Commande et estimation pour la robotique mobile

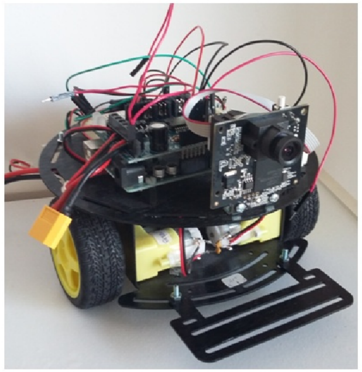
SYS5240

ESIEA 5A

S. Bertrand
sbertrand@esiea.fr

Commande d'un robot mobile

- Robot mobile à roues différentielles (*differential drive*)




Différents modes de locomotion : 2 roues , 2 chenilles

- Méthodes généralisables à d'autres modes de locomotion (ex : 4 roues motrices)

Commande d’un robot mobile

- Modélisation
- Problématique de la commande
- Commande en position et en orientation
  Commande hiérarchique
- Commande par points de passages
- Evitement d’obstacles
  Commande par champs de potentiels

Modélisation

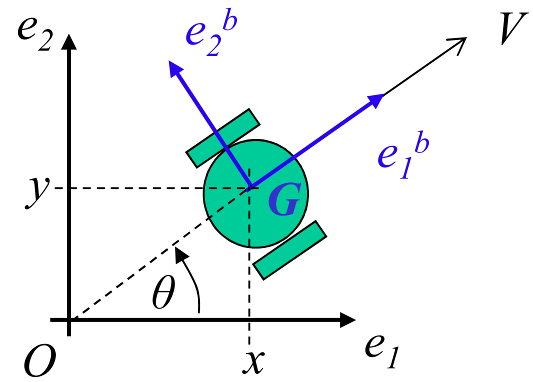
- Repère fixe de référence $\mathcal{R}_0 : (O, e_1, e_2)$
- Repère mobile lié au robot $\mathcal{R}_b : (G, e_1^b, e_2^b)$

Position : $p = \overrightarrow{OG} = \begin{pmatrix} x \\ y \end{pmatrix}_{(R_0)}$

Pose : $\begin{pmatrix} p \\ \theta \end{pmatrix}$

Orientation : $\theta = \widehat{\left(e_1, e_1^b\right)}$

$M(\theta)_{(R_b)->(R_0)} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$



Modèle cinématique :

$$\begin{cases} \dot{x} = V.\cos\theta \\ \dot{y} = V.\sin\theta \\ \dot{\theta} = \omega \end{cases}$$

Vecteur de commande : $u = \begin{bmatrix} V \\ \omega \end{bmatrix}$

Vecteur d'état : $X = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$

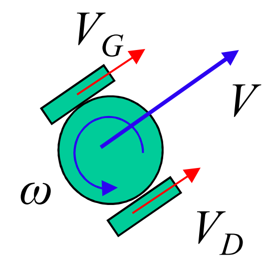
Représentation d'état :

$$\dot{X} = f(X, u)$$

Modélisation

- Réalisation de la vitesse et de la vitesse de rotation

Vitesses linéaires des roues $\left[\begin{array}{l} v_D = r_D.\omega_D \\ v_G = r_G.\omega_G \end{array}\right]$ Vitesses angulaires des roues



$$V = \frac{v_D + v_G}{2} = \frac{r_D\omega_D + r_G\omega_G}{2} = r\frac{\omega_D + \omega_G}{2}$$

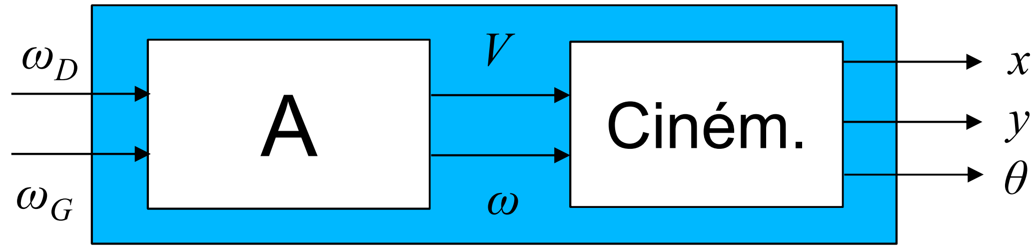
$$\omega = \frac{v_D - v_G}{d} = \frac{r_D\omega_D - r_G\omega_G}{d} = r\frac{\omega_D - \omega_G}{d}$$

H $r_G = r_D = r$

r : rayon d'une roue
d : distance entre les roues

$$\begin{bmatrix} V \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{d} & -\frac{r}{d} \end{bmatrix} . \begin{bmatrix} \omega_D \\ \omega_G \end{bmatrix}$$

$$u = A * u'$$



Problématique de la commande

- Différents problèmes de commande :

calculer $V(t)$ et $\omega(t)$ (equiv. $\omega_D(t)$ et $\omega_G(t)$) permettant au robot

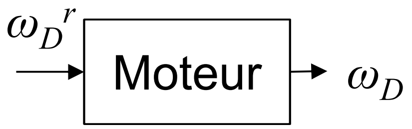
  - d'aller à une position fixe (x^r, y^r) donnée
  - d'atteindre une pose fixe donnée (x^r, y^r, θ^r)
  - de suivre une trajectoire donnée $(x^r(t), y^r(t))$
  - etc..

- Dans ce cours : commande vers une position fixe
  - $x(t) \rightarrow x^r = cste$ et $y(t) \rightarrow y^r = cste$ pour $t \rightarrow +\infty$

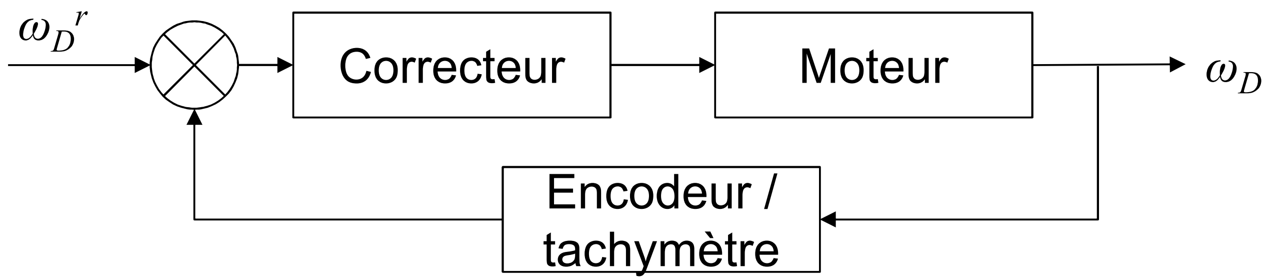
  => $\theta(t)$ est considéré comme une variable « interne »
  (on ne cherche pas à lui affecter une valeur particulière)

Hypothèses

- Remarque : qu'en est-il de la réalisation des vitesses de rotation des roues ?
  - Asservissement moteur déjà réalisé
    - ex : servomoteurs à rotation continue



  - Asservissement moteur à réaliser
    - ex : moteur CC



- Dans ce cours : on supposera l'asservissement en vitesse de rotation des roues réalisé

Commande

- Une stratégie de commande intuitive ?

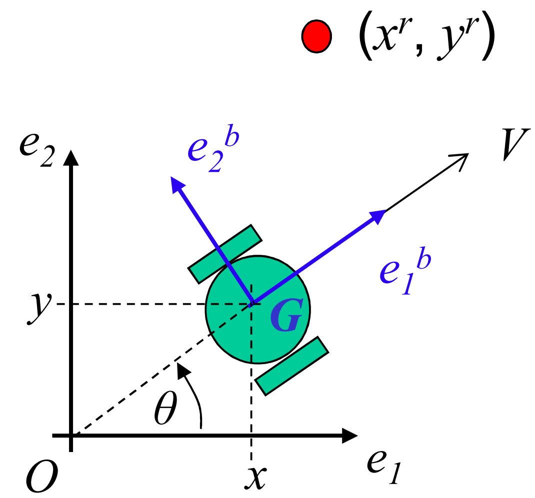
  1) pivoter le robot dans la direction de l'objectif $\theta \rightarrow \theta^r = \arctan\left(\frac{y^r - y}{x^r - x}\right)$

  (attention au domaine de validité de cette formule)

  $$\omega = k_1(\theta^r - \theta) \quad k_1>0 \qquad V = 0$$

  2) avancer jusqu'à l'objectif $x \rightarrow x^r$ et $y \rightarrow y^r$

  $$\omega = 0 \qquad V = k_2\sqrt{(x^r - x)^2 + (y^r - y)^2} \quad k_2>0$$



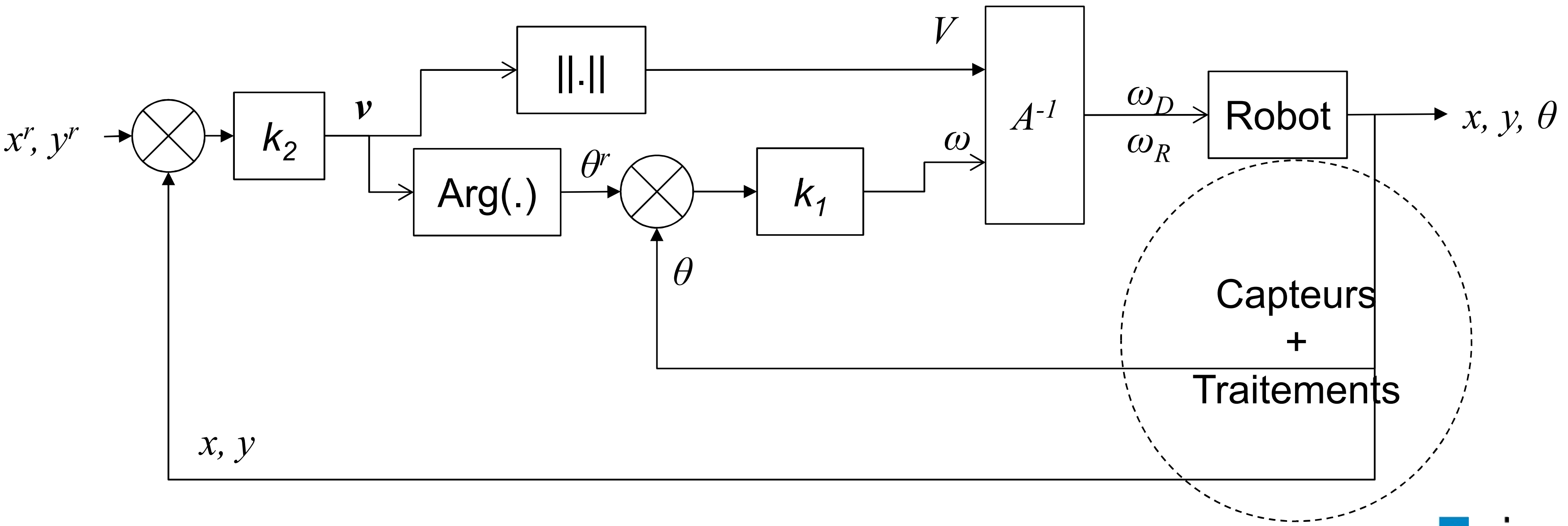
- Limitations :
  - Pas de correction de l'orientation possible durant la phase 2 (boucle ouverte)
  - erreur d'orientation (même faible) => forte erreur en position

- Solution : mettre à jour V , θ^r et ω à chaque instant

Commande

- Commande en cascade

1. Commande en position : déterminer $\boldsymbol{v} = [v_x \; v_y]^T$ tel que $x \rightarrow x^r$ et $y \rightarrow y^r$
2. En déduire $V = ||\boldsymbol{v}||$ et $\theta^r = \mathrm{Arg}(\boldsymbol{v})$
3. Commande en orientation : déterminer ω tel que $\theta \rightarrow \theta^r$



Commande

- Commande en position : $\mathbf{v} = k_2 \begin{bmatrix} x^r - x \\ y^r - y \end{bmatrix}$ $k_2>0$

- Détermination de V et de la direction de référence θ^r :

$$\mathbf{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix} = V \begin{bmatrix} \cos\theta^r \\ \sin\theta^r \end{bmatrix} = k_2 \begin{bmatrix} x^r - x \\ y^r - y \end{bmatrix}$$

$$\Rightarrow V = ||\mathbf{v}|| = k_2 \sqrt{(x^r - x)^2 + (y^r - y)^2}$$

$$\Rightarrow \theta^r = \arg \mathbf{v} = \arctan\left(\frac{v_y}{v_x}\right) = \arctan\left(\frac{y^r - y}{x^r - x}\right)$$

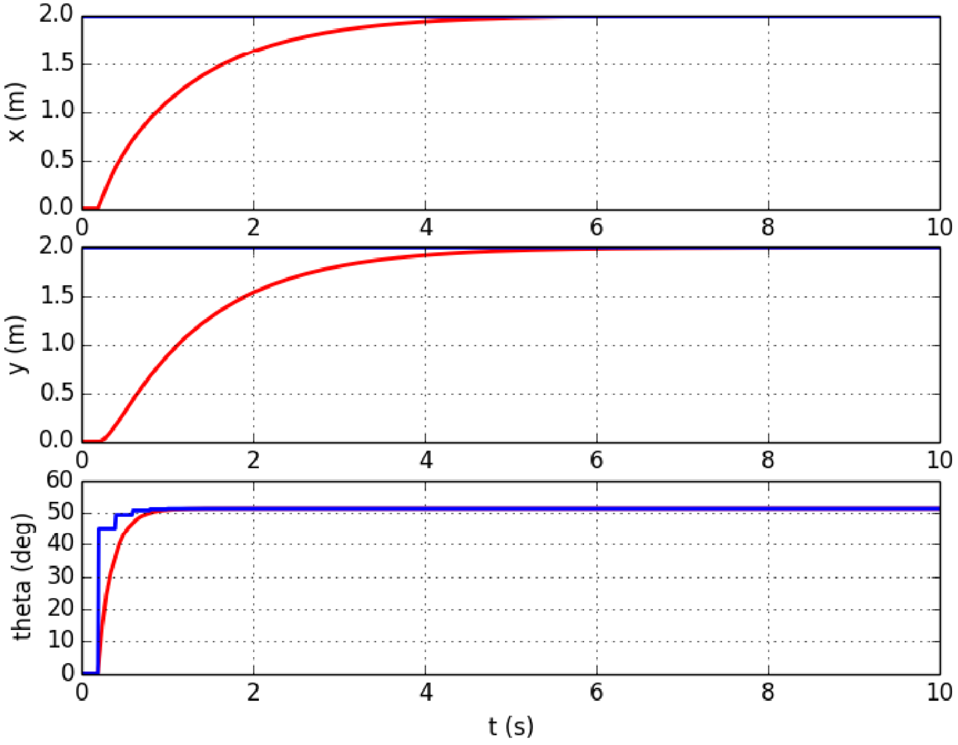
(attention au domaine de validité de cette formule)

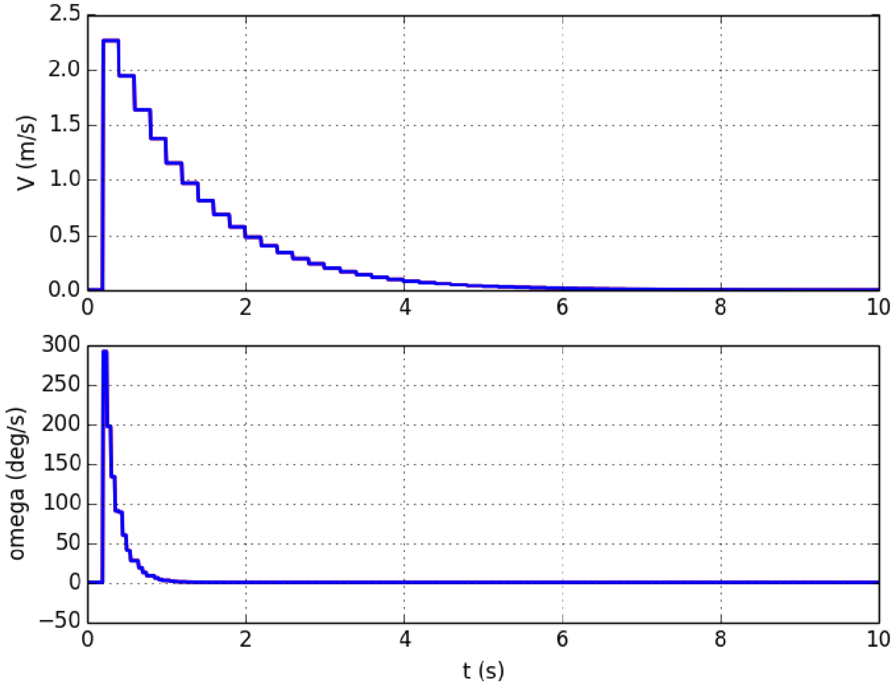
- Commande en orientation : $\omega = k_1(\theta^r - \theta)$ $k_1>0$

- Gains k_1 et k_2 à régler de manière à ce que la boucle de commande en orientation converge (beaucoup) plus rapidement que la boucle de commande en position

Commande

- Exemple : (x_0, y_0, θ_0)= (0,0,0) (x^r, y^r) = (2, 2)





Commande

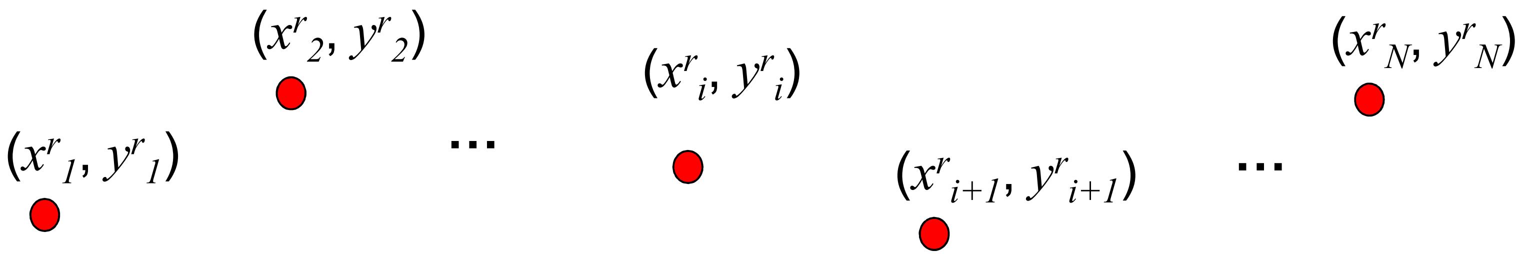
- On a vu comment calculer les commandes à appliquer au robot pour qu'il se déplace jusqu'à un point donné

- On souhaiterait maintenant :
  - Faire suivre au robot une séquence de points donnés

  - Permettre au robot d'éviter des obstacles non connus initialement et détectés en cours de route

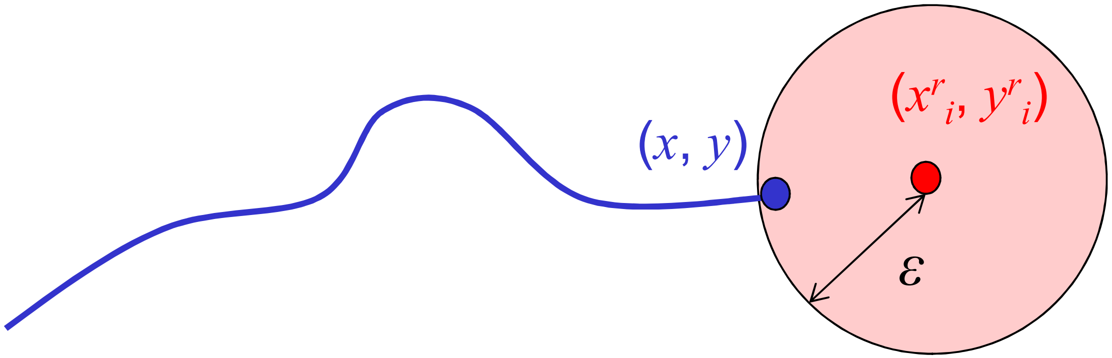
Navigation par points de passage

- Succession de N points de passage



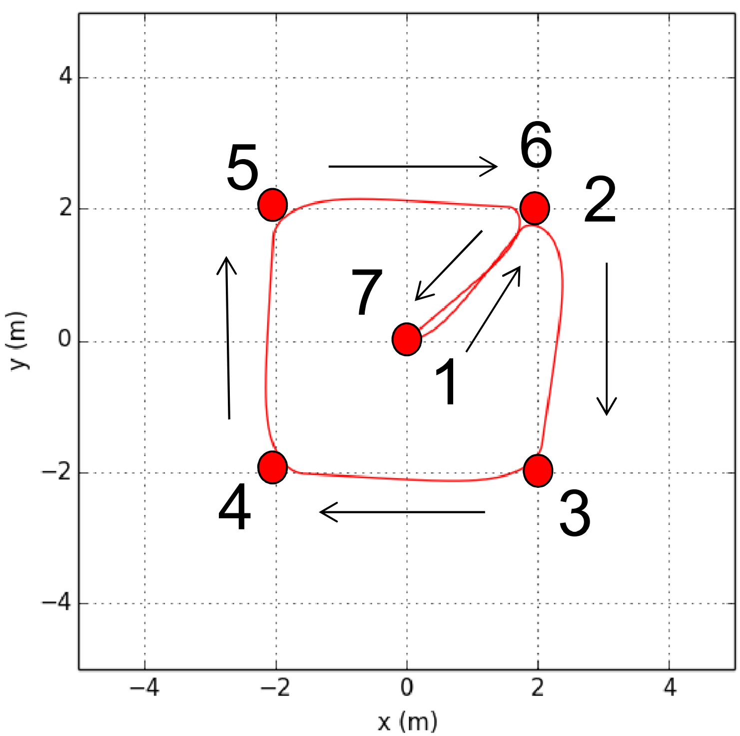
- Changement de référence :

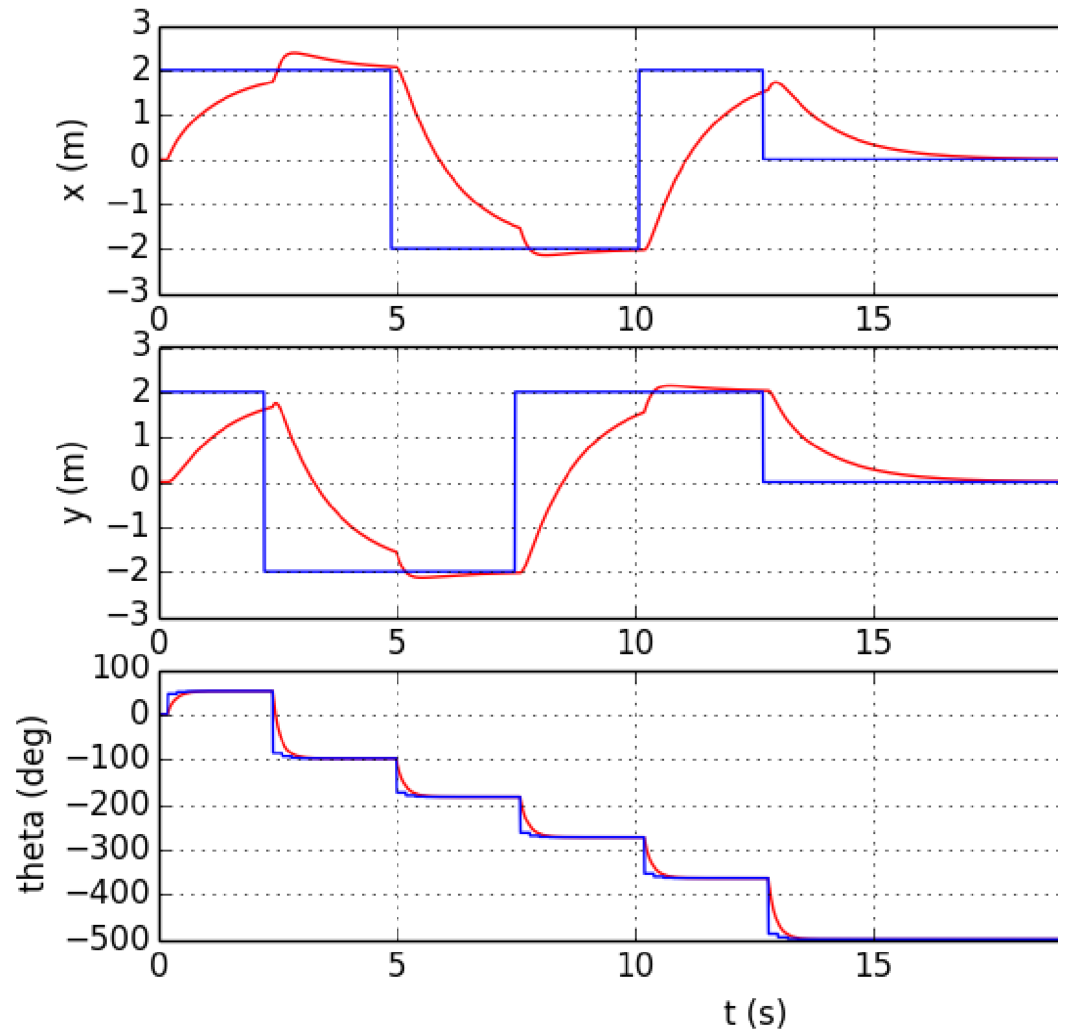
$$(x^r, y^r) = (x^r_{i+1}, y^r_{i+1}) \quad \text{si} \quad \left\| \begin{bmatrix} x - x^r_i \\ y - y^r_i \end{bmatrix} \right\| \leq \epsilon$$



Navigation par points de passage

Application de la loi de commande vue précédemment





Evitement d'obstacles

- Obstacles connus avant le départ du robot
  - intégrés à la carte
  - pris en compte dans la recherche d'une trajectoire initiale

- Obstacles découverts en cours de mission
  - détecter, mettre à jour la carte
  - planifier une nouvelle trajectoire

ou
  - conserver la trajectoire initiale en évitant l'obstacle

Navigation par champs de potentiels

- Calculer une force virtuelle $F(x, y) \in \mathbb{R}^2$ dont la direction et l'amplitude dépendent de la position (x,y) du robot

- Commande en position proportionnelle à cette force

$$\mathbf{v} = k.F(x, y) \quad \begin{cases} V = ||\boldsymbol{v}|| \\ \theta^r = \mathrm{Arg}(\boldsymbol{v}) \end{cases}$$

- Force calculée à partir d'un potentiel $U(x,y)$

$$F(x, y) = -\nabla U(x, y) = -\begin{bmatrix} \partial U / \partial x \\ \partial U / \partial y \end{bmatrix}$$

- Analogie électrostatique, gravitationnelle, etc …

Navigation par champs de potentiels

- Force calculée comme une combinaison de plusieurs composantes

$$F(x,y) = k_a.F_a(x,y,x^r,y^r) + \sum_{i=1}^{No} k_{ri}.F_{ri}(x,y,x_i^o,y_i^o)$$

force d'attraction vers l'objectif

force de répulsion liée à l'obstacle i

$$F_a(x,y,x^r,y^r) = -\nabla U_a(x,y,x^r,y^r)$$

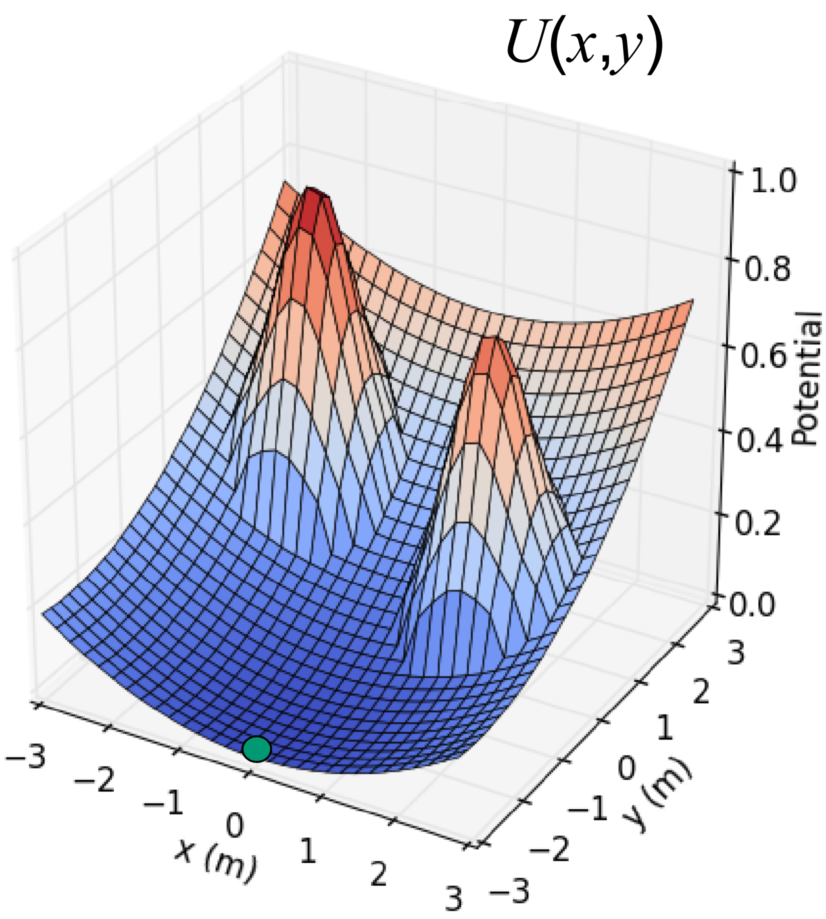
$$F_{ri}(x,y,x^r,y^r) = -\nabla U_{ri}(x,y,x_i^o,y_i^o)$$

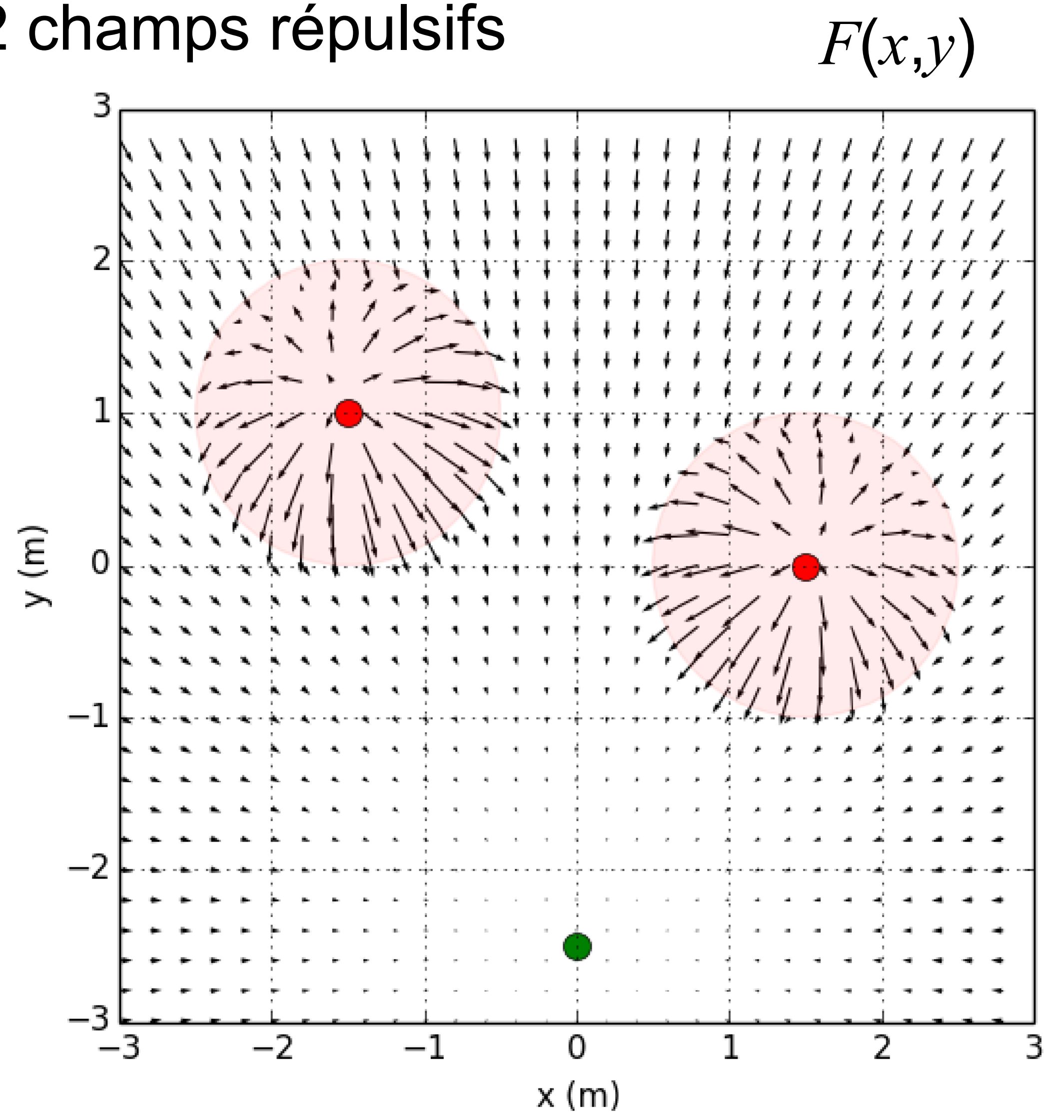
objectif : (x^r, y^r)

obstacle i : $(x^o{}_i, y^o{}_i)$

Navigation par champs de potentiels

- Ex : 1 champ attracteur, 2 champs répulsifs





Navigation par champs de potentiels

- Exemple de champ attracteur

$$U_a(x, y, x^r, y^r) = \frac{1}{2}\left((x - x^r)^2 + (y - y^r)^2\right)$$

-> Exercice : calculer l'expression de $F_a(x,y,x^r,y^r)$

- Exemple de champ répulsif (*)

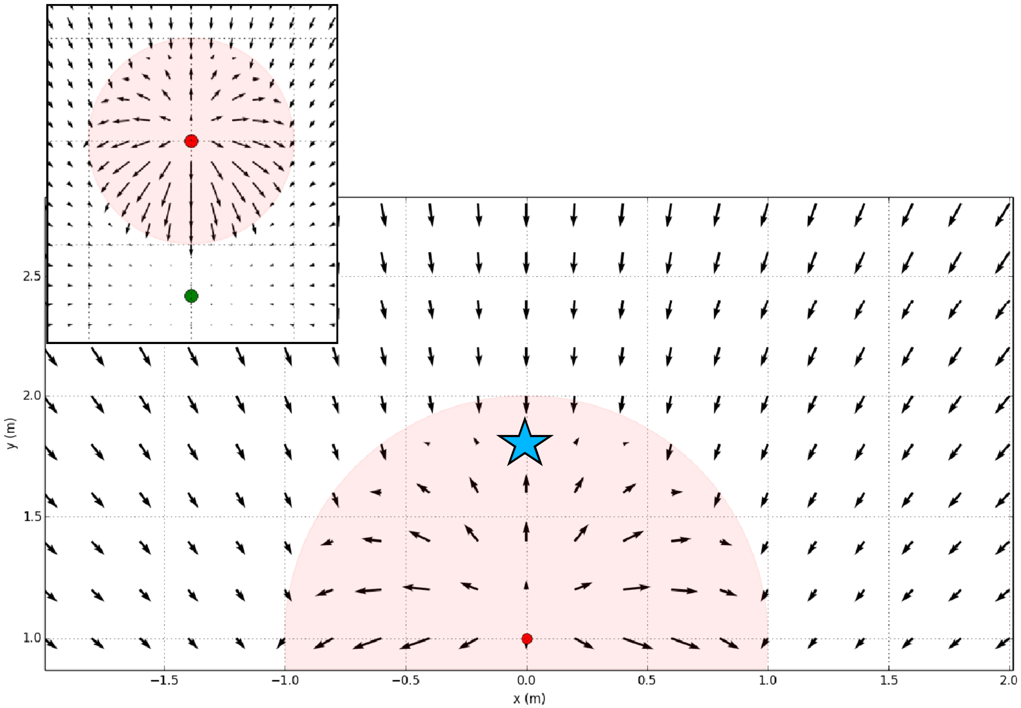
$$U_r(x, y, x^o, y^o) = \begin{cases} \frac{1}{2}\left(\frac{1}{\sqrt{(x - x_o)^2 + (y - y_o)^2}} - \frac{1}{d_o}\right)^2 & \text{si } d \leq d_o \\ 0 & \text{sinon} \end{cases}$$

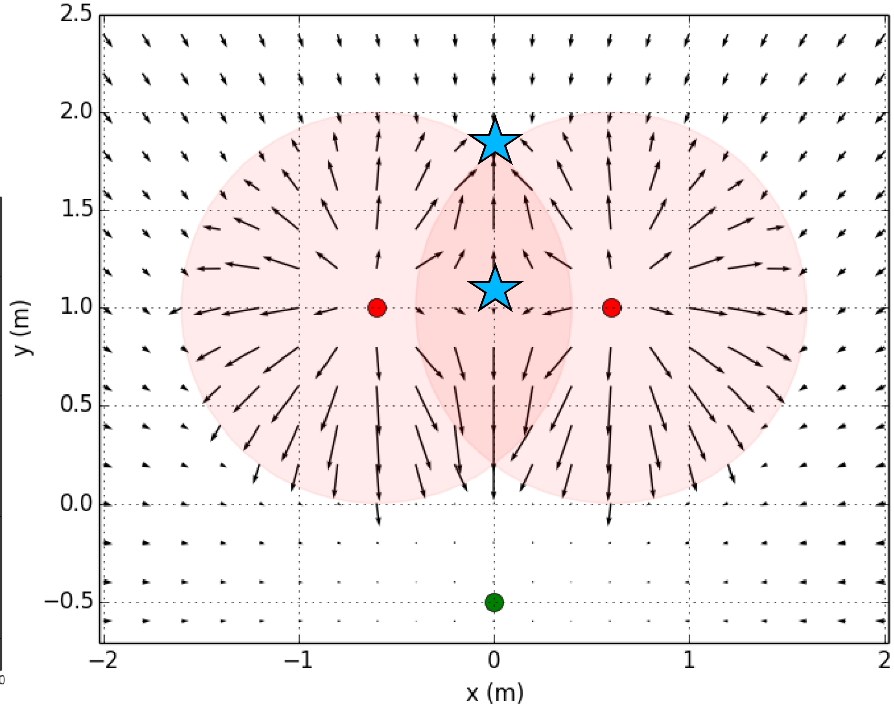
distance d'influence de l'obstacle

(*) O. Khatib, « Real-time obstacle avoidance for manipulators and mobiel robots », Proceedings of the IEEE International Conference on Robotics and Automation, pp. 500-505, 1985

Navigation par champs de potentiels

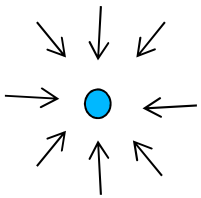
- Inconvénients de ce type d'approches
  - Existence possible de minima locaux où le robot peut rester bloqué
  - Difficultés dans les passages étroits entre obstacles rapprochés (oscillations, impossibilités de passer)

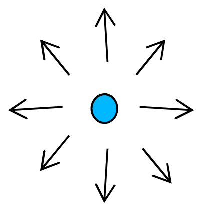


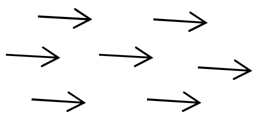


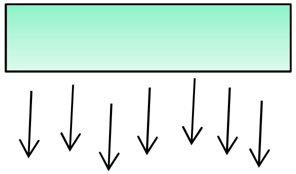
Navigation par champs de potentiels

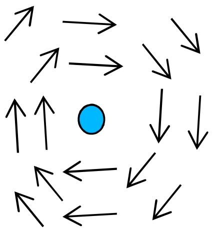
- Combiner différents types de potentiels pour obtenir le comportement souhaité

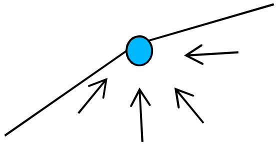

attraction

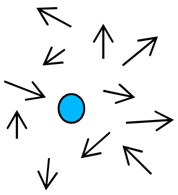

répulsion


directif


perpendiculaire


tangentiel


attraction
sélective


aléatoire

Introduit du « bruit » dans le déplacement mais peut aider à sortir de situations bloquantes

Avec pour chacun : zones d'influences bornées ou infinies, actions uniformes ou sélectives

Conclusion

- On a vu comment modéliser la dynamique d'un robot
  - Utilité : prédire/simuler, calculer la commande

- On a vu comment commander un robot mobile pour

  … rejoindre un point donné
    - Commande hiérarchique (position/orientation)

  … suivre une séquence de points donnés
    - Navigation par points de passage

  … rejoindre un point en évitant un obstacle
    - Commande par champs de potentiels

- **Mais** on a supposé qu'on sait parfaitement localiser le robot et que l'on a accès à toutes les informations souhaitées