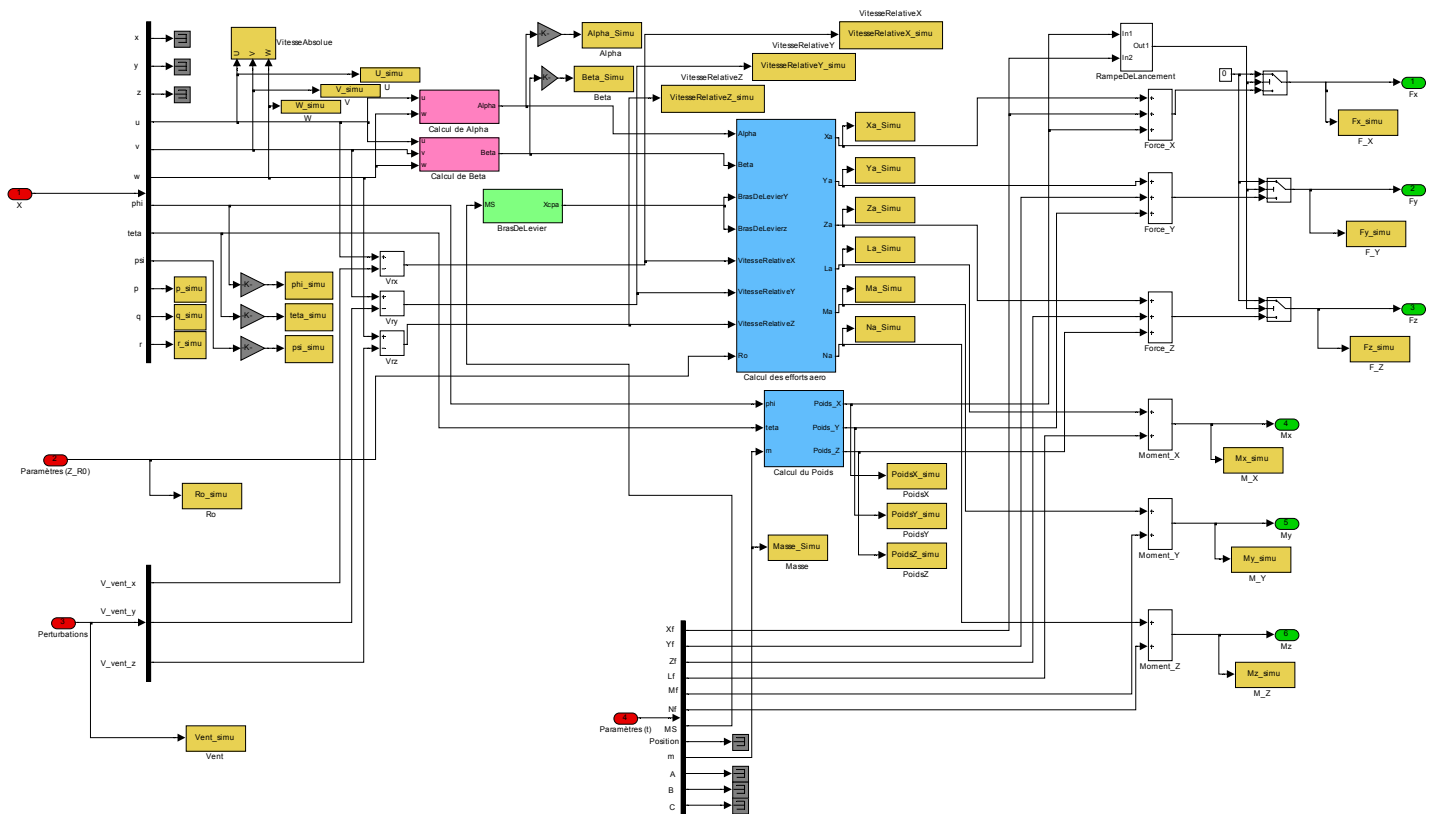
Si on étudie de plus près la structure de ce bloc, on observe le câblage présenté ci-dessous :



## 5 °) Le bloc « Calculs des F et M »

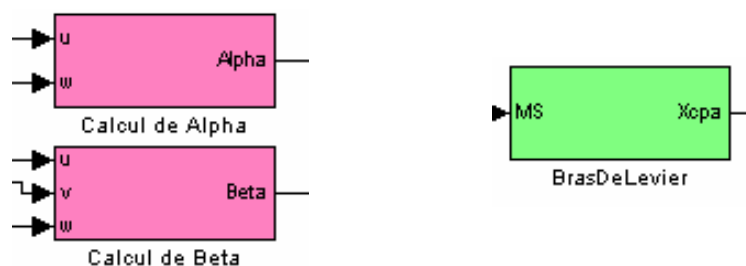
Ce bloc nous permet de déterminer les efforts et le moments qui s'appliquent sur la fusée dans le repère d'étude R. Nous retrouverons donc dans ce bloc le poids, les efforts aérodynamiques et les forces de propulsion.

Si on étudie de plus près la structure de ce bloc, on observe le câblage présenté ci-dessous :



On remarque la présence de nouveaux sous-systèmes à l'intérieur même de ce bloc. En effet, étant donné la complexité des équations mathématiques détaillés précédemment, nous avons décidé d'avoir recours à cette méthode, toujours dans le but de 'alléger et donc de faciliter la lecture du modèle.

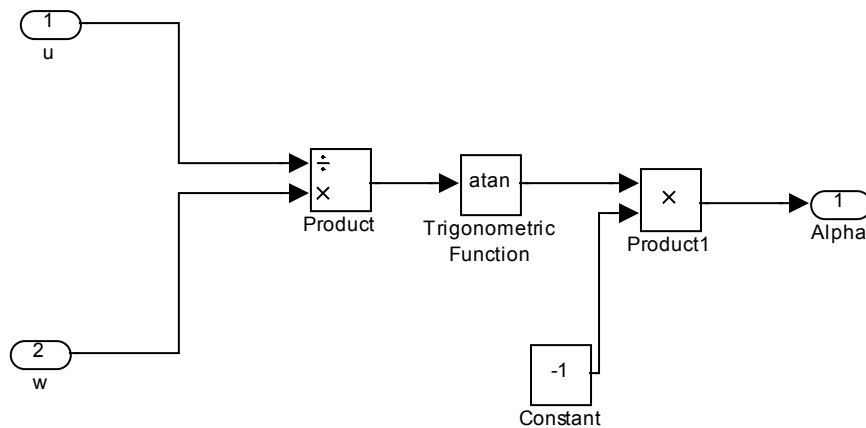
### 5.1 °) Incidence, dérapage et bras de levier



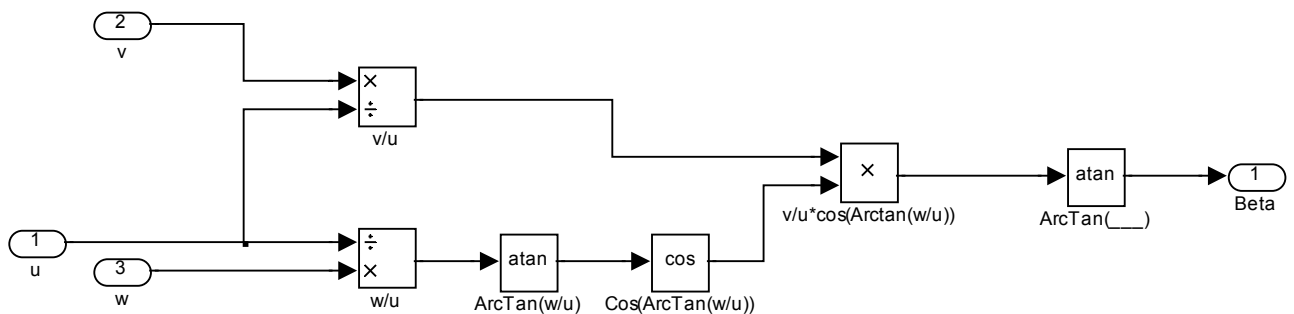
On peut remarquer qu'avant même de commencer à déterminer les composantes des efforts dans le repère fusée et d'en déduire les moments associés, nous avons besoin de déterminer un certain nombre de variable, comme l'angle d'incidence  $\alpha$  et dérapage  $\beta$ , ainsi que les bras de levier des moments

Si on étudie de plus près la structure de ces blocs, on observe les câblages présentés ci-dessous :

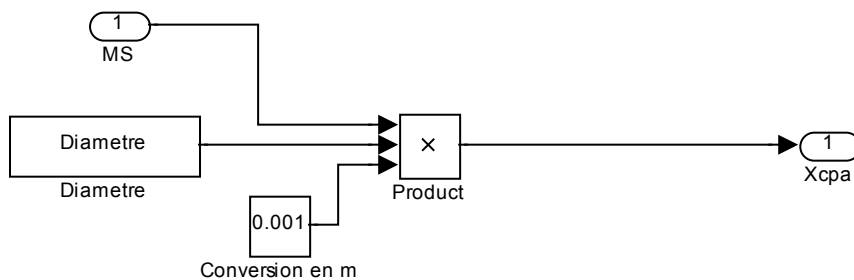
- Pour l'angle d'incidence  $\alpha$



- Pour l'angle de dérapage  $\beta$



- Pour le bras de levier

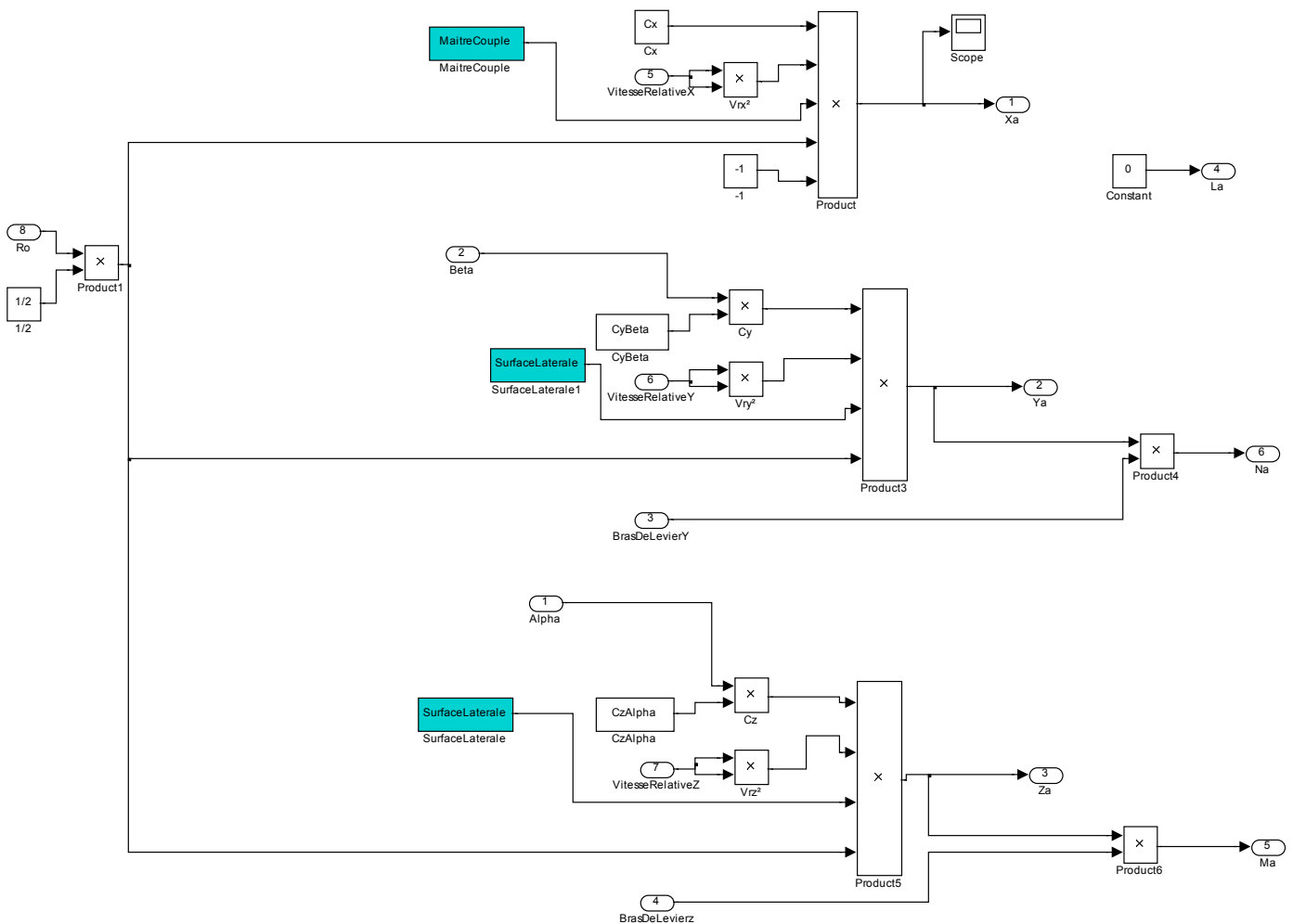




## 5.2 °) Les effort aérodynamiques

Maintenant que nous avons déterminé les angles d'incidence et de dérapage, qui lient le repère aérodynamique au repère de la fusée R, ainsi que le bras de levier pour les moments, nous pouvons donc déterminer les efforts aérodynamiques qui s'appliquent sur la fusée, et ce dans le repère d'étude R

Si on étudie de plus près la structure de ce bloc, on observe le câblage présenté ci-dessous :



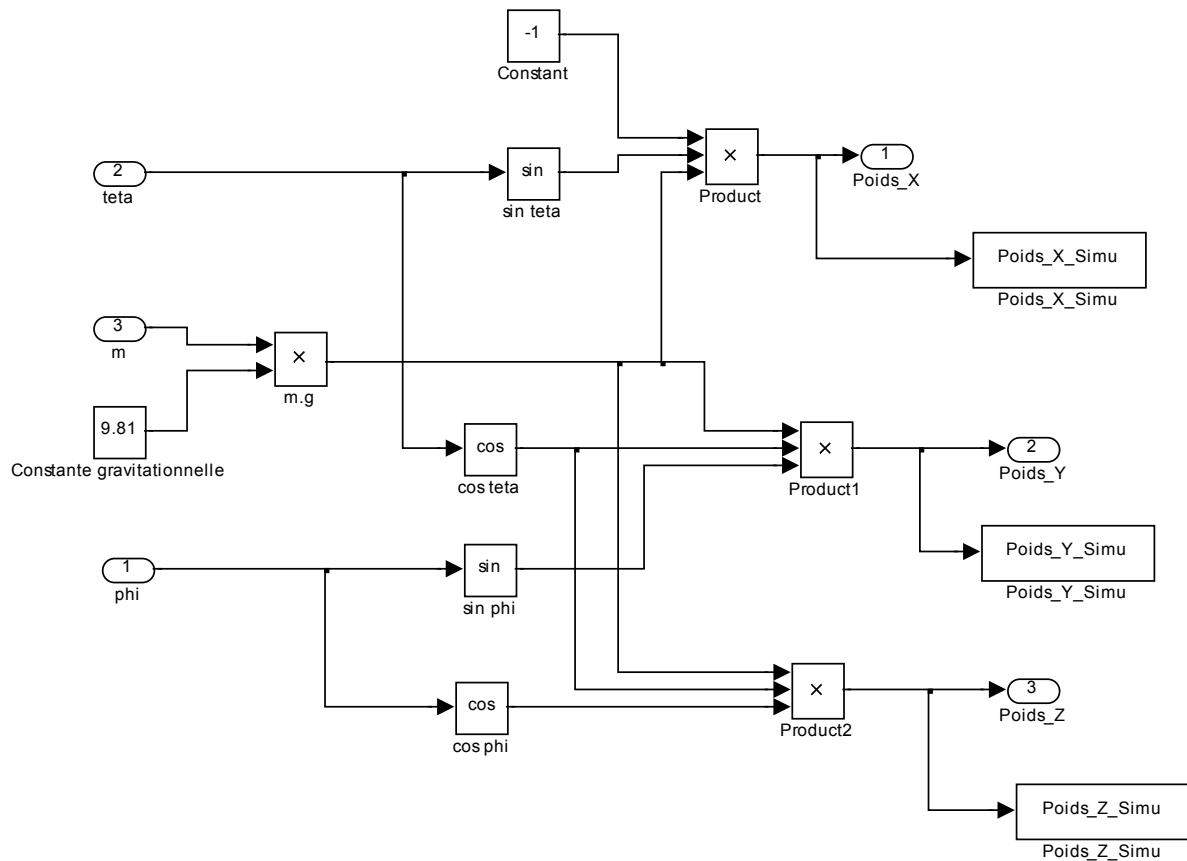
On remarque que les paramètres « MaitreCouple » et « SurfaceLaterale » sont introduites dans le modèle à l'aide de bloc « from Workspace ». En effet, les valeurs de ces surfaces ont été préalablement rentrées dans le fichier Matlab « .m » qui permet de lancer le modèle et donc la simulation.

Par ailleurs, on vérifie bien qu'en sortie de ce bloc, on récupère les valeurs des efforts aérodynamiques dans le repère de la fusée qui, rappelons-le, est notre repère d'étude.

### 5.3 °) Le poids

Traditionnellement, le poids est exprimé dans le repère absolu R0. Nous devons donc le transcrire dans le repère d'étude.

Si on étudie de plus près la structure de ce bloc, on observe le câblage présenté ci-dessous :



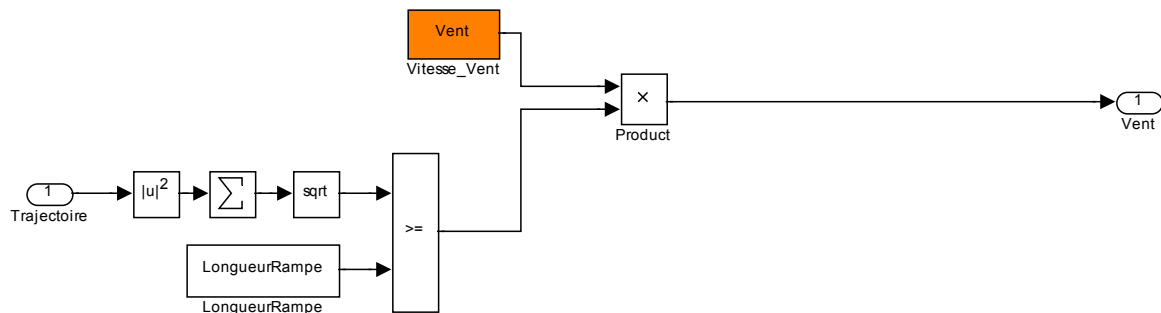
On remarque qu'en entrée de ce bloc, on utilise les valeurs des angles  $\theta$  (teta) et  $\phi$  (phi). Par ailleurs, on vérifie bien que l'on sort le poids exprimé dans le repère d'étude.

Les efforts de propulsion étant directement inclus dans le bloc « Calculs des F et M » par l'intermédiaire du vecteur « paramètres (t) », défini auparavant, et qui sont eux directement exprimé dans le repère fusée, alors on en déduit que l'on connaît les composantes de chaque force et de chaque moment dans le repère d'étude.

A l'aide des sommateurs présents à l'extrémité du bloc « Calculs des F et M », on en déduit que l'on sort bien de ce bloc l'ensemble des forces et des moments associés qui s'appliquent sur notre fusée, et ce dans le repère d'étude.

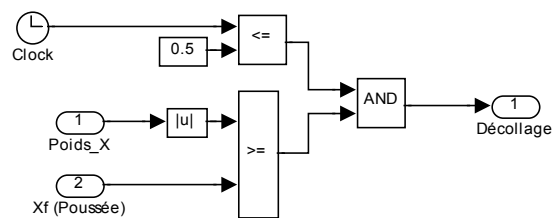
## 6 °) Réaction de la rampe

### 6.1 °) Masquage du vent



Ce bloc a pour objectif d'annuler la valeur du vent tant que la fusée est située dans la rampe. On compare donc la somme quadratique des coordonnées de la fusée dans le repère R0 avec la longueur de la rampe. Tant que la somme quadratique est inférieure à la longueur de la rampe, la vitesse du vent est fixée nulle.

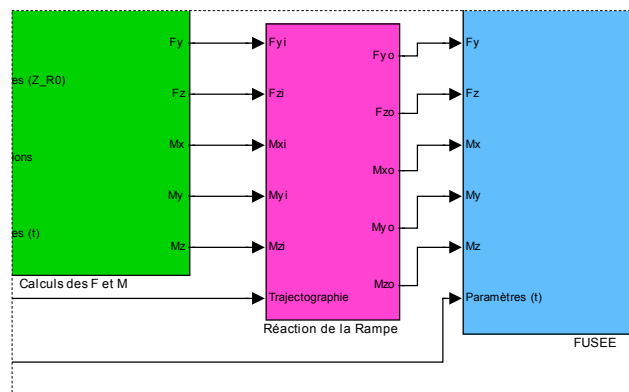
### 6.2 °) Réaction fond de rampe

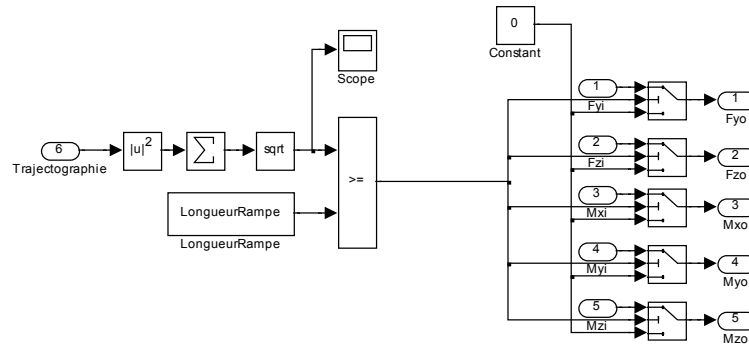


Ce bloc est destiné à simuler la réaction du fond de la rampe. Il évite que la fusée ne chute lorsque la poussée n'a pas encore atteint la valeur du poids sur l'axe x du repère fusée. Une condition temporelle supplémentaire est ajoutée pour que le cas ne se produise pas en fin de propulsion, c'est-à-dire lorsque le poids redevient supérieur à la poussée.

### 6.3 °) Réaction des patins de la rampe

Le but de ce bloc est de prendre en compte la réaction de la rampe sur la fusée.





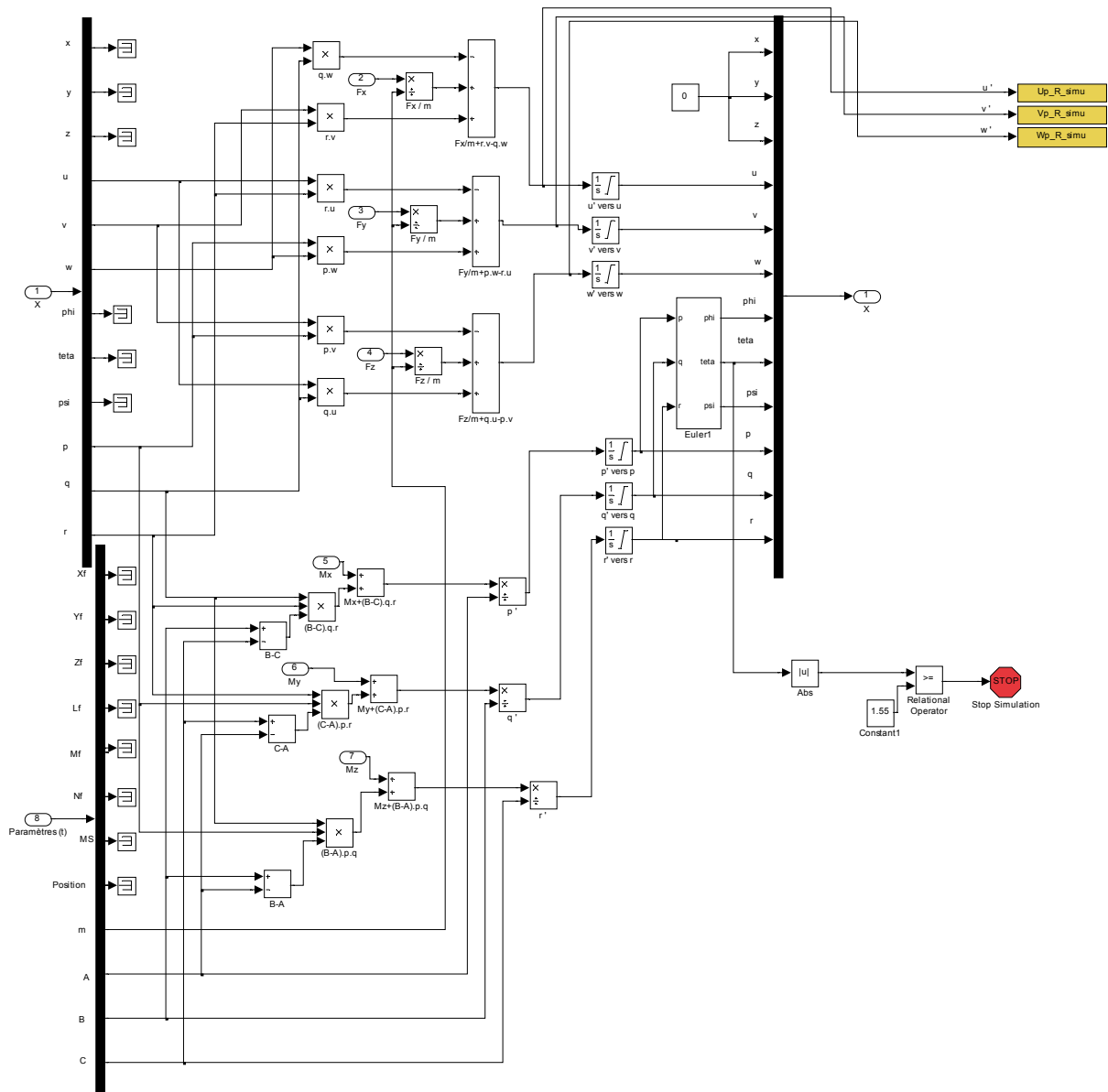
L'effet de ce bloc est d'annuler toute composante de force ou moment qui ne soit pas dans l'axe longitudinal de la fusée tant que cette dernière n'est pas sortie de rampe. Ainsi, on a :

$$\sqrt{X_{R0}^2 + Y_{R0}^2 + Z_{R0}^2} \leq \text{LongueurRampe} \Rightarrow \begin{cases} F_y = 0 \\ F_z = 0 \\ M_x = 0 \\ M_y = 0 \\ M_z = 0 \end{cases}$$

### 7 °) Le bloc « FUSEE »

Ce bloc nous permet tout simplement d'intégrer les équations de la représentation d'état non linéaire du système étudié, les équations ayant été explicité précédemment dans ce rapport.

Si on étudie de plus près la structure de ce bloc, on observe le câblage présenté ci-dessous :



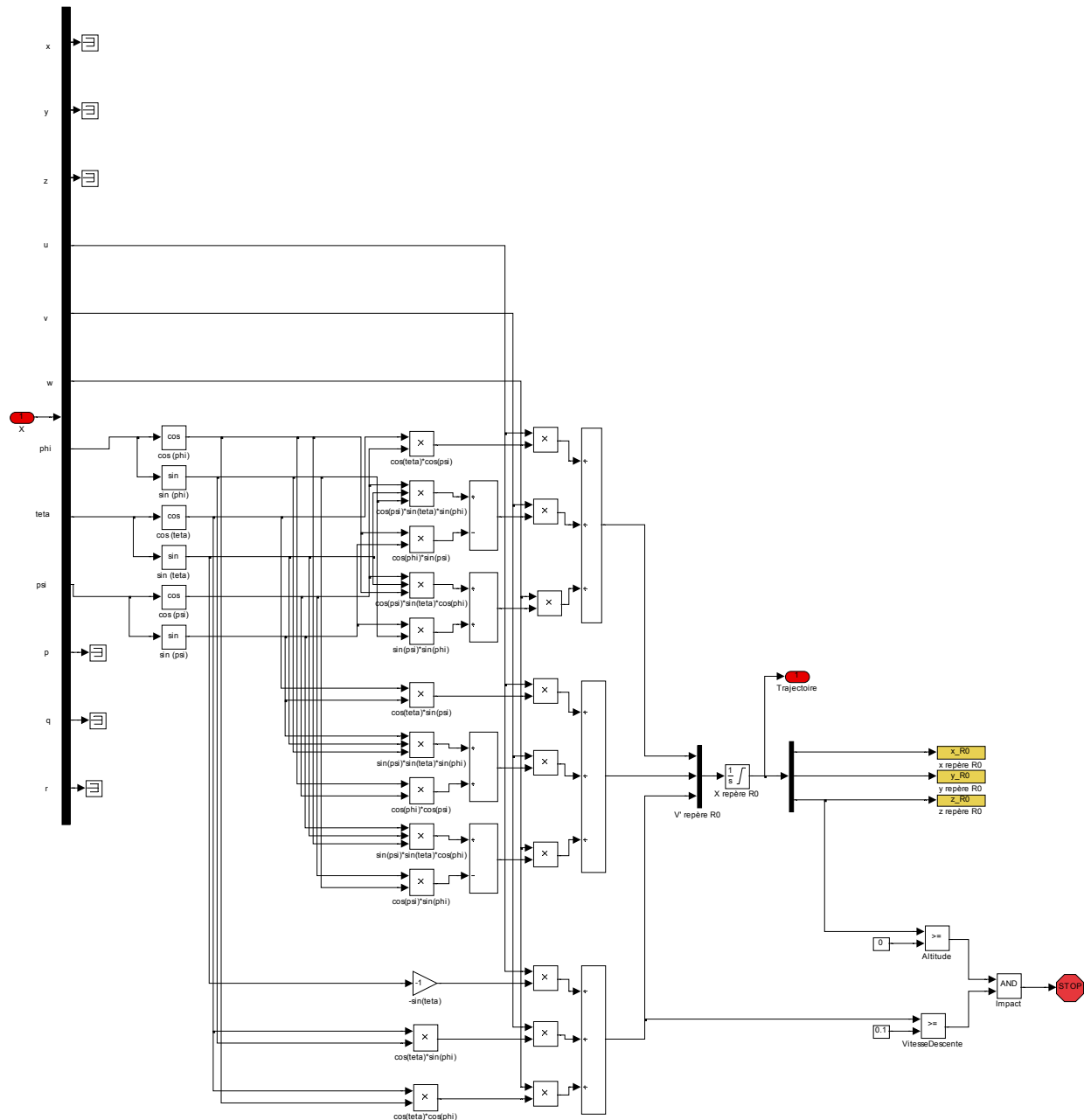
A la fin de notre représentation, on retrouve bien les composantes du vecteur d'état, ce qui nous permet donc de boucler ce vecteur sur l'entrée de notre modèle.

## 8 °) Le bloc « trajectographie »

A l'aide du bloc « FUSEE » détaillée précédemment, nous avons donc pu déterminer les valeurs des composantes de la vitesse de la fusée dans le repère d'étude R, ainsi que la valeur des angles reliant le repère de la fusée au repère absolu.

Néanmoins, si nous avons décidé d'effectuer nos calculs dans le repère de notre projectile, il serait maladroit de réaliser la trajectographie de la fusée dans ce repère. En effet, il est beaucoup plus simple et logique de réaliser celle-ci dans le repère absolu.

En utilisant la méthode décrite en VII, on obtient l'architecture suivante :



On vérifie bien que l'on sort les variables de position x, y et z dans le repère absolu R0, ce qui nous permet donc d'effectuer la trajectographie de la fusée.

### IX °) Validation : Comparaison de trajectoires issues de Trajec et de notre modèle.

Afin de pouvoir porter un jugement sur notre modèle, nous avons effectué des comparaisons entre les résultats de notre modèle et ceux de Trajec. Pour ce faire, nous avons lancé des calculs sur les deux codes avec des fusées de caractéristiques identiques.

Nous avons pris le soin de faire varier tous les paramètres des fusées ainsi que des conditions de lancement.

Nous présenterons alors ci-dessous des résultats portant sur 3 fusées chacune lancée par un propulseur différent.

## 1 °) Fusée Expérimentale « Chamois »

### 1.1 °) Caractéristiques

#### Géométrie de la fusée :

Propulseur :	Chamois
Masse sans propulseur :	8 Kg
Cx :	0.8
Diamètre (D) :	90 mm
Distance Ogive/Aileron (L) :	1600 mm
Distance Ogive/Propulseur (prop) :	1400 mm
Longueur Ogive (l) :	150 mm
Envergure d'un aileron (e) :	150 mm
Emplanture d'un aileron (m) :	150 mm
Profondeur du bord d'attaque (p) :	50 mm
Extrémité de l'aileron (n) :	100 mm
Epaisseur d'un aileron :	2
Nombre d'aileron :	4
Xcg sans propulseur :	1200 mm
Xcg avec propulseur :	1312 mm

#### Lancement :

Vitesses initiales sur les 3 axes :	0 m/s
Site de la rampe :	80 °

### 1.2 °) Résultats synthétiques de Trajec

calcul de trajectoire en cours									
t=0.000s	z-z0=	0m	v=	0m/s	x=	0m	y=	0m	g= 0m/s <sup>2</sup> $\alpha = 80^\circ$
sortie de rampe									
t=0.380s	z-z0=	4m	v=	21m/s	x=	1m	y=	0m	g= 63m/s <sup>2</sup> $\alpha = 80^\circ$
fin de propulsion									
t=2.900s	z-z0=	267m	v=	151m/s	x=	60m	y=	0m	g= 18m/s <sup>2</sup> $\alpha = 77^\circ$
culmination									
t=15.20s	z-z0=	1065m	v=	26m/s	x=	416m	y=	0m	g= 10m/s <sup>2</sup> $\alpha = -1^\circ$
impact									
t=30.80s	z-z0=	0m	v=	123m/s	x=	769m	y=	0m	g= 5m/s <sup>2</sup> $\alpha = -82^\circ$

### 1.3 °) Résultats synthétiques du modèle

Le tableau ci-dessous présente les valeurs synthétiques retournées par notre modèle correspondant à celle retournée ci-dessus par Trajec. L'écart en pourcentage entre les valeurs respectives est calculé en dessous de chaque valeur :

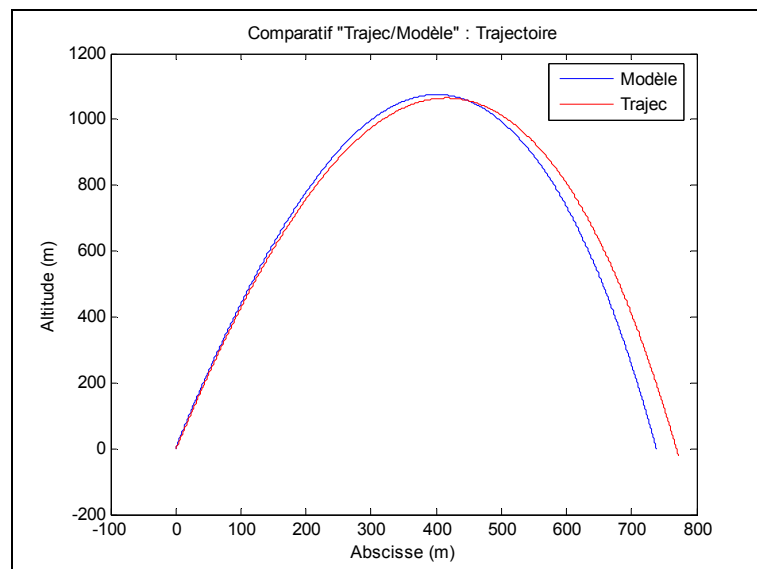


Evénements							
-----							
Début de calcul							
Valeur :	0	0	0,001	0	0	0	80
Ecart (%) :	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Sortie de rampe							
Valeur :	0,3846	3,9392	22,4072	0,6946	0	63,1159	80
Ecart (%) :	1,21%	-1,52%	6,70%	-30,54%	0,00%	0,18%	0,00%
Fin de propulsion							
Valeur :	2,8	239,2225	153,1967	51,2272	0	17,6358	76,9281
Ecart (%) :	-3,45%	-10,40%	1,45%	-14,62%	0,00%	-2,02%	-0,09%
Culmination							
Valeur :	15,2	1075,3	24,9	401,7	0	0,8	1,6
Ecart (%) :	0,00%	0,97%	-4,23%	-3,44%	0,00%	-92,00%	-260,00%
Impact							
Valeur :	30,9705	0	122,9624	738,1445	0	4,378	-82,2644
Ecart (%) :	0,55%	0,00%	-0,03%	-4,01%	0,00%	-12,44%	0,32%

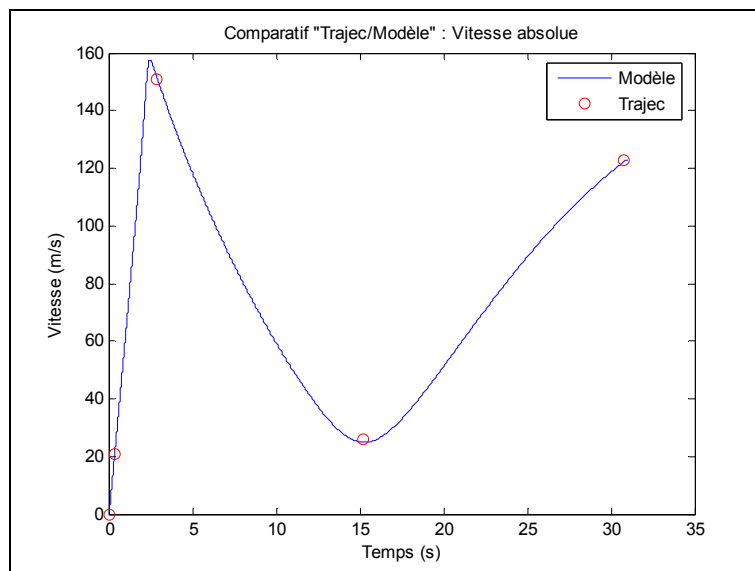
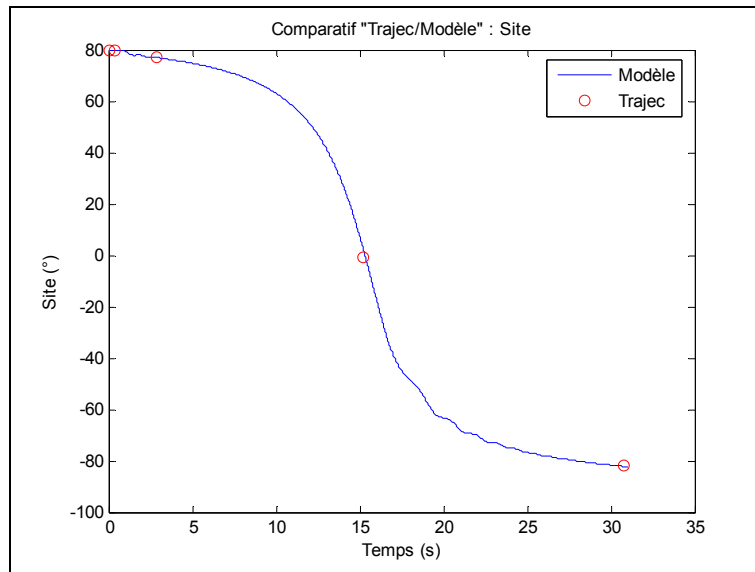
On peut ainsi constater que les écarts de valeurs n'existent que dans le cas où les valeurs retournées par Trajec sont fortement approximées. Cela pourra de plus être mis en évidence ci-dessous grâce aux graphiques retournés par le modèle. Seul l'écart de site à l'apogée est non négligeable, il est dû aux oscillations de la fusée lorsque sa vitesse est faible contrairement au vol de Trajec réalisé à incidence nulle.

#### 1.4 °) Résultats graphiques du modèle

Les résultats du modèle représentés sous la forme de quelques graphes ci-dessous sont complétés par la représentation des points caractéristiques du vol issus du calcul sous Trajec.



Note : La trajectoire affichée ci-dessus en rouge pour les résultats de Trajec est une interpolation polynomiale d'ordre 2 dont les points remarquables sont le décollage, la sortie de rampe, la fin de propulsion, la culmination et l'impact.



### 1.5 °) Comparaison énergétique

Afin d'estimer et de comparer les pertes énergétiques pendant la phase d'ascension entre les deux modèles, on calcule l'énergie cinétique à la culmination dans les deux cas de la manière suivante :

$$E = E_{Potentielle} + E_{Cinétique}$$

$$E = M * g * (z_{Culmination} - z_0) + 1/2 * M * V_{Culmination\_R0}^2$$

On a ainsi les résultats suivants :

Etude énergétique à culmination à partir du calcul de Trajec (J) :

1.1293e+005

Etude énergétique à culmination à partir du calcul du modèle (J) :

1.1370e+005

Ecart énergétique (%) :

0.6838

## 2 °) Mini Fusée « Cariatou »

### 2.1 °) Caractéristiques

#### Géométrie de la fusée :

Propulseur :	Cariatou
Masse sans propulseur :	1 Kg
Cx :	0.8
Diamètre (D) :	40 mm
Distance Ogive/Aileron (L) :	700 mm
Distance Ogive/Propulseur (prop) :	700 mm
Longueur Ogive (l) :	100 mm
Envergure d'un aileron (e) :	50 mm
Emplanture d'un aileron (m) :	60 mm
Profondeur du bord d'attaque (p) :	20 mm
Extrémité de l'aileron (n) :	50 mm
Epaisseur d'un aileron :	2
Nombre d'aileron :	4
Xcg sans propulseur :	500 mm
Xcg avec propulseur :	545 mm

#### Lancement :

Vitesses initiales sur les 3 axes :	0 m/s
Site de la rampe :	80 °

### 2.2 °) Résultats synthétiques de Trajec

calcul de trajectoire en cours									
t=0.000s	z-z0=	0m	v=	0m/s	x=	0m	y=	0m	g= 0m/s <sup>2</sup> A= 80°
sortie de rampe									
t=0.220s	z-z0=	4m	v=	36m/s	x=	1m	y=	0m	g=151m/s <sup>2</sup> A= 80°
fin de propulsion									
t=0.950s	z-z0=	61m	v=	110m/s	x=	11m	y=	0m	g= 19m/s <sup>2</sup> A= 79°
culmination									
t=9.830s	z-z0=	489m	v=	15m/s	x=	164m	y=	0m	g= 10m/s <sup>2</sup> A= 0°
impact									
t=20.41s	z-z0=	0m	v=	84m/s	x=	306m	y=	0m	g= 5m/s <sup>2</sup> A=-83°

### 2.3 °) Résultats synthétiques du modèle

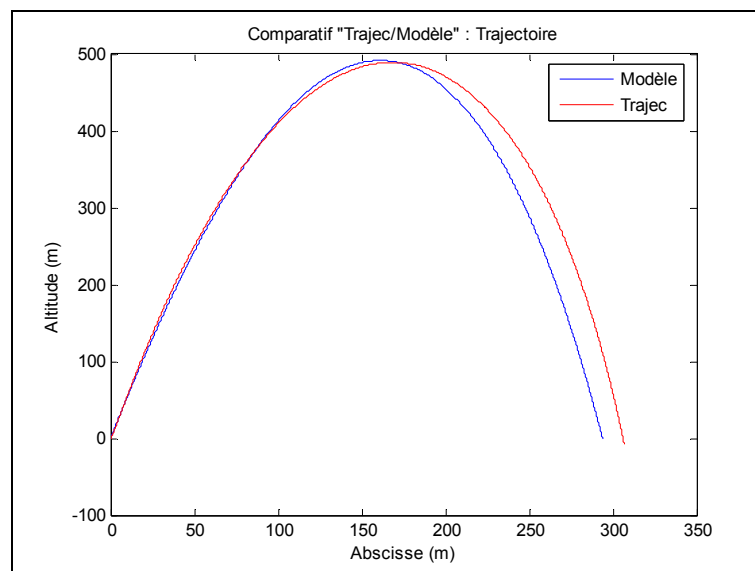
Le tableau ci-dessous présente les valeurs synthétiques retournées par notre modèle correspondant à celle retournée ci-dessus par Trajec. L'écart en pourcentage entre les valeurs respectives est calculé en dessous de chaque valeur :

Evénements							
-----							
Début de calcul							
Valeur :	0	0	0,001	0	0	0	80
Ecart (%) :	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Sortie de rampe							
Valeur :	0,2249	3,9392	35,5832	0,6946	0	150,0363	80
Ecart (%) :	2,23%	-1,52%	-1,16%	-30,54%	0,00%	-0,64%	0,00%
Fin de propulsion							
Valeur :	0,95	60,1209	110,0057	11,1427	0	10,1616	79,0981
Ecart (%) :	0,00%	-1,44%	0,01%	1,30%	0,00%	-36,49%	0,12%
Culmination							
Valeur :	9,8787	491,5931	14,7316	161,2269	0	0,2006	7,7196
Ecart (%) :	0,50%	0,53%	-1,79%	-1,69%	0,00%	-97,99%	0,00%
Impact							
Valeur :	20,4921	0	83,8794	293,7719	0	4,7719	83,9002
Ecart (%) :	0,40%	0,00%	-0,14%	-4,00%	0,00%	-4,56%	-201,08%

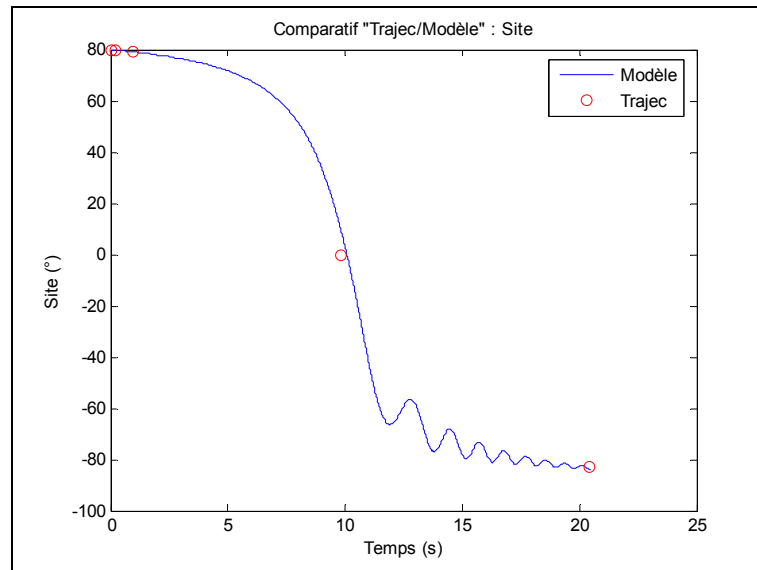
On peut ainsi constater que les écarts de valeurs n'existent que dans le cas où les valeurs retournées par Trajec sont fortement approximées. Cela pourra de plus être mis en évidence ci-dessous grâce aux graphiques retournés par le modèle.

#### 2.4 °) Résultats graphiques du modèle

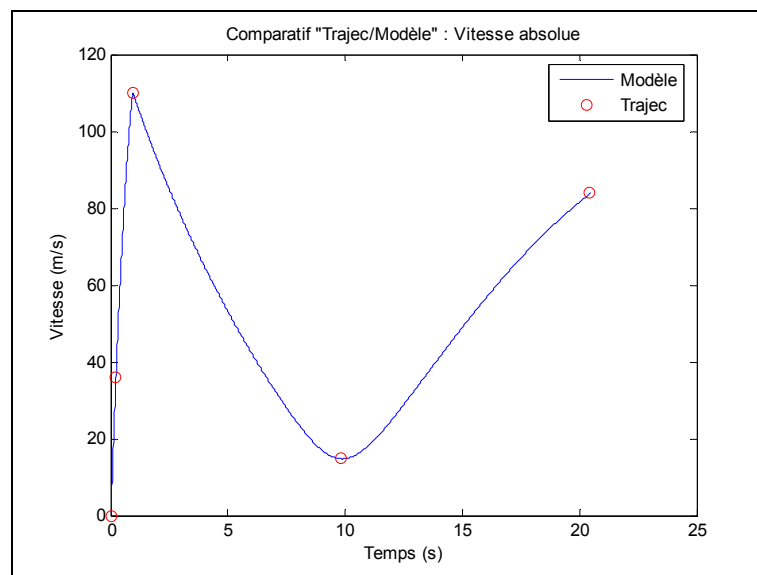
Les résultats du modèle représentés sous la forme des quelques graphes ci-dessous sont complétés par la représentation des points caractéristiques du vol issus du calcul sous Trajec.



Note : La trajectoire affichée ci-dessus en rouge pour les résultats de Trajec est une interpolation polynomiale d'ordre 2 dont les points remarquables sont le décollage, la sortie de rampe, la fin de propulsion, la culmination et l'impact.



L'oscillation mise en évidence pendant la descente est due à un  $C_n$  et une  $MS$  faibles. Ainsi lorsque la vitesse est encore faible en début de descente, l'effort de rappel pour des faibles incidences n'est pas suffisant ce qui amène à de telles oscillations. Elles s'atténuent lorsque la vitesse augmente.



## 2.5 °) Comparaison énergétique

Afin d'estimer et de comparer les pertes énergétiques pendant la phase d'ascension entre les deux modèles, on calcule l'énergie cinétique à la culmination dans les deux cas de la manière suivante :

$$E = E_{Potentielle} + E_{Cinétique}$$

$$E = M * g * (z_{Culmination} - z_0) + 1/2 * M * V_{Culmination\_R0}^2$$

On a ainsi les résultats suivants :

Etude énergétique à culmination à partir du calcul de Trajec (J) :	5.6460e+003
Etude énergétique à culmination à partir du calcul du modèle (J) :	5.6707e+003
Ecart énergétique (%) :	0.4369

### 3 °) Fusée Expérimentale « Isard »

#### 3.1 °) Caractéristiques

##### Géométrie de la fusée :

Propulseur :	Isard
Masse sans propulseur :	3.5 Kg
Cx :	0.75
Diamètre (D) :	80 mm
Distance Ogive/Aileron (L) :	1500 mm
Distance Ogive/Propulseur (prop) :	1400 mm
Longueur Ogive (l) :	85 mm
Envergure d'un aileron (e) :	125 mm
Emplanture d'un aileron (m) :	150 mm
Profondeur du bord d'attaque (p) :	10 mm
Extrémité de l'aileron (n) :	140 mm
Epaisseur d'un aileron :	2
Nombre d'aileron :	4
Xcg sans propulseur :	900 mm
Xcg avec propulseur :	1139 mm

##### Lancement :

Vitesses initiales sur les 3 axes :	0 m/s
Site de la rampe :	75 °

#### 3.2 °) Résultats synthétiques de Trajec

calcul de trajectoire en cours									
t=0.000s	z-z0=	0m	v=	0m/s	x=	0m	y=	0m	g= 0m/s <sup>2</sup> A= 75°
sortie de rampe									
t=0.290s	z-z0=	4m	v=	28m/s	x=	1m	y=	0m	g=121m/s <sup>2</sup> A= 75°
fin de propulsion									
t=1.800s	z-z0=	178m	v=	177m/s	x=	55m	y=	0m	g= 27m/s <sup>2</sup> A= 72°
culmination									
t=13.80s	z-z0=	1012m	v=	32m/s	x=	515m	y=	0m	g= 10m/s <sup>2</sup> A= 0°
impact									
t=29.60s	z-z0=	0m	v=	110m/s	x=	913m	y=	0m	g= 3m/s <sup>2</sup> A=-81°

#### 3.3 °) Résultats synthétiques du modèle

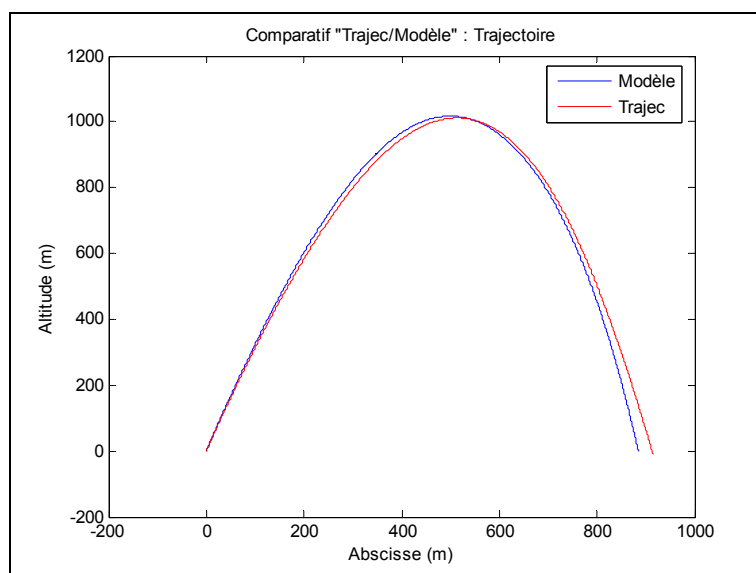
Le tableau ci-dessous présente les valeurs synthétiques retournées par notre modèle correspondant à celle retournée ci-dessus par Trajec. L'écart en pourcentage entre les valeurs respectives est calculé en dessous de chaque valeur :

Evénements							
-----							
Début de calcul							
Valeur :	0	0	0,001	0	0	0	75
Ecart (%) :	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Sortie de rampe							
Valeur :	0,2897	3,8637	31,0413	1,0353	0	117,74	75
Ecart (%) :	-0,10%	-3,41%	10,86%	3,53%	0,00%	-2,69%	0,00%
Fin de propulsion							
Valeur :	1,79	168,8273	177,3807	50,1864	0	27,3153	72,7086
Ecart (%) :	-0,56%	-5,15%	0,22%	-8,75%	0,00%	1,17%	0,98%
Culmination							
Valeur :	13,8	1017,3	30,5	500,6	0	0,6	0,1
Ecart (%) :	0,00%	0,52%	-4,69%	-2,80%	0,00%	-94,00%	0,00%
Impact							
Valeur :	29,6995	0	109,6569	884,5591	0	2,7464	-81,4822
Ecart (%) :	0,34%	0,00%	-0,31%	-3,12%	0,00%	-8,45%	0,60%

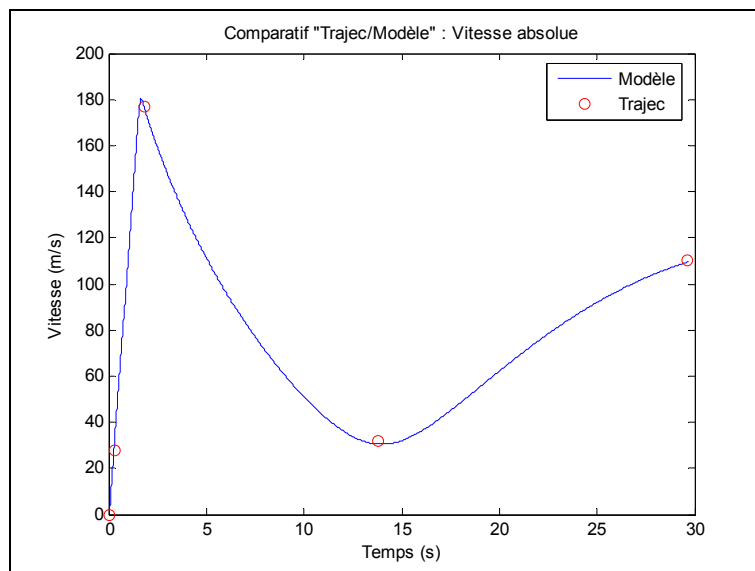
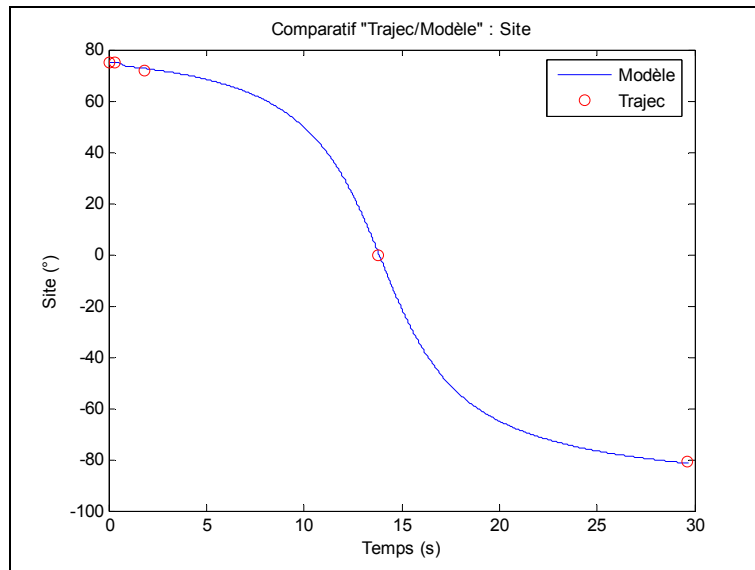
On peut ainsi constater que les écarts de valeurs n'existent que dans le cas où les valeurs retournées par Trajec sont fortement approximées. Cela pourra de plus être mis en évidence ci-dessous grâce aux graphiques retournés par le modèle.

### 3.4 °) Résultats graphiques du modèle

Les résultats du modèle représentés sous la forme des quelques graphes ci-dessous sont complétés par la représentation des points caractéristiques du vol issus du calcul sous Trajec.



Note : La trajectoire affichée ci-dessus en rouge pour les résultats de Trajec est une interpolation polynomiale d'ordre 2 dont les points remarquables sont le décollage, la sortie de rampe, la fin de propulsion, la culmination et l'impact.



### 3.5 °) Comparaison énergétique

Afin d'estimer et de comparer les pertes énergétiques pendant la phase d'ascension entre les deux modèles, on calcule l'énergie cinétique à la culmination dans les deux cas de la manière suivante :

$$E = E_{Potentielle} + E_{Cinétique}$$

$$E = M * g * (z_{Culmination} - z_0) + 1/2 * M * V_{Culmination\_R0}^2$$

On a ainsi les résultats suivants :

Etude énergétique à culmination à partir du calcul de Trajec (J) :

5.0633e+004

Etude énergétique à culmination à partir du calcul du modèle (J) :

5.0651e+004

Ecart énergétique (%) :

0.0370

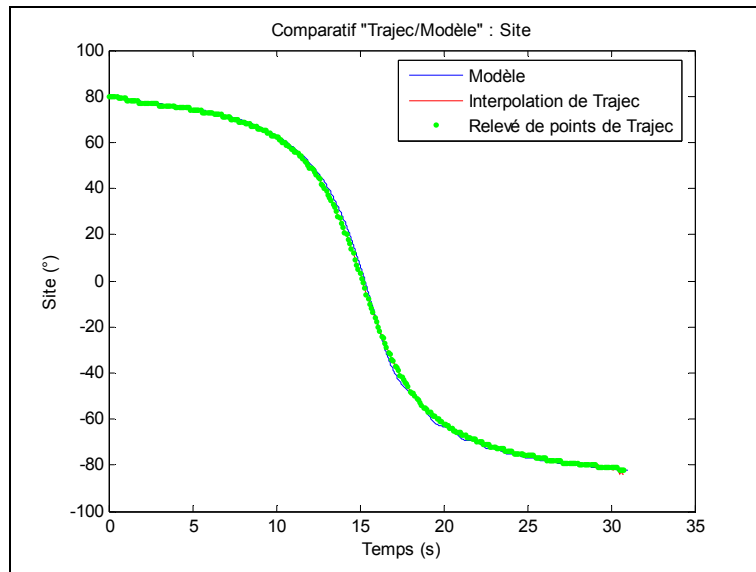


## 4 °) Interprétation des résultats

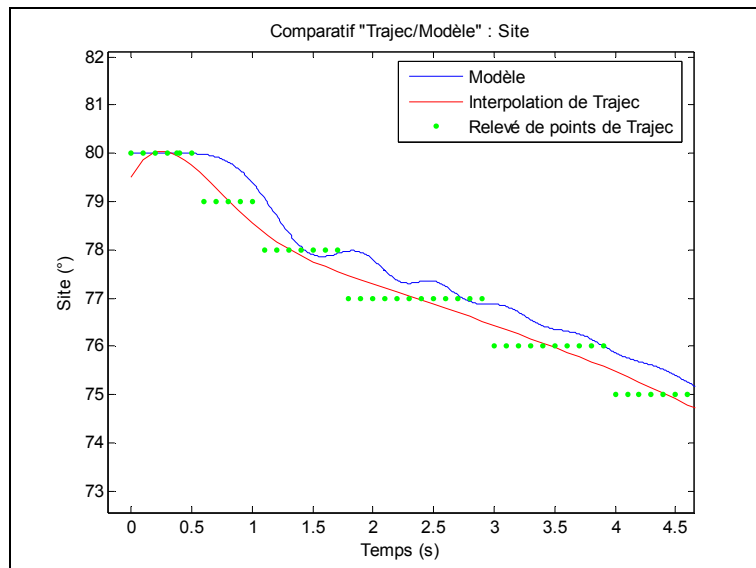
### 4.1 °) Comparatif Trajec/Modèle

Dans les 3 cas d'études précédents, on peut constater que les points caractéristiques issus de Trajec coïncident bien avec les résultats du modèle. Seul est visible l'écart de trajectoire entre Trajec et le modèle.

En important le fichier total des points de résultats de Trajec on obtient alors le tracé suivant pour le site :



Une vue agrandie de la sortie de rampe permet de mettre en évidence ceci :



Le site résultant du calcul du modèle est toujours supérieur à l'approximation retournée par Trajec (arrondi à 0 chiffres après la virgule). Ainsi l'intégration de la poussée mène à une trajectoire légèrement plus redressée, d'où l'apogée légèrement supérieure et la portée balistique légèrement inférieure.



#### 4.2 °) Comportement en sortie de rampe

En rampe, la fusée subit la réaction de la rampe qui vient annuler les composantes sur y et z dans le repère fusée.

*Lorsque la fusée sort de rampe, la fusée commence à être soumise au vent ambiant. Cependant dans les cas précédents, le vent choisi était nul.*

De plus lorsque la fusée sort de rampe, elle n'est plus soumise à la réaction de la rampe et ainsi est animée d'une vitesse composée de l'action de la poussée sur l'axe x, et de l'action de son poids sur les axes y et z..

Ainsi naît l'action des ailerons qui tendent à ramener la fusée dans son vent relatif et ainsi à diminuer le site de la fusée.

Dans Trajec, la fusée commence à diminuer de site dès sa sortie de rampe puisque l'orientation de la fusée résulte du calcul :  $\arctg\left(\frac{dV_{z_{R0}}}{dt} / \frac{dV_{x_{R0}}}{dt}\right)$  puisque l'incidence nulle est supposée.

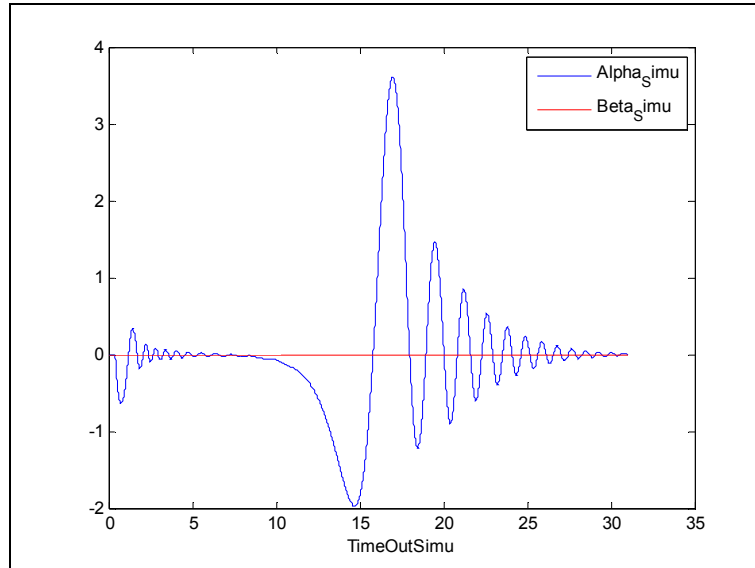
A l'inverse, dans le modèle, le site diminue puisque la vitesse sur les axes y et z augmente. En effet lorsque v ou w augmentent,  $\alpha$  et  $\beta$  augmentent également puisque :

$$\begin{array}{l} \alpha = \arcsin(v/V) \\ \beta = \arcsin(w/V) \end{array}$$

## 5 °) Etude de la stabilité

### 5.1 °) Cas avec vent face à la rampe

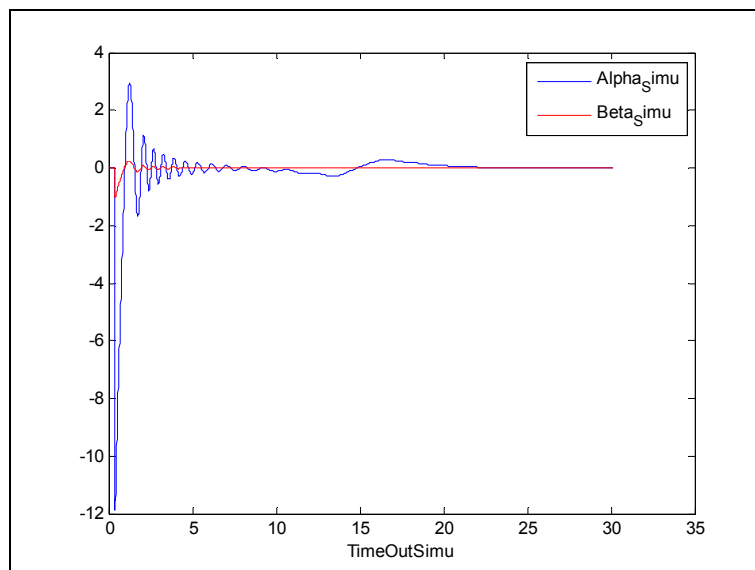
Dans le cas d'une fusée lancée avec un vent face à la rampe, on peut constater une évolution des paramètres  $\alpha$  et  $\beta$  comme ci-dessous :



Le vent provenant face à la fusée, l'angle  $\beta$  est nul, l'angle  $\alpha$  quand à lui converge vers 0 puisque la fusée est stable. L'oscillation visible aux alentours de t=15secs est due à la faible vitesse relative de la fusée à son apogée. Ceci est accentué par le fait que la fusée choisie pour cet exemple est lancée avec un site initial important.

### 5.2 °) Cas avec vent oblique

Dans le cas d'une fusée lancée avec un vent oblique par rapport à la rampe, on peut constater une évolution des paramètres  $\alpha$  et  $\beta$  comme ci-dessous :

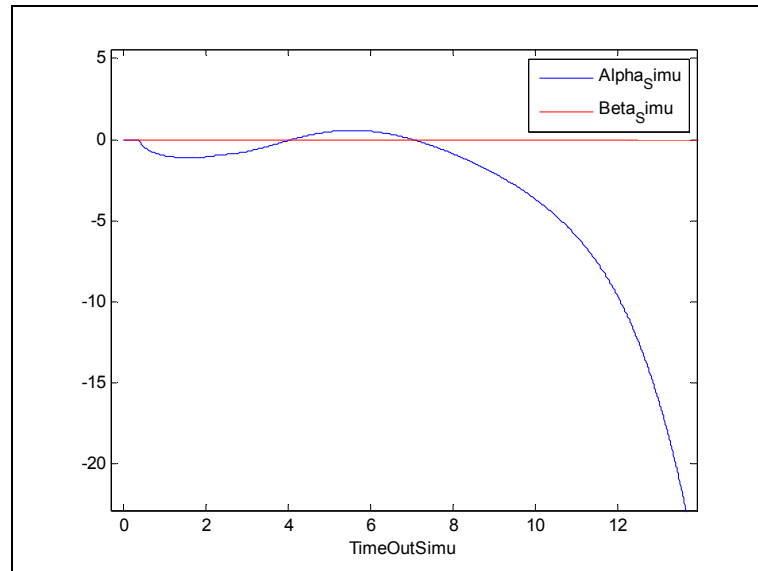


Contrairement au cas précédent avec un vent de face, l'angle  $\beta$  n'est pas constant et nul au

cours du temps. La fusée choisie étant stable, celui-ci ainsi que  $\alpha$  converge vers une valeur nulle.

### 5.3 °) Cas avec vent oblique

L'évolution des incidences  $\alpha$  et  $\beta$  sont, dans le cas d'une fusée instable, de la forme :

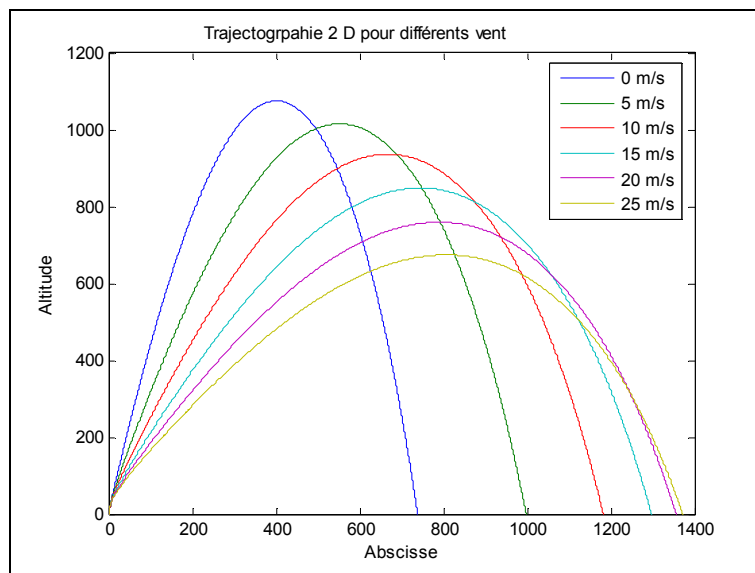


On voit alors que l'incidence  $\alpha$  de la fusée diverge. L'angle  $\beta$  reste quand à lui nul puisque le vent choisi ci-dessus était face à la rampe.

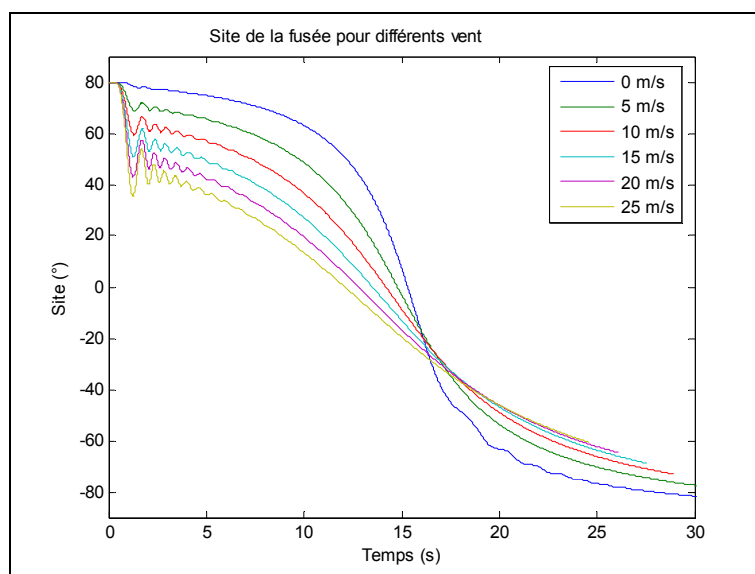
## 6 °) Etude de trajectoires d'une fusée soumis à un vent

### 6.1 °) Vent avec $\varepsilon=180^\circ$

Remarque : pour rappel, l'angle  $\varepsilon$  définit l'orientation du vent par rapport à l'axe  $x_{R0}$ , soit l'orientation de la rampe de lancement. Un angle  $\varepsilon=180^\circ$  correspond donc à un vent venant de face par rapport à la fusée.



On constate ici aisément que plus la vitesse du vent est élevée plus la portée balistique est importante et moins l'altitude est élevée. Cela s'explique par le fait que la fusée se couche dans le vent à sa sortie de rampe. Ainsi dans le cas présenté ci-dessus, la fusée est animée d'une vitesse de 22.4 m/s à sa sortie de rampe, dans le cas où la vitesse de vent est de 25 m/s, son incidence croît donc de façon instantanée vers une valeurs importante, le couple de rappel des ailerons contraint donc sa trajectoire à se pencher et le site diminue ainsi brusquement.

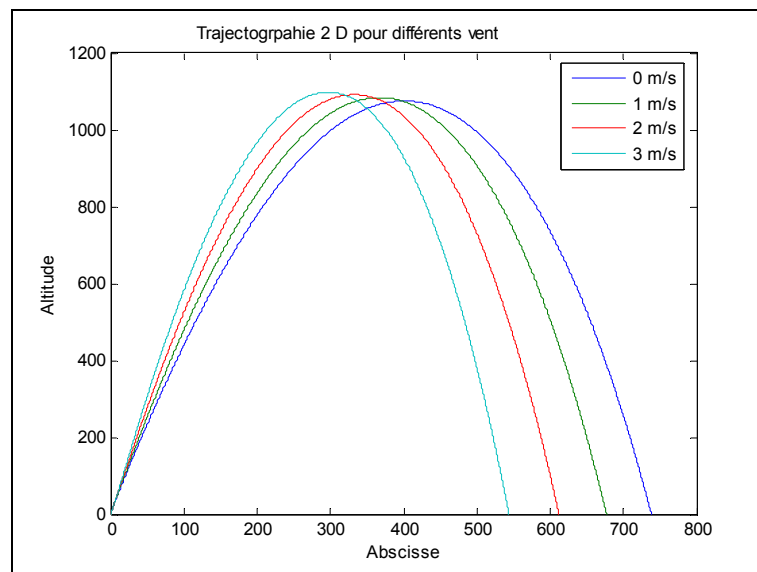


On constate de plus que plus la vitesse du vent est importante plus l'amplitude de la perturbation l'est également. Ainsi l'oscillation s'atténue dans un plus grand laps de temps.

Remarque : Ces calculs ont été effectués avec la fusée "Chamois" précédemment décrite, son  $C_{z_{\alpha}}$  est égale 23.14 et sa Marge Statique est comprise entre 2.31 et 2.59. Ces paramètres valident donc bien les critères de stabilité "statique", le temps de réponse à une perturbation est donc bien représentatif.

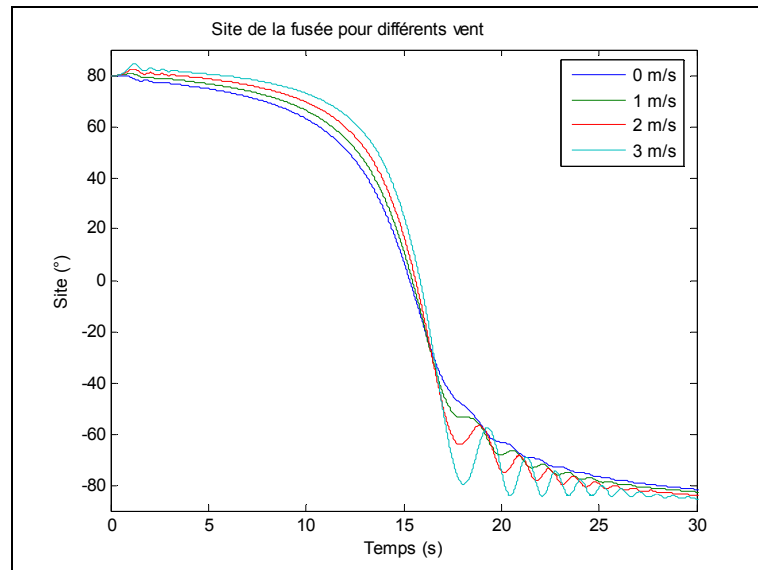
## 6.2 °) Vent avec $\varepsilon=0^\circ$

Remarque : pour rappel, l'angle  $\varepsilon$  définit l'orientation du vent par rapport à l'axe  $x_{R0}$ , soit l'orientation de la rampe de lancement. Un angle  $\varepsilon=0^\circ$  correspond donc à un vent venant dans le dos de la fusée.



De la même façon que lorsque  $\varepsilon=180^\circ$ , la vitesse du vent influe sur l'évolution du site en sortie de rampe. Ainsi plus la vitesse du vent est importante lorsque  $\varepsilon=0^\circ$ , plus la portée balistique est faible et plus l'altitude de culmination est importante.

On constate également que la présence d'un vent avec  $\varepsilon=0^\circ$  a plus d'impact à vitesse égale qu'un vent avec  $\varepsilon=180^\circ$ . Cela est en accord avec les raisons qui motivent le CNES et Planètes Sciences à procéder aux lancements des fusées expérimentales avec une rampe orientée face aux vents dominants.



L'oscillation mise en évidence pendant la descente est due au phénomène similaire à celui observé pour le vol de la fusée "Cariacou". En effet lorsque la vitesse est faible en début de descente, l'effort de rappel pour des faibles incidences n'est pas suffisant ce qui amène à de telles oscillations. Ces dernières s'atténuent lorsque la vitesse augmente.

Remarque : Comme pour l'étude effectuée avec  $\varepsilon=180^\circ$ , ces calculs ont été effectués avec la fusée "Chamois", le temps de réponse à une perturbation est donc bien représentatif.



## 7 °) Etude de trajectoire d'une fusée lancée à partir d'un drone

Cette étude porte sur la trajectoire de la fusée "Cariacou" décrite précédemment lancée à partir d'un drone avec les Conditions de lancements ci-dessous :

Site du guide de lancement :  $5^\circ$

Longueur du guide de lancement : 0.5 m

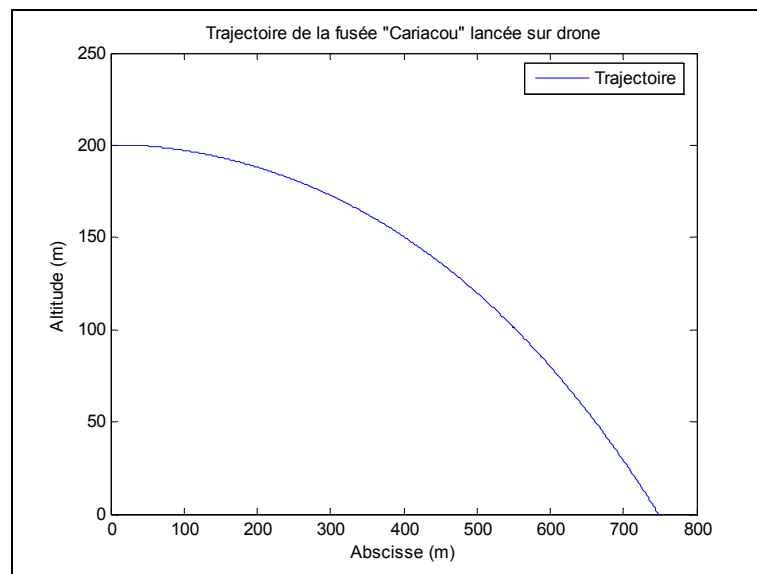
Vitesse du drone :

$U_0 = 20 \text{ m/s}$

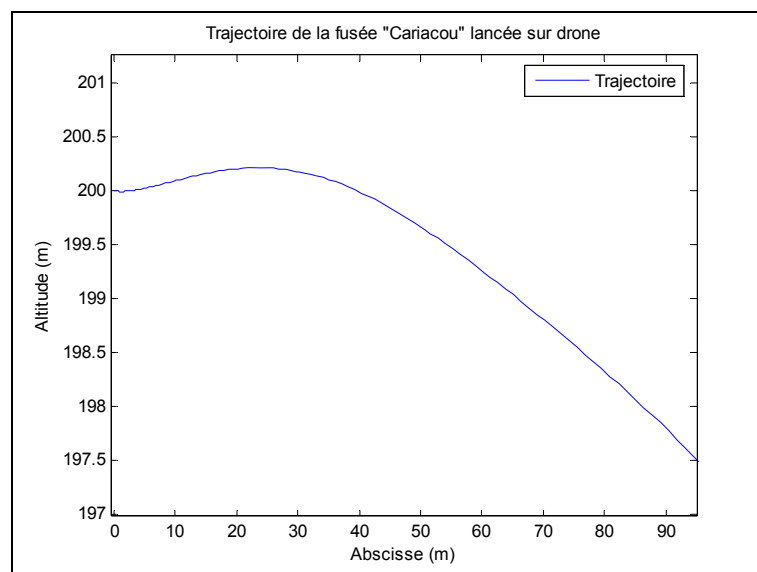
$V_0 = 0 \text{ m/s}$

$W_0 = 2 \text{ m/s}$

Altitude de lancement = 200 m

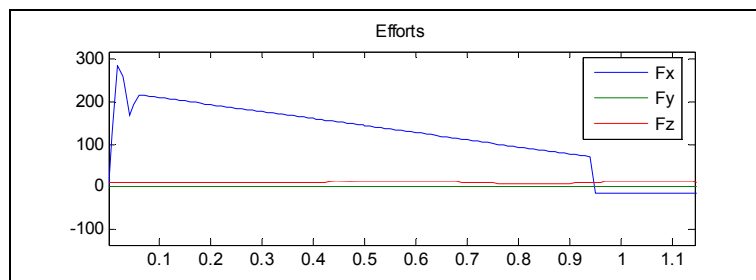
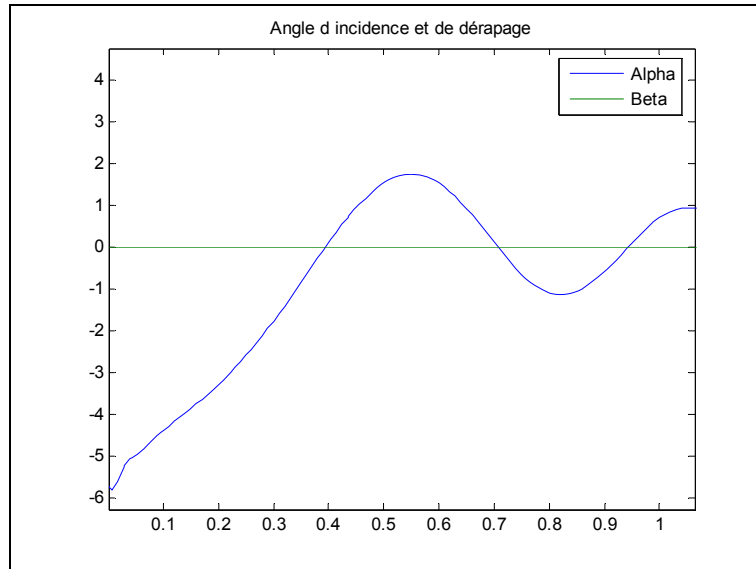


Un zoom sur le largage de la fusée permet de mettre en évidence le comportement de la fusée face à son vent relatif et la façon avec laquelle elle rejoint sa trajectoire finale.



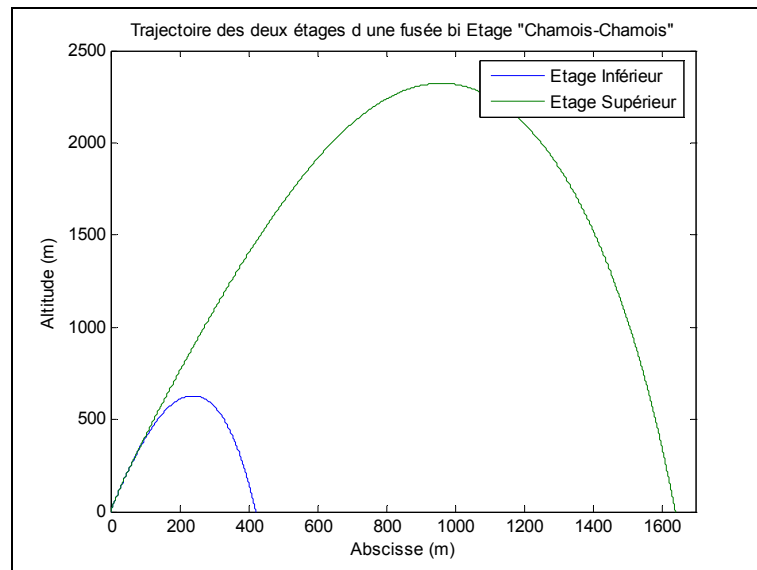
Analyse phase à phase du vol :

- La fusée décolle en incidence puisque elle est lancée avec un angle de  $5^\circ$  avec l'axe principal du drone.
- Elle se rétablit alors dans son vent relatif imposé par la vent relatif initial dû à la vitesse du drone et par la vitesse que la fusée acquière rapidement grâce à la poussée du moteur.

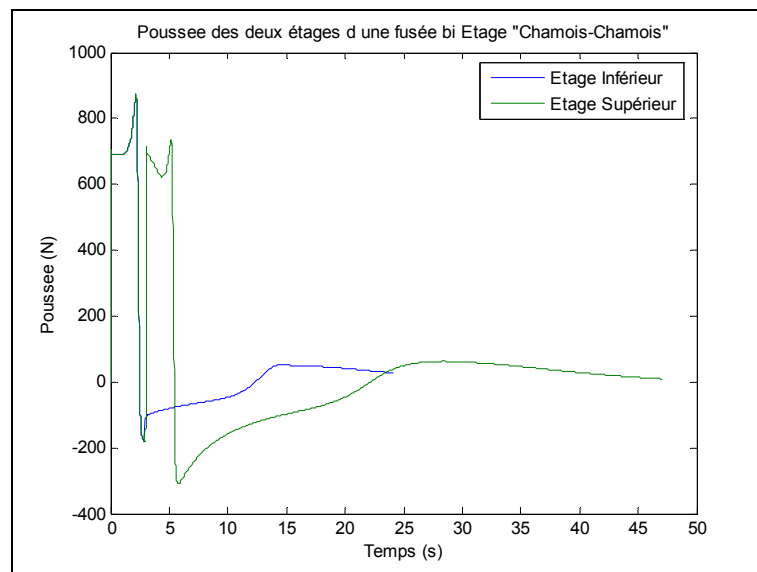


## 8 °) Etude de trajectoire d'une fusée bi étage Chamois Chamois

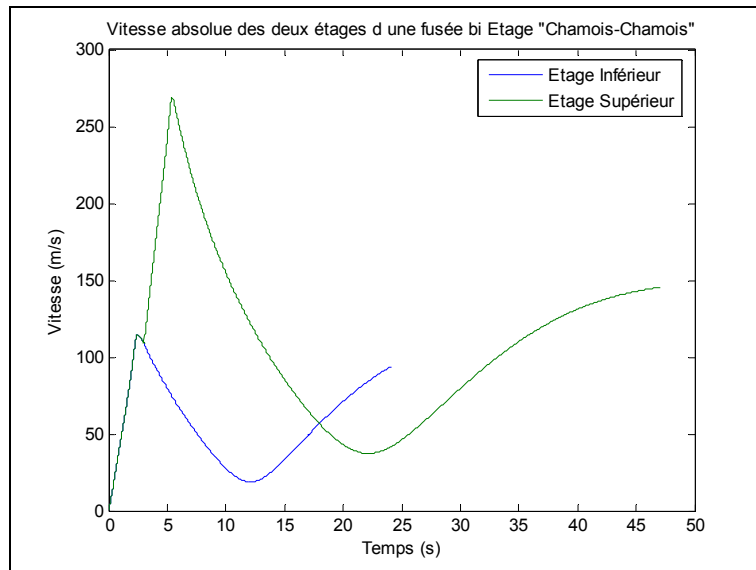
Bien que l'interface graphique ne soit pas encore adaptée à cet usage, le modèle peut convenir à l'étude de fusée bi étage. Chaque étage est alors simulé indépendamment en tenant compte de l'incrément de poussée ou du décrétement de masse apporté par l'autre étage. Ainsi on obtient pour une fusée bi étage de masse égale à 8Kg (sans propulseurs) la trajectoire ci-dessous :



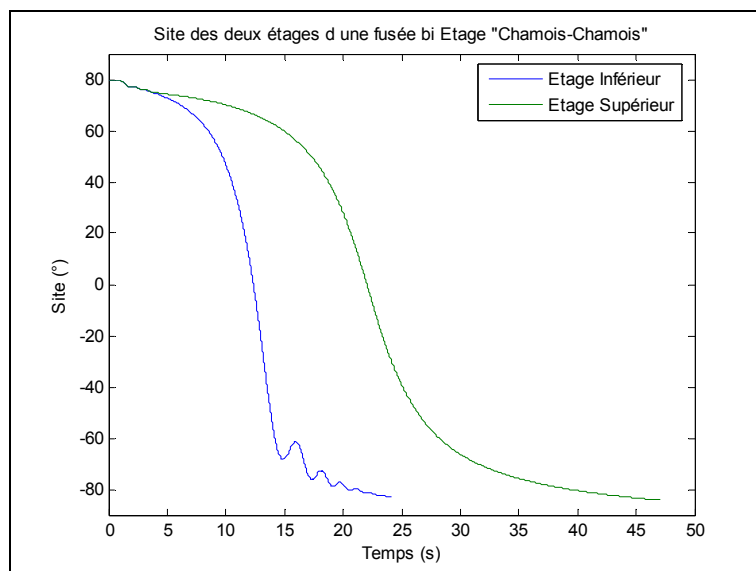
L'allumage du second étage est ici effectué 0.4 seconde après l'extinction du premier propulseur. On constate mieux plus facilement cet apport de poussée sur le graphique ci-dessous :



Cet apport de poussée se traduit donc par une accélération supplémentaire du second étage et donc une augmentation de sa vitesse comme le montre le graphe ci-dessous :



L'étude des sites respectifs des deux étages permet d'analyser les comportements respectifs des deux étages à leurs séparations :



Cette étude permet également de mettre une fois de plus en évidence l'oscillation en descente de l'étage inférieur lorsque son site devient inférieur à  $0^\circ$ . Cette oscillation n'est pas observable sur l'étage supérieur puisque sa vitesse est supérieure à celle du premier étage (voir graphique des vitesses ci-dessus).

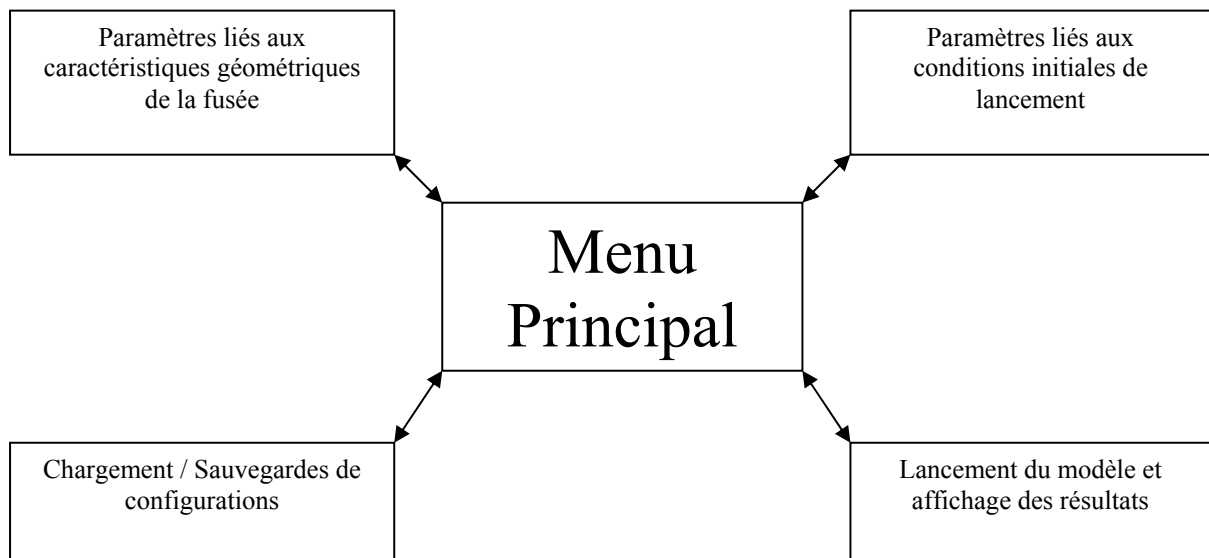
## X°) Réalisation de l'interface graphique

L'objectif à terme de ce projet étant de fournir un « logiciel » aux clubs aérospatiaux comme celui de l'ESTACA, qui permet la conception de fusée expérimental, il était inconcevable de fournir à ces membres une simulation dans laquelle il fallu modifier les paramètres des fusées au travers de ligne de code, d'autant plus que tout le monde n'est pas forcément familier avec les outils Matlab/Simulink. De ce fait, la mise en place d'une interface s'avérait donc nécessaire.

Dans ce chapitre, nous ne détaillerons évidemment pas l'ensemble des lignes de codes nécessaire à la création de cette interface. Après avoir brièvement justifier l'architecture choisie, nous nous contenterons d'en expliquer le principe de création, et de souligner les caractéristiques majeures de cette interface. Pour de plus amples informations relatives à l'utilisation de l'interface, merci de consulter la rubrique « manuel d'utilisation ».

### 1 °) Choix de l'architecture.

Avant de commencer quelconque codage, il est nécessaire de réfléchir à la façon dont seront agencé les différentes fenêtres de notre interface. En effet, devant le nombre conséquent de paramètres à rentrer, il paraît peu ergonomique de les afficher tous en même temps sur le même écran. De ce fait, nous avons donc l'architecture suivante :



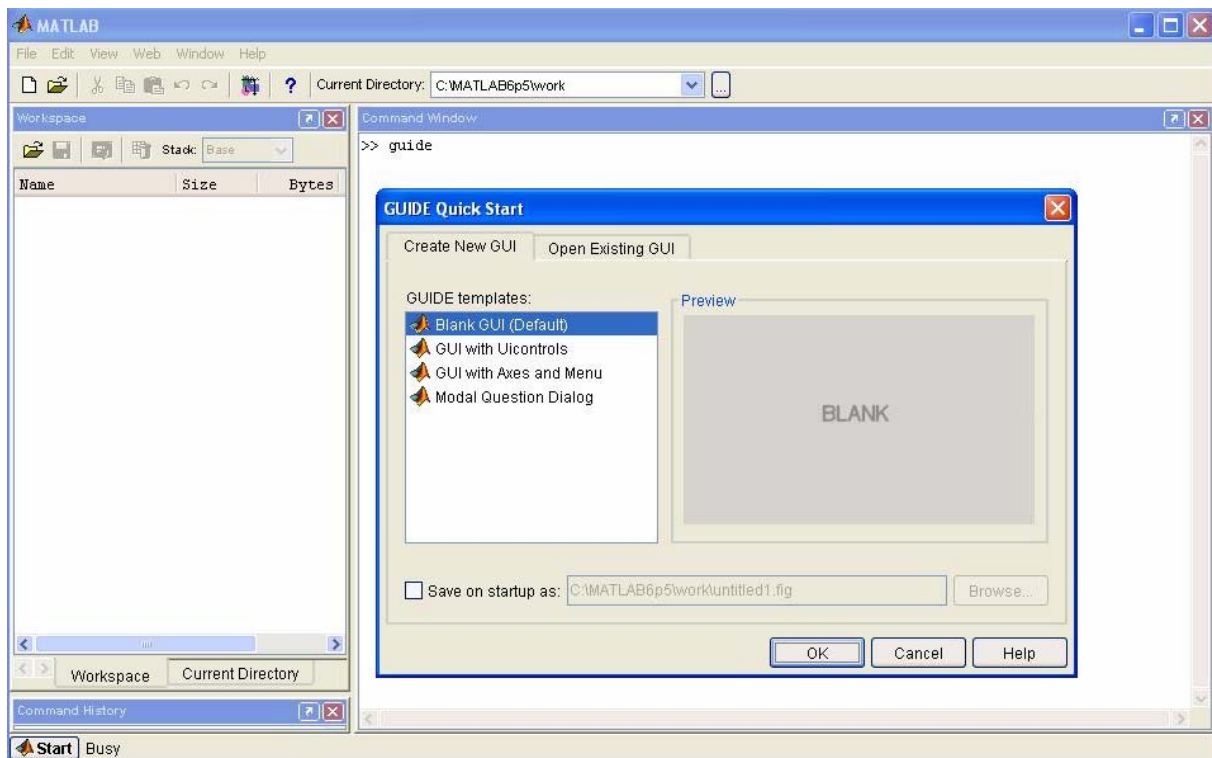
Il a donc été décidé d'articuler l'interface autour d'un menu principal. A partir de ce menu, l'utilisateur a accès à tous les paramètres configurables en fonction de leur nature. Par ailleurs, cette architecture lui permet d'optimiser facilement ses paramètres de lancement en fonction des résultats obtenus.

Par conséquent, l'architecture mise en place semble être relativement judicieuse, puisqu'elle permet à chacun d'accéder à l'ensemble des paramètres nécessaires sans pour autant se sentir perdu au milieu de ces derniers.

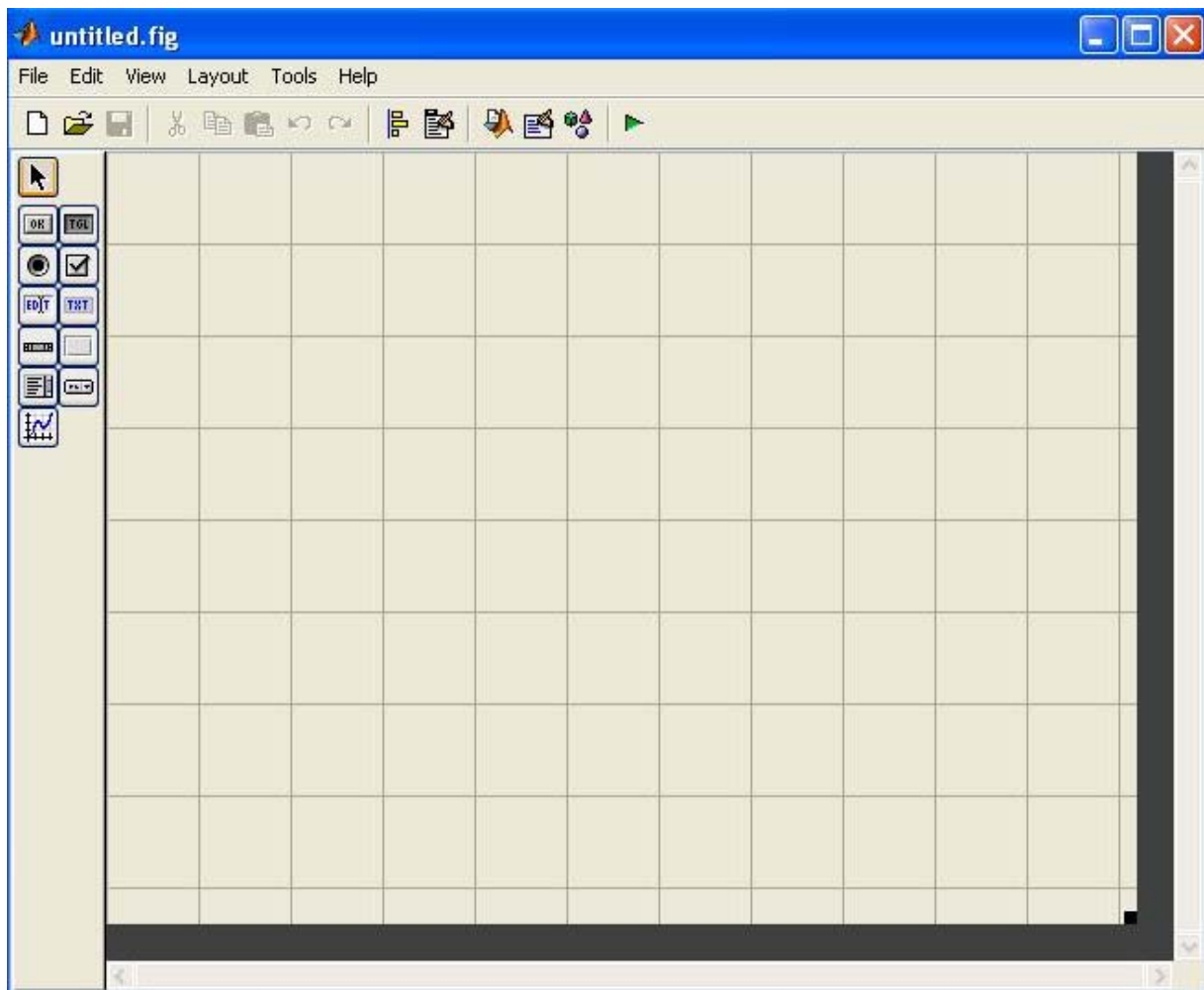
## 2 °) Mode de création des fenêtres

Comme nous venons de l'expliquer, l'interface repose sur la juxtaposition de plusieurs de fenêtres qui ensemble, permettent à l'utilisateur de configurer sa fusée et plus généralement, son lancement.

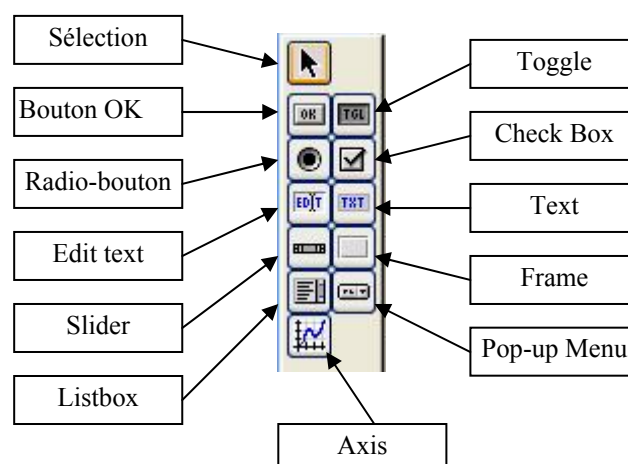
Pour la création de ces fenêtres, nous avons utilisé l'utilitaire « guide » (Graphical User Interface Development Environment ) inclus dans le logiciel Matlab :



Après avoir lancé l'outil à partir de la fenêtre de commande de Matlab, on sélectionne « Blank GUI (Default) », qui correspond à la création d'une nouvelle fenêtre. On arrive alors sur l'écran ci-dessous :



C'est à partir de ce menu que l'on va pouvoir créer nos différentes fenêtres. Pour ce faire, nous disposons de différents accessoires disponibles à partir de la barre de menu située à gauche de l'écran.

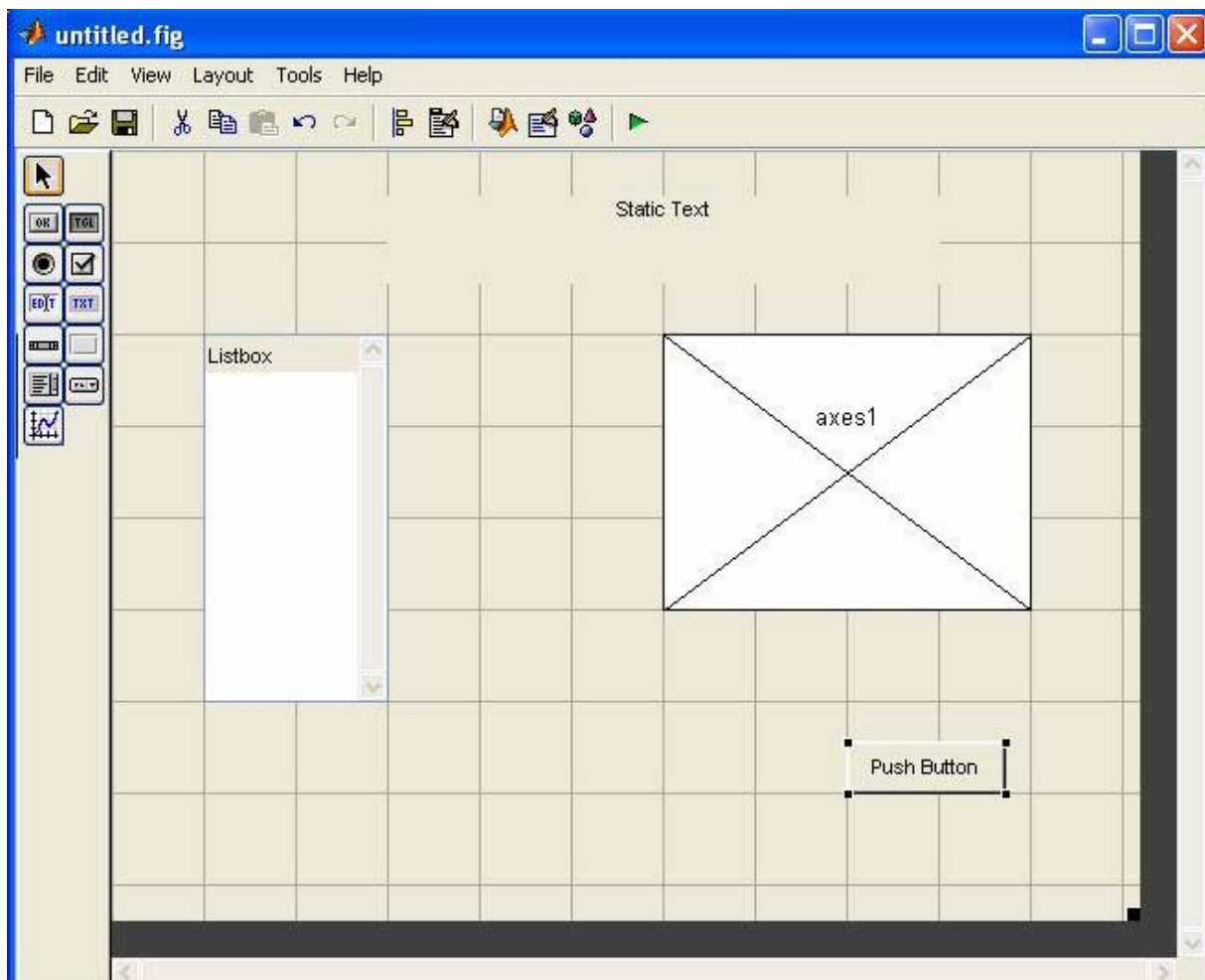


Expliquons le rôle des boutons que nous utiliserons par la suite :

- Bouton OK : Permet de créer un bouton, auquel on associe la fonction que l'on souhaite : validation, sauvegarde, chargement....

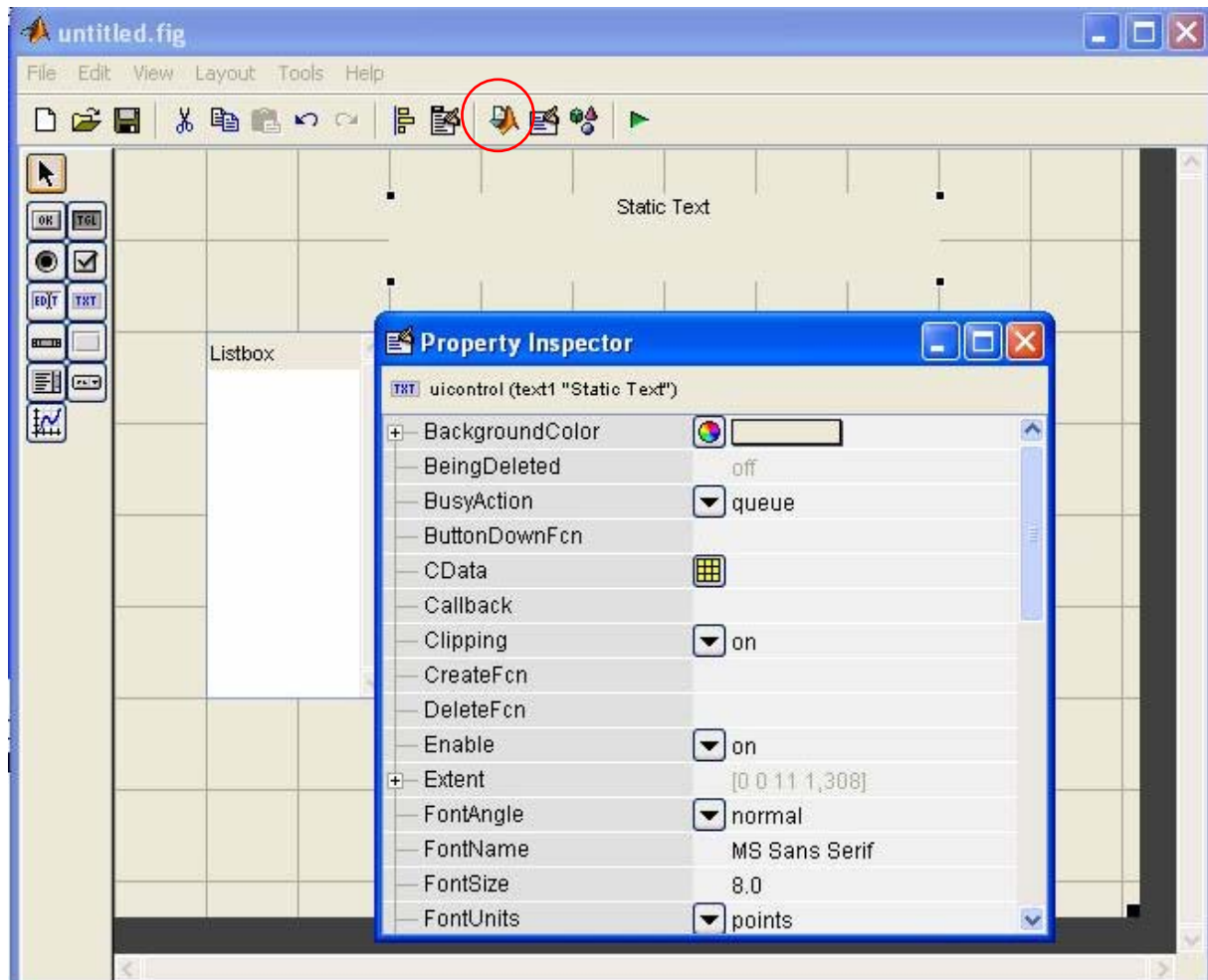
- Edit text : crée une zone de texte, accessible ou non à l'utilisateur : ces zones permettent soit de rentrer des valeurs de paramètres, soit d'afficher des résultats après calcul du modèle.
- Pop up Menu : Création de menu déroulant
- Text : création de texte dans l'interface
- ListBox : Création de fenêtres dans lesquelles s'afficheront un certains nombre de paramètres sélectionnables
- Axis : création de fenêtres graphiques

Par simple sélection de ces boutons, on peut alors créer l'allure de l'interface à sa guise :



Par suite il est possible de rentrer dans les propriétés de chaque éléments créés afin de changer le nom, la taille, la police.....Pour ce faire, il suffit de double-cliquer sur l'élément considéré.





Une fois la fenêtre graphique souhaitée obtenue, lancer le compilateur, afin que Matlab puisse générer le fichier source lié à cette fenêtre.

Dans la suite de ce paragraphe, nous n'explicitons pas le détail des fichiers créés, mais nous nous contenterons d'expliquer comment, à partir d'un objet créé graphiquement, il est possible de commander l'action à réaliser lorsque l'on clique dessus. Prenons par exemple le cas d'un « bouton OK », également appelé « PushButton ». Le code généré automatiquement par le compilateur est le suivant :

```

100
101 % --- Executes on button press in pushbutton1.
102 function pushbutton1_Callback(hObject, eventdata, handles)
103 % hObject    handle to pushbutton1 (see GCBO)
104 % eventdata  reserved - to be defined in a future version of MATLAB
105 % handles    structure with handles and user data (see GUIDATA)
106
107
108

```

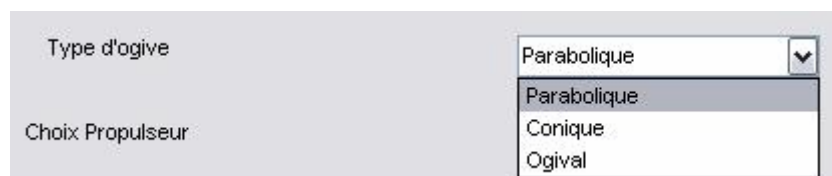
Le compilateur se contente de créer une fonction, dont le nom correspond au nom rentré dans les propriétés du bouton. A ce stade, aucune action ne se produit lorsque l'on clique sur le bouton. Pour changer cela, il suffit de remplir le champ correspondant à la fonction. Dans le

cas ci-dessous, nous désirons par exemple fermé la fenêtre graphique actuelle, que nous avons nommé « test ».

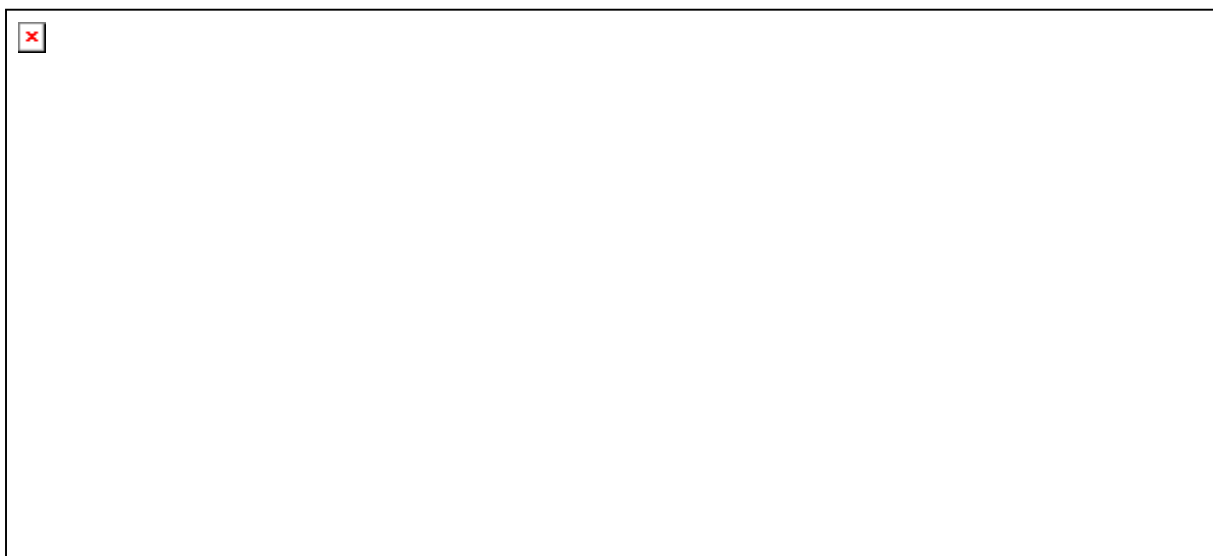
```
101 % --- Executes on button press in pushbutton1.  
102 function pushbutton1_Callback(hObject, eventdata, handles)  
103 % hObject    handle to pushbutton1 (see GCBO)  
104 % eventdata  reserved - to be defined in a future version of MATLAB  
105 % handles    structure with handles and user data (see GUIDATA)  
106  
107 -   close(test)|  
108
```

Désormais, la fenêtre en question se fermera lorsque l'on cliquera sur le bouton « pushbutton1 ».

Autre exemple utilisé dans le cas de notre interface graphique, le codage des « pop up menu », permettant à l'utilisateur de faire un choix entre différents paramètres :



Dans l'exemple ci-dessus, le code utilisé pour obtenir le choix de l'ogive est :



En fonction du type d'ogive choisie, la variable « TypeOgive » prend la valeur 1, 2 ou 3, puis est assigné dans le Workspace de base de Matlab, dont nous détaillerons l'accès dans la suite de ce rapport.

### 3 °) Caractéristiques de l'interface

Afin d'assurer le bon fonctionnement de l'interface, un certain nombre de précautions ont été prises concernant les règles de programmation de l'interface

#### 3.1 °) Accès aux différents paramètres

Lorsque l'on met traditionnellement en place un modèle Simulink associé à des fichiers sources, l'ensemble des variables mises en jeu sont répertoriées dans un espace de travail, le « Workspace ». Etant celui créé par défaut, il est automatiquement reconnu par Matlab comme étant l'espace « Base ».

Si ce rappel peut paraître anodin pour la plupart des utilisateurs, il prend néanmoins toute son importance lorsque l'on associe à ce modèles des interfaces graphiques, ce qui est le cas ici. En effet, pour chaque fenêtre d'interface créée, Matlab associe un espace de travail « virtuel » dans lequel il répertorie les variables utilisées par la fenêtre graphique en question. Par conséquent, nous sommes donc dans l'obligation de jongler entre différents workspaces, contenant des variables communes, et dont l'affectation de nouvelles valeurs dans un espace de travail ne se répercutera pas dans le second espace. Il faut donc trouver un moyen d'accorder tous ces workspaces entre eux.

Pour ce faire, deux méthodes sont envisageables :

##### ➤ Utilisation de variables globales

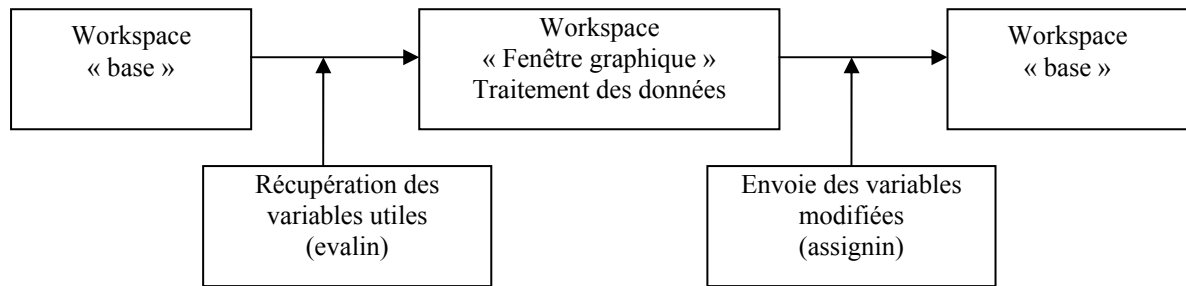
En affectant le type « Global » à l'ensemble de nos variables, celles-ci deviennent alors automatiquement accessibles à partir de n'importe quel espace de travail. Si cette méthode est relativement simple et pratique d'utilisation, il n'empêche qu'elle a le défaut d'affaiblir considérablement la robustesse des programmes. En effet, en supposant qu'une variable soit modifiée accidentellement à partir d'une interface, ses modifications seront automatiquement prises en compte par la suite ce qui peut amener à un plantage complet du système.

##### ➤ Utilisation d'un seul Workspace de référence

L'autre méthode envisageable est le chargement et la sauvegarde progressive des variables mises en jeu dans les différentes fenêtres graphique. Le principe est le suivant :

- A chaque fois qu'une fenêtre graphique est appelée, on charge seulement les variables qui seront utilisé à partir du Workspace « base » dans le Workspace virtuel de la fenêtre graphique ; Cette opération s'effectue à l'aide de la fonction « evalin »
- Une fois les variables chargées, on les modifie à notre guise dans ce Workspace virtuel, sans pour autant affecter ces modifications dans le Workspace initial, puisque aucune corrélation directe n'existe entre ces deux espaces de travail.
- Une fois l'ensemble des modifications effectuées, on enregistre celle-ci dans le Workspace initial « base » ; Cette opération s'effectue à l'aide de la fonction « assignin ».

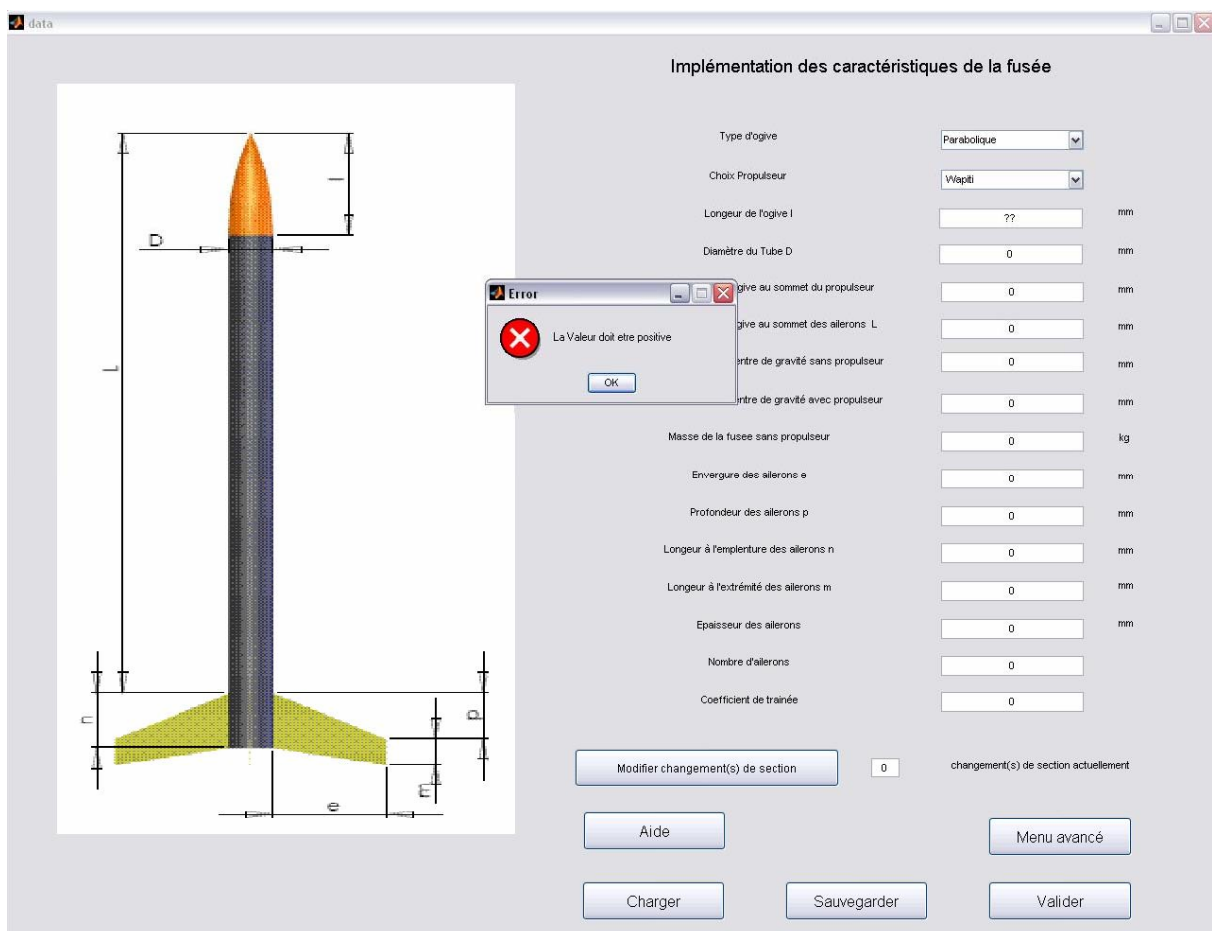
Ces opérations peuvent se résumer selon le schéma suivant :



Cette seconde méthode, plus lourde à mettre en œuvre, est néanmoins bien plus robuste que la première, puisqu'elle présente l'avantage de ne modifier en même temps que les variables utilisées dans la fenêtre graphique. Par ailleurs, la nécessité d'utiliser les fonctions « evalin » et « assignin » pour chacune des variables souhaitant être modifiée nous assure qu'aucune variable ne pourra être modifiée par erreur. C'est la raison pour laquelle nous avons donc opté pour celle-ci.

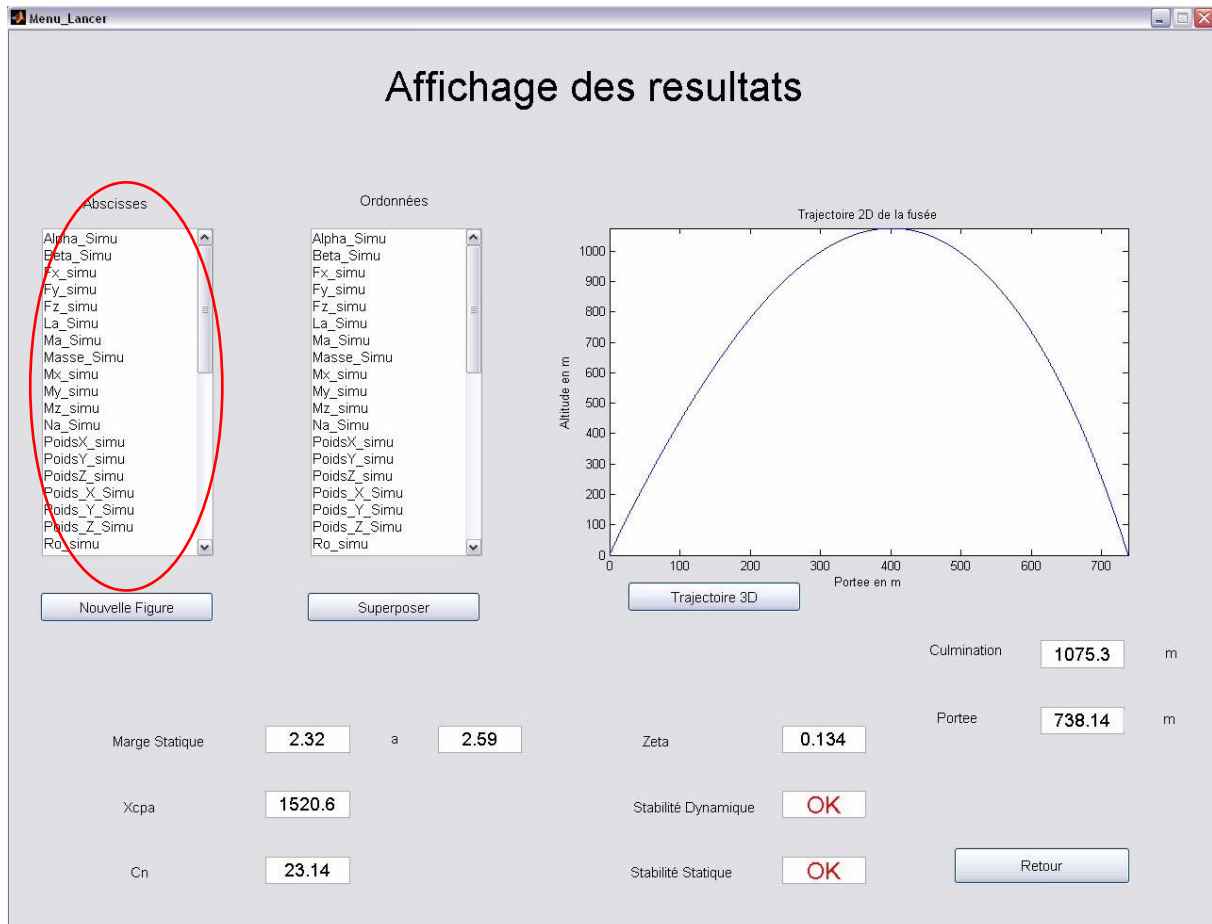
### 3.2 °) Gestion des messages d'erreur

Afin de s'assurer que l'utilisateur ne se trompe pas dans la saisie de données, ce qui pourrait conduire à un plantage complet de la simulation, il fut décidé de mettre en place des messages d'erreurs sur l'ensemble des variables paramétrables, dans le cas où celles-ci seraient incorrectes. Par exemple, un utilisateur qui rentre une ogive de longueur négative (ex : -150 mm) se verrait afficher le message suivant, qui refuserait l'enregistrement de sa donnée.



### 3.3 °) Affichage des données dans les listbox

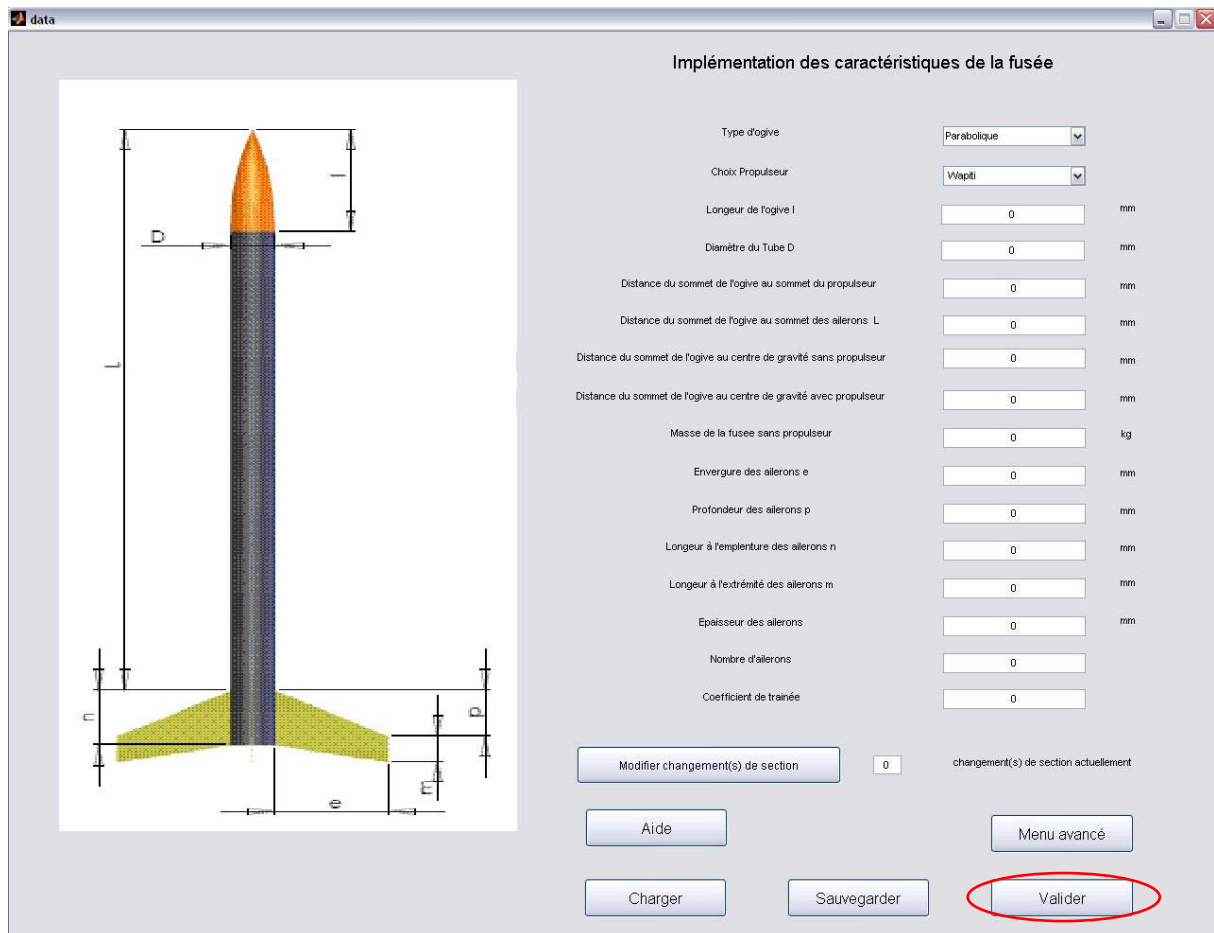
Comme nous l'avons vu précédemment, il est possible de mettre en place des « listbox », véritable fenêtre regroupant une liste de données accessible à l'utilisateur à partir de la fenêtre graphique considéré. Ceci est particulièrement visible lors de l'affichage des résultats obtenus après lancement de la simulation.



Néanmoins, de nombreuses variables ne sont que des calculs intermédiaires utilisés pour la détermination d'autres paramètres, et ne représentent en soit aucun intérêt pour l'utilisateur. Aussi a-t-il été décidé de restreindre les variables sélectionnables à celles présentes en sortie du modèle Simulink élaboré, et qui peuvent aider à l'interprétation des résultats obtenus.

### 3.4 °) Temps de calcul

Comme nous l'avons dit au début de ce chapitre, l'interface graphique est une succession de fenêtres donnant accès progressivement aux différents paramètres configurables par l'utilisateur. De ce fait, il fut décidé qu'une seule fenêtre devait être affichée en même temps à l'écran, toujours dans le but d'améliorer sa compréhension des saisies à effectuer. Dans le cas ci-dessous, nous sommes dans la fenêtre graphique « data » et nous souhaitons revenir à la fenêtre principale nommée « Menu » lorsque nous cliquons sur le bouton « Valider ».



Le code Matlab permettant d'assurer cette fonction est le suivant :

```

650
651 % --- Executes on button press in Valider.
652 function Valider_Callback(hObject, eventdata, handles)
653 % hObject    handle to Valider (see GCBO)
654 % eventdata  reserved - to be defined in a future version of MATLAB
655 % handles    structure with handles and user data (see GUIDATA)
656
657
658 - set(data,'Visible','off') %suite à la pression du bouton Valider on masque la fenetre data
659 - set(Menu,'Visible','on') % et on affiche la fenetre Menu
660

```

### 3.5 °) Gestion de la taille de l'écran

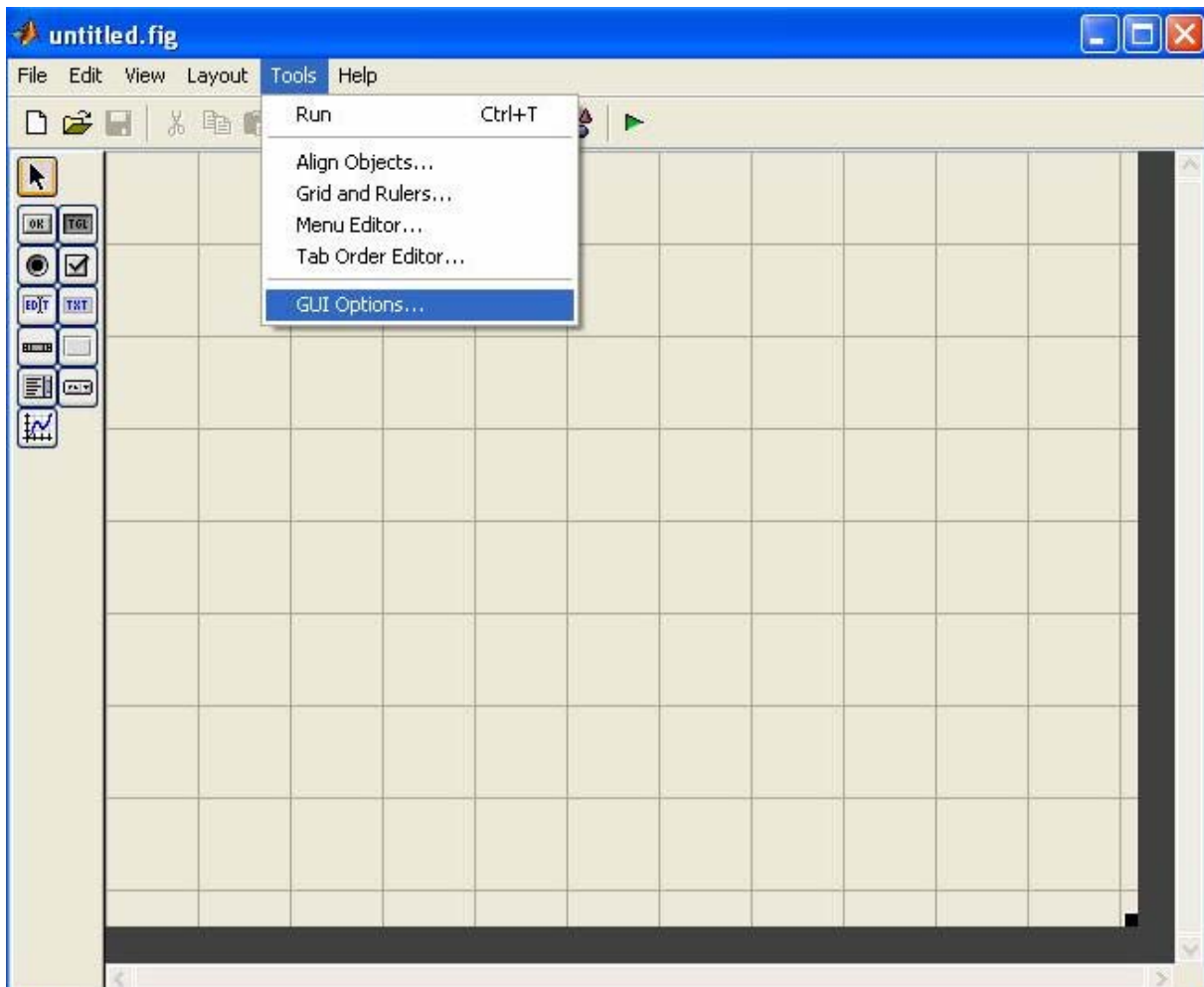
Lorsque nous créons notre interface graphique, Matlab fixe automatiquement les tailles en centimètres des différents objets créés. Néanmoins, si la conception des fenêtres est réalisée sur un écran 19 pouces par exemple, la conséquence visible est que lors de l'affichage sur un écran plus petit, une partie de la fenêtre créée ne sera plus visible. Pour remédier à cela, deux opérations majeures sont à effectuer :

➤ Gérer les unités de dimensionnement des fenêtres

La méthode la plus simple consiste à forcer les unités de taille dans le code de la fenêtre considéré, en indiquant le type d'unité et la position de la fenêtre. Choisissez le type « Normalized ». Pour ce qui est de la position, 0 correspond à la gauche de votre écran en référence horizontale, et au bas de celui-ci en référence verticale, 1 correspond à la droite de votre écran en référence horizontale, et au haut de celui-ci en référence verticale

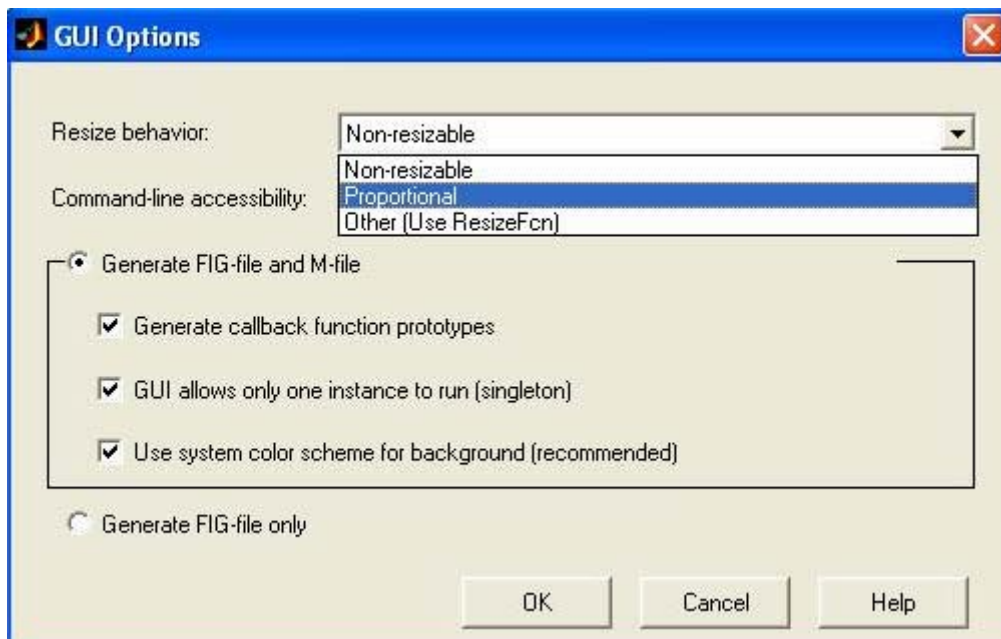
```
142  
143 - set(gcf,'Units','normalized','Position',[0.1 0.1 0.8 0.8])  
144  
145
```

- Assurez vous que les objets graphiques seront proportionnels à la taille de la fenêtre. Pour ce faire, dans la fenêtre du guide, cliquez sur « Tools » puis « GUI Options »



Vous arrivez alors sur la fenêtre suivante. Sélectionner l'option « Proportional » puis cliquez sur OK.





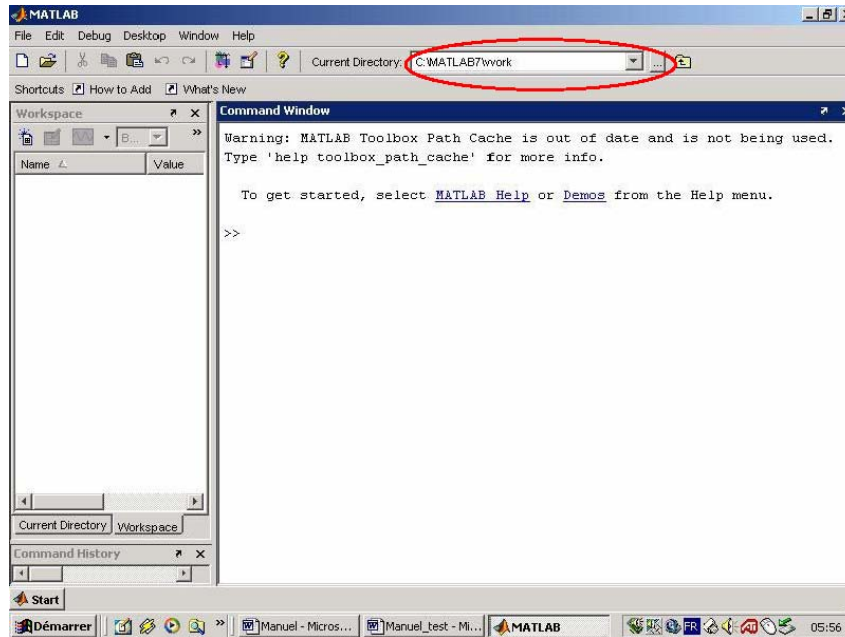


## XI ° ) Utilisation du logiciel

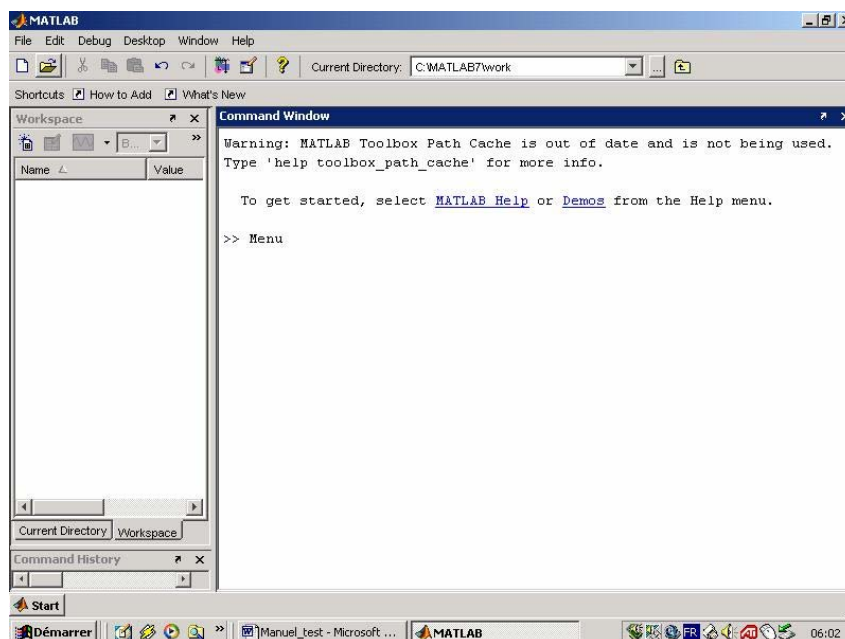
### 1 °) Lancement du logiciel

Ouvrez le logiciel Matlab.

Au niveau de la sélection du répertoire de travail « Current Directory », sélectionnez le répertoire où se trouve le logiciel de simulation fusée en cliquant sur « ... ».



Dans l'espace « Command Windows » de Matlab, tapez alors « Menu », puis appuyez sur la touche « Entrée ».



Vous lancez alors le programme et arrivez sur le menu suivant :



C'est à partir de ce menu que vous allez pouvoir configurer votre fusée et lancer votre simulation. Ce menu est constitué de 6 boutons :

-Caractéristiques de la fusée ; vous permet de configurer la géométrie de votre fusée, le choix du propulseur, vos éventuels changements de sections ....

-Conditions initiales ; vous permet de configurer les paramètres tels que l'inclinaison de la rampe de lancement, la vitesse et la position initiales de votre fusée ou encore le vent.

-Charger lancement ; vous permet de recharger instantanément l'ensemble des caractéristiques fusées et des conditions initiales à partir d'un fichier de sauvegarde antérieur.

-Sauvegarder lancement ; vous permet de sauvegarder instantanément l'ensemble des caractéristiques fusées et des conditions initiales.

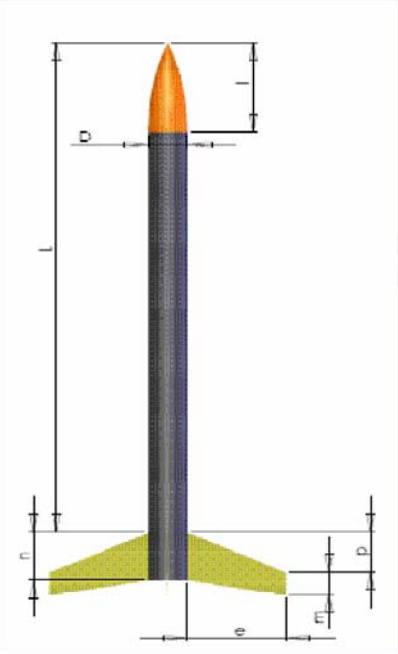
-Lancer Simulation ; Lancement de la simulation vous permettant de visualiser l'ensemble des paramètres du vol considéré.

## 2 °) Caractéristiques de la fusée

En cliquant sur cet onglet, vous arrivez sur la figure ci-contre :

data

### Implémentation des caractéristiques de la fusée



Type d'ogive	Parabolique	
Choix Propulseur	Chamais	
Longueur de l'ogive l	150	mm
Diamètre du Tube D	90	mm
Distance du sommet de l'ogive au sommet du propulseur	1400	mm
Distance du sommet de l'ogive au sommet des ailerons L	1600	mm
Distance du sommet de l'ogive au centre de gravité sans propulseur	1200	mm
Distance du sommet de l'ogive au centre de gravité avec propulseur	1312	mm
Masse de la fusée sans propulseur	8	kg
Envergure des ailerons e	150	mm
Profondeur des ailerons p	50	mm
Longueur à l'emplanture des ailerons n	150	mm
Longueur à l'extrémité des ailerons m	100	mm
Épaisseur des ailerons	2	mm
Nombre d'ailerons	4	
Coefficient de traînée	0.8	

Modifier changement(s) de section  changement(s) de section actuellement

Aide

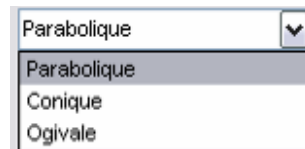
Menu avancé

Charger Sauvegarder Valider

## 2.1 °) Paramètres principaux

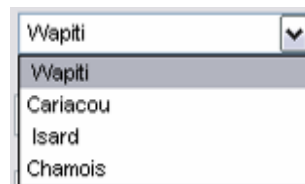
Cette fenêtre vous permet de configurer votre fusée. La correspondance des lettres entre le nom auxquelles elles renvoient et la figure de gauche vous permet d'identifier parfaitement chacun des paramètres. A noter que pour ce qui concerne le type d'ogive ainsi que le choix du propulseur, 2 menus déroulants sont à votre disposition vous permettant de choisir votre solution parmi les possibilités qu'offre le logiciel.

Si vous cliquez sur la flèche correspondante au menu déroulant du type d'ogive, vous obtiendrez la figure ci-dessous :



3 types d'ogives sont mis à votre disposition : parabolique, conique ou ogivale.

De même, si vous cliquez sur la flèche correspondant au menu déroulant du choix du propulseur, vous obtiendrez la figure ci-dessous :



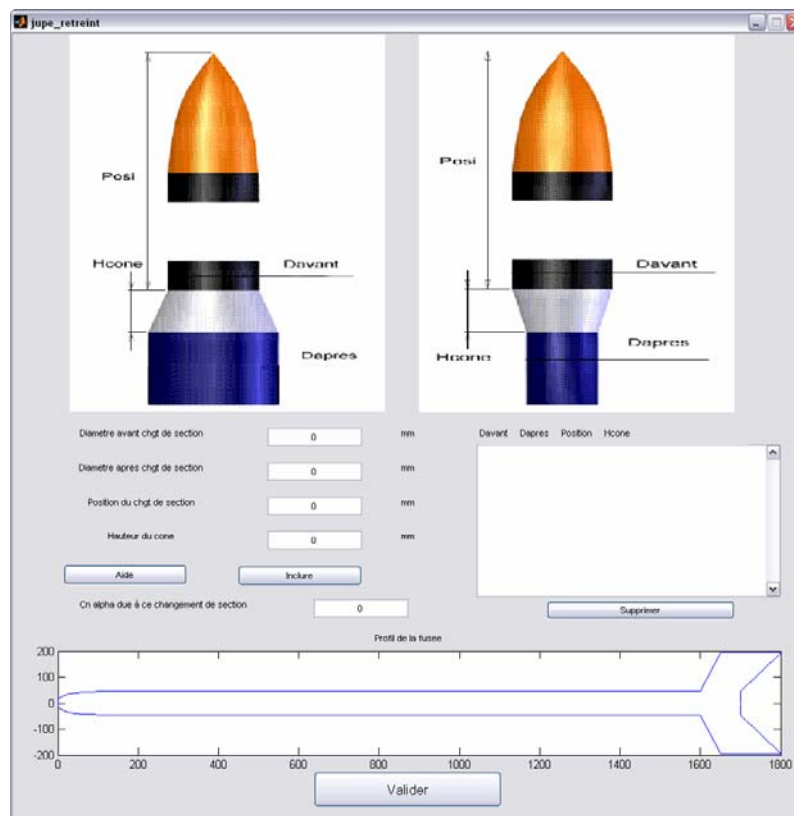
4 propulseurs sont mis à votre disposition : Wapiti, Cariacou, Isard et Chamois. Les autres valeurs des paramètres sont à rentrer dans les cases prévues à cet effet.

## 2.2 °) Changements de section

Vous pouvez également mettre en place des changements de section sur votre fusée, c'est à dire des jupes ou des rétreints. Pour ce faire, il vous suffit de cliquer sur l'icône « changements de section »

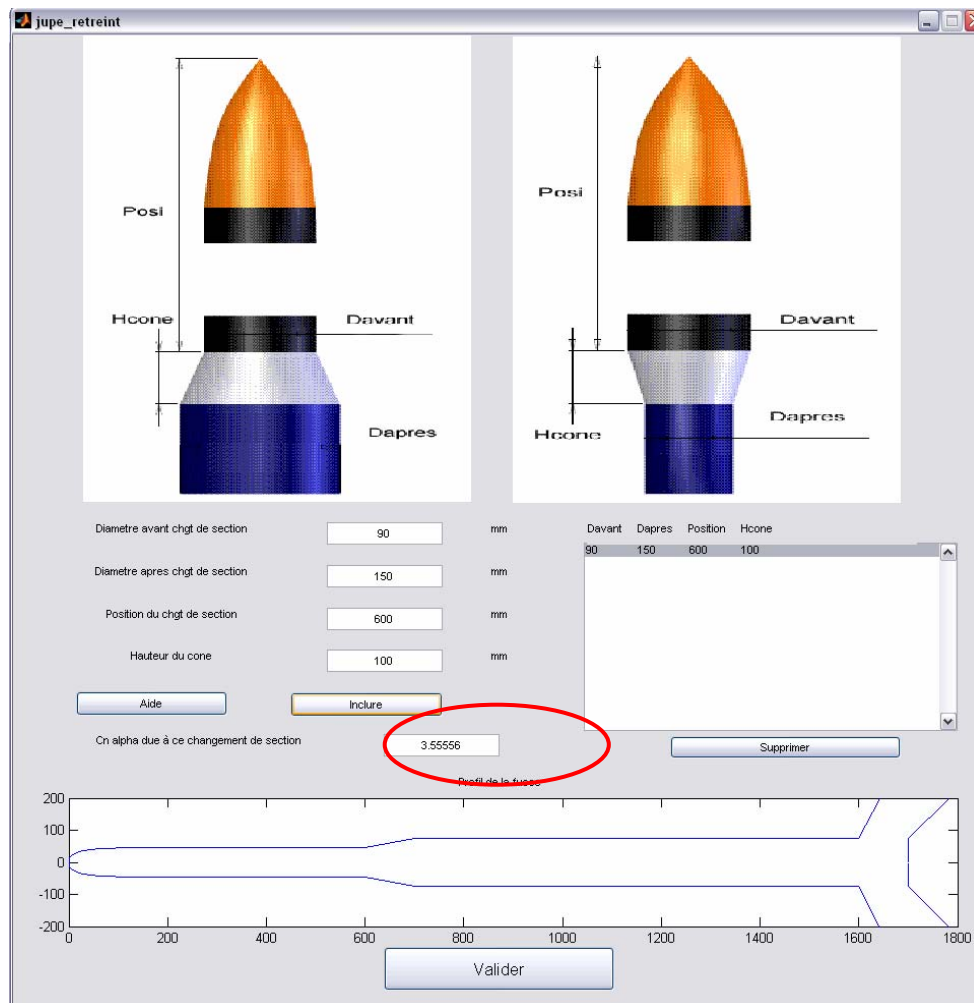


Vous arrivez sur la fenêtre ci-contre : Vos changements de section sont configurables au travers de 4 paramètres :



- Le diamètre avant le changement de section considéré
- Le diamètre après le changement de section considéré
- La position du changement de section : Par rapport au sommet de l'ogive, position à laquelle le début du changement intervient.
- Hauteur du cône : distance sur laquelle s'échelonne le changement de section

Une fois ce changement effectué, cliquez sur l'onglet inclure pour valider le changement de section :

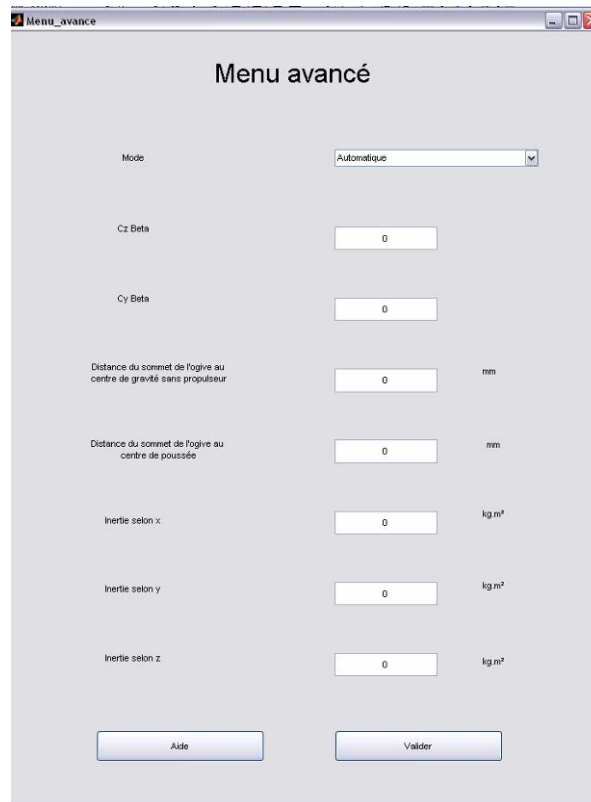


Vous pouvez voir apparaître en temps réel l'évolution du profil de votre fusée sur la figure en bas de l'écran. Par ailleurs, le  $C_n$  Alpha apporté par votre changement de section est automatiquement calculé.

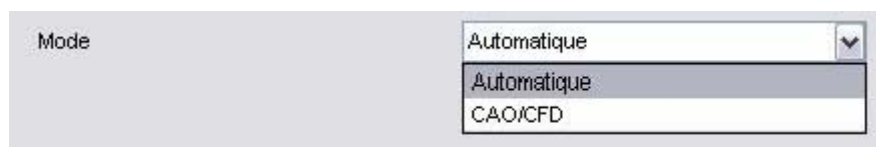
Pour supprimer un changement de section, sélectionnez-le puis cliquez sur l'onglet « Supprimer ». Vous pouvez ajouter et supprimer autant de changement de section que vous désirez. Une fois que le profil recherché de votre fusée est obtenu, cliquez sur l'onglet « Valider ». Vous revenez alors au menu « Caractéristiques de la fusée ».

### 2.3 °) Menu Avancé

Dans le cas de certains projets particulièrement complexe, il peut être utile de pouvoir rentrer certains paramètres à la main, après utilisation par exemple d'un logiciel de CAO et de CFD. Dans cette perspective, une fenêtre spéciale existe, et est accessible en cliquant sur le bouton « Menu Avancé ». L'écran ci-dessous apparaît alors :



Deux modes sont accessibles : Automatique et CAO/CFD



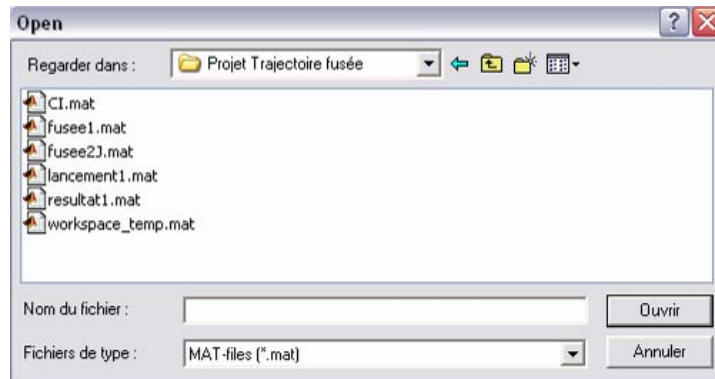
En mode automatique, cela revient à laisser calculer automatiquement les différents paramètres par le logiciel. Choisissez le mode CAO/CFD pour prendre en compte les variables que vous changez manuellement.

Ces variables sont :

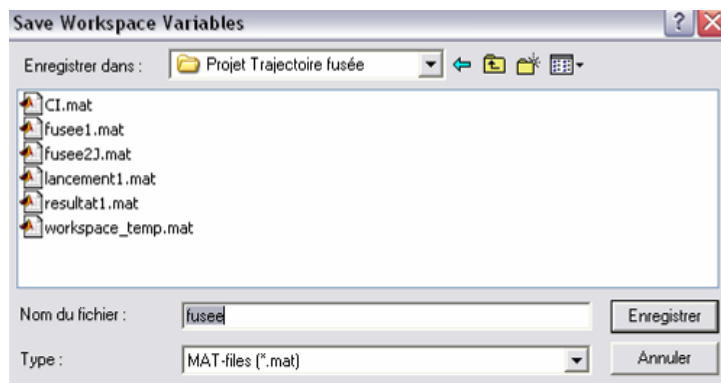
- Cza
- Cyβ
- La distance entre le sommet de l'ogive et le centre de gravité de la fusée
- La distance entre le sommet de l'ogive et le foyer aérodynamique de la fusée
- L'inertie selon l'axe x de la fusée
- L'inertie selon l'axe y de la fusée
- L'inertie selon l'axe z de la fusée

## 2.4 °) Charger/Sauvegarder la configuration

Une fois la configuration géométrique souhaitée obtenue, vous pouvez enregistrer celle-ci dans le cadre d'une utilisation future. Pour ce faire, il vous suffit de cliquer sur l'onglet « Enregistrer » présent dans la fenêtre « Caractéristiques de la fusée ».



Au contraire, si vous avez déjà enregistré une configuration de fusée par le passé, vous pouvez la recharger immédiatement. Pour ce faire, cliquez sur l'onglet « Charger ».



NB : Seule la configuration géométrique de votre fusée sera sauvegardée (Réciproquement chargée)! Les autres paramètres pouvant être issus des conditions initiales de lancement ne seront pas sauvegardés (réciproquement chargés). Pour les enregistrer (charger) également, consultez la rubrique

« sauvegarder lancement » et « charger lancement ».

Une fois toutes ces étapes effectuées, cliquez sur l'onglet « Valider ». Vous reviendrez alors au menu principal.



### 3 °) Conditions initiales.

En cliquant sur cet onglet, vous arrivez sur la fenêtre ci-dessous :

The screenshot shows a software window titled 'conditions\_initiales' with the subtitle 'Implémentation des conditions initiales et des caractéristiques du vent'. The window contains several input fields and buttons. The top section has four rows of inputs: 'Phi0 de la fusée = gîte' (0), 'Teta0 de la fusée = angle de la rampe = site' (80), 'Psi0 de la fusée = azimut' (0), and 'Longueur de la rampe' (4). The middle section is divided into 'Position' (X0, Y0, Z0) and 'Vitesse' (U0, V0, W0), all with input fields set to 0. The bottom section has 'Vitesse du vent' (0) and 'Orientation du vent par rapport au nord epsilon' (0). At the bottom are buttons for 'Aide', 'Charger', 'Sauvegarder', and 'Valider'.

3 éléments sont configurables au travers de cette fenêtre :

- la rampe
- la vitesse et la position initiale de la fusée
- le vent.

#### 3.1 °) La rampe

4 paramètres sont configurables :

- Gîte de la fusée dans la rampe (Phi0) : 0° par défaut
- Site de la rampe (Teta0) : 70 à 80° pour lancements classiques
- Azimut de la rampe (Psi0) : 0° = lancement plein nord, 90° = lancement plein est
- La longueur de la rampe : 2 à 3 m pour les minifs, 4m pour les fusexs

#### 3.2 °) La vitesse initiale et la position initiale de la fusée

Les vitesses selon chacun des axes de la fusée sont paramétrables ainsi que sa position initiale.

### 3.3 °) Le vent

Le vent est paramétrable au travers de deux variables :

- Sa valeur absolue
- L'orientation de sa provenance rapport au nord. ( $0^\circ$  = vent du nord)

### 3.4 °) Charger/Sauvegarder vos conditions initiales

De même que pour la fenêtre « Caractéristiques de la fusée », vous pouvez très bien enregistrer ou charger vos conditions de lancement pour éviter d'avoir à les réécrire à chaque fois. Pour ce faire, cliquez sur l'onglet « Sauvegarder » ou « Charger » selon l'opération souhaitée.

NB: Seuls les paramètres liés aux conditions initiales seront pris en compte. Les variables liées aux caractéristiques géométriques de votre fusée ne seront pas affectées!

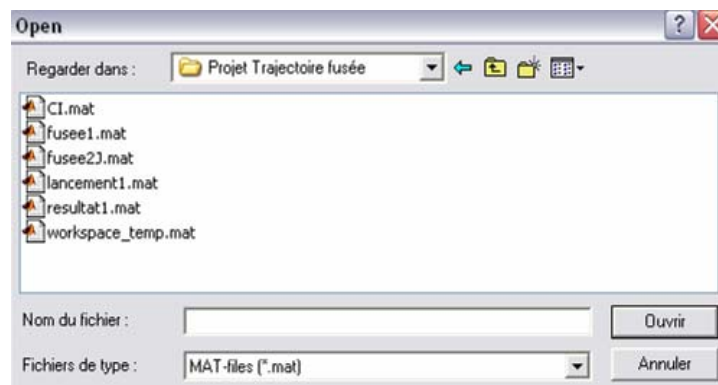
Une fois vos conditions initiales rentrées, cliquez sur l'onglet « Valider » pour revenir au menu principal.

#### 4 °) Charger / Sauvegarder lancement

Comme vous avez pu le constater en lisant les rubriques « Caractéristiques de la fusée » et « Conditions initiales », Vous pouvez enregistrer indépendamment les valeurs des paramètres liés à l'une ou l'autre des fenêtres. Néanmoins, il est possible de charger ou de sauvegarder l'ensemble des variables d'un lancement.

##### 4.1 °) Charger un lancement

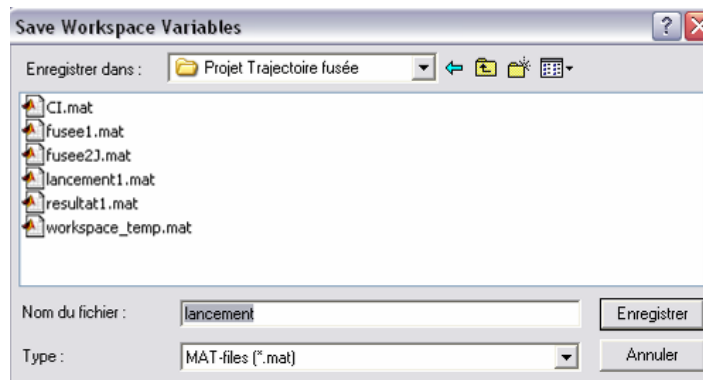
Pour charger un lancement, cliquez sur l'onglet « Charger lancement » du menu principal. Vous arrivez sur la fenêtre ci-dessous :



Sélectionnez le nom du lancement que vous devez sélectionner et cliquez sur « OK ». L'ensemble des paramètres sont alors automatiquement charger dans le programme. Vous pouvez même retourner dans les caractéristiques de la fusée où dans les conditions initiales pour visualiser ces valeurs.

#### 4.2 °) Sauvegarder un lancement

Pour sauvegarder un lancement, cliquez sur l'onglet « Sauvegarder lancement » du menu principal. Vous arrivez sur la fenêtre ci-dessous :



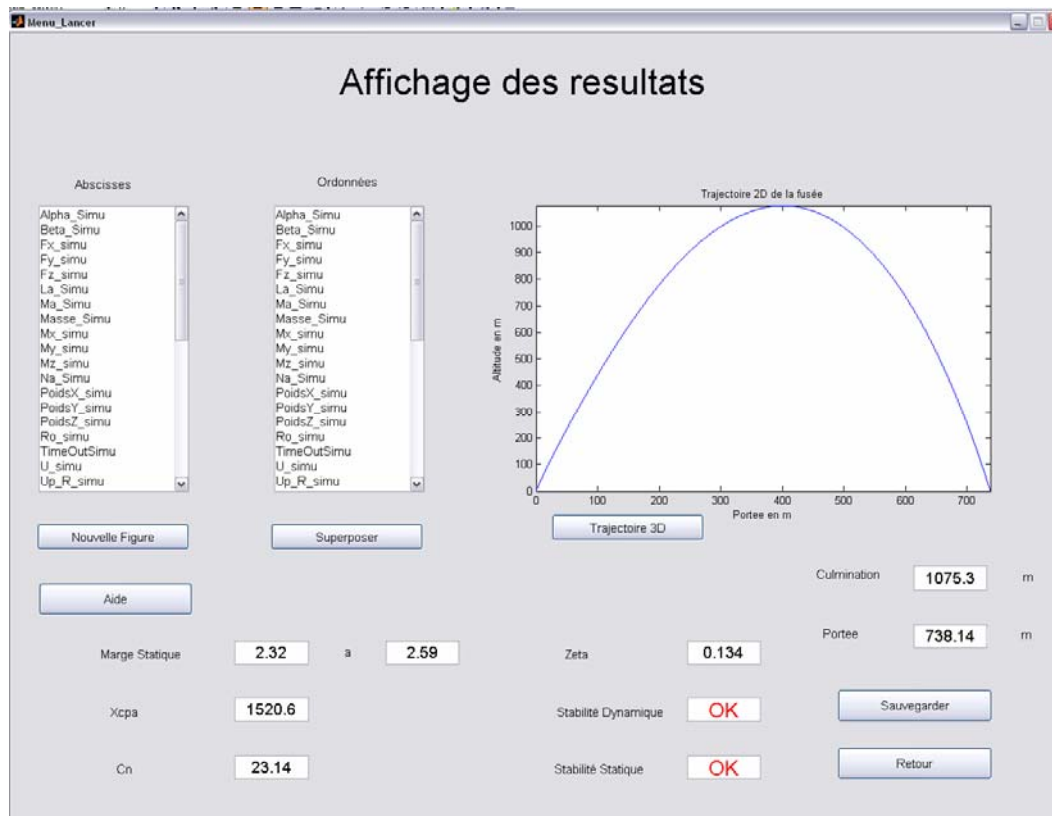
Choisissez le nom de votre lancement et le indiquez le répertoire où vous souhaitez l'enregistrer. Cliquez alors sur « OK ». Il vous est par la suite tout à fait possible de changer les valeurs enregistrées en les modifiant dans les menus de caractéristiques fusées ou de conditions initiales.

## 5 °) Lancement de la simulation

Une fois votre configuration entièrement paramétrée, vous pouvez lancer la simulation qui vous permettra d'étudier l'évolution des paramètres de votre fusée au cours de son vol. Pour ce faire, cliquez sur l'onglet « Lancer la simulation ».

NB : le temps de calcul peut prendre plus ou moins de temps en fonction de la puissance de votre ordinateur.

Vous arrivez alors sur la fenêtre suivante :



### 5.1 °) Résultats primaires

Vous pouvez observer en bas de la fenêtre l'affichage des différentes valeurs associées aux paramètres de la fusée : Marge statique, Xcpa, Cn, Zeta....Les résultats de stabilité statique et dynamique sont également indiqués....

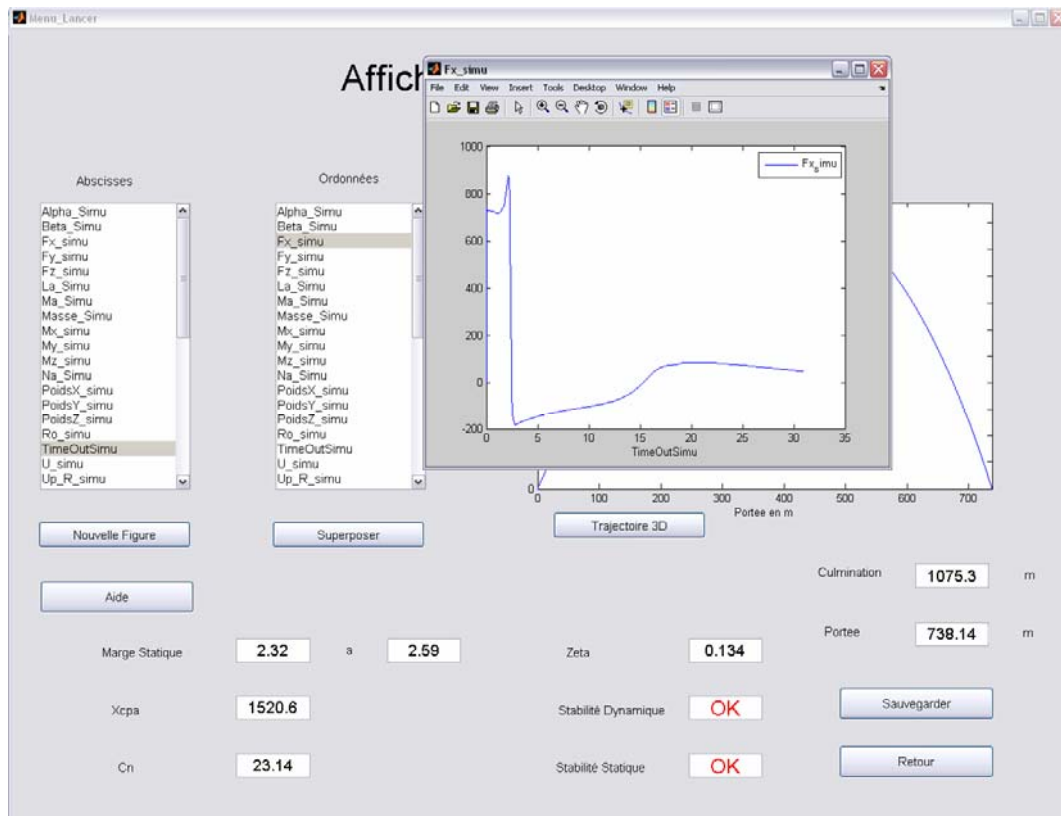
De plus, vous pouvez remarquer que la trajectographie 2D de la fusée s'affiche à l'écran, avec les données de culmination et de portée indiquées juste en dessous.

Ces résultats vous permettent d'avoir une vue globale du vol que suivra votre fusée, et permet ainsi que de vérifier si votre fusée est stable ou non, ce qui est l'un des points essentiel lors de la qualification d'une fusée lors de campagne de lancement.

## 5.2 °) Affichages complémentaires

En haut à gauche de l'écran apparaissent deux listes, identiques dans leur contenu, indiquant l'ensemble des paramètres affichables: la première correspond à la variable qui sera placée en abscisse de la figure, la seconde pour les ordonnées. La liste des variables ainsi que leur signification est indiquée en annexe.

Pour créer une nouvelle figure, sélectionner une variable dans chacune des listes, puis cliquez sur le bouton « nouvelle figure ».

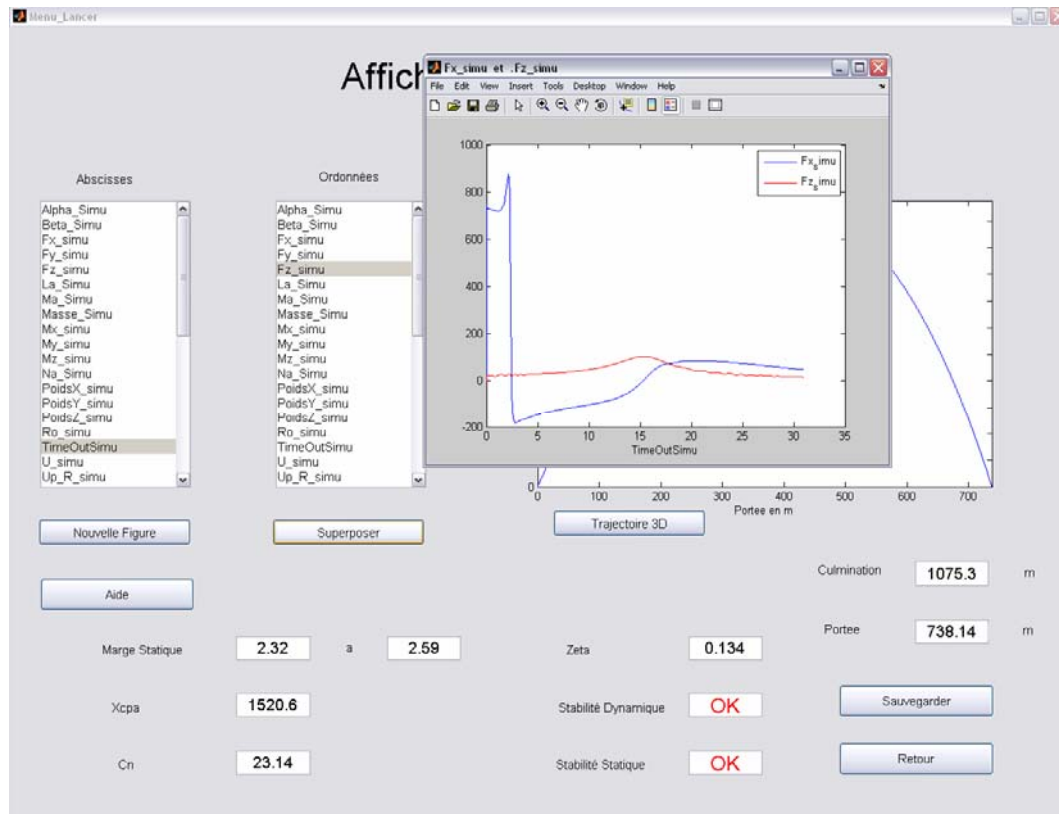


Vous pouvez créer autant de nouvelles figures que vous le souhaitez. Les précédentes ne seront pas effacées.

### 5.3 °) Superposition de résultats

Il peut s'avérer utile dans certains cas de superposer différentes courbes sur la même figure afin d'étudier l'évolution des paramètres.

Pour superposer une courbe à la figure en cours, sélectionnez une nouvelle variable en ordonnée, puis cliquez sur le bouton « superposer ».



ATTENTION :

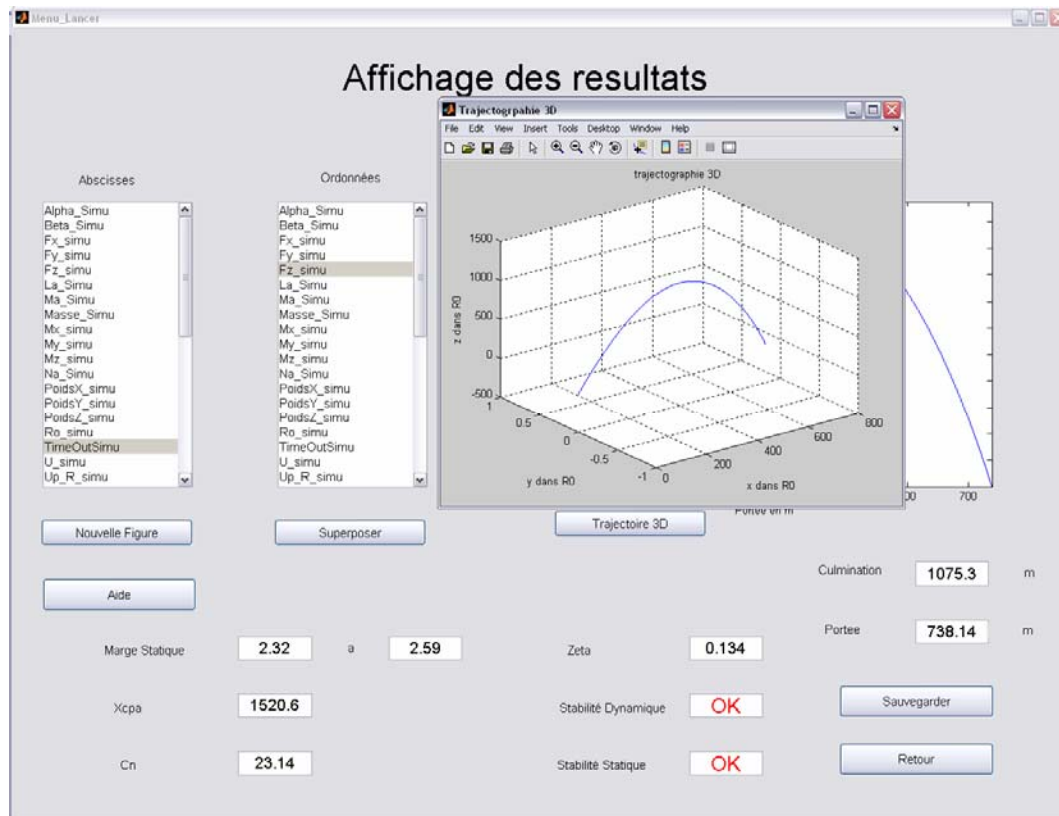
Pour superposer une courbe, vous devez vous assurer que :

- 1 : La variable en abscisse est la même que celle de la première courbe.
- 2 : Vous n'avez pas atteints le nombre maximal de courbes superposables : vous ne pouvez pas en afficher plus de 7 sur la même figure.
- 3 : La figure sur laquelle vous souhaitez superposer une courbe est la dernière figure créée.

Les figures créées étant des figures « classiques » de Matlab, vous pouvez à votre guise zoomer, insérer des curseurs, et effectuer l'ensemble des opérations courantes s'y rapportant. Une fois que vous avez observé l'ensemble de vos paramètres, vous pouvez tout à fait changer de configuration sans pour autant quitter le programme. Pour ce faire, il vous suffit de cliquer sur le bouton « retour » pour revenir au menu principal et sélectionner vos nouveaux paramètres.

### 5.4 °) Trajectoire 3D

Vous pouvez également apercevoir sur cette fenêtre la présence d'un bouton nommé « Affichage 3D » : cliquez dessus pour obtenir, comme son nom l'indique, la trajectoire réelle de la fusée selon les 3 axes.





### 5.5 °) Listes de variables affichables dans la simulation

Voici la liste des paramètres que vous pouvez afficher à l'aide de cette fenêtre. Toutes les unités des variables ci-dessous sont celles du Système International :

Alpha\_Simu : Incidence de la fusée au cours du vol  
Beta\_Simu : Dérapiage de la fusée au cours du vol  
Fx\_simu : Bilan des efforts selon x dans le repère fusée  
Fy\_simu : Bilan des efforts selon y dans le repère fusée  
Fz\_simu : Bilan des efforts selon z dans le repère fusée  
La\_Simu : Bilan des moments aérodynamiques autour de x dans le repère fusée  
Ma\_Simu : Bilan des moments aérodynamiques autour de y dans le repère fusée  
Masse\_Simu : Masse de la fusée au cours du vol  
Mx\_simu : Bilan des moments autour de x dans le repère fusée  
My\_simu : Bilan des moments autour de y dans le repère fusée  
Mz\_simu : Bilan des moments autour de z dans le repère fusée  
Na\_Simu : Bilan des moments aérodynamiques autour de z dans le repère fusée  
PoidsX\_simu : Poids de la fusée selon x dans le repère fusée  
PoidsY\_simu : Poids de la fusée selon y dans le repère fusée  
PoidsZ\_simu : Poids de la fusée selon z dans le repère fusée  
Ro\_simu : Densité de l'air  
TimeOutSimu : Temps  
U\_simu : Vitesse de la fusée selon x dans le repère fusée  
Up\_R\_simu : Accélération selon l'axe x de la fusée  
V\_simu : Vitesse de la fusée selon y dans le repère fusée  
Vabs\_simu : Valeur absolue de la vitesse de la fusée  
Vent\_simu : Vitesse du vent  
VitesseRelativeX\_simu : Vitesse relative vue par la fusée selon x dans le repère fusée  
VitesseRelativeY\_simu : Vitesse relative vue par la fusée selon y dans le repère fusée  
VitesseRelativeZ\_simu : Vitesse relative vue par la fusée selon z dans le repère fusée  
Vp\_R\_simu : Accélération selon l'axe y de la fusée  
W\_simu : Vitesse de la fusée selon z dans le repère fusée  
Wp\_R\_simu : Accélération selon l'axe z de la fusée  
Xa\_Simu : Bilan des efforts aérodynamiques de la fusée selon x dans le repère fusée  
Xf\_simu : Poussée du propulseur  
Ya\_Simu : Bilan des efforts aérodynamiques de la fusée selon y dans le repère fusée  
Za\_Simu : Bilan des efforts aérodynamiques de la fusée selon z dans le repère fusée  
p\_simu : Vitesse de rotation de la fusée autour de x dans le repère fusée  
phi\_simu : Roulis de la fusée  
psi\_simu : Lacet de la fusée  
q\_simu : Vitesse de rotation de la fusée autour de y dans le repère fusée  
r\_simu : Vitesse de rotation de la fusée autour de x dans le repère fusée  
teta\_simu : Tangage de la fusée  
x\_R0 : Position de la fusée selon x0 dans le repère absolu  
y\_R0 : Position de la fusée selon y0 dans le repère absolu  
z\_R0 : Position de la fusée selon z0 dans le repère absolu

## Conclusion

A travers ce projet, commencé il y a tout juste un an, nous avons pu mettre en place un logiciel aidant à la conception de fusée expérimentale. Ce projet se décomposa en plusieurs phases : La mise en place des équations régissant le vol, la modélisation de la fusée sous MATLAB SIMULINK, la validation du modèle obtenu et la mise en place d'une interface graphique.

La validation théorique du modèle a été réalisée en comparaison aux résultats du logiciel équivalent CNES, Trajec. Par ailleurs, des calculs atypiques, tels que des fusées bi étages ou larguées en altitude, effectués sur notre modèle ont retourné des comportements en vols cohérents.

La validation expérimentale du modèle aura quant à elle lieu lors de la campagne nationale de lancement qui aura lieu à la Courtine (23) en août en 2007. L'exploitation consistera en la comparaison des points d'impacts des vols balistiques et de ceux calculés par notre modèle. Un seul facteur n'a pas pu encore être mesuré à l'heure actuelle. Il s'agit du coefficient de frottement sec de la fusée en rampe. Ce coefficient sera lui aussi mesuré lors de la campagne de lancement.

Ce modèle a de plus été doté d'une interface graphique fonctionnelle et robuste. Cette dernière permet une utilisation aisée du logiciel pour les usagers ayant pour habitude d'utiliser Trajec. Cependant, elle permet également un paramétrage plus fin pour les variables pouvant être, par exemple, calculées par CAO ou CFD.

D'autre part, cet outil contrairement à celui disponible, permet l'étude de la stabilité dynamique de l'engin. Dans la mesure où ce critère est moins contraignant que celui de stabilité statique, cela octroiera donc une plus grande souplesse à l'association quant à la conception de ces fusées.

Par la suite ce modèle pourra être codé dans un langage universel tel que le langage C pour permettre au logiciel d'être portable, c'est-à-dire exécutable depuis n'importe quelle plateforme.

Outre l'usage auquel ce modèle est destiné, ce dernier pourra également être utilisé pour l'implémentation de lois de commande, telles que commande en roulis, pilotage, etc... Une des modifications qui pourra être effectuée sur ce modèle est la possibilité de lancement de fusée bi étage.

La présentation de ce modèle à Planète Sciences et au CNES a débutée et sa validation à leurs yeux accompagnée des modifications jugées nécessaires par ces deux organismes permettra peut-être à terme le remplacement du logiciel Trajec.

Ce projet nous a permis de couvrir l'ensemble des étapes nécessaires à la mise en place d'un nouveau modèle comme cela aurait pu être le cas dans un milieu industriel.

Nous espérons, dès alors, que le devenir de notre logiciel sera prometteur...