Reinforcement Learning for Adaptive Operator Selection in Memetic Search Applied to Quadratic Assignment Problem

Stephanus Daniel Handoko*
dhandoko@smu.edu.sg

Duc Thien Nguyen*
dtnguyen@smu.edu.sg

Zhi Yuan**
yuanz@hsu-hh.de

Hoong Chuin Lau*
hclau@smu.edu.sg

*School of Information Systems, Singapore Management University, Singapore
**Department of Mechanical Engineering, Helmut Schmidt University, Hamburg, Germany

ABSTRACT

Memetic search is well known as one of the state-of-the-art metaheuristics for finding high-quality solutions to NP-hard problems. Its performance is often attributable to appropriate design, including the choice of its operators. In this paper, we propose a Markov Decision Process model for the selection of crossover operators in the course of the evolutionary search. We solve the proposed model by a Q-learning method. We experimentally verify the efficacy of our proposed approach on the benchmark instances of Quadratic Assignment Problem.

1. INTRODUCTION

Evolutionary search has been widely used to find solutions to many NP-hard problems [1]. Rudimentarily, it evolves population of individuals which represent the candidate solutions to some problem. Central to the evolution of these individuals are variation operators and selection processes. The former enables search method to sample various regions of the search space to escape from some local optima. The latter biases the evolution towards some promising regions where fitter individuals may be identified. These evolutionary operators therefore constitute the algorithmic core of an evolutionary search technique. The quality of the evolutionary operators is thus critical for the performance of the search algorithm. While the efficiency of a specific evolutionary search method may be well-established on a number of problem instances, its performance normally depends on the correct settings of its components; and domain knowledge is often essential to good performance.

The No Free Lunch theorem [2] suggests that it is difficult to anticipate the efficiency of an algorithm on any instances of a wide class of problems without preliminary experiment or learning process. Some off-line tuning algorithms [3] may be used to adjust parameters by running the search on training instances prior to using it for solving new problems. This does not, however, exploit the fact that the usefulness of the operators often varies during the course of evolution. Additionally, dependencies between operators may exist and hence multiple operators may provide better results through interactions rather than when they are applied individually. Setting the application rates of the variation operators can be achieved by using methods that control—in the course of searching for the solution to the problem—which variation operator should be applied based on the recent performance of all available operators. Commonly referred to as Adaptive Operator Selection (AOS) [4], such control provides some adaptive mechanism for selecting one suitable operator for each iteration of the evolutionary search. Recent approaches [5, 6] have proposed such adaptive mechanisms for general evolutionary search with possibly many variation operators whose behavior may be unknown, giving rise to uncertainty. Our recent study [7] has confirmed that adapting operator choices well in accordance with the different search stages of an algorithm is important for the effectiveness of an AOS method. Herein, we propose to formulate the AOS problem as a Markov Decision Process (MDP) [8] by automatically mapping search stages into states and solve the MDP by means of reinforcement learning.

2. ADAPTIVE OPERATOR SELECTION

The development of an adaptive operator selection (AOS) method includes a credit assignment mechanism that evaluates the quality of operator applied in the last decision stage, and an adaptation mechanism that decide which operator to choose based on the evaluated quality. Herein, we reward an operator only if it improves over its better parent, and the amount fo reward depends on its comparison with the current best solution:

$$\text{reward} = \frac{\text{cost}_{\text{best}}}{\text{cost}_{\text{child}}} \cdot I(\text{cost}_{\text{parent}} - \text{cost}_{\text{child}}) \qquad (1)$$

The indicator $I(\cdot)$ returns 1 if the child improves its better parent and 0 otherwise. After the quality evaluation of the operator applied in the current generation, there exist various established adaptation mechanisms: Probability Matching (PM), Adaptive Pursuit (AP), and Multi-Armed Bandit (MAB). An extensive experimental study of these methods can be found in [9, 7].

GECCO'14, July 12–16, 2014, Vancouver, BC, Canada.
ACM 978-1-4503-2881-4/14/07.
http://dx.doi.org/10.1145/2598394.2598451.

3. A MARKOV DECISION PROCESS FOR ADAPTIVE OPERATOR SELECTION

The established AOS methods mentioned above are stateless. A state-based MDP is defined by the sets of states, actions, and transitions. The states include:

1. *Restart.* Binary state variable defining whether it is just restarted. In our implementation, restart is triggered to diversify the current population when the diversity level falls below a threshold. Intuitively, a more explorative crossover operator may be preferred to explore the surroundings.

2. *Fitness Improvement.* Binary state variable defining whether a new restart best solution is just found. A fitness improvement may indicate that the search has entered a promising region. Intuitively, an exploitive crossover operator is preferred to find the best solution therein.

3. *Diversity Level.* Three discretized diversity levels are considered: low (1/8 the maximum diversity or less), medium (1/8 to 1/4 the maximum diversity), and high (1/4 the maximum diversity or more). The maximum diversity is computed as $n \cdot p \cdot (p-1)/2$ where n is the size of the QAP instance and p is the population size.

A total of $2 \times 2 \times 3 = 12$ states are used for defining different search stages. Besides, each action is defined as using one of the available operators. By modelling the evolutionary search as a Markov Decision Process, the expected value Q of each state-action (s,a) pair can be formulated using a recursive equation

$$Q(s,a) = R(s,a) + \sum_{s'} P(s',a,s) \max_{a'} Q(s',a'), \qquad (2)$$

where $P(s',a,s)$ is the probability to go into a new state s' and $R(s,a)$ is the expected reward from executing action a when in the current state s. In evolutionary search, $Q(s,a)$ is instance-specific value. We consider herein the Q-learning method, which allows us to explore the Q value at the same time as solving it. To implement the reinforcement learning method, we maintain and update $Q(s,a)$ iteratively at each generation such that

$$Q_{t+1}(s_t,a_t) = Q_t(s_t,a_t) + \alpha \cdot \qquad (3)$$
$$\Big[R_{t+1}(s_t,a_t) + \delta \max_a Q_t(s_{t+1},a) - Q_t(s_t,a_t)\Big]$$

where α and δ are the learning and discount rates.

4. EXPERIMENTS

For assessing the aforementioned AOS methods, we followed the implementation of a memetic algorithm for the QAP [10], and consider the selection of its crossover operators. Two different scenarios were considered herein. The first requires the AOS to select from 3 crossover operators with comparable performances: CX, OX, and PMX. The second requires the AOS to select from a pool with one additional crossover operator, DPX, which demonstrates some outlier performance. In each scenario, each of the five AOS methods (the proposed RL, PM, AP, MAB, and the "naive" approach N) was run 10 times on each of the 137 benchmark QAP instances from QAPLib. The Friedman test was used to assess the pairwise statistical significance so as to compare among the various AOS methods under the two scenarios. The mean ranks and the 95% confidence intervals obtained are shown in Figure 1. An overlap on the intervals of any two methods indicates their performance difference is not statistically significant. The proposed RL approach is found to outperform its counterparts. And interestingly, while an additional outlier operator worsens the performance of the other three AOS methods PM, AP, and MAB, the performance of RL is rather robust and scales well as the number of operators increases.

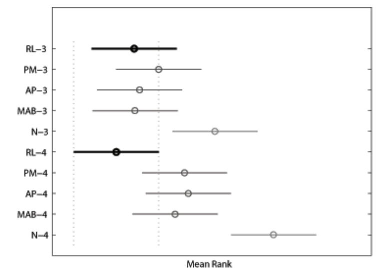

Figure 1: Mean ranks and 95% confidence intervals (by Friedman test) of various AOS methods (RL, PM, AP, MAB, and N) under two scenarios (3 or 4 crossover operators) on 137 QAPLib instances.

5. ACKNOWLEDGMENT

This research is supported by Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office, Media Development Authority (MDA).

6. REFERENCES

[1] K.A. De Jong. *Evolutionary Computation: A Unified Approach*, MIT Press, 2006.

[2] D.H. Wolpert and W.G. Macready. No Free Lunch Theorems for Optimization. *IEEE T. Evolut. Comput.*, 1(1):67–82, 1997.

[3] M. Birattari, et al. F-Race and Iterated F-Race: An overview. *Experimental Methods for the Analysis of Optimization Algorithms*, 311–336, 2010.

[4] E. Krempser, et al. Adaptive Operator Selection at the Hyper-level. *LNCS*, 7492:378–387, 2012.

[5] Á. Fialho, et al. Extreme Value Based Adaptive Operator Selection. *LNCS*, 5199:175–184, 2008.

[6] J. Maturana, et al. Autonomous Operator Management for Evolutionary Algorithms. *J. Heuristics*, 16(6):881–909, 2010.

[7] Z. Yuan, et al. An Empirical Study of Off-line Configuration and On-line Adaptation in Operator Selection. *LNCS*, in Press.

[8] M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Wiley, 1994.

[9] G. Francesca, et al. Off-line and On-line Tuning: A Study on Operator Selection for A Memetic Algorithm Applied to the QAP. *LNCS*, 6622:203–214, 2011.

[10] P. Merz and B. Freisleben. Fitness Landscape Analysis and Memetic Algorithms for the QAP. *IEEE T. Evolut. Comput.*, 4(4):337–352, 2000.