

# **Autonomous Robot Manipulation in a Simulated Kitchen Environment Using Webots and Behavior Trees**

## **1. Introduction**

Autonomous robots are increasingly deployed in structured environments such as homes, warehouses, and kitchens to assist in routine tasks. Effective deployment requires integrating capabilities like mapping, navigation, obstacle avoidance, and object manipulation. This project demonstrates an autonomous mobile manipulator operating in a simulated kitchen environment using Webots. The robot is designed to perform pick-and-place tasks using a modular and reactive behavior tree-based control architecture.

## **2. Webots Simulator Overview**

Webots is an open-source 3D robot simulator developed by Cyberbotics. It supports a wide variety of mobile and humanoid robots, offering physics-based simulation, customizable environments, and real-time visualization. Webots provides interfaces with standard robot controllers, sensors (e.g., LIDAR, cameras), and actuators, making it a powerful tool for robotics development, testing, and validation.

## **3. Project Environment and Robot Platform**

The project is set in the Webots kitchen simulation world. The robot used is the TIAGo mobile manipulator, which features a differential drive base, a 7-DOF front arm, and an RGB-D camera mounted on the head. This setup allows the robot to navigate, perceive its surroundings, and interact with kitchen objects such as jars and bottles.

## **4. Software Architecture**

The control system is implemented using the `py_trees` Python library, which enables the design of reactive, modular, and hierarchical behavior trees. This architecture promotes clean separation of tasks and allows for easy integration of sensing, decision-making, and actuation layers.

## **5. Implemented Capabilities**

### **5.1 Mapping and Localization**

An occupancy grid map is generated using LIDAR-based sensing. The robot fuses sensor data to update the environment model and its own pose within the map. This allows for continuous localization and situational awareness.

## **5.2 Navigation and Path Planning**

A custom A\* path planner is used to generate feasible trajectories from the robot's current location to its goal destination. The map is discretized into a cost space to account for obstacles and free space, and the planned path is executed by a waypoint-following behavior.

## **5.3 Obstacle Avoidance and Replanning**

A reactive obstacle avoidance system is implemented using frontal LIDAR data. If an unexpected obstacle is detected on the planned path, the robot halts and triggers a replanning behavior to find an alternative route.

## **5.4 Robot Pose Updates**

The robot's position and orientation are continuously updated by integrating odometry with LIDAR-based corrections. This ensures accurate navigation and decision-making.

## **5.5 Stuck Watchdog**

A dedicated watchdog monitors the robot's motion. If the robot remains in the same position for an extended period (indicating it is stuck), a recovery procedure is initiated, involving map updates and global replanning.

## **5.6 Pick and Place Behavior**

Pick-and-place tasks are handled through a combination of object detection using a camera capabilities.

## **6. Conclusion**

This project showcases a complete mobile manipulation pipeline within a simulated domestic setting using Webots. The integration of perception, planning, and manipulation with behavior trees results in a robust and flexible architecture capable of handling complex real-world tasks like pick-and-place. Future work may include expanding object categories, enhancing visual perception, and integrating voice commands via a natural language interface.