

# Simulation des systèmes de solides rigides polyarticulés

Modélisation et analyse du mouvement d'un bras robot 2ddl

## Contexte et problématique

Dans le cadre de cet exercice, nous allons nous focaliser sur un modèle de robot plan (que l'on retrouve par exemple dans les robots manipulateurs SCARA) qui va nous permettre de caractériser pour des tâches de levage et de déplacement de charge les couples moteurs et actions de liaison mis en jeu dans les axes. L'objectif est de mettre en équation ce modèle à l'aide du principe fondamental de la dynamique et du principe des puissances virtuelles sous le formalisme de Lagrange., et d'ensuite utiliser Matlab pour simuler le comportement du robot.

## Modélisation du robot plan 2R

On considère le modèle plan de bras robot suivant :

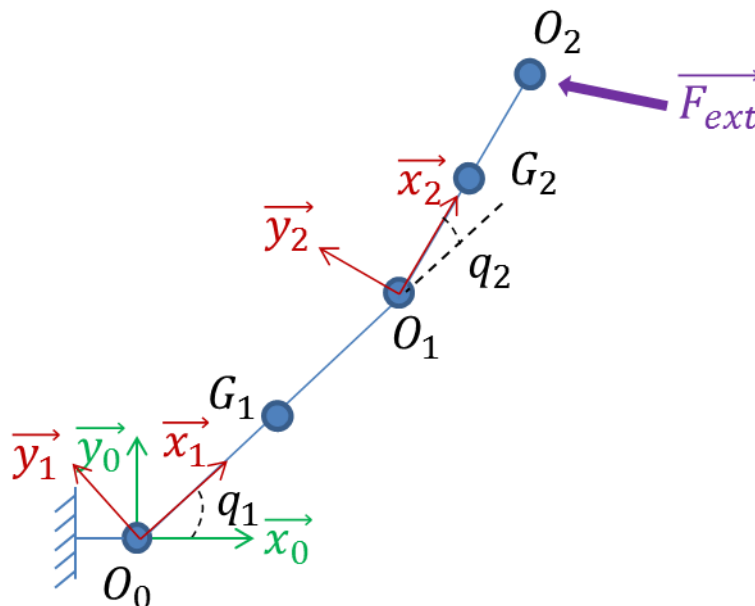


Figure 1 – Modèle plan du bras robot

On considère le robot comme étant composé de deux segments : le bras  $S_1$  et l'avant-bras  $S_2$ . On considère que le robot a seulement deux degrés de liberté, la flexion de

l'épaule  $q_1$  et la flexion du coude  $q_2$ . Lorsque ces deux angles sont nuls, le robot est tendu vers l'avant (dans la direction  $\vec{x}_0$ ).

Les solides sont paramétrés de la manière suivante :

$$\begin{aligned}\overline{O_0O_1} &= l_1 \vec{x}_1 \\ \overline{O_1O_2} &= l_2 \vec{x}_2 \\ \overline{O_0G_1} &= a \vec{x}_1 \\ \overline{O_1G_2} &= b \vec{x}_2\end{aligned}$$

Segment 1 (bras) :  $l_1, m_1, I_1$

Segment 2 (avant-bras) :  $l_2, m_2, I_2$

Le système est soumis à aux efforts suivants :

- Gravité :  $T\{\bar{S} \xrightarrow{g} S_i\} = \left\{ \begin{array}{c} -m_i g \vec{y}_0 \\ \vec{0} \end{array} \right\}_{G_i}$
- Effort extérieur généralisé en  $O_2$  :  $T\{\bar{S} \xrightarrow{\text{action ext}} S_2\} = \left\{ \begin{array}{c} F_x \vec{x}_0 + F_y \vec{y}_0 \\ \vec{0} \end{array} \right\}_{O_2}$
- Couples articulaires générés par les moteurs :  
 $T\{S_0 \xrightarrow{\text{moteur}} S_1\} = \left\{ \begin{array}{c} \vec{0} \\ \Gamma_1 \vec{z}_0 \end{array} \right\}_{O_0}$  et  $T\{S_1 \xrightarrow{\text{moteur}} S_2\} = \left\{ \begin{array}{c} \vec{0} \\ \Gamma_2 \vec{z}_0 \end{array} \right\}_{O_1}$

## Partie 1 : PFD

1. Réaliser le bilan des actions mécaniques extérieures appliquées aux solides  $S_1$  et  $S_2$ . Ne pas oublier d'exprimer les efforts de liaison.
2. Décrire à partir des paramètres du mouvement  $q_1$  et  $q_2$  les vitesses des centres de masse  $G_1$  et  $G_2$  des solides  $S_1$  et  $S_2$ .
3. A partir de la réponse précédente, exprimer les torseurs dynamiques des solides  $S_1$  et  $S_2$  exprimés respectivement en  $G_1$  et  $G_2$ .
4. En isolant successivement les solides 2 puis 1 (ou {1+2}), exprimer le principe fondamental de la dynamique et en déduire les expressions des couples articulaires des articulations de l'épaule et du coude ainsi que les efforts de réaction articulaires associés.

## Partie 2 : PPV

5. Décrire à partir des paramètres du mouvement  $q_1$  et  $q_2$  les vitesses des centres de masse  $G_1$  et  $G_2$  des solides  $S_1$  et  $S_2$ .
6. Exprimer l'énergie cinétique et l'énergie potentielle du système

7. Exprimer les coefficients énergétiques des actions non-conservatives agissant sur le système.
8. Exprimer les équations de Lagrange associées au système.
9. Ces équations ne donnent pas les actions de liaison. Comment procéder pour les obtenir à partir de la méthode de Lagrange ?

## Partie 2 : MATLAB

Dans cette partie, nous nous intéressons à un simulateur permettant de simuler la dynamique inverse du robot. Le principe est le suivant : à partir de la définition d'un mouvement du robot dans l'espace opérationnel (trajectoire du point  $O_2$  dans le plan  $(\vec{x}_0, \vec{y}_0)$ ), on cherche à obtenir les couples articulaires et les efforts de réaction inter-articulaires. Pour cela, deux opérations distinctes sont réalisées:

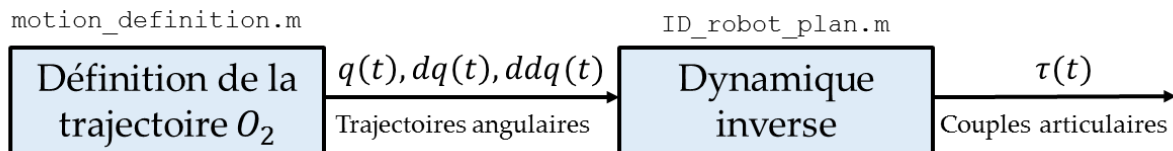


Figure 2 – Définition du simulateur de dynamique inverse

S'ajoutent à cela deux fonctions supplémentaires : un script permettant de paramétrer le modèle (`definition_modele.m`) et un script permettant d'afficher le mouvement simulé (`draw_planar_arm.m`). Le script de paramétrage du modèle est réglé par défaut pour un bras prédéfini.

La fonction définissant le mouvement s'appelle `motion_definition.m`. Pour le moment, il s'agit d'un simple mouvement circulaire du point  $O_2$ . La fonction prend en entrée le pas de temps, et rend en sortie  $q(t)$ ,  $\dot{q}(t)$ ,  $\ddot{q}(t)$  et le vecteur temps  $t$ .

La fonction de dynamique inverse permet de calculer à tout instant de la simulation les couples moteurs mis en jeu à partir du mouvement. C'est sur cette fonction que vous allez travailler dans un premier temps.

10. A partir de la méthode vue en cours, écrire un script permettant d'obtenir le modèle dynamique analytique du robot à l'aide de la méthode de Lagrange. Comparer votre résultat avec celui obtenu aux questions 4 et 8.
11. Ouvrez la fonction `ID_robot_plan.m`. Cette fonction, dans un premier temps, doit pouvoir rendre pour un instant donné les couples moteurs mis en jeu. Implémentez les équations obtenues à la question n°10 dans cette fonction, en prenant bien garde à respecter la syntaxe permettant d'accéder aux paramètres du modèle (pour cela, exécuter le script `definition_modele.m` et examiner la structure « specs » dans le workspace de Matlab).

12. Exécutez le programme principal et observez le résultat obtenu. Comparez-le à la référence proposée ci-dessous. Si vous n'obtenez pas ce résultat, vous avez sans doute fait une erreur dans l'implémentation ou dans l'établissement des équations.

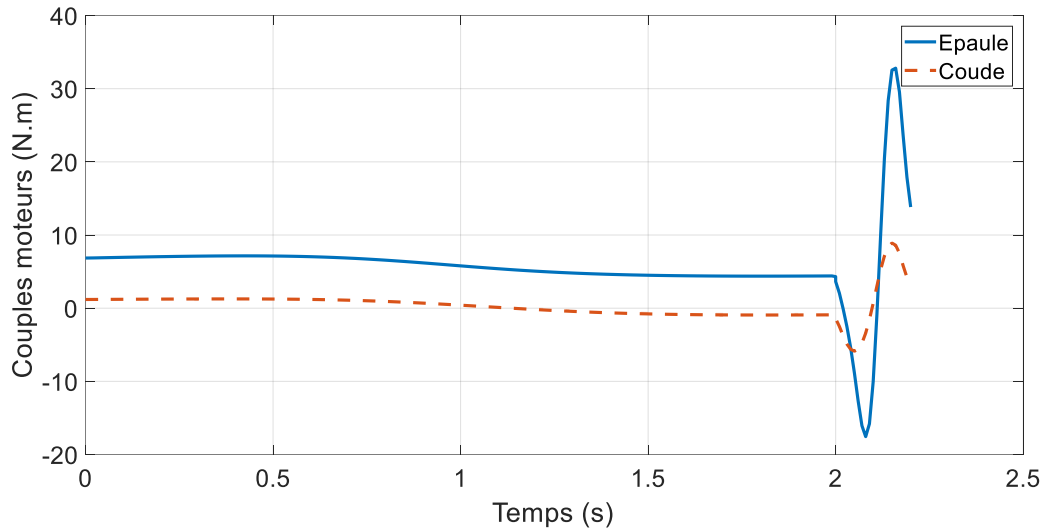


Figure 3 – Couples moteurs à obtenir pour le mouvement de référence

Maintenant que nous avons des équations du mouvement fonctionnelles, il est nécessaire de définir des mouvements réalistes à étudier. Nous allons donc mettre en œuvre une tâche de déplacement entre un point A et un point B à l'aide d'une trajectoire minimisant les à-coups (minimum jerk). D'un point de vue mathématique, pour un mouvement dans le plan  $(\vec{x}_0, \vec{y}_0)$ , cela revient à écrire que la fonction suivante est minimisée :

$$C = \frac{1}{2} \int_0^{t_f} \left( \left( \frac{d^3x}{dt^3} \right)^2 + \left( \frac{d^3y}{dt^3} \right)^2 \right) dt$$

Peut graphiquement représenter ce mouvement de la manière suivante :

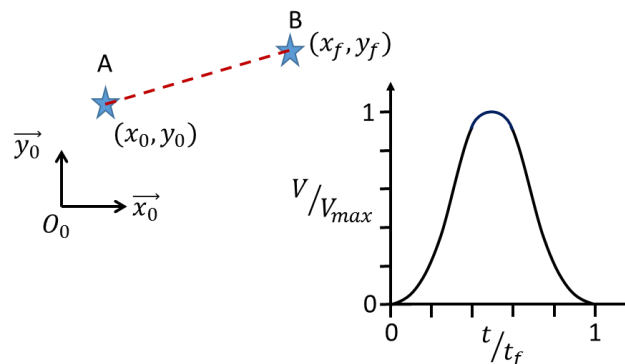


Figure 4 – Définition de la trajectoire libre en minimum jerk et profil de vitesse associé

Comme on peut le voir sur la figure 4, la formulation de la loi en minimum jerk mène à un profil de vitesse caractéristique que l'on nomme habituellement « bell shape » (soit forme en cloche). D'un point de vue mathématique, cela revient à exprimer la trajectoire sous la forme suivante :

$$\begin{cases} x(t) = x_0 + (x_f - x_0)(15\tau^4 - 6\tau^5 - 10\tau^3) \\ y(t) = y_0 + (y_f - y_0)(15\tau^4 - 6\tau^5 - 10\tau^3) \end{cases} \text{ avec } \tau = \frac{t}{t_f}$$

On comprend alors qu'en définissant un point de départ  $(x_0, y_0)$ , un point d'arrivée  $(x_f, y_f)$ , un temps de parcours  $t_f$  et un échantillonnage  $dt$  on est capable de définir complètement une trajectoire  $X(t) = \begin{cases} x(t) \\ y(t) \end{cases}$  pour le point  $O_2$  au cours du temps.

Une fonction appelée `motion_definition_min_jerk.m` permet de définir le mouvement de lever que nous cherchons à étudier.

13. Simulez le mouvement suivant : lever d'un poids d'une position initiale  $q_{ini} = [-70^\circ \ 90^\circ]$  à une position finale  $q_d = [-20^\circ \ 100^\circ]$ . Simulez le mouvement pour un poids de 1, 2, 5, 10, 15 et 20 kilogrammes (modifier en conséquence le champ `Fext` du script) et pour un temps en un temps  $t_f = \{0.5, 1, 2, 5\}$ s. Pour chacune des simulations, calculer le couple moyen et le couple maximum développé au coude et à l'épaule (utiliser pour cela les fonctions `max` et `mean`).
14. Simulez le mouvement suivant (déplacement vers l'avant) : déplacement d'un poids d'une position initiale  $q_{ini} = [-70^\circ \ 90^\circ]$  à une position finale  $q_d = [-20^\circ \ 100^\circ]$ . Simulez le mouvement pour un poids de 1, 2, 5, 10, 15 et 20 kilogrammes (modifier en conséquence le champ `Fext` du script) et pour un temps en un temps  $t_f = \{0.5, 1, 2, 5\}$ s. Pour chacune des simulations, calculer le couple moyen et le couple maximum développé au coude et à l'épaule.
15. A partir des résultats précédents, spécifier les caractéristiques que devraient respecter les actionneurs et leur commande afin de pouvoir réaliser l'ensemble des mouvements précédemment testés.