WS 2022-2023, SMT-ROB, DISAL-MP50 Project Summary

Automatic Design of Flexible Behavioral Arbitrators for Khepera IV Robots

Lucas Bost

Professor : Alcherio Martinoli
Assistant(s) : Cyrill Baumann

Automatic design of controllers is a dominant topic in robotics. In distal control architectures, pre-coded basic behaviors are assembled into finite state machines (hereafter "FSM"), or other switching structures. In the recent years, a new approach has appeared and relies on behavior trees (hereafter "BT"). BTs are task switching structure organized in a tree composed of nodes. A tick signal propagates through the tree and a node is executed when it receives the tick signal. A node can be a control flow node, which controls the propagation of the tick, a condition node, which corresponds to the transition condition or an action node, which corresponds to the basic behaviors. BTs have a built-in hierarchical structure, where tasks can be composed of subtasks and trees can be composed of subtrees. They also have an easily understandable graphical representation and they offer a lot of flexibility by allowing to run several states at once.

We created architectures for BTs with fixed number of levels and fixed maximum number of children allowed per node and we defined a grammar associated to these architectures. Unfortunately this grammar contains constraints and leads to very large search spaces. One of the main drawback of this approach is that the size of the set of unconstrained combinations decreases when the size of the tree increases.

The experiments have been carried out using a Khepera IV robot involved in 2 different scenario, a scenario where the robot must explores an arena while avoiding obstacles and stop on a black area ("Blackstop") and a scenario where the robot must find black floor before reaching a white area which is placed under a light source and is difficult to access ("Foraging"). We generated BT controllers using the defined grammar, simulations in Webots simulator, cost functions associated to each scenario and a mixed-discrete particle swarm optimization algorithm enhanced with an optical computing budget allocation scheme (hereafter "MDPSO-OCBA").

For each scenario, we generated controllers with 2 different tree architectures and compared them to a manual solution and a FSM-based controller.

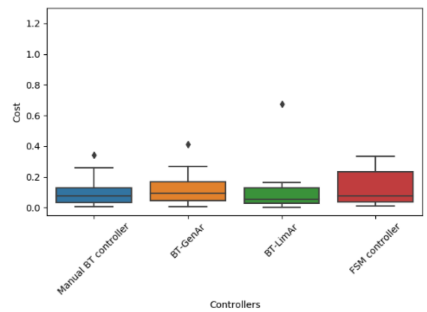

Figure 1 – Results of the evaluation of the different controllers for scenario "Blackstop"

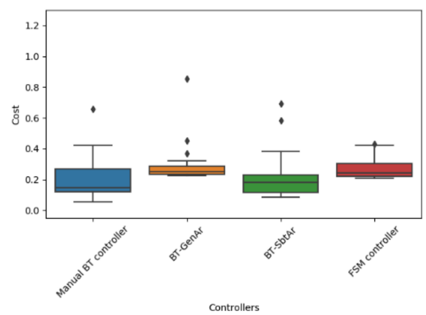

Figure 2 – Results of the evaluation of the different controllers for scenario "Foraging"

For the scenario "Blackstop", a Friedman test between the 4 controllers showed that the 4 controllers have similar performance. For the scenario "Foraging", a Friedman test demonstrated that there are significant differences between the performance of the 4 controllers. We performed a Dunn's post-hoc test which showed that the manual solution outperforms the first BT controller generated and the FSM controller and that the second BT controller generated have a performance similar to the manual solution and outperforms the FSM controller. We managed to recreate results that already existed with FSM and to show that flexibility of BT can be an advantage. However the algorithm had difficulties to find solutions and it would be interesting to further investigate the approach.