Université Mohammed V de Rabat
Atelier de robotique

Année univ. 2023-2024
Pr. Younès RAOUI (FSR)
Le 20 mai 2024

Lab d'Initiation et prise en main du logiciel CoppeliaSim

Durant cette session de partage sur le logiciel Coppeliasim, vous allez apprendre à utiliser ce logiciel et ses outils de simulation avec Python.

Exercice 1:

1. Importer un objet Cuboid:
- Ouvrez CoppeliaSim et créez une nouvelle scène.
- Importez un objet tel qu'une forme primitive (par exemple un cube, une sphère) ou un modèle existant dans la scène.
2. Exercice de modification de la taille :
- Sélectionnez l'objet importé.
- Localisez et modifiez les propriétés d'échelle de l'objet pour le redimensionner. Vous pouvez le faire soit dans le panneau des propriétés de l'objet.
3. Exercice de changement de couleur :
- Sélectionnez l'objet.
- Changez la couleur de l'objet en modifiant ses propriétés matérielles. Vous pouvez ajuster les couleurs diffuses, spéculaires ou d'émission pour obtenir l'effet désiré.
4. Deplacer l'objet en faisant des translations et des rotations

Exercice 2:

Ci-dessous, les fonctions que nous allons utiliser :

Function name	Explanation
sysCall_init()	This function initializes various global variables,
sysCall_actuation() ○	This function is used to perform the main code execution, typically in a loop.
sysCall_sensing()	This function is responsible for handling sensing-related tasks.

sim.getObject(object_name)	This function is used to retrieve the handle of an object in the simulation environment.
sim.setJointTargetVelocity(joint_handle, velocity)	This function sets the target velocity for a specific joint in the simulation.
sim.addGraphStream(graph_handle, name, unit, data_title)	This function adds a stream to a graph with the specified name, unit, and data title.
sim.getSimulationTime()	This function retrieves the current simulation time in CoppeliaSim. It is used in the `sysCall_actuation()` function to get the current simulation time for decision-making.
sim.getObjectPosition(handle, relative_to_handle):	This function returns the position of an object in the global/world reference frame or relative to another object. It is utilized in the `sysCall_actuation()` function to get the position of the robot.
sim.getObjectOrientation(handle, relative_to_handle):	This function returns the orientation (rotation) of an object in the global/world reference frame or relative to another object.

1. Copiez le programme sur navigation_autonome.py sur Coppeliasim
2. Ajoutez un robot pioneer et plusieurs objets de votre choix dans la scène.
3. Adapter la taille et la position des objets
4. Assurer la sauvegarde de la scène
5. Pour exécuter le programme, il vous suffit d'appuyer sur le bouton exécuter.
6. Changez le seuil de détection d'obstacles et ré-exécutez.
7. Modifiez les vitesses des deux roues et ré-exécutez.

Exercice 3 :

1. Construire une nouvelle scène
2. Ajouter un objet "vision sensor" et configurer-le en ajustant sa position et sa taille. Inclure également une sphère.

3. Déplacer la sphère pour la voir sur la caméra.
4. Mettre la caméra sur le robot Pioneer et le déplacer.
5. Créer un nouveau script Python, copier le programme simulation_capteur.py et exécuter le.
6. Obtenir la position et l'orientation de la caméra en fonction du temps

Exercice 4:

1. Lancer un IDE adapté à Python, comme VScode.
2. Téléchargez le programme read_poses.py et ouvrez le sur l'éditeur
3. Ouvrez sur Vrep la scène navigation.ttt
4. Observez la trajectoire du robot Pioneer et des objets sur Matplotlib