Ant-Q Hyper Heuristic Approach applied to the Cross-domain Heuristic Search Challenge problems

Imen Khamassi
University of Tunisia
Higher Institute of Management of Tunis
imenkhamassi@yahoo.fr

Abstract. The first Cross-domain Heuristic Search Challenge (CHeSC 2011) is an international research competition aimed at measuring hyper-heuristics performance over several problem domains. Hyper-heuristics are new approaches which aim at raising the level of abstraction when solving combinatorial optimisation problems. During this competition, we have applied the Ant-Q hyper-heuristic approach, proposed by Khamassi et al., for building good sequences of low-level heuristics aimed at finding optimal solutions. This approach was applied to the MAX-SAT Problem, the One Dimensional Bin Packing Problem, the Permutation Flow Shop Problem and the Personnel Scheduling Problem; and tested through a large set of benchmark problems.

1 Introduction

Hyper-heuristics encompass a set of approaches which aim at raising the level of generality when dealing with hard computational search problems by exposing much more independency to problem specification [1].

The Ant-Q hyper-heuristic approach [2] is a constructive hyper-heuristic model, which transliterates the significant learning ability of Ant-Q algorithm [3] for building good sequences of low-level heuristics aimed at gradually finding optimal solutions. Notice that the Ant-Q algorithm is an agent based algorithm that combines Ant colony based methods [4] with the reinforcement learning technique [5].

In order to highlight the different improvement stages of Ant-Q hyper-heuristic approach, we have applied it to solve the MAX-SAT Problem, the One Dimensional Bin Packing Problem, the Permutation Flow Shop Problem and the Personnel Scheduling Problem during the Cross-domain Heuristic Search Challenge [6].

2 Ant-Q Hyper Heuristic Approach

In Ant-Q hyper-heuristic, the problem is exposed as a graph where vertices represent low-level heuristics related with directional edges. A number of ants are initially dispersed on the vertices and proceed by gradually constructing sequences of heuristics by traversing the graph. Each ant chooses the next vertex to move on, traverses the
related edge and applies the corresponding heuristic. At each move, the ant lays a quantity of pheromone on the crossed edge in order to mark its own path which could be followed by other members of the colony.

In order to guarantee the efficiency of the proposed approach, the ants ought to intelligently explore the heuristic search space through a learning mechanism that can adaptively guide the search. In fact, the ants learn by interacting with their environment in the sense that they take account from the previous results and attempt to predict the future utility of each action’s choice. The learning technique investigated in our approach is based on Ant-Q algorithm proposed in [3].

2.1 Formulation

The equations used in our approach were transliterated from those used in Ant-Q algorithm [3] with some modifications.

Equation (1) presents the state transition rule in which we manage a fixed set of low-level heuristics \( H \). \( HE(r,s) \) is a heuristic information which indicates how useful is to apply heuristic \( h_s \) after heuristic \( h_r \). \( AQ(r,s) \) represents the pheromone laid on the edge \((r,s)\) which can be considered as a learning value. Parameters \( \delta \) and \( \beta \) weigh the relative importance of \( AQ(r,s) \) and \( HE(r,s) \), \( q \) is a random value with uniform probability in the range \([0,1]\), \( q_0 \) \((0 \leq q_0 \leq 1)\) is a parameter such that the higher \( q_0 \) the smaller the probability to make a random choice and \( S \) is a random heuristic which is selected according to a given distribution of probabilities described by:

\[
p_q(r,s) = \frac{[AQ(r,s)]^\delta \cdot [HE(r,s)]^\beta}{\sum_{u \in H}[AQ(r,u)]^\delta \cdot [HE(r,u)]^\beta}
\]  

(2)

The update rule, given by (3), is respectively composed of the discounted old value which refers to the pheromone evaporation; the reinforcement term, namely delayed reinforcement reward, which adjusts the new information learned towards the old one; and the discounted evaluation of the next state which takes into account the importance of the future rewards. \( \alpha \) and \( \gamma \) are respectively the pheromone evaporation and the discount factor parameters

\[
AQ(r,s) \leftarrow (1 - \alpha)AQ(r,s) + \alpha \left( \Delta AQ(r,s) + \gamma \cdot \max_{z \in H} AQ(r,z) \right)
\]  

(3)

\( \Delta AQ(r,s) \) is computed according to (4) where \( k_{best} \) is the ant which has made the best tour in the current cycle; \( f_{best} \) is its fitness function defined according to problem specification and \( W \) is a parameter used to adjust \( f_{best} \) values.

\[
\Delta AQ(r,s) = \begin{cases} 
W \cdot f_{best} & \text{if } (r,s) \notin \text{ tour done by ant } k_{best} \\
0 & \text{otherwise}
\end{cases}
\]  

(4)
2.2 Algorithm

At step 1, an initial quantity of pheromone \( AQ_0 \) is laid on all the edges and each ant applies an initial heuristic \( h_i \) to its set of initial solutions. At step 2, a cycle is defined in which each ant chooses the next heuristic \( h_k \) according to (1) and (2); and applies it on its best solution. After each move, the quantity of pheromone \( AQ(r,s) \) is locally updated according to (3) in which \( \Delta AQ(r,s) \) is null. At step 3, when all the ants have finished their tours, the best results are kept and the quantity of pheromone \( AQ(r,s) \) is globally updated according to (3) in which \( \gamma \max_{z \in S} AQ(r,z) \) is null. Notice that, in (4), the solutions are evaluated by means of a fitness function which varies according to the problem specification. This fitness function is used to compute the delayed reinforcement reward \( \Delta AQ(r,s) \) which refers to the additional quantity of pheromone laid only on the paths that have provided the best solutions. Finally, at step 4, if a termination condition is not met the algorithm continues to execute other cycles.

3. Experiments

The values of the standard parameters, respectively described in (1), (2), (3) and (4), were set to \( \delta=1; \beta=2; \alpha=0.1; \gamma=0.3; q_0=0.9 \) and \( W=10 \). These values were optimised for the Ant-Q algorithm [7] and were kept for the Ant-Q hyper-heuristic [2], since early experiments have shown that the proposed hyper-heuristic is less sensitive to the changes of these parameters.
Concerning the number of ants, we have remarked that better solutions are reached when the total number of ants is increased. For this experimental investigation we have found that 2 ants per heuristic are quite sufficient to reach good results.

Concerning the initial quantity of pheromone, we have assigned to $AQ_0$ the inverse of the best objective function value of the initial solutions:

$$AQ_0 = \frac{1}{f_{\text{best}}}$$  \hspace{1cm} (5)

4. Conclusions and future work
During the Cross-domain Heuristic Search Challenge, we have developed the Ant-Q hyper heuristic, which investigates the Ant-Q algorithm as means of constructing effective sequences of heuristics moves; and applied it to the MAX-SAT Problem, the One Dimensional Bin Packing Problem, the Permutation Flow Shop Problem and the Personnel Scheduling Problem in order to highlight its apparent qualities for solving combinatorial optimisation problems.

References