Behavior Trees Demo

Concepts

https://docs.nav2.org/concepts/index.html#behavior-trees

https://docs.nav2.org/behavior_trees/overview/nav2_specific_nodes.html

https://docs.nav2.org/behavior_trees/overview/detailed_behavior_tree_walkthrough.html

Démo avec le Turtlebot3

https://github.com/sea-bass/turtlebot3_behavior_demos

Usage sans docker

- Installation

  https://github.com/sea-bass/turtlebot3_behavior_demos?tab=readme-ov-file#local-setup

- Vérifier que Gazebo fonctionne en lançant le noeud de base :
  `ros2 launch turtlebot3_gazebo turtlebot3_world.launch.py`
- Éteindre le noeud

On lance l'environnement de simulation lié à la démo de Behavior Tree :

- `ros2 launch tb3_worlds tb3_demo_world.launch.py`
  Le robot navigue en des positions connues avec pour but de trouver un cube d'une couleur spécifiée (rouge, vert ou bleu). La détection d'objets est faite par un simple seuillage en couleurs HSV avec des valeurs calibrées.

Démos de Behavior Trees en Python

On regarde le fichier `turtlebot3_behavior_demos/docker-compose.yaml` pour déterminer les commandes Bash correspondant aux commande docker indiquées dans le dépôt.

Dans un second terminal, on lance une des démos suivantes :

- `ros2 launch tb3_autonomy tb3_demo_behavior_py.launch.py tree_type:=queue enable_vision:=true target_color:=green`
  Démo avec `py_trees`

Les fichiers source de la démo sont :

- tb3_demo_behavior_py.launch.py
- autonomy_node.py
  - navigation.py
    - test_move_base.py
  - vision.py
    - test_vision.py

Revision #6
Created 17 October 2024 10:22:47 by admin_idf
Updated 22 October 2024 14:08:24 by admin_idf