Evolving Self-Adaptive Tabu Search Algorithm for Storage Location Assignment Problems

Jing Xie
School of Computer Science
and IT, RMIT University
jing.xie@rmit.edu.au

Yi Mei
School of Computer Science
and IT, RMIT University
yi.mei@rmit.edu.au

Andy Song
School of Computer Science
and IT, RMIT University
andy.song@rmit.edu.au

ABSTRACT

We propose a novel grammar guided Genetic Programming method to solve a real world problem, the Storage Location Assignment Problem (SLAP) with Grouping Constraints. Self-adaptive Tabu Search Algorithms are evolved by this approach as solvers for SLAP. The quality of evolved solutions and execution speed are both considered in the evaluation function. So our method can evolve efficient and effective Tabu Search algorithms for for solving a given SLAP instance. The experimental results show that better Tabu search algorithms can be found by this approach. They can outperform manually designed Tabu Search method.

Categories and Subject Descriptors

G.1.6 [**Optimization**]: Stochastic Programming
; I.2.8 [**Problem Solving, Control Methods and Search**]: Heuristic Methods

General Terms

Algorithms, Experimentation

Keywords

Storage Location Assignment Problem, Tabu Search, Self-Adaptive, Genetic Programming

1. INTRODUCTION

The Storage Location Assignment Problem (SLAP) is one of the prominent problems in warehouse optimization. The problem is to rearrange the inventory layout to minimize the cost of product storage and retrieval [1]. The basic concept of how to solve this problem is relatively simple. Products of high demand should be placed near the loading dock and correlated products should be stored close to each other [2]. However, finding a solve is difficult in practice due to the following reasons. Firstly, the modelling of the problem is a challenging task by itself because of the problem-specific constraints. Secondly, the problem is associated with other operations in the warehouse. Thirdly, the problem is dynamic as parameters such as demand may change over time.

In this paper we propose a grammar guided Genetic Programming method which is to evolve Tabu search algorithms to address the above challenges. A novel representation with the grammars are presented. The experiments show that better Tabu algorithms can be found which can outperform a Tabu search which has been carefully and manually designed for SLAPs.

GECCO'15, July 11-15, 2015, Madrid, Spain.

2. PROBLEM DESCRIPTION

The SLAP instance addressed by this study is for a real wholesaler warehouse which mainly stores garments. The task is to assign a set of products to a set of locations. Each product consists of a number of items in different color and size. The overall goal is to reduce the total distance for picking operations, e.g. the total distance for warehouse operators (pickers) to travel between these locations and the loading dock. Items of a same product may have significantly different demand for example trousers of size XS, color pink are unpopluar. Items of the same product are preferred to be placed in adjacent locations, while assigning items strictly according to demand may lead to an awkward arrangment. Hence, each product is allowed to be split into at most two subgroups and stored separately in the warehouse.

The problem description and the Integer Linear Programming model was firstly proposed in [3]. A GP-based hyper-heuristic method was proposed in the same paper to evolve reusable matching functions for this problem. This method was then combined with two sampling techniques to scale up the solutions for larger problems [4]. In a recent study, a Restricted Neighbourhood Tabu Search algorithm was proposed [5] that was found to be the most efficient algorithm for solving this problem. The design of the algorithm was based on the intuitive understanding and the theoretical analysis of the problem characteristics. In this study, we hybridize this Tabu Search method with GP to automatically design a more efficient self-adaptive Tabu Search method.

3. METHODOLOGY

The grammar of the evolved Tabu Search method is shown in Fig. 1. Under a standard Tabu Search framework, the search starts with an empty Tabu list and an initial solution. In each iteration, the best non-tabu neighbor of the current solution is selected as the next solution. The parameter setting of the algorithm remains the same throughout the

```
1 <start>     ::= Init() Repeat_Unitl(<do>)
2 <do>        ::= if(<condition>, <do>, <do>) | <tabu>
3 <condition> ::= true | false | leq(curIter, <threshold>)
4 <tabu>      ::= set(<num>, <num>, <threshold>)
5                 search() updateTabuParams()
6 <num>       ::= (2) | (4) | (6) | (8) | (16)
7                 | (24) | (32) | (48) | (64)
8 <threshold> ::= (0.1) | (0.2) | (0.3) | (0.4) | (0.5)
9                 | (0.6) | (0.7) | (0.8) | (0.9) | (1.0)
```

Figure 1: Grammar of the Self-Adaptive Tabu Search Algorithm for SLAP with Grouping Constraints

whole search procedure. In the proposed self-adaptive Tabu search method, the parameter setting changes based on the feedback from a validator which is evolved by GP. This approach enables the evolved algorithm to be informed of the changes during the searching process, so the search process can adapt to those changes. For example, the diversity of the visited solutions is more significant than other factors at the beginning of the search. The intensification strategy is preferred when there is not much computational budget left [6]. This type of adjustment can be achieved by changing the amount of effort spent on generating the neighbors of a solution, or changing the percentage of neighborhood to be visited. The first parameter of method set (Line 4 on Fig 1) refers to the size of the tabu list, which is also equal to the tabu tenure, i.e., the iterations a solution can be kept in the tabu list. The second parameter of set is the iterations of local search to be performed for each solution. The percentage of neighborhood to be explored is determined by the parameter threshold.

We consider two factors when evolving the algorithm. They are the quality of the solution and the time spent on finding that solution. The fitness function used in the experiment can be denoted as the following:

$$\sum P \times D + \beta * time \tag{1}$$

where P and D refer to the picking frequency of an item and the distance of its location to the loading dock, $time$ is the elapsed time taken to complete a search with a certain number of iterations.

4. EXPERIMENTAL RESULTS

Two data sets, each with 100 items, are used as the training data. The first data set is trained with $\beta = 0.1$. The second data set is trained with $\beta = 0.01$. The change of β affects the magnitude of the effect of the speed of the algorithm. Each data set is trained 30 times to choose the best and the most comprehensible individual. We have an evolved program $A1$ using the first data set, and $A2$ by the second data set. The evolved programs $A1$ and $A2$ are evaluated on four test problems. These test problems all have 400 items to be assigned to a warehouse with 20 shelves. Each test is assigned 190 seconds for the search process. This parameter setting is consistent with other researches in the literature. Each method runs for 30 times.

Table 1 shows the results from A1, A2 and RNTS, a manually designed Tabu method . The best results are in bold. It can be seen that among the three methods, $A2$ was successful on 3 out of 4 problems (see the MIN of Table 1). In addition $A2$ achieved the best maximum, average and standard deviation on two problems.

Table 1: Comparison of the Evolved Algorithms and the Existing Method

	No.	*RNTS*	*A1*	*A2*
MIN	1	195511	**195509**	**195509**
	2	**133373**	133381	133374
	3	58194	58202	**58193**
	4	289316	289319	**289310**
MAX	1	195536	195551	**195529**
	2	133392	133403	**133388**
	3	**58210**	58222	58230
	4	**289398**	289433	289410
MEAN	1	195519.7	195527.4	**195518.4**
	2	133383.0	133392.2	**133381.5**
	3	**58202.4**	58212.4	58208.4
	4	**289342.0**	289380.6	289347.7
STDEV	1	5.9	9.0	**5.7**
	2	4.6	5.7	**3.5**
	3	**3.8**	4.657092	8.6
	4	**19.4**	23.6	24.18

5. CONCLUSION

This paper presents a novel grammar guided GP method for self-adaptive Tabu Search. The preliminary experimental results shows that it can outperform human designed RNTS method for some of the test problems.

6. REFERENCES

[1] Jinxiang Gu, Marc Goetschalckx, and Leon F McGinnis. Research on warehouse operation: A comprehensive review. *European Journal of Operational Research*, 177(1):1–21, 2007.

[2] EA Frazele and Gunter P Sharp. Correlated assignment strategy can improve any order-picking operation. *Industrial Engineering*, 21(4):33–37, 1989.

[3] Jing Xie, Yi Mei, Andreas T Ernst, Xiaodong Li, and Andy Song. A genetic programming-based hyper-heuristic approach for storage location assignment problem. In *Evolutionary Computation (CEC), 2014 IEEE Congress on*, pages 3000–3007. IEEE, 2014.

[4] Jing Xie, Yi Mei, Andreas T Ernst, Xiaodong Li, and Andy Song. Scaling up solutions to storage location assignment problems by genetic programming. In *Simulated Evolution and Learning*, pages 691–702. Springer, 2014.

[5] Jing Xie, Yi Mei, Andreas T Ernst, Xiaodong Li, and Andy Song. A restricted neighbourhood tabu search for storage location assignment problem. In *Evolutionary Computation (CEC), 2015 IEEE Congress on*. IEEE, 2015.

[6] Fred Glover. Tabu search-part i. *ORSA Journal on computing*, 1(3):190–206, 1989.