



[Contrôle du robot JOG]

Fabrice LE BARS

Introduction

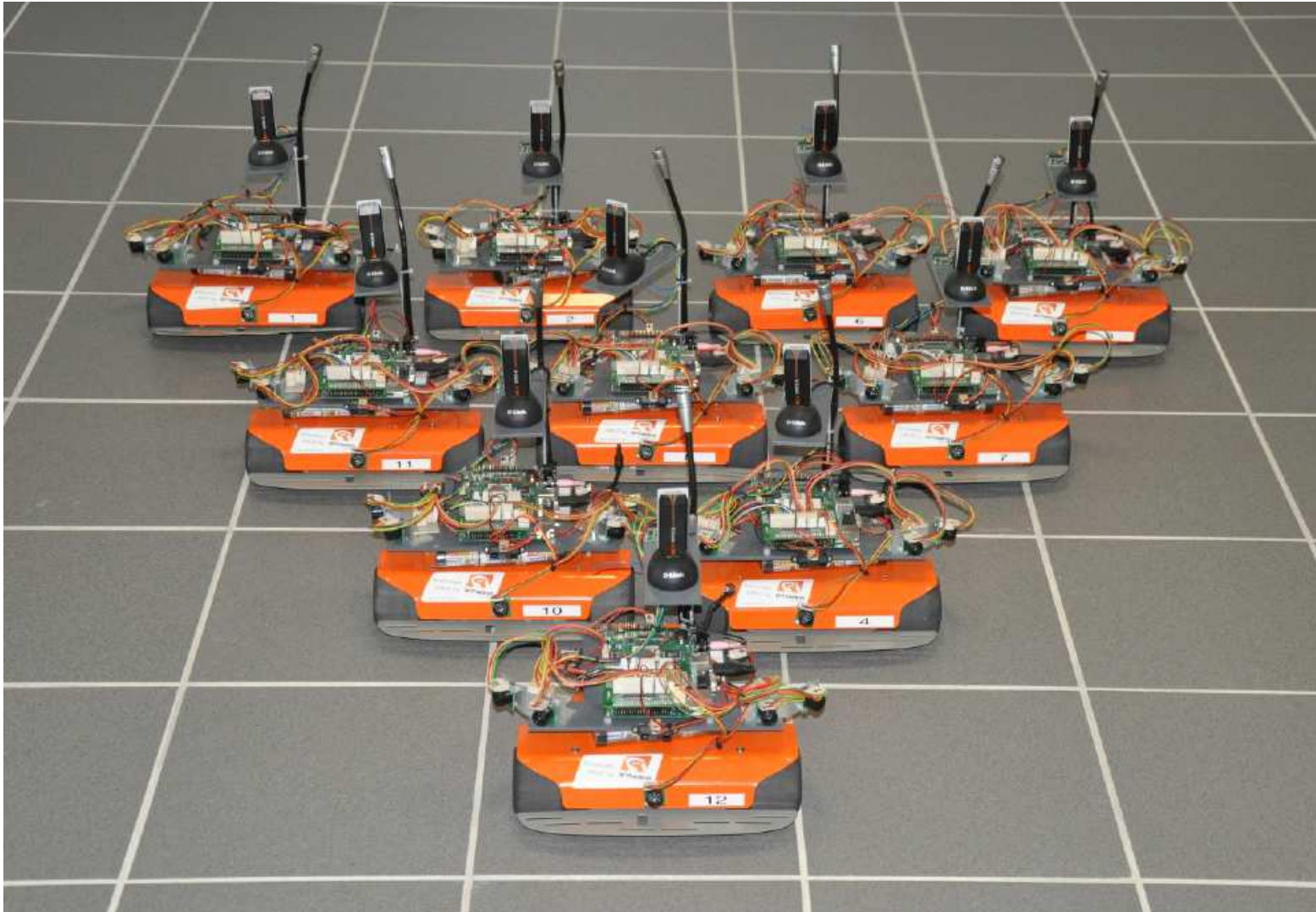


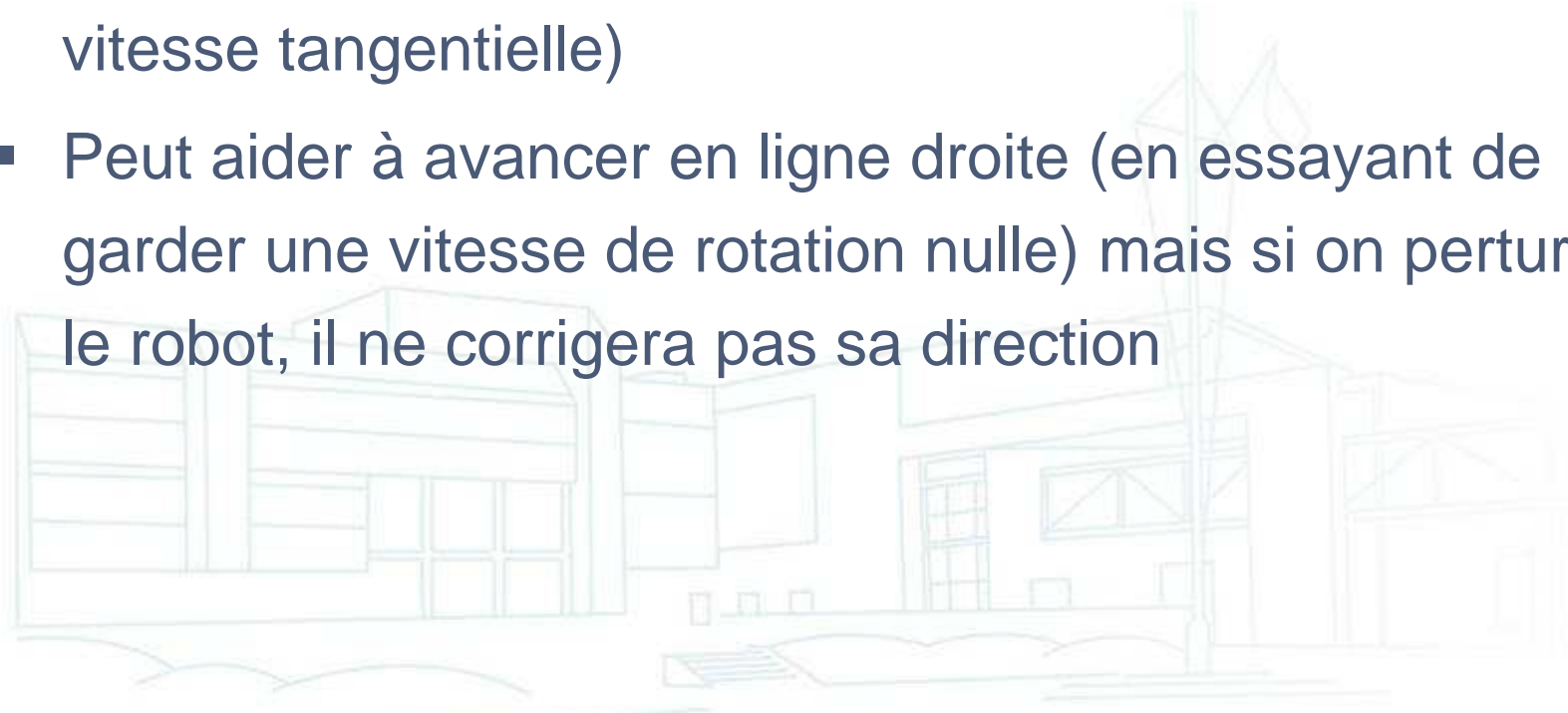
Schéma du système



- Capteurs : boussole et odomètres
- Commande : vitesse de chaque roue
(approximativement, $\bar{\omega}_1 = \alpha_1 u_1$ et $\bar{\omega}_2 = \alpha_2 u_2$, avec α_1 et α_2 à déterminer empiriquement)

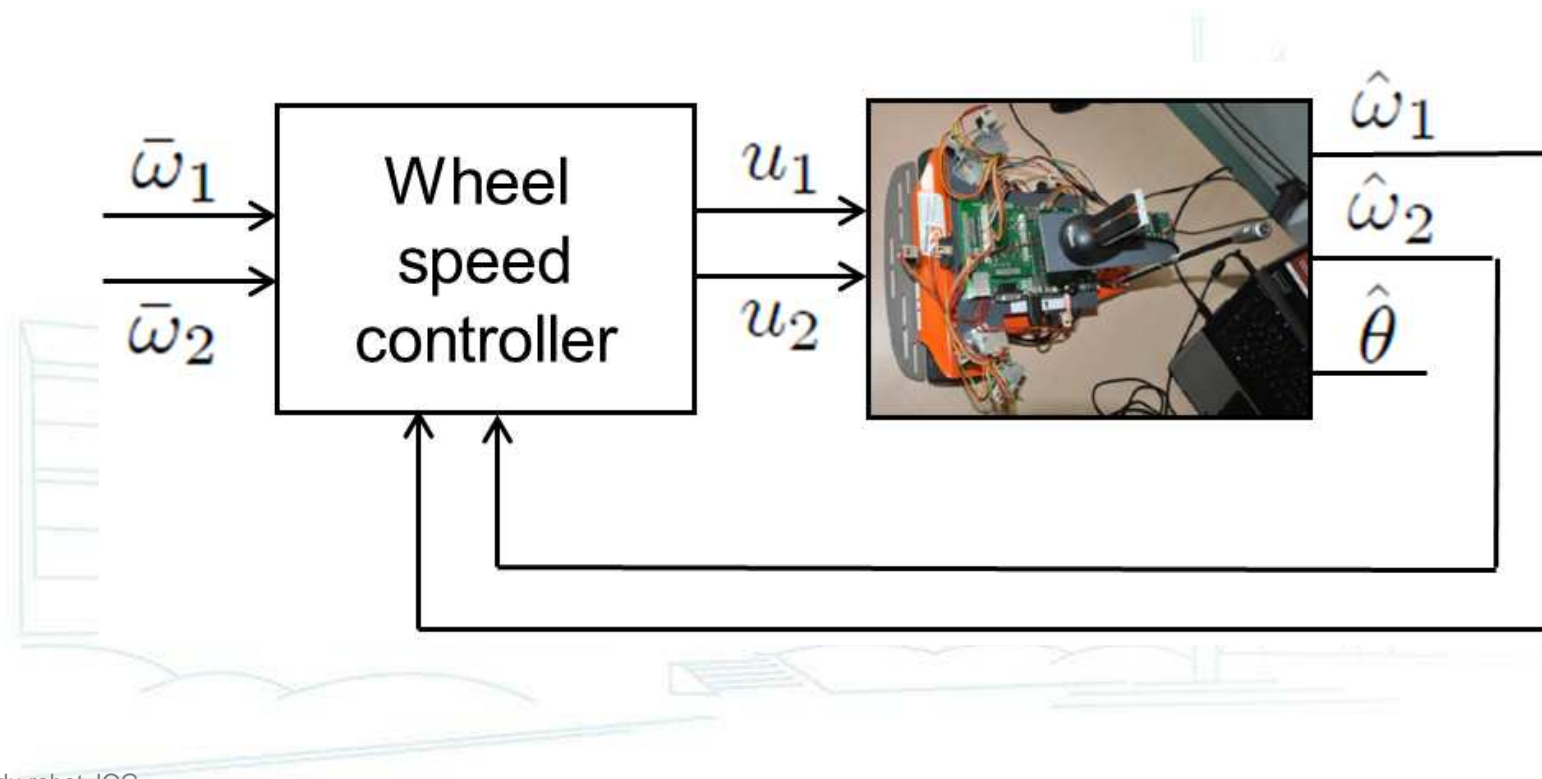
Avantages/inconvénients odomètres

- Suppose qu'il y a non-glissement
- Ne permet pas directement de s'orienter à un cap voulu absolu (on mesure plutôt une vitesse de rotation ou une vitesse tangentielle)
- Peut aider à avancer en ligne droite (en essayant de garder une vitesse de rotation nulle) mais si on perturbe le robot, il ne corrigera pas sa direction



Régulation des vitesses de rotation des moteurs grâce aux odomètres

- Une des 1^{ère} chose qui peut être faite est d'utiliser les odomètres pour pouvoir contrôler le plus exactement possible la vitesse de chaque roue...



Régulation des vitesses de rotation des moteurs grâce aux odomètres

Boucle faisant :

$$u_1 = K_1(\bar{\omega}_1 - \hat{\omega}_1)$$

$$u_2 = K_2(\bar{\omega}_2 - \hat{\omega}_2)$$

➔ Commande grand gain (mais risque d'être saccadée)

➔ Pour adoucir :

$$u_1 = \frac{\bar{\omega}_1}{\alpha_1} + K_1(\bar{\omega}_1 - \hat{\omega}_1)$$

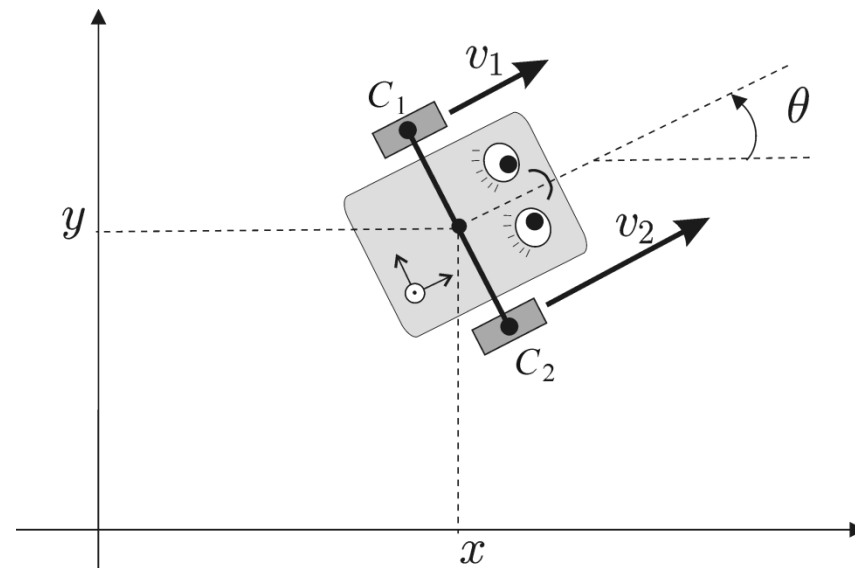
$$u_2 = \frac{\bar{\omega}_2}{\alpha_2} + K_2(\bar{\omega}_2 - \hat{\omega}_2)$$

➔ Ceci n'est pas vraiment une régulation du char, mais plutôt de ses moteurs

Modèle d'état du robot et relations géométriques

JOG : modèle de type char

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



r Rayon des roues
 l Distance entre roues

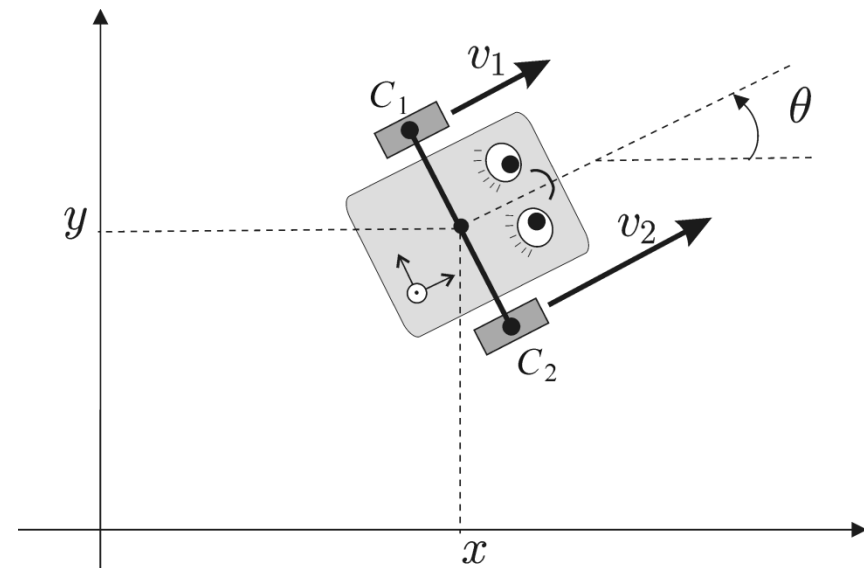
Modèle d'état du robot et relations géométriques

$$v = \frac{v_1 + v_2}{2}$$

$$\omega = \frac{v_2 - v_1}{l}$$

$$v_1 = r\bar{\omega}_1$$

$$v_2 = r\bar{\omega}_2$$



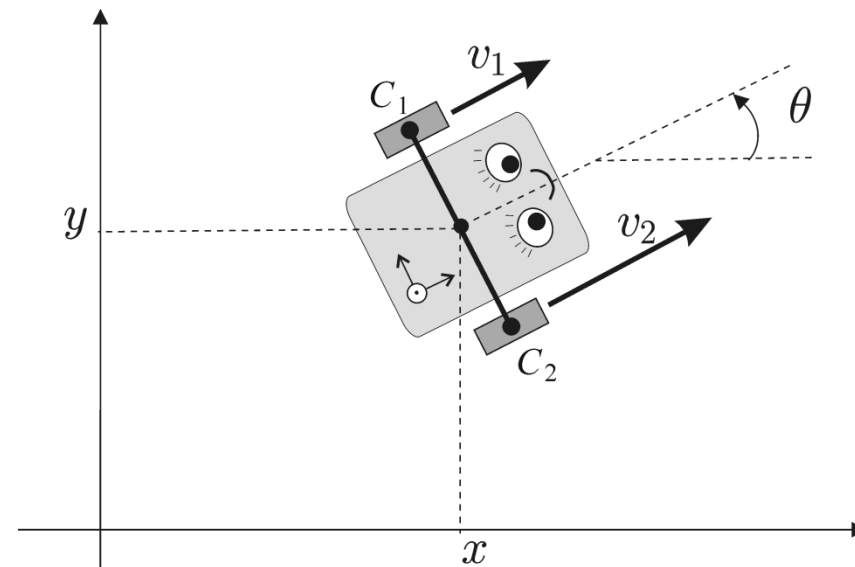
$$\begin{cases} \dot{x} = v \cos \theta \\ \dot{y} = v \sin \theta \\ \dot{\theta} = \omega \end{cases}$$

r Rayon des roues

l Distance entre roues

Modèle d'état du robot et relations géométriques

$$\begin{cases} \dot{x} = \frac{r\bar{\omega}_1 + r\bar{\omega}_2}{2} \cos \theta \\ \dot{y} = \frac{r\bar{\omega}_1 + r\bar{\omega}_2}{2} \sin \theta \\ \dot{\theta} = \frac{r\bar{\omega}_2 - r\bar{\omega}_1}{l} \end{cases}$$



$$v = \frac{v_1 + v_2}{2}$$

$$\omega = \frac{v_2 - v_1}{l}$$

$$v_1 = r\bar{\omega}_1$$

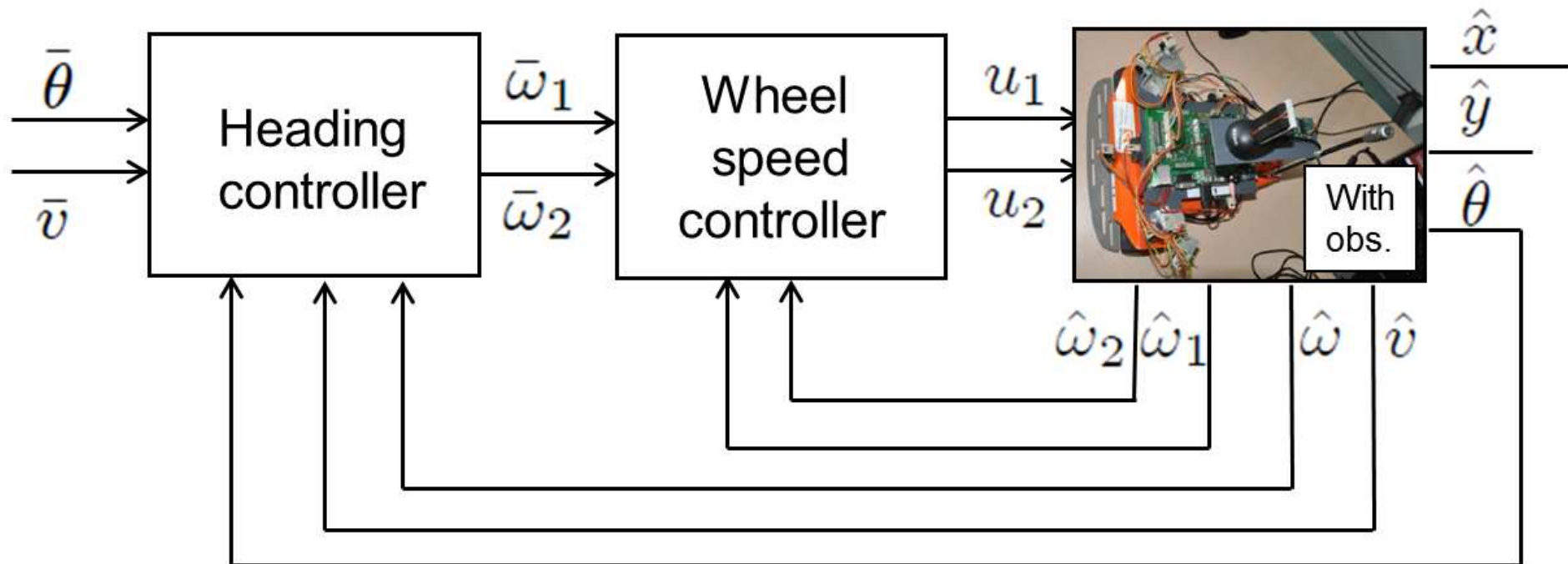
$$v_2 = r\bar{\omega}_2$$

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

r Rayon des roues

l Distance entre roues

Schéma du système pour la régulation en cap et vitesse



Régulation à une orientation et vitesse voulues grâce à la boussole

- La boussole nous donne l'angle au Nord en degrés $\hat{\theta}$
- Régulation à un cap voulu $\bar{\theta}$:
 - Commande bang-bang : on fait tourner le robot à la vitesse de rotation maximale lorsqu'il est tourné dans le mauvais sens par rapport au cap voulu

- Proportionnelle à l'erreur autrement:

$$\bar{\omega} = K_p (\bar{\theta} - \hat{\theta}) \quad \text{avec}$$

$$\bar{\omega}_1 = \frac{\bar{v} - \frac{\bar{\omega} l}{2}}{r}$$

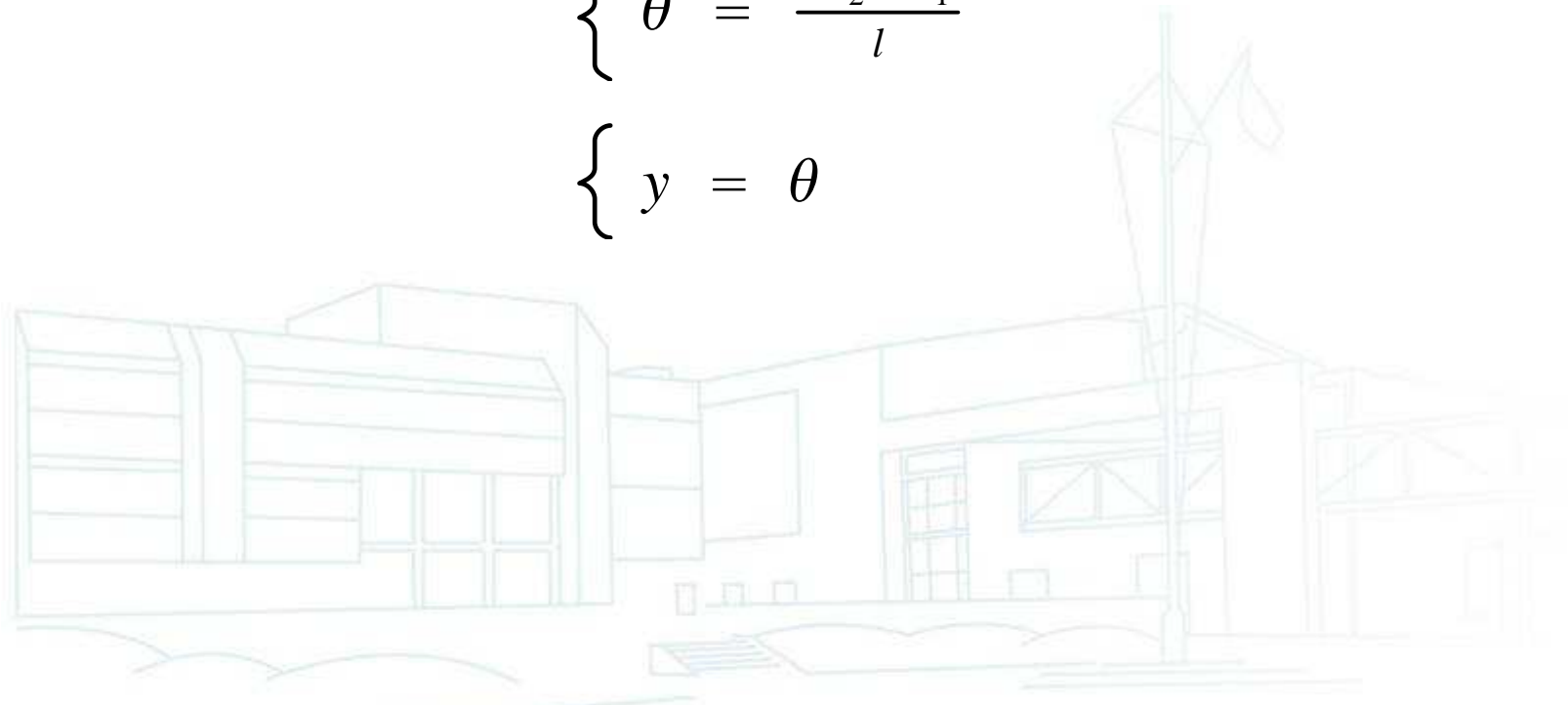
$$\bar{\omega}_2 = \frac{\bar{v} + \frac{\bar{\omega} l}{2}}{r}$$

- Attention aux problèmes de modulo 2π : utiliser des sin et cos par exemple, voir aussi http://www.ensta-bretagne.fr/lebars/Share/fmod_360.zip

Régulation à une orientation et vitesse voulues

- Eventuellement : amélioration de l'estimation de theta avec un filtre de Kalman dédié

$$\begin{cases} \dot{\theta} = \frac{r\bar{\omega}_2 - r\bar{\omega}_1}{l} \\ y = \theta \end{cases}$$



Régulation à une orientation et vitesse voulues

- Eventuellement : amélioration de la régulation en prenant en compte la dérivée (mesurée via odomètres)

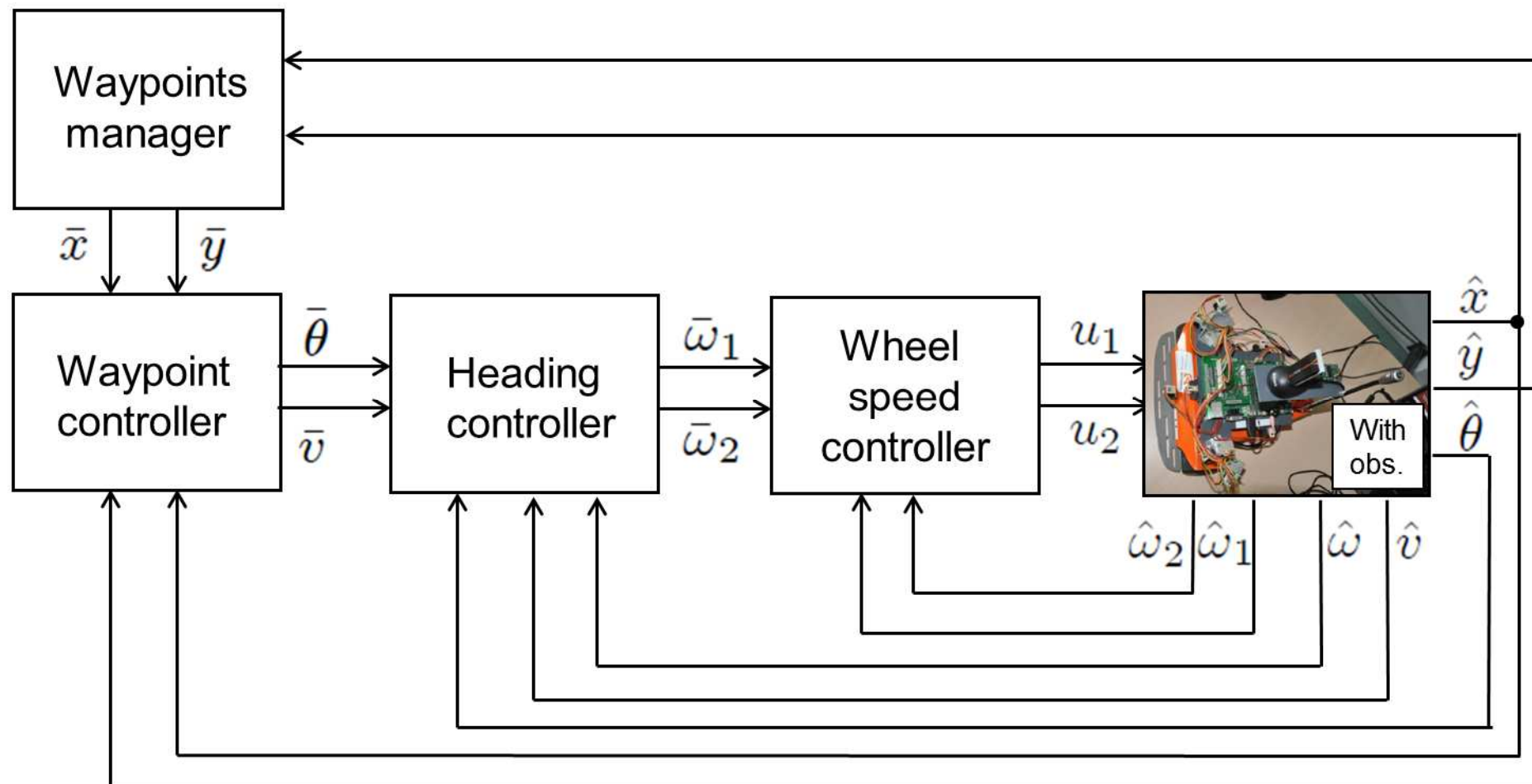
$$\bar{\omega} = K_p (\bar{\theta} - \hat{\theta}) - K_d \hat{\omega}$$

- Avec

$$\hat{\omega} = \frac{(\hat{\omega}_2 - \hat{\omega}_1)r}{l}$$

- Remarque : on peut aussi calculer $\hat{v} = \frac{(\hat{\omega}_1 + \hat{\omega}_2)r}{2}$ avec les odomètres

Schéma du système pour le suivi de waypoints



Observateur : utilisation e.g. du filtre de Kalman pour estimer x et y

