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

This study proposes 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 and it can be used as solvers for SLAPs. A novel self-adaptive Tabu Search framework is proposed that key configurations of the algorithm are determined based on the problem-specific characters, and these configurations are changed dynamically during the search process. In addition, both the quality of the solutions and the execution speed are considered in the evaluation function. The experimental results show that more efficient Tabu Search algorithms can be found by this approach comparing to a 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 closer to each other [2].

Copyright is held by the owner/author(s).

GECCO '15 Companion, July 11-15, 2015, Madrid, Spain ACM 978-1-4503-3488-4/15/07 http://dx.doi.org/10.1145/2739482.2764896. However, finding a solution 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 that is to evolve Tabu Search algorithms for the SLAP with Grouping Constraints. A novel representation of the grammars is presented. The experimental results indicate that better Tabu algorithms can be found which can outperform a Tabu Search method which has been carefully and manually designed for SLAPs.

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, specifically, the total distance for pickers to travel between these locations and the loading dock. Items of a same product may have significantly different demand. Items of the same product are preferred to be placed in adjacent locations, while assigning items strictly according to demand may lead to an unreasonable arrangement. 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 hyperheuristic 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. In 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

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

```
::= Init() Repeat Unitl(<do>)
1<start>
             ::= if(<condition>, <do>, <do>) | <tabu>
2 <do>
3 <condition> ::= true | false | leq(curIter, <threshold>)
4 <tabu>
             ::= set(<num>, <num>, <threshold>)
                 search() updateTabuParams()
5
                 (2) (4) (6) (8) (16)
6 < num>
             ::=
                 (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

setting of the algorithm remains the same throughout the whole searching 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 that the algorithm can adapt to them. For example, the diversity of the visited solutions is more significant than other factors at the beginning of the search, while 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 equal to the tabu tenure (the iterations a solution can be kept in the tabu list). The second parameter 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.

The evaluation 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 [5]. 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.

5. CONCLUSION

This paper presents a novel grammar guided GP method

he 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

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

for evolving self-adaptive Tabu Search algorithms. The preliminary experiment shows that it can outperform human designed RNTS method for some of the test problems.

23.6

24.18

19.4

6. **REFERENCES**

4

- 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.