A Surrogate-assisted Tabu Search Algorithm for Large-scale Storage Location Assignment Problems with Grouping Constrains

Shengran Hu

Abstract—**The storage location assignment problem with grouping constraints is a widespread problem in the field of logistics and warehousing, which can be represented as a bi-level optimization problem to be solved. However, since the solution of the lower-level optimization problem is based on local search, the time overhead for solving the problem increases rapidly with the problem size. To solve this problem, this paper proposes the Surrogate-assisted Tabu Search (SA-Tabu) algorithm, which achieves the improvement of sampling efficiency by evaluating the neighborhood of the solution with the assistance of an surrogate model. At the same time, this method can be conceptually applied to other expensive optimization problems. Comparison experiments with the traditional Tabu Search Algorithm in large-scale Storage Location Assignment Problems show that SA-Tabu can significantly accelerate the convergence of the search without affecting the convergence of the algorithm to the global optimal solution. Finally, an ablation study for the agent model further illustrates the reliability and rationality of the SA-Tabu algorithm.**

Index Terms—**Surrogate Model, Tabu Search, Storage Location Assignment Problem, Grouping Constraint**

I. Introduction

In industrial scenarios of logistics and warehousing, the storage location assignment problem with grouping constraints is a common and important optimization problem. [1] represents such problems as bi-level optimization problems, which are solved by applying the Tabu Search Algorithm. However, in practical industrial logistics and warehousing problems, the size of such problems is often large. Since the solution of the lower-level optimization problem is based on local search, the time overhead of the problem will increase rapidly with scale. In [1], the actual warehousing problem with 10,000 Items requires 70 hours of running on the server to solve.

To address this problem, the Surrogate-assisted Tabu Search (SA-Tabu) algorithm is proposed in this paper, which achieves the improvement of sampling efficiency by evaluating the neighborhood of the solution with the assistance of a surrogate model. Comparison experiments on our randomly generated large-scale test cases containing 3000 and 5000 Items show that the SA-Tabu algorithm can significantly accelerate the convergence of the search compared to the conventional Tabu search algorithm without affecting the convergence of the algorithm to the global optimal solution.

Shengran Hu was with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, Guangdong, 518055 China e-mail: 11711011@mail.sustech.edu.cn.

Manuscript submitted May 6, 2021.

The main contributions of this paper are summarized as follows.

- In this paper, we introduce a surrogate model to the Tabu Search Algorithm framework and propose the SA-Tabu algorithm, which enables the sampling efficiency of the search to be improved.
- We further test the SA-Tabu algorithm on a large-scale test sample of the storage location assignment problem with grouping restrictions. The experiments show that the SA-Tabu algorithm can significantly accelerate the convergence of the search without affecting the convergence to the global optimal solution compared with the traditional Tabu search algorithm.
- The ablation studies of the surrogate model proves the superiority of our chosen RBF model for this problem and illustrates the reliability of the surrogate model for the prediction of the solution.

The remainder of the paper is organized as follows: in Section II, the procedure of the SA-Tabu algorithm is presented; in Section III, it is tested on a large-scale sample of the storage location assignment problem with grouping constraints, along with an ablation study of the surrogate model; in Section IV, possible future work of the paper is described; and finally, a summary of the paper is given in Section V.

II. The Proposed Method

The storage location assignment problem with grouping constraints can be represented as a bi-level optimization problem [1],

$$
\begin{aligned}
\underset{G}{\text{minimize}} \quad & F(G,S) = \phi(S) \\
\text{subject to} \quad & S \in \text{argmin } \phi(S')
\end{aligned}
$$

where $F(G,S)$ is the objective function, $\phi(S)$ is the weighted distance of all goods to the pickup point, G is the upper variable that determines the grouping, S is the lower variable that indicates the location of the goods, and S' is the valid location of the goods under the condition of G.

For the solving of the lower-level problem, a local search method is used [1]. This also leads to the fact that the solution of the lower-level problem becomes expensive in large-scale problems.

Compared with the conventional Tabu algorithm that evaluates all neighborhoods [1], the SA-Tabu algorithm proposed

Algorithm 1: Surrogate-assisted Tabu Search

Input : Max number of evaluation $maxFEs$
Output: Best found grouping G^* and corresponding lower level solution S^*.

$nbEvaluate \leftarrow 0$;
Initialize best solution G^*, S^* to random generated solution;
$G_{next} \leftarrow G^*$;
Initialize $tabuList \leftarrow \emptyset$;
while $nbEvaluate < maxFEs$ **do**
  $\mathcal{N} \leftarrow$ neighbor(G_{next});
  $\mathcal{N}_{evaluated} \leftarrow$ randomly choose $r_1\%$ of $\mathcal{N}$ to be evaluated;
  $Model \leftarrow$ bulid local surrogate model based on $\mathcal{N}_{evaluated}$;
  $\mathcal{N}_{predicted} \leftarrow r_2\%$ of neighbors from $\mathcal{N} \setminus \mathcal{N}_{evaluated}$ with best prediction with $Model$;
  Evaluate $\mathcal{N}_{predicted}$ and add them into $\mathcal{N}_{evaluated}$;
  Choose best non-tabu neighbor from $\mathcal{N}_{evaluated}$ as G_{next} and update G^*, S^*;
  Update $tabuList, nbEvaluate$;
end
return G^*, S^*.

in this paper predicts the status of the overall neighborhood from the partially evaluated neighborhoods, and the algorithm procedure is shown in Algorithm 1. Specifically, SA-Tabu builds the local surrogate model [2] for the unevaluated neighborhoods after real evaluation of $r_1\%$ neighborhoods, and subsequently the neighborhoods with the best prediction performance of $r_2\%$ are truthfully evaluated with local search. Finally, among all truthfully evaluated individuals, the best non-tabu solution is selected as the individual for the next iteration.

III. Experimental Results and Discussion

A. Benchmark Problems and Algorithm Configuration

To test the performance of the algorithm proposed in this paper on the problem, we randomly generated several large-scale test problems. The statistics of these problems are presented by Table. I. In [1], the Large-scale dataset contains 900 items, while the large-scale dataset we generated contains 2520 and 5040 items. The rules for randomly generated data are designed based on the statistics in [1] for real-world industrial data.

The parameters used in the experiments are shown by Table. II, and the rest of the algorithm parameters are consistent with [1].

The final test results and statistical tests are shown by Table. III. Fig. 1-4 shows the convergence curves on the four test instances. The experiments use # of evaluation as the metrics of algorithm execution cost, because the time consumption of all steps in the algorithm except evaluation is negligible, including the training and querying of the surrogate model. From the experimental results, SA-Tabu significantly accelerates the convergence of the search compared to the

TABLE I
Statistic on generated benchmarking instances.

Instance	# of items	# of products	# of shelves	Size of shelves
1 2	2520	63 66	18	140
3 4	5040	127 128	36	140

TABLE II
Parameter Settings for SA-Tabu.

Parameter	Value
Proportion of neighbors got evaluated to build surrogate $r_1\%$	0.1
Proportion of neighbors got evaluate after prediction $r_2\%$	0.05
Which kind of surrogate model is used	RBF

baseline algorithm without degrading the search performance at a higher number of evaluations.

B. Ablation study on the choice of surrogate models.

We conducted an ablation study with the choice of the surrogate model. We chose to use RBF and other common surrogate models for comparison. The surrogate models were randomly sampled throughout the decision space and trained using ten-fold cross-validation, and the results are shown by Fig. 5. The horizontal axis is the number of samples and the vertical axis is the Spearmanr rank-order correlation coefficient, which characterizes the positive correlation in ranking between the predicted and true values of the surrogate model. From the experiments, it is clear that RBF has better performance on this problem for different number of samples compared to other surrogate models. The [3] in shows that RBF has better performance on high dimensional space, and ours is also based on the higher dimensionality of decision variables in our problem to choose RBF as a proxy model. Also, in the construction of local surrogate, the typical sample size is

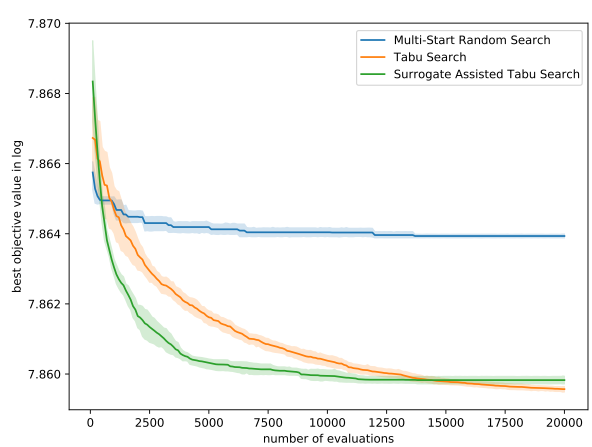

Fig. 1. The comparison on the speed of convergence between alrgoithm for instance 1 with 2050 items.

TABLE III

Experimental results on generated large-scale benchmark problems. Each experiment repeats for 5 times. The Wilcoxon signed ranks test (α= 0.1) is applied between SA-Tabu and other two comparison algorithms.

Instance	# of items	# of evaluations	Multi-Start Random Search mean	Multi-Start Random Search std	Tabu Search mean	Tabu Search std	SA-Tabu mean	SA-Tabu std
1	2520	5000	7.3146e+07 ($-$)	3.0574e+04	7.2714e+07 ($-$)	3.9104e+04	7.2497e+07	2.9456e+04
2			7.1563e+07 ($-$)	4.7324e+04	7.1151e+07 ($-$)	1.0735e+05	7.0752e+07	2.3472e+04
1	2520	10000	7.3121e+07 ($-$)	2.4234e+04	7.2507e+07 ($-$)	3.0500e+04	7.2434e+07	2.9785e+04
2			7.1502e+07 ($-$)	2.4118e+04	7.0866e+07 ($-$)	3.4453e+04	7.0742e+07	2.2291e+04
1	2520	20000	7.3103e+07 ($-$)	9.9827e+03	7.2372e+07 ($+$)	1.3728e+04	7.2415e+07	2.0840e+04
2			7.1456e+07 ($-$)	4.1320e+04	7.0709e+07 ($\approx$)	4.5713e+03	7.0721e+07	1.3751e+04
3	5040	10000	1.6496e+08 ($-$)	4.8060e+04	1.6426e+08 ($-$)	1.2167e+05	1.6311e+08	3.9934e+04
4			1.6539e+08 ($-$)	6.9521e+04	1.6469e+08 ($-$)	3.2122e+05	1.6352e+08	3.7600e+04
3	5040	25000	1.6494e+08 ($-$)	7.0755e+04	1.6360e+08 ($-$)	5.8044e+04	1.6301e+08	3.1013e+04
4			1.6536e+08 ($-$)	7.41E+04	1.6398e+08 ($-$)	1.1026e+05	1.6339e+08	1.6394e+04
3	5040	50000	1.6494e+08 ($-$)	7.0755e+04	1.6322e+08 ($-$)	3.4146e+04	1.6301e+08	3.1013e+04
4			1.6536e+08 ($-$)	7.4055e+04	1.6363e+08 ($-$)	8.3067e+04	1.6339e+08	1.6394e+04

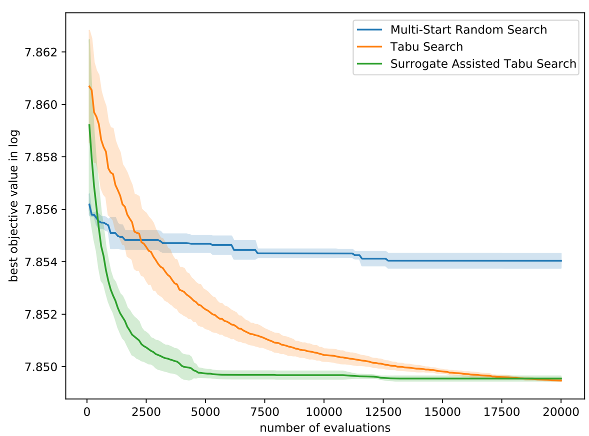

Fig. 2. The comparison on the speed of convergence between alrgoithm for instance 2 with 2050 items.

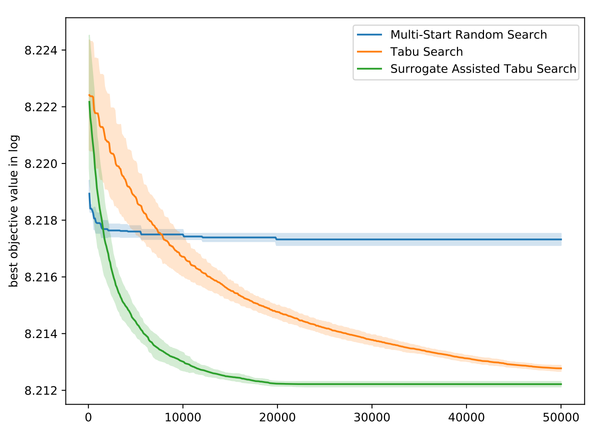

Fig. 3. The comparison on the speed of convergence between alrgoithm for instance 3 with 5040 items.

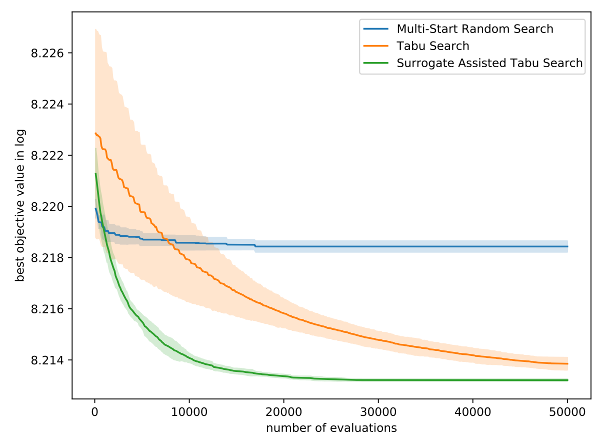

Fig. 4. The comparison on the speed of convergence between alrgoithm for instance 4 with 5040 items.

about 100 to 300. As shown by Fig. 5, the global surrogate model already has better predictive power with samples in this interval. In SA-Tabu, the local surrogate is established, i.e., the sample has a strong localization, when the reliability of the model may be further improved [2]. The implementation of the surrogate model used in the experiments uses a machine learning framework in Python [4] and Java [5].

IV. Future Work

In this paper, the settings of the two scaling parameters used by the surrogate model in exploring the neighborhood were not validated. Future work should test for the sensitivity of these two parameters.

V. Conclusion

To solve the large-scale storage location assignment problem with grouping restrictions, the Surrogate-assisted Tabu Search (SA-Tabu) algorithm is proposed in this paper. The method achieves the improvement of sampling efficiency by evaluating the neighborhood of the solution with the assistance

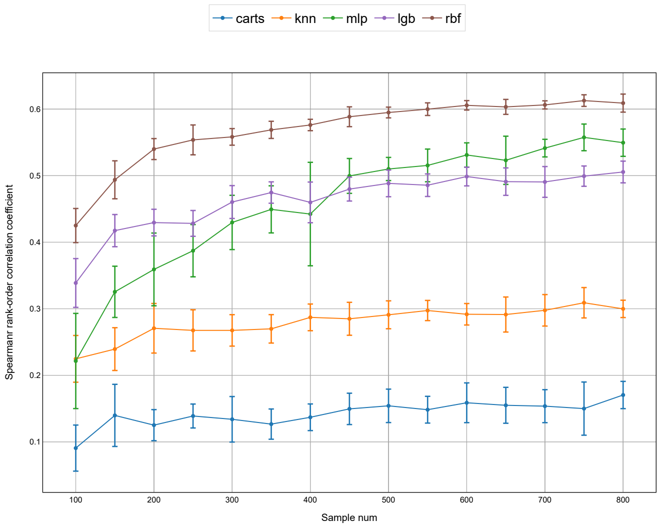

Fig. 5. The ablation study on the choice of surrogate models. The experiment conducts on instance 1 with 2520 items and 63 products

of an agent model. At the same time, this method can be conceptually applied to other expensive optimization problems. Comparison experiments with the conventional Tabu Search Algorithm in Large-scale Storage Location Assignment Problems show that SA-Tabu can significantly accelerate the convergence of the search without affecting the convergence of the algorithm to the global optimal solution. Finally, an ablation study for the surrogate model further illustrates the reliability and rationality of the SA-Tabu algorithm.

Appendix A
Guideline for Running the Code

A. Environment Setup

- Config **JDK 8** (8u291).
- Build maven environment based on **pom.xml**.
- Import **OpenTS.jar** to current project dependency.

B. Results Reproduction and Instances Generation

- To reproduce the search with SA-Tabu on instances given in report: Run **SATabu.java [instanceFileName]**. Notice: the output would be redirect to a file.
- To generate new instances: Run **DataGenerator.java** after modifying variables # of items **N** and # of samples **sampleNum** in DataGenerator.java.

References

[1] J. Xie, Y. Mei, A. T. Ernst, X. Li, and A. Song, "A bi-level optimization model for grouping constrained storage location assignment problems," *IEEE transactions on cybernetics*, vol. 48, no. 1, pp. 385–398, 2016.

[2] X. Wang, G. G. Wang, B. Song, P. Wang, and Y. Wang, "A novel evolutionary sampling assisted optimization method for high-dimensional expensive problems," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 815–827, 2019.

[3] A. Díaz-Manríquez, G. Toscano-Pulido, and W. Gómez-Flores, "On the selection of surrogate models in evolutionary optimization algorithms," in *Proceedings of IEEE congress of evolutionary computation*, 2011, pp. 2155–2162.

[4] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[5] H. Li, "Smile," https://haifengl.github.io, 2014.