SS 2020-2021, SMT, DISAL-SP153 Project Summary

Automatic Design of Behavioral Arbitrators for Khepera IV Robots: Comparing Optimization Algorithms to Generate a Finite State Machine

Arnaud Guibbert

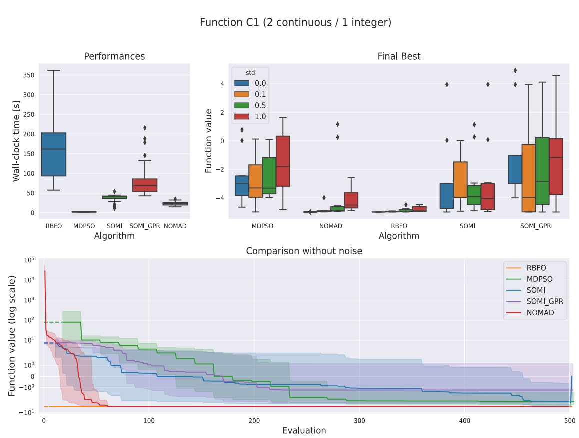
Professor : Alcherio Martinoli
Assistant(s) : Cyrill Baumann

Most of the probabilistic finite state machine (PFSM) are mainly designed by hand. However, in recent years, many algorithms have been developed to automatically achieve this task. The question then arises whether computers can do better than humans. For the time-being, most of PFSMs designed by hand are performing better than those designed by computers. The aim of this project was therefore to search for promising techniques/algorithms that would be able to beat humans performances.

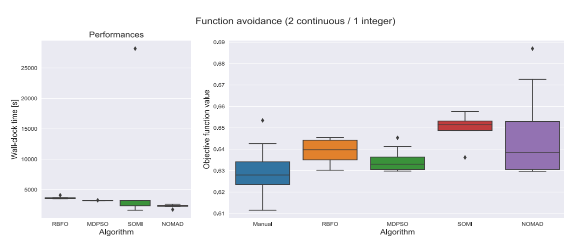
At the beginning of the project, three algorithms were already tested by the DISAL laboratory: MDPSO, NOMAD and IRACE. Exploring new promising ways leads us to the algorithms using surrogate model. Known as SOMI and RBFO algorithms, both are using radial basis function interpolant as surrogate models. They were selected for their ability to find global minimum in a limited number of steps. In addition to these two algorithms, one variant of SOMI, called SOMI GPR was developed in the scope of this project to provide diversity in the choice of the surrogate model type. This variant is using gaussian process regressor as surrogate model instead of radial basis function one. Such a choice was motivated by the fact that gaussian process regression is returning, in addition to an interpolant, a measure of the uncertainty about its predictions.

Once implemented, the performances of these were first evaluated on benchmark functions to confirm their good functioning and test their robustness against noise. This was also the opportunity to have insights about other parameters such as time complexity, sensitivity to the input dimension as well as the convergence speed. This first stage highlighted the power of the RBFO algorithm as it was returning the best results on each benchmark function.

The second step was to evaluate the performances of the algorithm on the real cost function, the one that assesses the efficiency of a PFSM with respect to a task. Three different tasks were considered and surprisingly RBFO failed to keep pace with MDPSO. One of the main reasons that explains it is the fact that MDPSO took advantage of its noise resistance. The noise added on the benchmark functions was lower than the one present in the simulations. The results also showed that algorithms were not able to do better than humans.



Evaluation of the performances of the algorithm on a benchmark function (C1)



Evaluation of the performances of the algorithm on the real cost function