Learning Automata for Hyperheuristic Selection

T. Wauters[1,2], K. Verbeeck[1,2], B. Bilgin[1,2], and P. Demeester[1,2]

[1]KaHo Sint-Lieven, Vakgroep IT
{tony.wauters,katja.verbeeck,burak.bilgin,peter.demeester}
@kahosl.be
[2]K.U.Leuven, Departement Computerwetenschappen

This abstract introduces a novel approach for selecting heuristics within a hyperheuristic framework. In the literature we can find heuristic selection mechanisms like Simple-Random, Random Descent, Random Permutation, Permutation Descent and Reinforcement based Tabu Search. Our approach applies classical Reinforcement Learning algorithms for Finite Action Learning Automata to select the heuristics. Learning Automata can be described as methods that choose an action at every time step ($t = 0, 1, 2, \ldots$) according to a probability vector $p(t)$. This probability vector has to be updated. A popular update mechanism is Lineair Reward-Penalty. It has two parameters: the reward and penalty parameter. Intuitively this mechanism implements the hypotheses that whenever the selected actions result in favorable reinforcement, the action probability is increased; otherwise it is decreased.

We introduce two new selecting methods for heuristic selection that are both based on Learning Automata. The first one chooses a heuristic at every selection step. We call it Learning Automata Selection. The second method chooses a new order of n heuristics at every n selection steps. This one is called Order based Learning Automata Selection. Notice that this order can be more than just a permutation of the n heuristics, for example, an alternation or a sequence of n similar heuristics, but also variable length orders are possible. Both selection methods update their Learning Automata at every timestep using the quality of the chosen heuristics.

To validate our method we added it to an existing hyperheuristic that was used for the patient admission scheduling problem. The best performing combination was a Simple-Random selection method together with a Simulated Annealing acceptance criteria. We compared the method with other selection methods and it performs very well. When evaluating the results in function of the number of iterations, we notice large improvements. However, when evaluated in function of the computation time, the improvements are much smaller due to the overhead of the learning. The latter is an important challenge for future research.