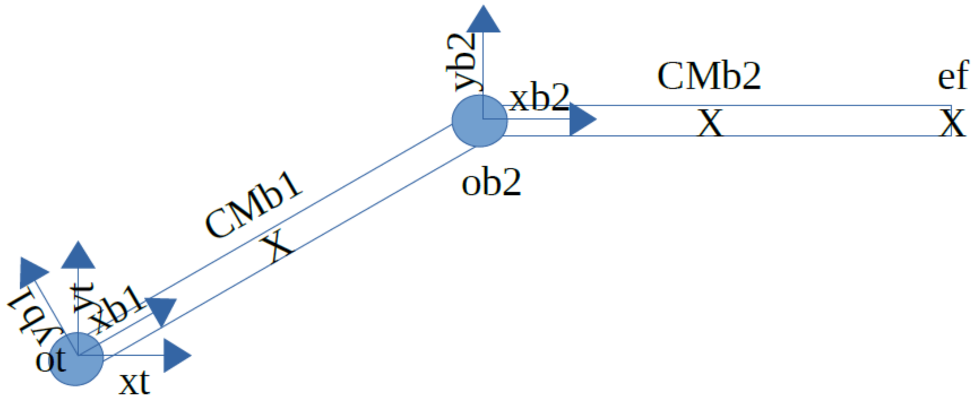
Commande d'un bras de robot à 2 degrés de liberté

1- Présentation du système

L'objectif du TP consiste à dimensionner la loi de commande d'un bras de robot à 2 degrés de liberté étudié en CAO, tel que représenté ci-dessous.



Le repère *t* représente le repère lié à la terre (supposé galiléen), dont l'axe *yt* est dirigé positivement vers le haut.

Le repère *b1* représente le repère lié au bras 1, d'origine *ob1=of* située à l'extrémité fixe du bras1

[l'angle *q1* entre les repères *t* et *b1* est piloté par un moteur *m1*, de masse et d'inertie négligeables, produisant un couple moteur *Fq1*]

Le repère *b2* représente le repère lié au bras 2, d'origine *ob2* située à l'extrémité mobile du bras 1, confondue avec l'extrémité correspondante du bras 2.

[l'angle *q2* entre les repères *b1* et *b2* est piloté par un moteur *m2*, de masse et d'inertie négligeables, produisant un couple moteur *Fq2*]

Les bras ont pour longueurs respectives : *Lb1*, *Lb2*

pour masses respectives : *Mb1*, *Mb2*

pour tenseurs principaux d'inertie respectifs : $Ib1 = \begin{bmatrix} Ixb1 & 0 & 0 \\ 0 & Iyb1 & 0 \\ 0 & 0 & Izb1 \end{bmatrix}$ $Ib2 = \begin{bmatrix} Ixb2 & 0 & 0 \\ 0 & Iyb2 & 0 \\ 0 & 0 & Izb2 \end{bmatrix}$

Les centres de masses CMb1et CMb2 sont situés au milieu des bras.

On suppose également que le mouvement des moteurs produit une puissance de dissipation proportionnelle au carré de leur vitesse angulaire :$P_{dissip\_m1} = \frac{1}{2}.f_{m1}.\frac{dq_1}{dt}^2$,$P_{dissip\_m2} = \frac{1}{2}.f_{m2}.\frac{dq_2}{dt}^2$

De la même façon on suppose que le déplacement de l'outil produit une puissance de dissipation proportionnelle au carré de la norme de sa vitesse de translation par rapport au repère terre :

$$P_{dissip\_outil} = \frac{1}{2}.f_{outil}.\|\overrightarrow{v_{outil/Rt}}\|^2$$

2- Partie symbolique : modèles du robot

<u>*Attention:* l</u>*e calcul symbolique doit être effectué en employant le formalisme des transformations homogènes, aucune expression directe des coordonnées dans le repère 0, vitesses ou énergies ne sera acceptée.* <u>*Dans cette partie il vous appartient de vérifier la cohérence entre les résultats obtenus et attendus en analysant des cas simples avec la fonction* ***subs****: aucune expression fausse de position, vitesse ou énergie ne sera tolérée.*</u>

Compléter le programme *robot_symbolic.m*, dans le dossier *partie_1_etudiant* [dans lequel vous avez le droit de copier vos propres fonctions et programmes au préalable]

de façon à générer [en choisissant *q1* et *q2* comme degrés de liberté]

1- la fonction matlab *clc_coords_outil* déterminant les coordonnées de l'effecteur dans le repère 0 en fonction de *q1*,*q2* et des paramètres du robot.

3- les fonctions matlab *clc_A.m* , *clc_H.m* calculant les matrices *A* et *H* du modèle dynamique du robot, obtenu avec le formalisme de Lagrange.

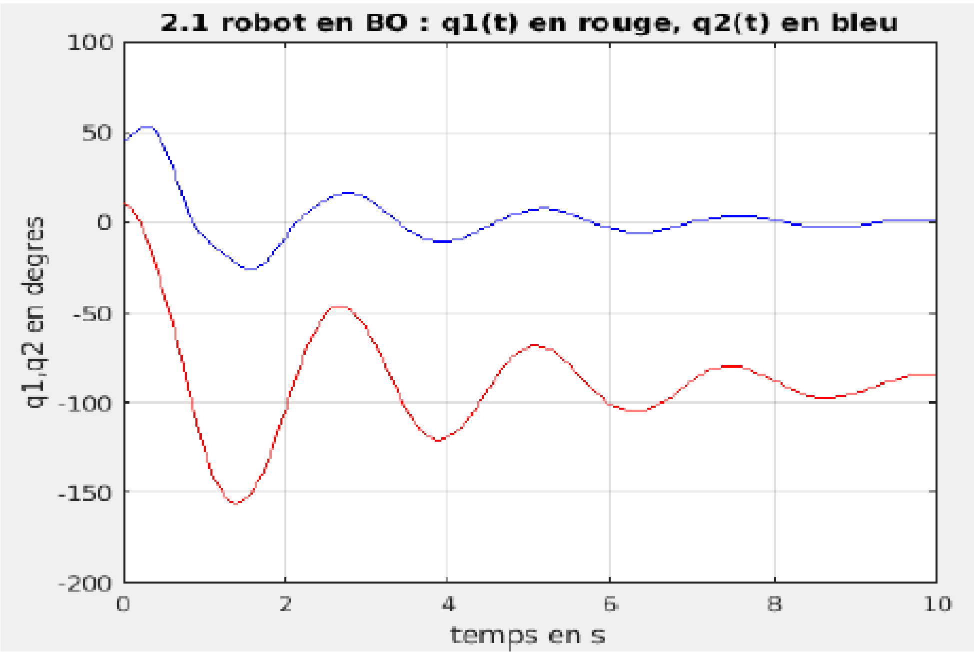
3- Partie numérique : analyse du système

Dans le dossier *partie_2_etudiant*, on vous fournit une trame du programme *robot_numeric.m*, un schéma simulink du robot en boucle ouverte *schema_robot.slx* {les entrées sont *Fq1* et *Fq2*} ainsi que les fonctions *clc_coords_outil* , *clc_A.m* , *clc_H.m, partiellement intégrées au schéma*

3.1 Simulation en boucle ouverte

Dans la partie correspondante (*if simuBo==1*), simuler l'évolution du robot pendant 10 s, sous le seul effet de la force de gravité, à vitesses initiales nulles, depuis les positions initiales q1(0) = 45°, et q2(0) = 10°, Fq1 et Fq2 étant identiquement nuls.

Tracer l'évolution des angles en degrés en fonction du temps, dans une figure telle que ci-dessous.



3.2 Linéarisations en boucle ouverte

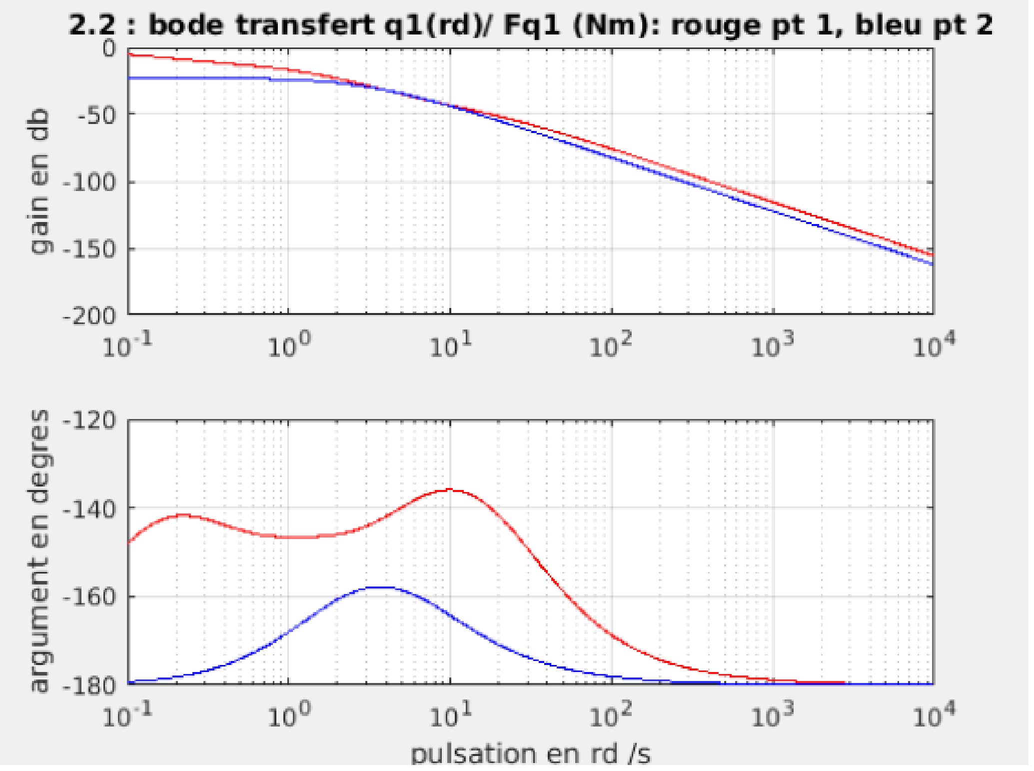
Dans la partie correspondante "*if (partie_22_lin_bo==1), "* linéariser le système autour des 2 points suivants :

Point 1 : $q1 = 10°, q2 = 10°, \frac{dq1}{dt} = 0, \frac{dq2}{dt} = 0, Fq1 = 0, Fq2 = 0$

Point 2 : $q1 = 80°, q2 = -80°, \frac{dq1}{dt} = 0, \frac{dq2}{dt} = 0, Fq1 = 0, Fq2 = 0$

Tracer le diagramme de Bode de la fonction de transfert de *Fq1 vers q1* de chacun des modèles linéarisés, [en rouge pour le point 1, bleu pour le point 2], dans un graphique tel que ci-après

Conclure sur la linéarité du système (afficher la conclusion avec la fonction *disp*)



4- Partie numérique : computed torque

La méthode du <u>*computed torque*</u> permet de linéariser le système en calculant les couples articulaires F_q depuis des accélérations de référence $\frac{d^2 q_{ref}}{dt^2}$, depuis le modèle dynamique du robot. De cette façon le système d'entrée $accq_{ref} = \frac{d^2 q_{ref}}{dt^2}$, et de sortie q, se comportera comme un ensemble d'intégrateurs découplés, et chaque articulation pourra être commandée indépendamment des autres.

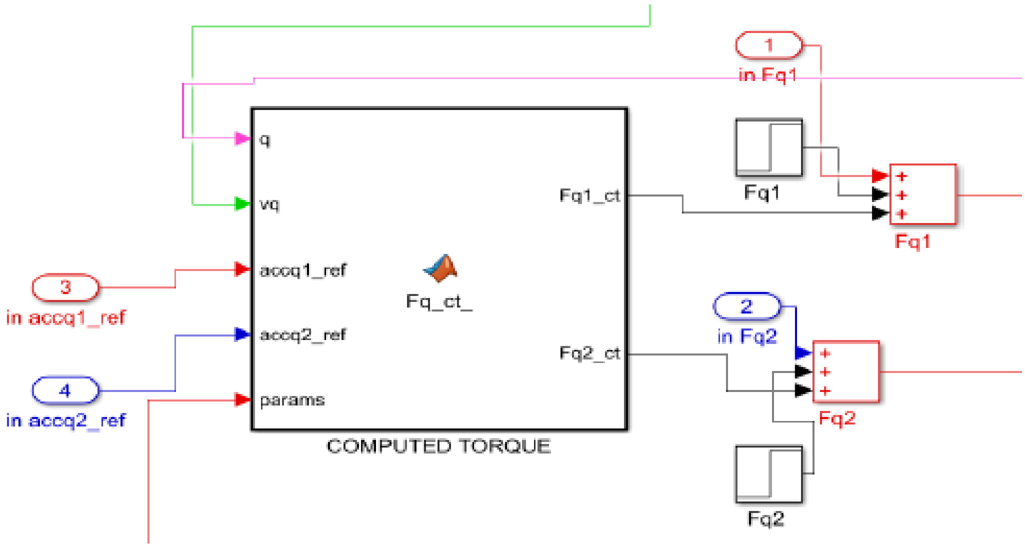
4.1- Enrichissement du schéma Simulink : commande en accélération articulaire

Rappeler l'équation dynamique du système, en fonction de *A*, *H*, *Fq*, $\frac{d^2 q}{dt^2}$

En déduire l'expression de *Fq* en fonction de *A*, *H*, $\frac{d^2 q}{dt^2}$

Remplacer $\frac{d^2q}{dt^2}$ $par accq_{ref}$ dans cette expression, et vous obtiendrez alors l'équation de base du computed torque, d'entrée $accq_{ref}$ et de sortie *Fq (notée Fq_ct par la suite, pour Fq calculé par computed torque)*

Compléter le schéma simulink et le programme pour y ajouter les nouvelles entrées $accq1_{ref}$, $accq2_{ref}$, les inports correspondants, et la fonction calculant la sortie *Fq_ct* du computed torque, que l'on ajoutera à l'ancienne entrée *Fq, conformément au schéma de principe suivant*



4.2- Linéarisation

Linéariser le robot autour du premier point d'équilibre en 2.2, et vérifier que les fonctions de transfert :

entre $q\,1/accq\,1_{ref}$; $q\,2/accq\,2_{ref}$ sont bien celles attendues [il n'est pas demandé de tracer leur diagramme de Bode]. Vous devez afficher les fonctions de transfert à l'écran après simplification, et expliquer pourquoi ce sont bien celles attendues]

5- Loi de commande dans l'espace articulaire

On considère que les ' nouvelles entrées de commande' sont $accq1_{ref}, accq2_{ref}$, et que les sorties à réguler sont $q1, q2$.

5.1 Calcul du régulateur

Déterminer une loi de commande pour obtenir une marge de phase raisonnable à la pulsation au gain unité : wu=60rd./s.(on pourra se baser sur le modèle théorique des fcts de transfert $qi/accqi_{ref}$ pour calculer C)

5.2 Simulation en boucle fermée

Une fois le(s) régulateur(s) calculé(s), compléter votre schéma avec la loi de commande en boucle fermée :

- ajouter les matrices des représentations d'état des régulateurs RB.ssC1, RB.ssC2

- ajouter les entrées de référence pour q1 et q2

Simuler la loi de commande en boucle fermée dans les conditions suivantes à t=0s on part des conditions initiales $q1_{ref} = 10°, q1 = 5°, q2_{ref} = 20°, q2 = 5°, \frac{dq1}{dt} = 0, \frac{dq2}{dt} = 0,$

$Fq1 = 0, Fq2 = 0$

à t=10/wu , on demande $q1_{ref} = 30°, q2_{ref} = 40°$

à t= 20/wu , une perturbation Fq1 est appliquée : $Fq1 = 1Nm, Fq2 = 0$

à t= 30/wu , une perturbation Fq2 est appliquée : $Fq1 = 1Nm, Fq2 = 2Nm$

à t=40/wu, on peut arrêter la simulation

Tracer sur la même figure les courbes suivantes : *ref_q1*(t), rouge avec des +, *q1*(t) (rouge continu), *ref_q2* (t) (bleu avec des +) *q2*(t) (bleu)

6- Prise en compte d'une trajectoire dans l'espace opérationnel

Compte tenu du modèle géométrique du robot, créer dans le modèle de simulation une fonction permettant de déterminer des consignes de variables articulaire permettant de réaliser une trajectoire dans l'espace opérationnel. Si nécessaire, on pourra s'inspirer des travaux présentés dans https://www.ensta-bretagne.fr/jaulin/mastersds_cours_robot_boimond.pdf

L'efficacité du système de commande réalisé sera alors testé pour une trajectoire de l'effecteur réalisée à vitesse constante et définissant 2 fois un losange de sommets successifs : (0.5,0), (1, 0.5), (0.5, 1) et (0, 0.5).

Comparer l'évolution de la trajectoire obtenue à celle attendue. Examiner l'influence du temps de parcours demandé sur la précision réalisée.