A Novel Genetic Programming Algorithm with Knowledge Transfer for Uncertain Capacitated Arc Routing Problem

Mazhar Ansari Ardeh, Yi Mei, and Mengjie Zhang

Victoria University of Wellington, Wellington, New Zealand
{mazhar.ansariardeh, yi.mei, mengjie.zhang}@ecs.vuw.ac.nz

Abstract. Uncertain Capacitated Arc Routing Problem (UCARP) is a challenging optimization problem. Genetic Programming (GP) has been successfully applied to train routing policies (heuristics to make decisions in real time rather than a fixed solution) to respond to uncertain environments effectively. However, the effectiveness of routing policy is scenario dependent, and it takes time to train a new routing policy for each scenario. In this paper, we investigate GP with knowledge transfer to improve the training efficiency by reusing useful knowledge from previously solved related scenarios. We propose a novel knowledge transfer approach which our experimental results show that it obtained significantly higher training efficiency than the existing GP knowledge transfer methods, and the vanilla training process without knowledge transfer.

Keywords: Uncertain Arc Routing · Genetic Programming · Hyper-Heuristics · Transfer Learning

1 Introduction

Uncertain Capacitated Arc Routing Problem (UCARP) has many important real-world applications in supply chain and logistics. In UCARP, a graph $G(V, E)$ is given, where V and E are the set of nodes and edges. Each edge $e \in E$ has a positive stochastic deadheading cost $dc(e)$, a non-negative serving cost $sc(e)$, and a non-negative stochastic demand $d(e)$. An edge with positive demand is called a *task*. A number of vehicles with capacity Q are located at the depot $v_0 \in V$. The problem is to find the optimal routes for the vehicles subject to the constraints: (1) each vehicle needs to start and end its route at the depot; (2) between two refills, the total demand served by each vehicle cannot exceed its capacity.

There have been several studies dedicated to solving UCARP (e.g. (Mei et al., 2010)), among which the Genetic Programming (GP) based approaches have achieved great success. GP evolves (trains) routing policies, which are decision-making heuristics, rather than solutions. A routing policy can generate the solution in an online fashion based on the latest information, and thus is effective to handle uncertain environments.

The effectiveness of routing policies depends on the problem scenario (e.g. the topology of the graph, and the number of vehicles to be used). The performance of a routing policy can dramatically decrease when changing from one scenario to another. Intuitively, one can retrain the routing policy in the new scenario from scratch. However, it can be time consuming and inefficient. In this case, we propose GP with transfer learning to improve the efficiency of the retraining.

Transfer learning can be defined as "the improvement of learning in a new task through the transfer of knowledge from a related task that has already been learned" (Torrey and Shavlik, 2010). For transfer learning in GP, a commonly used strategy is to transfer sub-trees from the source domain to the target domain. Intuitively, different subtrees in the source domain should have different levels of importance and should be more likely to be transferred. However, it is challenging to quantitatively measure the re-usability of a subtree. Existing studies mostly select the subtree in promising individuals randomly (e.g. (Dinh et al., 2015)), which is not an optimal strategy. Another measure of defining importance of subtrees is to consider the number of times that they appeared in the source domain (Ansari Ardeh et al., 2019). However, frequency may be misleading because the final GP tree may have many redundant branches and some frequent subtrees can be in the redundant branches. In addition, subtrees can be structurally different but essentially the same.

In this paper we aim to propose a novel GP with subtree transfer to improve the effectiveness of retraining routing policies for UCARP. The research objectives that we follow in this paper are (1) propose a new and more accurate measure for the reusability of subtrees based on their contribution to the individuals; (2) develop a novel GP with knowledge transfer based on the new reusability measure to transfer subtrees from source domains to the target domain of UCARP; (3) verify the efficacy of the proposed algorithm on different transfer scenarios.

2 Novel Subtree Transfer for Genetic Programming Hyper-Heuristic

We propose a novel method for filtering good transferable knowledge by evaluating the reusability of subtrees to distinguish their potential for transfer. We choose the final GP population in the source domain as the knowledge source.

When identifying transferable subtrees, it is natural to conjecture that individuals with good fitness value are better sources for knowledge extraction. Therefore, we consider the subtrees of the top 50% individuals in the final population in terms of their test performance in the source domain.

To form the pool of the candidate subtrees, we adopt the following two strategies that are commonly used by existing works: (1) **All**: All the subtrees of all the considered individuals are included in the pool; (2) **Root Subtrees**: Immediate subtrees of the roots of the considered individuals are included in the pool. The subtrees in the pool have different reusability. To select the transferred subtrees more intelligently, we propose a new reusability measure based on the contribution of a subtree to its tree (Mei et al., 2017).

Given a GP tree x, the contribution $\xi(x,\tau)$ of its subtree τ is defined as:

$$\xi(x,\tau) = fit(x|\tau = 1) - fit(x). \quad (1)$$

Then, the weight (importance) of the subtree τ is defined as follows:

$$w(\tau) = \sum_{x \in \Omega} \xi(x,\tau) pow(x), pow(x) = \frac{g(x) - g_{min}}{g_{max} - g_{min}}, \quad (2)$$

in which Ω is the set of all the considered individuals, and $pow(x)$ is the normalised fitness of individual x. Let $fit(x)$ be the fitness of individual x and Ψ the set of all individuals that were evaluated in the source domain,

$$g(x) = \frac{1}{(1 + fit(x))}, g_{min} = min\{g(x)|x \in \Psi\}, g_{max} = max\{g(x)|x \in \Psi\} \quad (3)$$

The motivation behind Eq. (2) is to let good subtrees of good individuals have higher weights. In the target domain, the subtrees in the pool are sorted by their weights and the top subtrees form 50% of the initial population. The corresponding algorithms are named (1) *ContribSub-all* and (2) *ContribSub-subtree*.

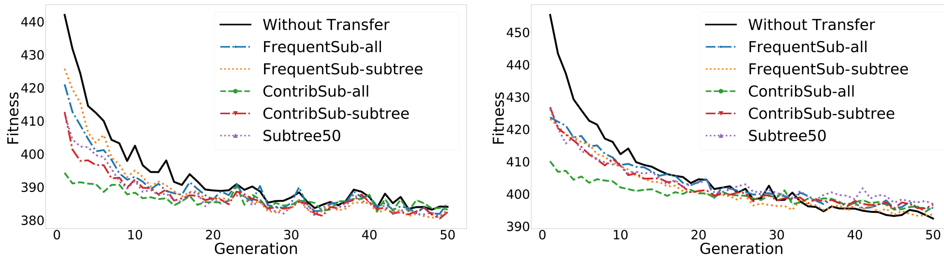
3 Experimental Studies

A collection of experimental source and target domain settings are designed to evaluate the proposed methods. In our design, the difference between source and target domain is in terms of the number of vehicles. Several UCARP instances with different sizes are chosen to have a thorough investigation of knowledge transfer in different scenarios. GP settings and datasets in this paper are based on the work in (Mei and Zhang, 2018). All algorithms are run 30 times independently. The compared algorithms include FrequentSub-all, FrequentSub-subtree (Ansari Ardeh et al., 2019), SubTree50 (Dinh et al., 2015), ContribSub-all and ContribSub-subtree and GPHH without any knowledge transfer. The reason for including SubTree50 is that it has the same pool of candidate trees as FrequentSub-subtree and ContribSub-subtree.

Figs. 1 and 2 show the convergence curves of the test performance in the target domain. We conducted Wilcoxon's rank sum test to compare between the final test performance of the algorithms, and the results showed no significant difference.

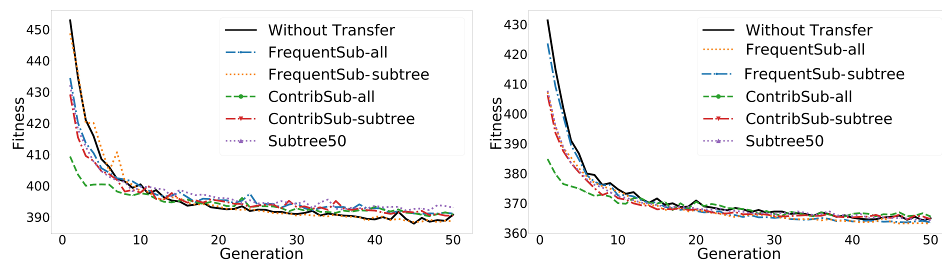
Overall, we have the following observations:

– Subtree transfer can improve the efficiency of the retraining process of routing policies in the target UCARP domain.
– The contribution measure is an effective indicator for the reusability of subtrees, and can identify better subtrees to the target domain than the random selection and frequency-based selection.



(a) (b)

Fig. 1 *Convergence curves of the compared algorithms on **gdb9** from 10 to (a) 9 and (b) 11 vehicles.*



(a) (b)

Fig. 2 *Convergence curves of the compared algorithms on **val9C** from 5 to (a) 4 and (b) 6 vehicles.*

– The frequency measure for both the “All” and “root” pools performed comparable with the random subtree selection for the “root subtrees” pool.

The possible reasons for the above observations are that the subtrees of the root are large and may not appear more than once. Thus, frequency-based method is very similar to random selection. If considering all the subtrees, then the small subtrees are more likely to have higher frequency, and tend to be selected. However, the frequency can be misleading as the occurrences can be in redundant branches but the contribution-based measure can handle this, and identify the truly important subtrees regardless of their frequency. Therefore, the contribution-based transfer methods can work better.

In our experiments, we noticed subtrees could receive different opinions from the frequency and contribution measures. For example, the subtree *max(max(min(FUT, FRT), CFH/FULL), CFH/FULL)* appeared 47 times in the source domain and was transferred by the FrequentSub-all method. However, its contribution to one of its trees *min(CTT1, max(min(FUT, (max(DEM, DC) / max (CFR1 / CTT1))/((FRT * CR) / DEM1)), min(FUT, FRT))) + (((CR + FULL) * (FUT / CTD) * (FULL * max(DC, FUT))) * ((max(max(min(FUT, FRT), CFH/FULL), CFH/FULL)) / (max(min(CFD, RQ), RQ / DEM))/ RQ)* was -47.57. Thus, ContribSub-all considered it as useless and did not transfer it. Note that the the subtree had a complex structure and thus, the commonly considered algebraic simplification (Zhang et al., 2005) plus frequency measure cannot effectively detect the important subtrees for transfer.

4 Conclusions and Future Works

In this paper, we proposed a new GP with knowledge transfer for retraining routing policies for UCARP. To reduce the noise caused by random selection, we proposed two strategies to detect more useful subtrees. Our experiments showed that subtree transfer can effectively improve the efficiency of the retraining process, making GP achieve the desired performance in a much shorter time. Specifically, the contribution measure showed better efficiency and effectiveness than random selection and thus achieved much better convergence speed in the retraining process. In the future, we will develop more advanced tree transformation techniques to reduce the noise. We will also consider clustering methods to cluster the similar subtrees together to avoid transferring redundant knowledge.

References

M. Ansari Ardeh, Y. Mei, and M. Zhang. Transfer learning in genetic programming hyper-heuristic for solving uncertain capacitated arc routing problem. In *IEEE Congress on Evolutionary Computation*, 2019.

T. T. H. Dinh, T. H. Chu, and Q. U. Nguyen. Transfer learning in genetic programming. In *IEEE ongress on Evolutionary Computation*, 2015.

Y. Mei and M. Zhang. Genetic programming hyper-heuristic for multi-vehicle uncertain capacitated arc routing problem. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 141–142, 2018.

Y. Mei, K. Tang, and X. Yao. Capacitated arc routing problem in uncertain environments. In *IEEE Congress on Evolutionary Computation,*, 2010.

Y. Mei, S. Nguyen, B. Xue, and M. Zhang. An efficient feature selection algorithm for evolving job shop scheduling rules with genetic programming. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 1(5): 339–353, 2017.

L. Torrey and J. Shavlik. Transfer learning. In *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, pages 242–264. IGI Global, 2010.

M. Zhang, Y. Zhang, and W. Smart. Program simplification in genetic programming for object classification. In R. Khosla, R. J. Howlett, and L. C. Jain, editors, *Knowledge-Based Intelligent Information and Engineering Systems*, pages 988–996, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.