# Tutorial HPP

Florent Lamiraux et Steve Tonneau
CNRS-LAAS, Toulouse, France

## 1 Steering method

Une méthode de guidage `hpp::core::SteeringMethod` est une classe qui construit des chemins entre des paires de configurations d'un robot. Son rôle est de prendre en compte la cinématique particulière du robot en question. Par exemple, un robot de type voiture ne peut pas exécuter n'importe quel mouvement, à cause des contraintes de roulement sans glissement des roues.

L'objectif de ce tutoriel est de développer une méthode de guidage respectant les contraintes de roulement sans glissement d'un robot de type voiture.

Pour cela, les étapes suivantes sont nécessaires :

1. dériver la classe `hpp::core::Path` afin de modéliser des trajectoires respectant les contraintes de roulement sans glissement de la voiture,
2. dériver la classe `hpp::core::SteeringMethod` afin de créer une méthode de guidage produisant des chemins du type défini à l'étape précédente entre deux configurations données,
3. rendre cette méthode de guidage accessible dans les scripts python.

Le point 1 ci-dessus fait l'objet de la partie suivante.

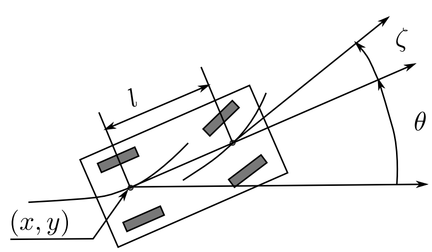


FIGURE 1 – Modèle cinématique de voiture. La configuration de la voiture est représentée par le vecteur $(x, y, \cos\theta, \sin\theta, \zeta)$.

## 1.1 Trajectoires respectant les contraintes de roulement sans glissement

### Platitude différentielle

Considérons la voiture représentée sur la figure 1. C'est un système dit différentiellement plat, dont la sortie plate est le centre de l'axe des roues arrières $(x, y)$. Cela signifie que la trajectoire dans le plan de $(x, y)$ permet de reconstruire la configuration du système uniquement par dérivation.

En effet, soit

— $(x, y)$ la position du centre de l'axe des roues arrières,

— $(\dot{x}, \dot{y})$ sa vitesse,

— $(\ddot{x}, \ddot{y})$ son accélération.

A condition que la vitesse soit non-nulle, on peut établir les relations suivantes :

$$\cos(\theta) = \frac{\dot{x}}{\sqrt{\dot{x}^2 + \dot{y}^2}} \qquad \sin(\theta) = \frac{\dot{y}}{\sqrt{\dot{x}^2 + \dot{y}^2}} \tag{1}$$

La dérivée première de la sortie plate donne l'orientation du véhicule,

$$\kappa = \frac{\tan(\zeta)}{l} = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{\frac{3}{2}}} \tag{2}$$

La dérivée seconde via la courbure donne l'angle de braquage.

## Mise en oeuvre

Etant données deux configurations définies par les vecteurs $(x_1, y_1, c\theta_1, s\theta_1, \zeta_1)$ et $(x_2, y_2, c\theta_2, s\theta_2, \zeta_2)$, nous proposons de définir une trajectoire faisable entre ces deux configurations par l'intermédiaire d'un polynome défini sur l'intervalle $[0, 1]$ à valeurs dans $\mathbb{R}^2$ et décrivant la trajectoire de la sortie plate du système. Nous notons $P$ ce polynôme. Les relations (1) et (2) induisent des équations non-linéaires sur les coefficients du polynôme $P$. Pour rendre ces équations linéaires, il suffit d'ajouter les contraintes suivantes :

1. la dérivée première de la sortie plate en 0 et en 1 a pour norme 1,
2. la dérivée seconde de la sortie plate en 0 (resp. 1) est orthogonale à la dérivée première en 0 (resp. 1).

**Question 1 :** écrire les équations sur les cofficients du polynôme $P$ en fonction des configurations initiales et finales $(x_1, y_1, c\theta_1, s\theta_1, \zeta_1)$ et $(x_2, y_2, c\theta_2, s\theta_2, \zeta_2)$.

**Question 2 :** quel degré de $P$ permet d'avoir autant d'équations que d'inconnues.

**Question 3 :** résoudre le système d'équation et exprimer les coefficients de $P$ en fonction des configurations initiales et finales.