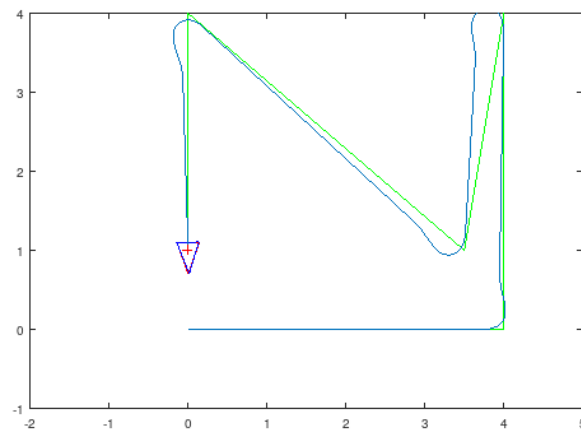


Dynamique et commande prédictive

ROB316 - Planification et contrôle

Bastien HUBERT

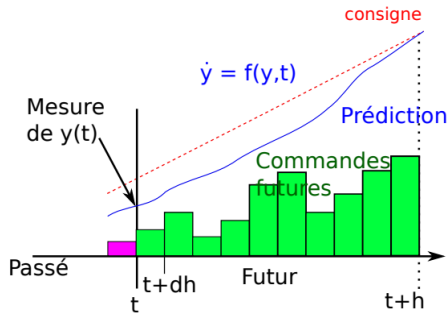


ENSTA Paris - décembre 2022

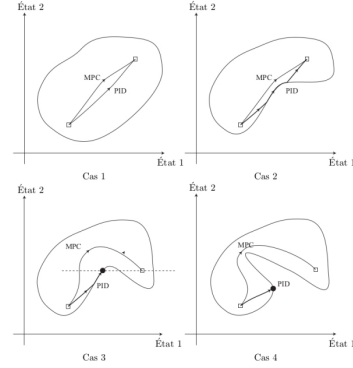
1 Rappels de cours

Le but de ce cours était de présenter un modèle dynamique permettant d'étudier le déplacement d'un robot. Issues du principe fondamental de la dynamique, les équations de ce modèle sont des équations différentielles ordinaires présentant des non-linéarités et sont de la forme $\ddot{q} = F(q, \dot{q}, t)$.

La résolution de ces équations permet de prédire l'évolution de la trajectoire du robot au prix d'un besoin de connaissance du système plus important. Grâce à ces prédictions, il est alors possible de choisir par anticipation la commande qui permet de suivre au mieux la trajectoire désirée, comme illustré ci-dessous :



Principe de la commande prédictive et comparaison avec le PID



Le principe du *model predictive command* (MPC) consiste à chercher la commande optimale au sens de :
$$\min_{\hat{u}} J(x, \hat{u}, T_p) = \int_{t+T_p}^t (\|\hat{x}(\tau; x(t), t)\|_Q^2 + \|\hat{u}(\tau)\|_R^2) d\tau + \|\hat{x}(t+T_p; x(t), t)\|_P^2$$
 dans un système linéaire à contraintes et coût linéaires. Ce système est transformé en équation vectorielle de la forme : $X(k) = \hat{A} \tilde{x}(k) + \tilde{B} U = 0$, et la commande optimale est approximée au sens des moindres carrées par : $U = -B^\# \hat{A} \tilde{x}(k)$.

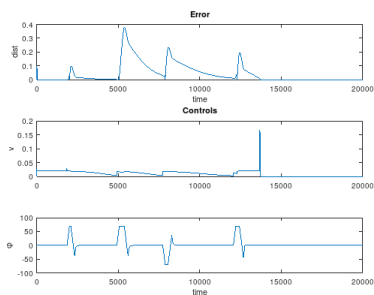
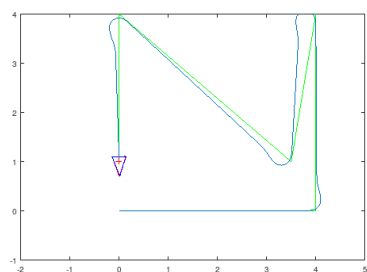
Des variantes, comme NMPC, AMPC, ou GSMPC permettent d'étendre le MPC hors du cadre linéaire convexe dans lequel il s'applique. En particulier, on peut citer la linéarisation par développement de Taylor à l'ordre 1, la discrétisation en pas de temps finis ou encore l'algorithme de Chen & Allgower.

2 Commande avec anticipation, commande prédictive

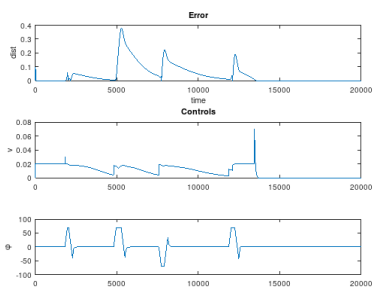
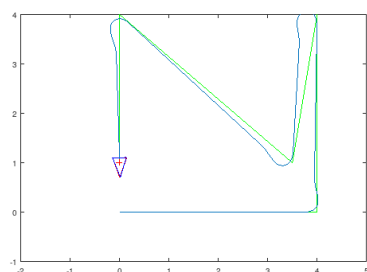
Le but de ce TP était d'implémenter sous Octave les effets d'une commande anticipative et d'une commande prédictive dans le cas du robot bicyclette du précédant TP.

2.1 Anticipation

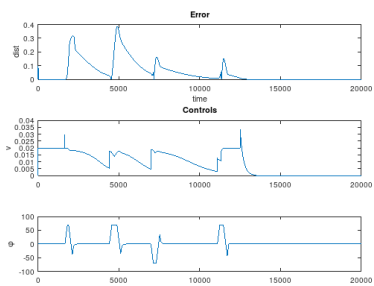
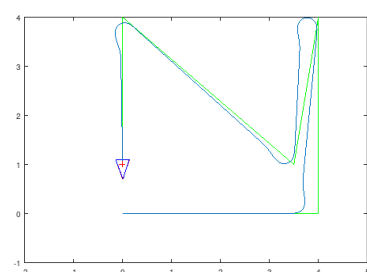
Les figures suivantes montrent les trajectoires obtenues pour différentes valeurs d'horizon :



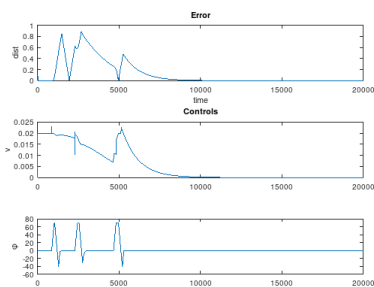
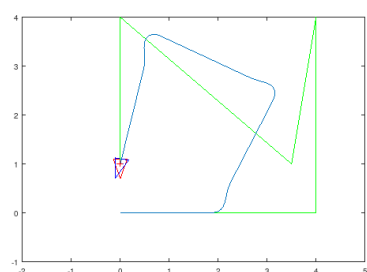
window_size = 1, Total error = 865.9341



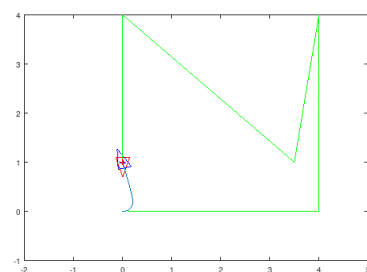
window_size = 5, Total error = 865.9341



window_size = 20, Total error = 1015.5479



window_size = 100, Total error = 2405.0702



window_size = 1000

On voit que pour un horizon de 1, la commande n'est pas anticipée et correspond à une commande réactive. Un horizon de 5 anticipe légèrement tandis qu'un horizon supérieur à 20 ne suit plus la trajectoire désirée, jusqu'à un horizon de 1000 pour lequel le robot se dirige directement vers son objectif.

La trajectoire optimale a lieu pour un horizon entre 5 et 20. En dessous de ce seuil, la trajectoire n'est pas assez anticipée, et au dessus, elle est tronquée et prend par ailleurs bien trop de temps à calculer.

2.2 zone de stabilité d'une commande prédictive

La fonction `verif_stability` permet de trouver la commande qui stabilise le système :

$$\begin{cases} \dot{x}_1 &= x_2 + u \times (\mu + (1 - \mu) x_1) \\ \dot{x}_2 &= x_1 + u \times (\mu - 4(1 - \mu) x_2) \end{cases}$$

Mathématiquement, on trouve que la commande stabilisant le système est $K = [-2.1180, -2.1180]^T$ en tous points où il est stabilisable. Cet ensemble est calculable grâce à la fonction `plot_zone` :

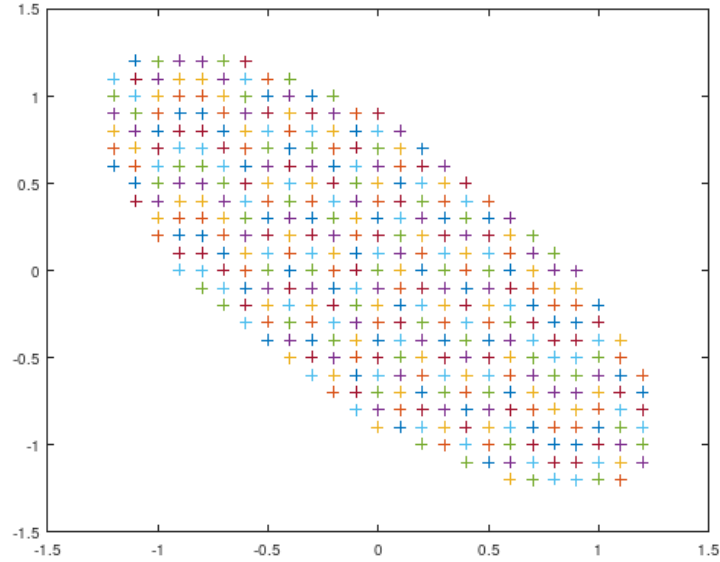


Figure 3: Ensemble des conditions initiales depuis lesquelles le système est stabilisable

2.3 Commande prédictive

En utilisant la commande déterminée à l'étape précédente, on peut donc stabiliser le système à partir de certaines conditions initiales et déterminer sa trajectoire tel que montré ci-dessous :

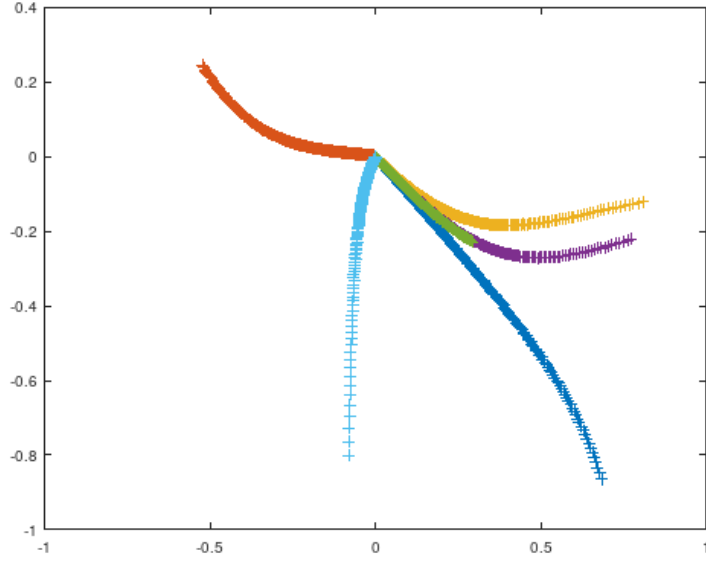


Figure 4: Exemple de trajectoires stabilisées pour différentes conditions initiales avec $K = [-2.1180, -2.1180]^T$

Dans un second temps, nous avons choisi de calculer une commande prédictive en utilisant la technique de MPC par vectorisation vue en cours. Pour les mêmes conditions initiales, la commande prédictive parvient également à stabiliser le système, ainsi que montré dans la figure suivante :

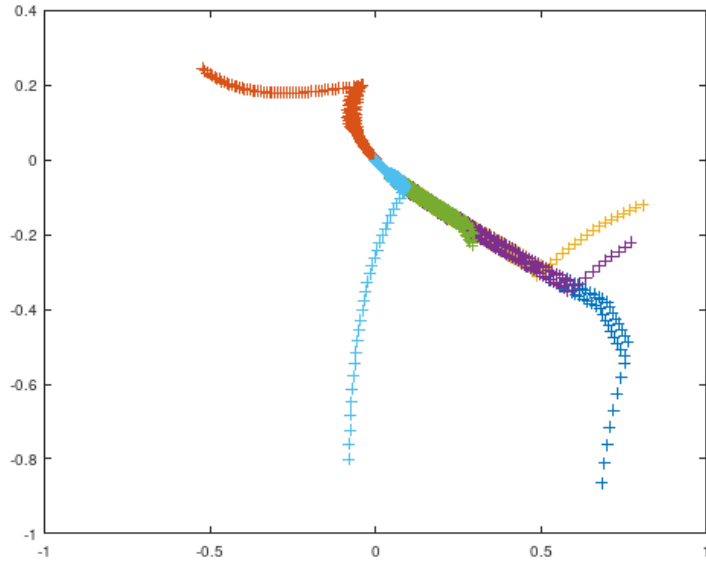


Figure 5: Exemple de trajectoires stabilisées pour différentes conditions initiales avec commande prédictive

On remarque que les trajectoires obtenues sont moins lisses que celles précédemment calculées, mais ceci est vraisemblablement dû aux approximations causées par la discrétisation par pas de temps du problème et le calcul du pseudoinverse de B . La commande prédictive permet donc bien de stabiliser le système.