SS 2019-2020, SMT-ROB, DISAL-SP136 Project Summary

Mixed-Discrete Particle Swarm Optimization, High-Dimensional Performance Evaluation and Comparison

Eric Bergkvist

Professor : Alcherio Martinoli
Assistant(s) : Cyrill Baumann

Classical Particle Swarm Optimization (PSO) is a popular metaheuristic to solve continuous optimization problems. A mixed-discrete version (MDPSO) was implemented by Chowdhury et al. In this project, its performance was evaluated on 9 benchmark functions from literature, based on the original paper. These problems are called Mixed-Integer Nonlinear Programs (MINLPs). From these, 11 more problems were generated to test the algorithm on higher-dimensional problems, as well as on categorical problems. The latter are particularly interesting as real robotic experiments can be categorical, which means one or more variables are integers which are not ordered; they represent the belonging to a category.

MDPSO was then compared to a state-of-the-art method for solving MINLPs: Mesh Adaptive Direct Search (MADS). It was chosen as it can tackle all kinds of optimization problems, as MDPSO. Both methods are considered derivative-free and they can therefore be applied to problems where no derivative information is retrievable or where the function's value (called "fitness") is long (and therefore inconvenient) to compute. MADS is an iterative algorithm composed of a explorative search step – which looks in the whole search space for a solution with a better fitness than the current best – followed by an exploitative poll step, which looks for a better solution in the neighborhood of the current best.

The main objectives were to analyze the impact of selected hyperparameters of each method and then to compare their performances with optimal hyperparameters. For each setting tested, 20 runs of MDPSO were performed, while for MADS 10 runs were done each time. The results of the experiments were analyzed through the average feasible fitness and the standard deviation of all runs (the fitness of the runs that did not reach the feasible set are not taken into account), the best feasible fitness reached by any run and the percentage of runs that reached the feasible set. These values were compared for each setting and were then interpreted.

The impact of the number of particles on MDPSO's optimization performance was analyzed. On most problems, the results obtained were significantly better when using *5N* particles, with *N* being the problem's dimension. This result is surprising as a rule of thumb for classical PSO is to use only N particles. We then analyzed the effect of the diversity-preservation mechanism on MDPSO's performance. We realized that it does not systematically improve the algorithm's performance as can be seen in Figure 1.

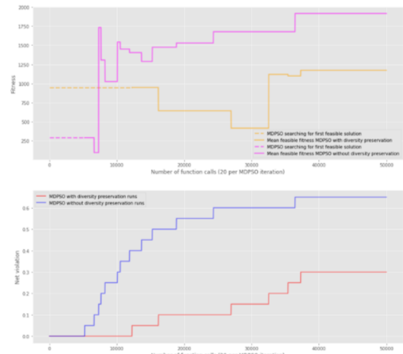

Figure 1: The fitness and the feasibility of MDPSO with and without diversity preservation, in function of the number of function evaluations.

The effect of the balance between the exploration and exploitation properties of MADS on its performance were analyzed. A large allocation to initialization together with a search step composed of Latin Hypercube searches (sophisticated grid search) was chosen based on the results obtained.

The performances of MDPSO and MADS were then compared on the set of benchmark functions, using the results of the previous experiments on each method. MADS seems to be a close winner as it outperforms MDPSO more often than not. However, in general the two methods reach similar fitness values, often close to global optima (in the problems tested). From the results obtained, no clear pattern stands out to distinguish on which types of problems one algorithm works better than the other.