MASTER INGÉNIERIE DE SYSTÈMES COMPLEXES
PARCOURS ROBOTIQUE ET OBJETS CONNECTÉS

MODÉLISATION DE SYSTÈMES MÉCANIQUES
SYSTÈMES MARINS
TRAVAIL PRATIQUE

Modeling and control of underwater vehicle : Nessie V

Étudiant :
Otmane ATTOU

Enseignant :
Mathieu RICHIER

Table des matières

Table des figures

1 Introduction :

Les robots marins (sous l'eau, mais aussi sur l'eau) sont en plein essor.Ces derniers ont acquis désormais une certaine maturité industrielle et scientifique, notamment Nessie V. Leur usage est largement répandu, principalement pour des applications pétrolières off-shore, pour les applications militaires (surveillance, deminage, etc.) et pour les applications scientifiques (océanographie, climatologie, etc.).

Le but de ce TP est de réaliser la commande de ce scénario suivant :
1. Nous mettons l'AUV dans l'eau.
2. Se plonger dans l'eau de 5 M en faisant un mouvement droite.
3. Avancer avec une constante vitesse de 1m.s^{-1}.
4. Arrêter AUV quand il arrive à la position.

Afin d'atteindre cet objectif, nous présentons l'étude et la modélisation de Nessie V. Ensuite, nous exposons le simulateur sur Simulink pour la valider par la suite. Ainsi, nous concevons l'observateur permettant de mesurer certains paramètres pour constituer une commande par retour d'état.

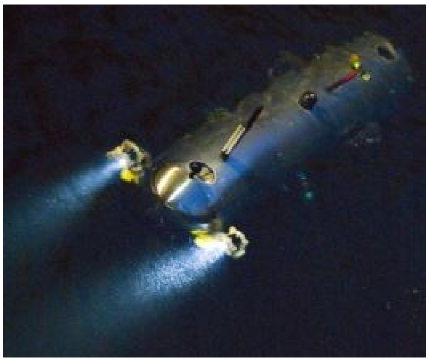

FIGURE 1 – Nessie V après avoir fait une mission dans la nuit

2 Étude et modélisation de Nessie V :

2.1 Les paramètres fixes :

Pour cette étude, les paramètres que nous considérons fixes sont présentées dans le tableau suivant :

ρ_w	La masse volumique de l'eau	$1000\ kg.m^{-3}$
R	Rayon du véhicule	$0.15\ m$
L	Longueur du véhicule	$1.72\ m$
m	Masse	$55\ Kg$
V	Coefficient de volume de flottabilité	$34\ \%$
g	La gravité	$9.81\ m.s^{-2}$
P	Le poids du véhicule	$P = m.g$

2.2 Notations :

Les notations vectorielles et matricielles qui seront utilisées sont les suivantes :

G^b	Vecteur de gravité
B^b	Vecteur de flottabilité
U^b	Thrust mapped vector
K^b	Matrice de friction
E^b	matrice de cartographie
$M_G^b = M_B^b + M_A^b$	Matrice de masse généralisée
M_B^b	Matrice de masse du véhicule
M_A^b	Matrice de la masse ajoutée
$C_G^b = C_B^b + C_A^b$	Matrice de Coriolis généralisée
v	Vecteur de vitesse dans le repère du véhicule
η	Vecteur de position dans le repère fixe
J_θ	Matrice de rotation généralisée

2.3 Cinématique :

2.3.1 Notations :

Dans la référence NED, nous notons la position du centre de gravité de Nessie et son orientation par les vecteurs p_b^n et Θ_{nb} avec :

$$p_b^n = \begin{pmatrix} N \\ E \\ D \end{pmatrix}$$

$$\Theta_{nb} = \begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix}$$

2.3.2 Transformations entre le repère lié au Nessie V et le repère NED :

D'après les calculs fait en cours, nous trouvons les composantes de p_b^n et Θ_{nb} que nous notons par f_i :

$$N' = f_1(t) = u.cos\phi.cos\theta + v.(cos\psi.sin\theta.sin\phi - sin\psi.cos\phi) + w(sin\psi.sin\phi + cos\psi.cos\phi.sin\theta) \quad (1)$$

$$E' = f_2(t) = u.sin\phi.cos\theta + v.(cos\psi.cos\phi + sin\phi.sin\theta.sin\phi) + w(sin\theta.sin\psi.cos\phi - cos\psi.sin\phi) \quad (2)$$

$$D' = f_3(t) = -u.sin\theta + v.cos\theta.sin\phi + w.cos\theta.cos\phi \quad (3)$$

$$\phi' = f_4(t) = p + q.sin\phi.tan\theta + r.cos\theta \quad (4)$$

$$\theta' = f_5(t) = q.cos\phi - r.sin\phi \quad (5)$$

$$\psi' = f_6(t) = q.sin\phi/cos\theta + r.cos\phi/cos\theta \quad (6)$$

2.4 Dynamique :

L'objet de cette partie est d'exprimer le vecteur de forces généralisé $\overrightarrow{\Lambda}$:

— F_a^b Masse ajoutée ($F_a^b = -M_a^b \dot{v} - C_a^b v$)
— $G^b + B^b$: Gravité et flottabilité
— K^b : Friction
— U^b : Thruster
Avec,

$$\overrightarrow{\Lambda} = F_a^b + G^b - B^b + K^b + U^b \tag{7}$$

2.4.1 Masse ajoutée :

la masse ajoutée peut être considérée comme une force et un moment induits par la pression créés par un mouvement harmonique forcé d'un corps. Cette pression est proportionnelle à l'accélération du corps.

Pour calculer la masse ajoutée pour Nessie, nous faisons l'hypothèse de mouvements à basse vitesse et puisque le véhicule a trois plans de symétrie, nous pouvons obtenir la matrice diagonale suivante de masse ajoutée :

$$M_A^b = \begin{bmatrix} Ma_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & Ma_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & Ma_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & Ma_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & Ma_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & Ma_{66} \end{bmatrix} \tag{8}$$

Nous pouvons obtenir ces termes en appliquant la théorie de la bande à un véhicule mince submergé. Les coefficients suivants sont les résultats pour un cylindre :

$$M_A^b = \begin{bmatrix} 0.1 * m & 0 & 0 & 0 & 0 & 0 \\ 0 & \pi\rho_w R^2 L & 0 & 0 & 0 & 0 \\ 0 & 0 & \pi\rho_w R^2 L & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{12}(0.4mR^2 + \pi\rho_w R^2 L^3 & 0) \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{12}(0.4mR^2 + \pi\rho_w R^2 L^3) \end{bmatrix} \tag{9}$$

2.4.2 Gravité et flottabilité :

Les vecteurs de gravité G^b et de flottabilité B^b peuvent être définis par :

$$G^b = \begin{pmatrix} \overrightarrow{F}_g^b \\ \overrightarrow{r}_g \wedge \overrightarrow{F}_g^b \end{pmatrix} = \begin{pmatrix} R_\theta^{-1}\overrightarrow{F}_g^e \\ \overrightarrow{r}_g \wedge R_\theta^{-1}\overrightarrow{F}_g^e \end{pmatrix} \tag{10}$$

et

$$B^b = \begin{pmatrix} \overrightarrow{F}_b^b \\ \overrightarrow{r}_b \wedge \overrightarrow{F}_b^b \end{pmatrix} = \begin{pmatrix} R_\theta^{-1}\overrightarrow{F}_b^e \\ \overrightarrow{r}_b \wedge R_\theta^{-1}\overrightarrow{F}_b^e \end{pmatrix} \tag{11}$$

Où $\overrightarrow{F}_g^e$ et $\overrightarrow{F}_b^e$ sont respectivement le vecteur de la force de gravité et la poussée d'Archimède dans le repère fixe.

Donc, $\overrightarrow{F}_g^e = -m.g.\overrightarrow{z}$ et , $\overrightarrow{F}_b^b = \rho_{eau}.V.g.\overrightarrow{z}$
Avec R_θ est la matrice de rotation suivante :

$$R_\theta = \begin{bmatrix} cos(\psi).cos(\theta) & -sin(\psi).cos(\phi) + cos(\psi).sin(\phi).sin(\theta) & sin(\psi).sin(\theta) + cos(\psi).cos(\phi).sin(\theta) \\ sin(\psi).cos(\theta) & cos(\psi).cos(\phi) + sin(\psi).sin(\phi).sin(\theta) & -cos(\psi).sin(\theta) + sin(\psi).cos(\phi).sin(\theta) \\ -sin(\theta) & cos(\theta).sin(\phi) & cos(\theta).cos(\phi) \end{bmatrix} \tag{12}$$

2.4.3 Force de frottement :

Ici, nous ne considérerons que la force de traînée, car pour les systèmes sous-marins, seule la perte de tourbillon contribue à la force de portance. Nous négligeons donc les forces vortex.

La force de traînée peut être écrite comme suit :

$$K^b = -K_l v - K_q |v| v \tag{13}$$

Où K_l et K_q sont respectivement des coefficients de traînée pour le frottement cutané linéaire et le frottement quadratique :

$$K_l = \begin{bmatrix} k_{lu} & 0 & 0 & 0 & 0 & 0 \\ 0 & k_{lv} & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{lw} & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{lp} & 0 & 0 \\ 0 & 0 & 0 & 0 & k_{lq} & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{lr} \end{bmatrix} \qquad K_q = \begin{bmatrix} k_{qu} & 0 & 0 & 0 & 0 & 0 \\ 0 & k_{qv} & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{qw} & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{qp} & 0 & 0 \\ 0 & 0 & 0 & 0 & k_{qq} & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{qr} \end{bmatrix}$$

Les valeurs de coefficients de friction sont :

$$K_l = diag(2, 23, 0, 10, 0, 0)$$
$$K_q = diag(32, 223, 263, 0, 40, 40)$$

2.4.4 Effets d'actionneurs :

Nessie V contient 6 propulseurs. Par conséquent, les forces et les moments du propulseur peuvent être représentés par l'équation suivante :

$$U^b = E^b F_T^b = E_{6*6}^b \begin{bmatrix} F_{T1}^b \\ F_{T2}^b \\ F_{T3}^b \\ F_{T4}^b \\ F_{T5}^b \\ F_{T6}^b \end{bmatrix} \tag{14}$$

E^b doit être déterminé grâce à la position du propulseur. Par exemple, le moment produit par le propulseur 1 est $\overrightarrow{r}_{t1}^b \wedge \overrightarrow{F_{T1}^b}$.

Nous considérons que les 6 propulseurs engendrent les forces suivantes :
- F_1 et F_2 dirigées par l'axe $\overrightarrow{x}$.
$-$F$_3$ et F_4 dirigées par l'axe $\overrightarrow{y}$.
$-$F$_5$ et F_6 dirigées par l'axe $\overrightarrow{z}$.

Ainsi, le tableau suivant montre les positions de propulseurs :

Motors	Coordonnées dans la basse b	Valeurs en m
Motor 1	$\overrightarrow{r}^b_{t1} = (d_{1x}, d_{1y}, d_{1z})^T$	$(-0.270, +0.175, 0.000)$
Motor 2	$\overrightarrow{r}^b_{t2} = (d_{2x}, d_{2y}, d_{2z})^T$	$(-0.270, -0.175, 0.000)$
Motor 3	$\overrightarrow{r}^b_{t3} = (d_{3x}, d_{3y}, d_{3z})^T$	$(+0.530, +0.000, 0.000)$
Motor 4	$\overrightarrow{r}^b_{t4} = (d_{4x}, d_{4y}, d_{4z})^T$	$(-0.715, +0.000, 0.000)$
Motor 5	$\overrightarrow{r}^b_{t5} = (d_{5x}, d_{5y}, d_{5z})^T$	$(+0.430, +0.000, 0.000)$
Motor 6	$\overrightarrow{r}^b_{t6} = (d_{6x}, d_{6y}, d_{6z})^T$	$(-0.620, +0.000, 0.000)$

Les moments générés par les moteurs sont :
- Par le propulseur 1 est : $M_1 = -7/40 * F_1 \overrightarrow{z}$
- Par le propulseur 2 est : $M_2 = 7/40 * F_2 \overrightarrow{z}$
- Par le propulseur 3 est : $M_3 = 53/100 * F_3 \overrightarrow{z}$
- Par le propulseur 4 est : $M_4 = -143/200 * F_4 \overrightarrow{z}$
- Par le propulseur 5 est : $M_5 = -43/100 * F_5 \overrightarrow{y}$
- Par le propulseur 6 est : $M_6 = 31/50 * F_6 \overrightarrow{y}$

Donc, la matrice U^b est la suivante :

$$U^b = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -43/100 & 31/50 \\ -7/40 & 7/40 & 53/100 & -143/200 & 0 & 0 \end{bmatrix}$$

2.4.5 Les équations d'accélérations :

Nous en déduisons les équations d'accélérations suivantes, et nous les notons par f_i :

```
%%Les fonctions permettant de faire les transformations entre Body et Ned
f7(X')=(F1+F2+g_nu(1)+Kb(1))/M(1,1)+v*r-w*q;          %u'
f8(X')=(F3+F4+g_nu(2)+Kb(2))/M(2,2)+w*p-u*r;          %v'
f9(X')=(F5+F6+g_nu(3)+Kb(3))/M(3,3)+u*p-v*p;          %w'
f10(X')=((M(5,5)-M(6,6))*q*r+g_nu(4)+Kb(4))/M(4,4);          %p'
```

```
f11(X')=1/M(5,5)*(-(M(4,4)-M(6,6))*p*r-d5x*F5+d6x*F6+g_nu(5)+Kb(5)); %q'
f12(X')=1/M(6,6)*(-(M(5,5)-M(4,4))*p*q+d2y*F1+d2y*F2+d3x*F3+d4x*F4+g_nu
    (6)+Kb(6)); %r'
```

2.5 Modèle d'état :

2.5.1 Le choix du vecteur d'état :

Nous choisissons le vecteur d'état suivant :

```
X=[N,E,D,phi,theta,psi,u,v,w,p,q,r]';
```

Les sorties cherchées sont :

```
Y=[u,v,w,p,q,r]';
```

et les entrées du système sont :

```
 U=[F1,F2,F3,F4,F5,F6]'
```

2.5.2 Le système dans l'espace d'état :

Nous obtenons la matrice d'état A et la matrice d'entrée par calculer les jacobiennes suivantes en utilisant ces commandes :

```
%La jacobienne de f par rapport    N E D 0 0 0 u v w p q r

f=[f1 ;f2; f3 ;f4 ;f5 ;f6; f7; f8; f9; f10; f11; f12];
J_A = jacobian(f', [N E D phi theta psi u v w p q r] );
J_B = jacobian(f', [F1 F2 F3 F4 F5 F6] );
```

La matrice de sortie C est la suivante :

```
C=[zeros(6),eye(6)]
```

La matrice de couplage D est :

```
D=zeros(6)
```

3 L'analyse du système en boucle ouverte :

3.1 L'étude du système

Nous travaillons sur l'interface de Matlab afin d'analyser les différentes caractéristiques du système étudié.
Dans un premier temps, nous définissons les différentes paramètres fixes du problème.

Ceci est représenté dans le fichier paramètres.Ici, nous calculons les valeurs numériques de A, B, C et D.

3.2 La stabilité :

En calculant les valeurs propres de A, par la commande de $eig(A_mat)$, nous trouvons que les valeurs propres sont inférieurs ou égales à 0. Donc le système est à sa limite de stabilité.

3.3 Commandabilité et observabilité :

Nous calculons la matrice de commandabilité, et nous remarquons que le système n'est pas complètement commandable. Car le rang de cette dernière est 10.

Donc nous ne sommes pas capables de mesurer les positions de notre véhicule dans le repère earth. Pour cela, nous utilisons un centrale optique à bord de AUV avec un LBL (Long Base Line). De même pour l'observabilité, le système n'est observable et la matrice d'observabilité est de rang 8.

3.4 Réponse du système :

Nous complétons le schéma bloc donné par ajouter le bloc $PosE_M$ concernant la position et l'orientation, ainsi le bruit correspondant. Ensuite, les différents blocs d'affichage.

Nous modifions également le bloc de l'observateur afin d'obtenir toutes les données.

3.4.1 Positions de Nessie V :

Nous utilisons ce code permettant de présenter la position du véhicule en fonction du temps.

```
%G n ration axe des abscisses (Time)
for i = 1 : size(PosE_M(:,1),1)
    Time(i) = 0.01*i ;
end



%Les diff rents positions de Nessie V
Pos_X=PosE_M(:,1);    % positions X en fonction du temps
Pos_Y=PosE_M(:,2);    % positions Y en fonction du temps
Pos_Z=PosE_M(:,3);    % positions Z en fonction du temps
Orientation_X=PosE_M(:,4);    % phi autour de X en fonction du temps
Orientation_Y=PosE_M(:,5);    % theta autour de X en fonction du temps
Orientation_Z=PosE_M(:,6);    % psi autour de X en fonction du temps



figure(1)
grid on;

subplot(211)
plot(Time,Pos_X,'r',Time,Pos_Y,'m+',Time,Pos_Z,'g*')

legend('Pos_X : Position selon X','Pos_Y : Positiob selon Y','Pos_Z :
    Position selon Z','Location','northwest')


title('Position De Nessie V en fonction du temps ','FontSize',23,...
        'FontWeight','bold','FontName',...
        'Times New Roman','Color','k')
xlabel('Time','FontSize',15,...
        'FontWeight','bold','FontName',...
        'Times New Roman','Color','b')
ylabel('Positions en m','FontSize',15,...
        'FontWeight','bold','FontName',...
```

```
        'Times New Roman','Color','b')

subplot(212)
plot(Time,Orientation_X ,'y—',Time,Orientation_Y ,'cx',Time,Orientation_Z ,
    'bp','LineWidth',2)

legend('Phi : Orientation autour X','Theta : Orientation autour Y','Psy :
    Orientation autour Z','Location','northwest')


title('orientations angulaires autour de  X, Y et Z en fonction du temps'
    ,'FontSize',23,...
        'FontWeight','bold','FontName',...
        'Times New Roman','Color','k')
xlabel('Time','FontSize',15,...
        'FontWeight','bold','FontName',...
        'Times New Roman','Color','b')
ylabel('orientations angulaires','FontSize',15,...
        'FontWeight','bold','FontName',...
        'Times New Roman','Color','b')
```

Nous obtenons pour une entrée de $[10000000]$, les courbes suivantes :

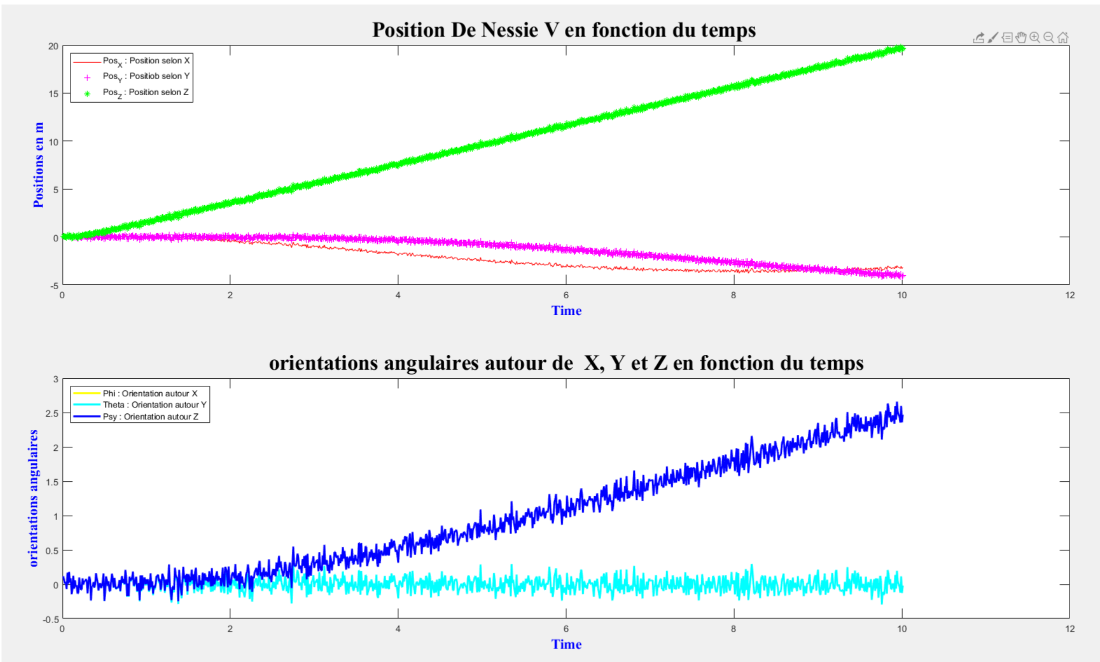

FIGURE 2 – Les différents positions de Nessie V en fonction du temps à une entrée donnée

3.5 Affichage des accélérations vitesses et positions :

Le bloc *scope* permet d'obtenir les différentes vitesses et accélérations. Donc, nous les plaçons comme la figure 3 montre :

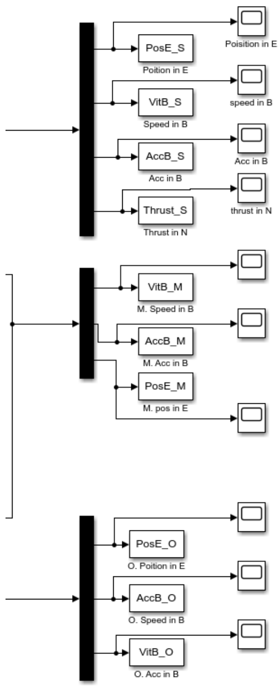

FIGURE 3 – Affichage des différentes positions, vitesses et accélérations

4 Observateur de Luenberger :

4.1 Principe et objectif :

Nous nous intéressons à développer un observateur permettant de récupérer le mouvement de cavalement et pilonnement.
Pour cela, nous choisissons en début les pôles de notre observateur comme :

$$Lobs = [1, 1, 1, 1, 1, 1, 10, 1, 20, 1, 1, 1]$$

Et nous rappelons que :

$$\dot{\hat{X}} = A.X + B.U + L * (y - \hat{y})$$

4.2 L'algorithme :

Le calcul de se fait par :

- Détermination de la valeur numérique de A, B et C.
- Détermination de la valeur de x mesurée correspondante à $PosE_M$.
- Choisir une matrice Lobs.
- Calculer $X(k+1) = PosE_M(k) + A.PosE_M(k) + B.U(k) + L(C.PosE_M(k) - C.PosE_S(k))$.

La difficulté principale trouvée était le choix du Lobs, et comment obtenir les valeurs estimées.

5 Commande par retour d'état :

5.1 Principe et objectif :

Nous développons une commande par retour d'état pour le mouvement de cavalement et pilonnement.

5.2 L'algorithme :

Le calcule de L se fait en quatre étapes :

- Détermination du polynôme caractéristique en boucle ouvert obtenue grace à la commande $poly(A)$.
- Détermination du polynôme caractéristique désiré, autrement dit les pôles désirées, ce qui permet de trouver le polynôme désiré par la commande $poly(poles)$.
- Mettre sous forme compagne commandable le système.
- Calculer de Lcom dans la base de commandabilité.
- Calculer de L dans la base initiale.

6 Conclusion :

Le travail réalisé dans ce TP consolide nos connaissances sur la partie théorique faite en cours. Il nous a permis de modéliser et d'étudier un système complexe tel que Nessie V en utilisant les outils Matlab et Simulink. Malgré les efforts faites dans le cadre de ce TP, je n'ai pas pu de réaliser un observateur et la commande par retour d'état de Nessie.