

# TPs de modélisation et commande en robotique à l'aide de Matlab/Simulink

PROGRAMME  
UNIT-GDR ROBOTIQUE



GDR  
Robotique

Lounis ADOUANE  
Novembre 2010  
[Lounis.Adouane@lasmea.univ-bpclermont.fr](mailto:Lounis.Adouane@lasmea.univ-bpclermont.fr)

Fondation  
**unit**  
Université Numérique  
Ingénierie et Technologie

**GDR**  
ROBOTIQUE

 **POLYTECH**  
CLERMONT-FERRAND

# Mise en place

**Séances :** TPs (3x4h) de Robotique sous forme d'un mini-projet

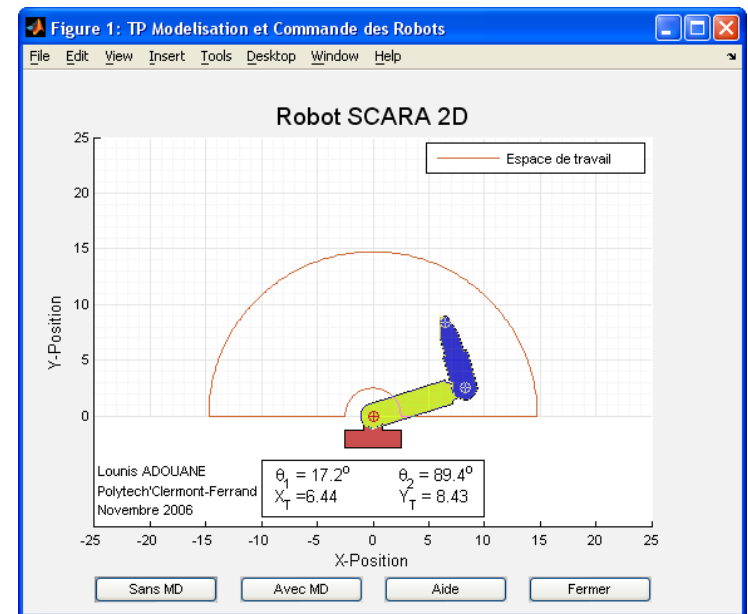
**Formation :** Département Génie Electrique - Polytech'Clermont-Ferrand

**Public concerné :** 3<sup>ème</sup> année (Bac + 5)

**Supports :** Programmes Matlab/Simulink + énoncés de TP

## Concepts illustrés :

- Modélisation géométrique
- Modélisation cinématique
- Modélisation dynamique
- Génération de trajectoires
- Commande par : PID, modèle inverse
- Evitement d'obstacles

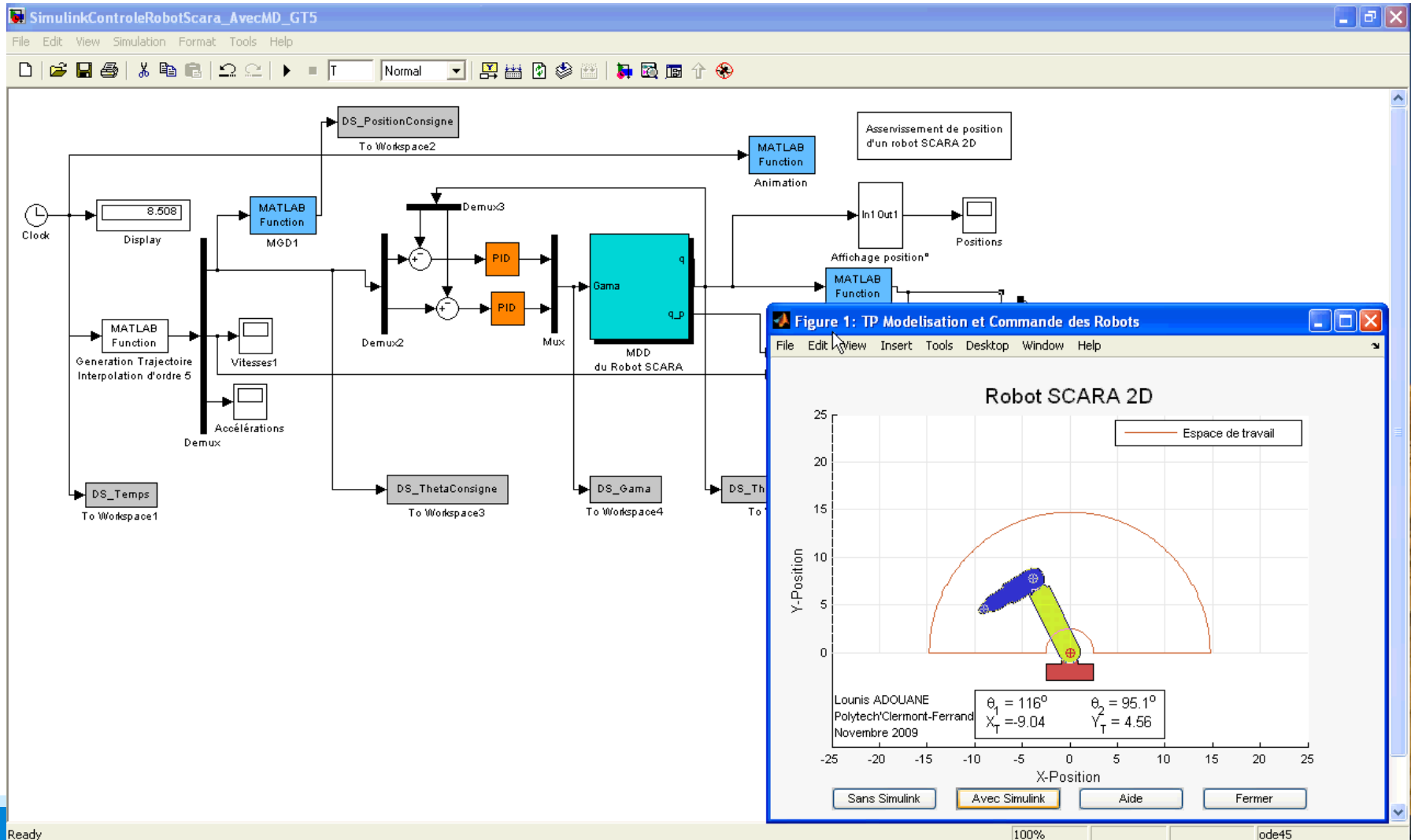


Robot SCARA

# Compétences développées

- ✓ Maîtrise et implémentation des modèles géométriques, cinématiques et dynamiques de robots,
- ✓ Savoir comment générer des consignes (articulaires, opérationnelles) pour la commande de robots,
- ✓ Commande de robots par PID, modèle inverse, etc.,
- ✓ Ouverture vers la recherche (e.g., stratégie d'évitement d'obstacles).

# Le logiciel en question



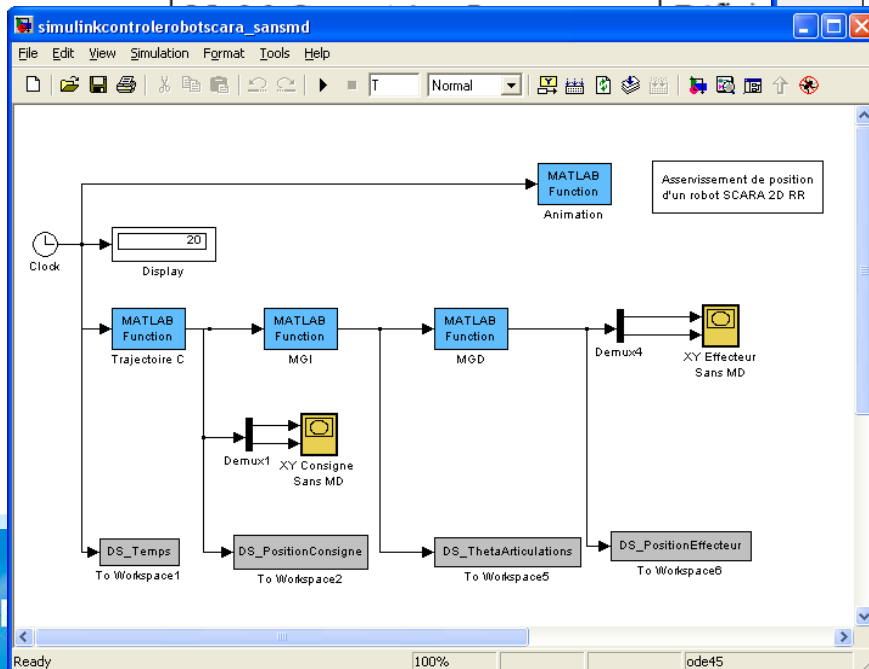
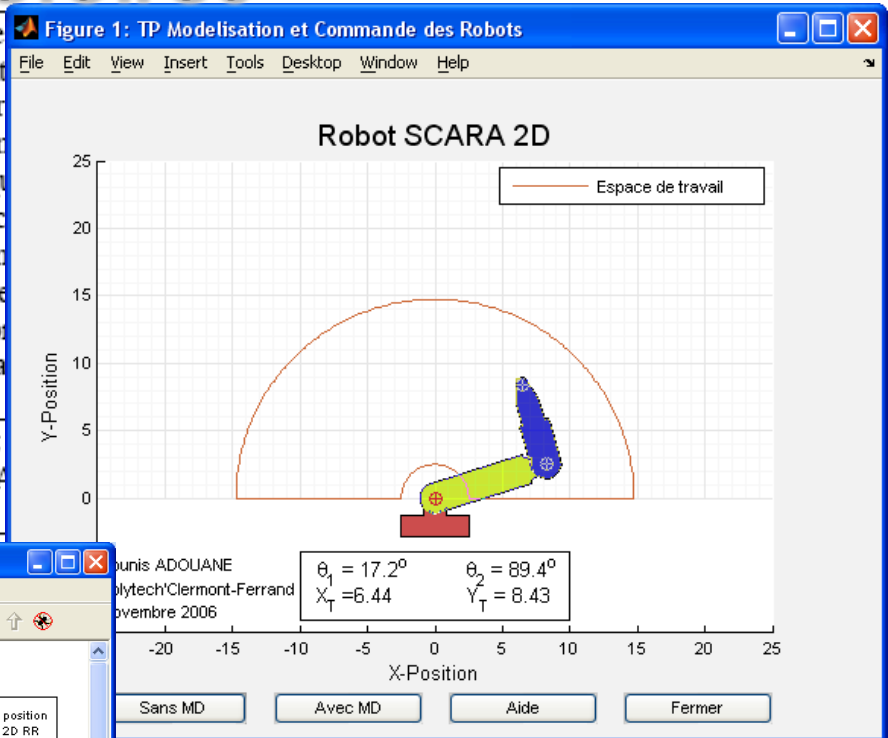
# Plan du TP

---

- Partie I : modélisation et génération de trajectoires
- Partie II : commande des robots et évitement d'obstacles

# Partie I : Modélisation et génération de trajectoires

SimulationRobotSCARA.m	C'est le une int simuler Simul ainsi qu interfac événem vous pe des co termina
ModeleGeometriqueDirect.m	Définit SCARA



événements (clic de souris, bouton enfoncé, etc.)  
 tre GUI (Graphical User Interface)

mettre à jour la représentation graphique du robot  
 que l'affichage des informations articulaires et la  
 ecteur (Figure. 2)

lk (Figure. 3) qui permet en l'interfaçant avec  
 décrits ci-dessus de commander le mouvement  
 fin de suivre une trajectoire consigne par

# Partie I : Modélisation et génération de trajectoires

## I.1. Modèle Géométrique Direct

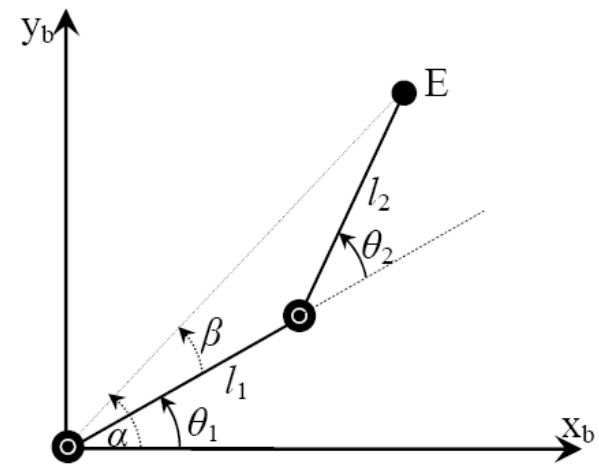
$$T0Tn = MGD\_General(alpha, d, theta, r)$$

Avec (alpha, d, theta, r) vecteurs des paramètres géométriques du robot (chaque vecteur correspond à une colonne du tableau de Denavit-Hertenberg modifié).

## I.2. Modèle Géométrique Inverse

$$[Q1, Q2, err] = ModeleGeometriqueInverse\_Bis(X)$$

Avec : **Q1** la solution avec  $\theta_2 > 0$  (coude bas), **Q2** la deuxième solution (coude haut), **err** indique la faisabilité des deux solutions.

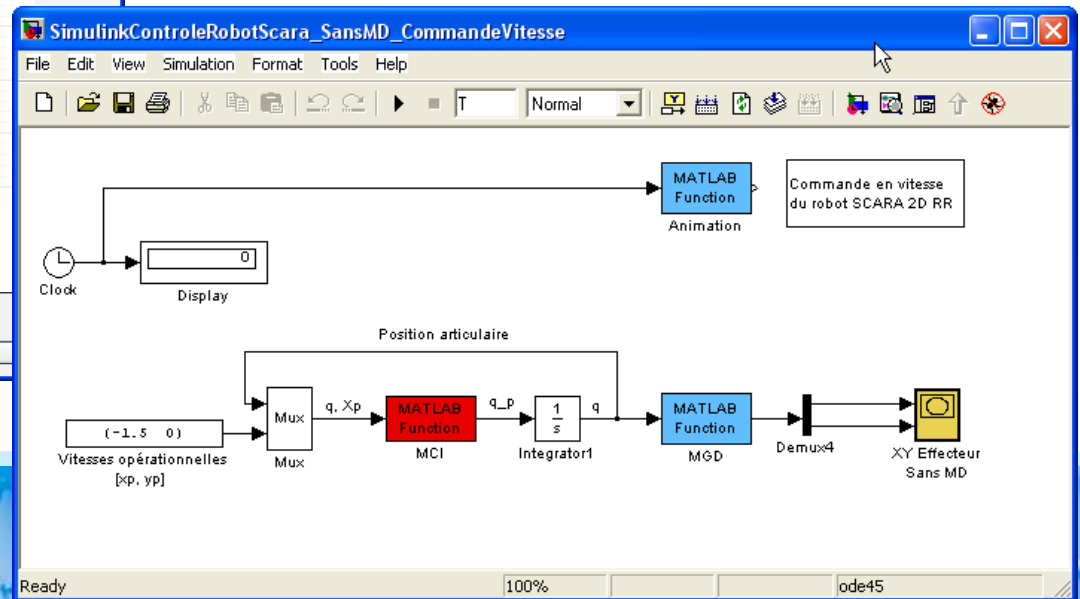
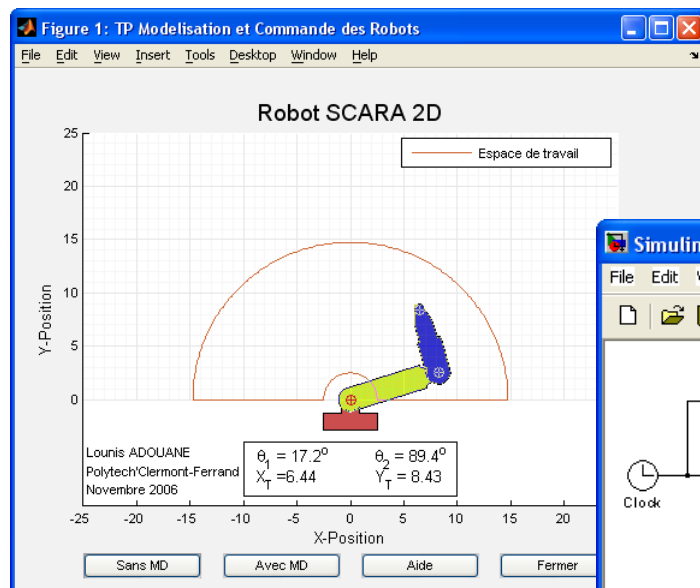




# Évaluation et génération de trajectoires

### I.3. Modèle Cinématique Direct

- Implémentation du modèle cinématique direct et inverse et test avec l'interface Matlab
- Gestion des singularités.

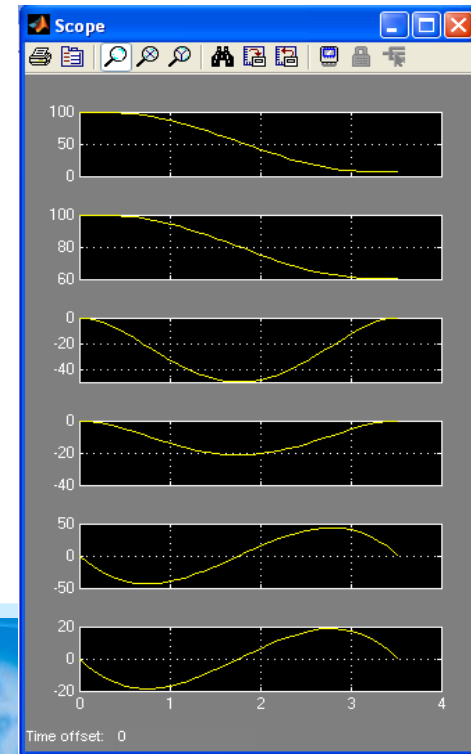
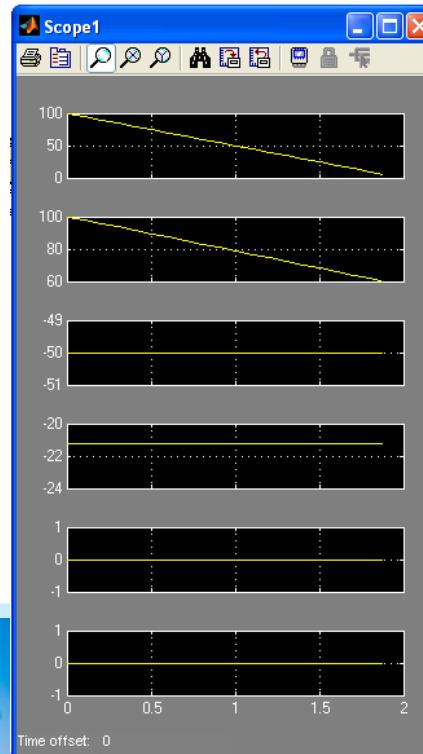




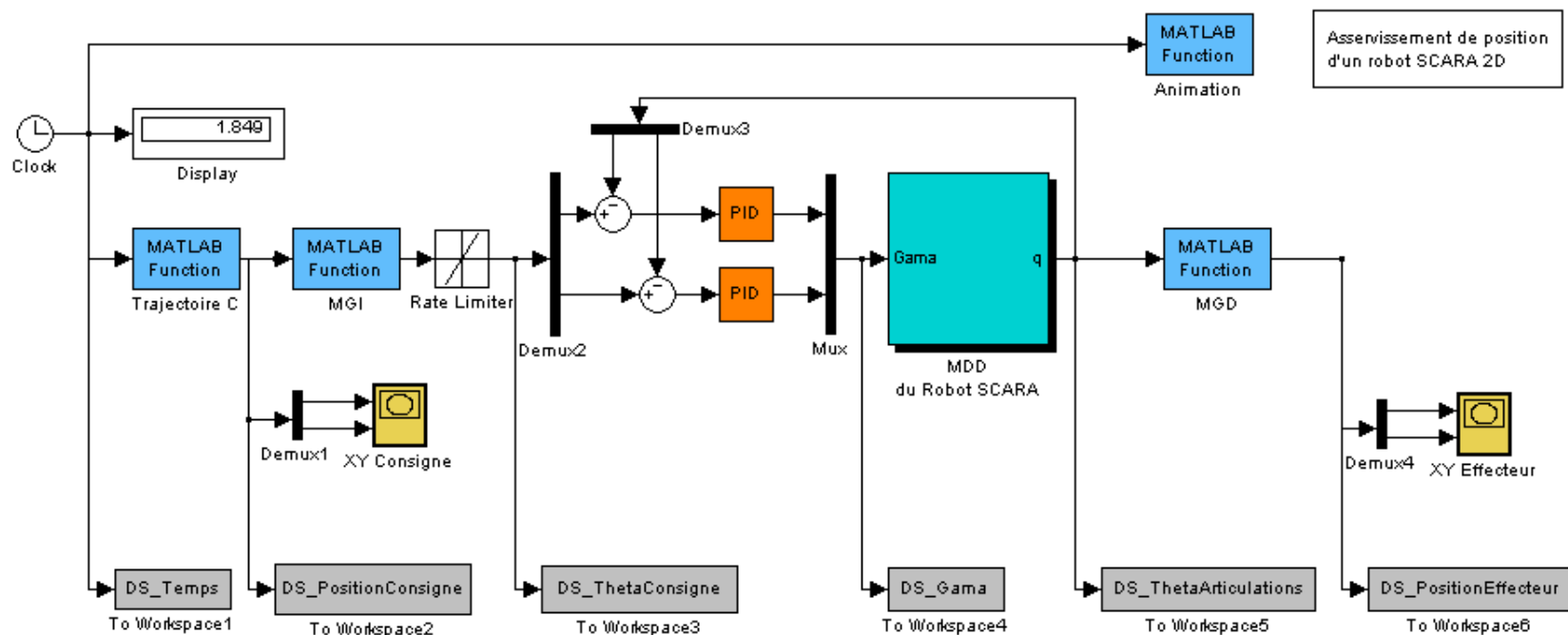
# Partie I : Modélisation et génération de trajectoires

## I.4. Génération de trajectoires

- Implémentation d'un interpolateur d'ordre 1 et 5 dans le domaine articulaire,
- Synchronisation des mouvements articulaires,
- Génération de mouvements géométriques élémentaires (cercle, carré, etc.) de l'effecteur.



## PROGRAMME UNIT-GDR ROBOTIQUE



# Partie II : Commande des robots et Evitement d'obstacles

## II.1. Commande par PID

- Synthèse d'un contrôleur PID,
- Commande dans l'espace articulaire et opérationnel.

## II.2. Evitement d'obstacles

