ISSN 2658-5014 (Print)
2024. Vol. 6, No. 3 (18). Pp. 55–60

SIIT
SYSTEMS ENGINEERING & INFORMATION TECHNOLOGIES
Scientific message

ISSN 2686-7044 (Online)
http://siit.ugatu.su

UDC 004.65 DOI 10.54708/2658-5014-SIIT-2024-no3-p55 EDN XGGYUO

Exponential neighborhood for Bin Packing Problem

A. R. Usmanova • Yu. I. Valiakhmetova

Abstract. The article is devoted to the well-known problem of one-dimensional packing – Bin Packing Problem (BPP). The Bin Packing Problem can be found widely in different branches of industry and technique. BPP is NP-hard, so the set of solutions has exponential cardinality in relation to packed items. The authors consider modified model of problem – in fact, authors solve the Flow-shop scheduling problem. The goal is to vanish so called total overfilling (TO) – the sum of differences between the bin capacity and weights of matched items to each bin. The different methods using polynomial neighborhoods requires a lot of time. The authors offer exponential neighborhood that require polynomial time to find the best solution. The linear Assignment Problem is considered to construct an exponential neighborhood. Despite the fact that there are n! solutions, the optimal solution can be found in O(n3). The authors consider several algorithms for constructing an exponential neighborhood. The main idea is to remove one item from each container in some feasible solution. And then we should reassign such unpacked items to used containers so that TO will be minimal. However, the proposed method of constructing exponential solutions does not allow you to directly change the number of items in the bin. Therefore, it is advisable to combine the search in exponential neighborhood with some strategies that allow you to change the number of items associated with each container. The results of a numerical experiment are comparing the search in polynomial neighborhoods and the proposed exponential one.

Key words: Bin Pacing Problem; exponential neighborhood; optimization; local optimum, Assignment Problem.

Introduction

Let us describe the Bin Packing Problem (BPP): the set $L = \{w_1, w_2, \ldots, w_n\}$ of nonnegative weights of items and positive number C – the bin capacity. It is necessary to find such partition L into the minimal number of disjoint subsets, that the sum of weights in each subset does not exceed the bin capacity C. Let us formulate one of mathematical definitions of BPP. We are given the set $L = \{w_1, w_2, \ldots, w_n\}$ of item weights and n bins of capacity C. Let us assign each item to one and only one bin, i.e. pack item to bin, so that the total weight packed in any bin does not exceed the capacity, and the number of used bins will be minimal. We suppose that the bin containing at least one item is used, otherwise bin is not used. Now let us introduce two binary vectors. Let $y_i \in \{0,1\}$, $i \in N = \{1, 2, ..., n\}$, where

$$y_j = \begin{cases} 1, & \text{if } j\text{-th bin is used,} \\ 0, & \text{otherwise,} \end{cases}$$

and let $x_{ij} \in \{0,1\}$ $i, j \in N$, where

$$x_{jj} = \begin{cases} 1, & \text{if } i\text{-th item is packed into bin } j, \\ 0, & \text{otherwise.} \end{cases}$$

Then we can formulate BPP as Integer Linear Program (ILP):
minimize $z = \sum_{j=1}^{n} y_j$ with respect to

$$\sum_{i=1}^{n} y_i x_{ij} w_i \le C, j \in N \quad (1)$$

and $\sum_{i=1}^{n} x_{ij} = 1, j \in N$.

In this paper we use modified model of problem. We pack items to fixed number of containers and try to converse a packing plan so that condition (1) is satisfied. Let introduce the value total overfilling (TO) by the next way:

Note the weight of each container as S_j, $j = 1, \ldots, m$, where m is number of used containers. Set values t_j, $j = 1, \ldots, m$, to characterize numerically the overfilling of containers:

$$t_j = \begin{cases} S_j - C, & \text{if } C < S_j, \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where C is capacity of container. Then total overfilling will be

$$TO = \sum_{j=1}^{m} t_j. \tag{3}$$

Let describe now linear assignment problem (LAP). The assignment problem is a special case of a transportation problem. Its name arose from the following interpretation: there are m positions and m applicants for these positions. The appointment of the i-th applicant to the j-th position leads to losses $c[i, j]$. It is required to distribute applicants by position so that the total loss is minimal. This problem is equivalent to the problem of finding the maximum matching of the minimum weight in a bipartite graph.

Let us formulate mathematical model. Given the matrix of integer nonnegative numbers $c[i, j]$, $i, j \in \{1, \ldots, m\}$. It is required to choose m items – one in each row and each column, so that their sum is minimal.

LAP

Find vector $x = (x_1, \ldots, x_m)$, (4)

where $x \in \{1, \ldots, m\}$ (5)

to minimize function $\sum_{i=1}^{n} C_{ix_i}$ (6)

and satisfying conditions $x_i \neq x_j$ if $i \neq j, i, j \in \{1, \ldots, m\}$. (7)

Vector (4) is called assignment, vector (4) satisfying condition (7) is called feasible assignment.

The solution algorithm using the described model one can find, for example, at [1, 2]. The main idea is to converse vector (4) so that at each step the condition (7) keeps, and function (3) improves. Another widespread method of solving this problem is the Hungarian method [3, 4]. It uses a slightly different formulation of the problem, in particular, the solution is not a vector, but a binary matrix consisting of zeros and ones, and having exactly one unit in each row and each column. To find the optimal solution, identical steps are also performed, at each of which the loss matrix changes in a certain way. At each step the same number is added to all the elements of any row or column of the matrix. Since exactly one element from this row (or column) must be selected in a feasible solution, the value of the objective function is changed by the same number. The time complexity of Hungarian method is $O(n^3)$.

EXPONENTIAL NEIGHBORHOOD CONSTRUCTION

Suppose there is some feasible solution to the BPP problem. It is proposed to build an exponential neighborhood by removing one item from each container. Then, solving the assignment problem, match one of the unpacked items to each container again. The set of solutions – neighborhood has cardinality $n!$, because we have n unpacked items and n containers. As you know, the factorial can be approximated by an exponential function, so we called such neighborhood exponential.

The use of the assignment problem was proposed earlier [5] for the TSP problem – "the traveling salesman problem". However, it is impossible to draw a strict analogy between this task and the BPP task, and the development of an algorithm for the packaging problem presented a significant difficulty.

However, unlike the polynomial neighborhoods discussed in [6, 7], the proposed method of constructing exponential solutions does not allow you to directly change the number of items in the container. Therefore, it is advisable to combine the search in exponential neighborhood with some strategies that allow you to change the number of items associated with each container. And although it is possible to propose as an independent heuristic algorithm a method similar to the local descent algorithm described in [6], which would build exponential neighborhoods for various initial solutions and find local optima in them, a combination of search in both exponential and polynomial neighborhoods is assumed to be more effective.

So, we some initial packing plan. Then we should select one item from each container. Each of these items must be re-matched to one of the containers. The loss matrix is calculated according to the following rule

$$c[i,j] = \begin{cases} C - (S_j + w_i), \text{ if } (S_j + w_i) > 0, \\ 0, \text{ if } (S_j + w_i) \le 0, \end{cases} \quad (8)$$

where C is capacity of container, S_j is total weight of items packed to j-th container (before matching), w_i is weight of matched item.

This means that the loss from assigning the i-th item to the j-th container is the difference between the weight of the container together with the assigned item and the container capacity, that is, the amount of overflow of the container. If there is no overflow, then the corresponding element of the matrix is zero. After that, the assignment problem is solved, minimizing the total overflow across all containers. It is clear that the resulting solution will be no worse than the initial one, since in the absence of a better one, the same initial solution will turn out.

The main question is how exactly to choose the items that will be reassigned. Next, three algorithms are considered.

ALGORITHMS OF CONSTRUCTING EXPONENTIAL NEIGHBORHOOD

The easiest way is to select each item from the container with an equal probability equal to $1/d_j$, where d_j is the quantity of items in the j-th container. This algorithm is most effective if the packing plan is far from the local optimum. In this case, it makes it possible to obtain a solution with a small value of total overfilling (7) much faster than when search in polynomial neighborhoods [6]. Here is a description of the RI (Random Item) algorithm.

Algorithm RI

1. For each j-th container ($j = 1, \dots, m$) generate random integer number $r(j)$, taking with an equal probability value in range $1, \dots, d(j)$; where $d(j)$ is quantity of items matched to j-th container.
2. Remove $r(j)$-th item from j-th container, write its weight is $w[j]$.
3. Calculate loss matrix $c[i, j]$ according to formula (8).
4. Solve assignment problem, get optimal solution vector x.
5. Repack to each j-th container item with index $x[j]$ ($j = 1, \dots, m$).

The time complexity of this algorithm is $O(m)$ – for choosing removed items and as mentioned above $O(m^3)$ – for solving assignment problem. Since the procedures for removing and assigning items are performed sequentially, the algorithm as a whole has cubic complexity.

The next algorithm removes an item with the maximum weight from each container. Let's call this algorithm the MI (Maximal Item) algorithm. The difference from the described RI algorithm will be that in steps 1–2, an item with the maximum weight is removed from each container, after which steps 3–5 are performed similarly to the steps of the RI algorithm. The complexity of the MI algorithm is the same as that of the RI – $O(m^3)$ algorithm. This algorithm assumes a situation of local optimum; often in this case, overflowing containers contain two rather heavy items of similar weight. Moreover, in local optima, the overflow of each container is usually small – 1–2 units. That is, it makes sense to try to replace one of these items in overflowing containers with an item only slightly lighter, this item should also be contained among heavy items.

The third Chain algorithm (CI) removes items from containers in such a way that the weights of these items form the maximum length of the chain of decreasing weight items. That is, from the first container (containers are sorted by weight) we remove the heaviest object from the second, the heaviest of the objects not exceeding the weight of the first, and so on. If the next container does not contain an item with a weight less than the last item removed, then the heaviest item is removed from it and the chain is built anew.

The basis for such an algorithm is the following consideration: assuming the local optimum situation described above, it is desirable to put an object with a weight slightly less than the remote one in each container. Namely, in the first container – the second deleted item, in the second container – the third item, and so on, and in the last container from the chain – the first deleted item from the chain. The difficulty of selecting the object to be deleted can be considered linear from the number of containers, although the number of operations here will be greater than for the two previous algorithms. For each container, it is necessary to search for the item needed for the chain among all the items in the container. But the number of items in a container does not depend on the number of all items, nor on the number of containers, and in practice does not exceed 5–6 items.

Table

Comparison of exponential neighborhood constructing algorithms

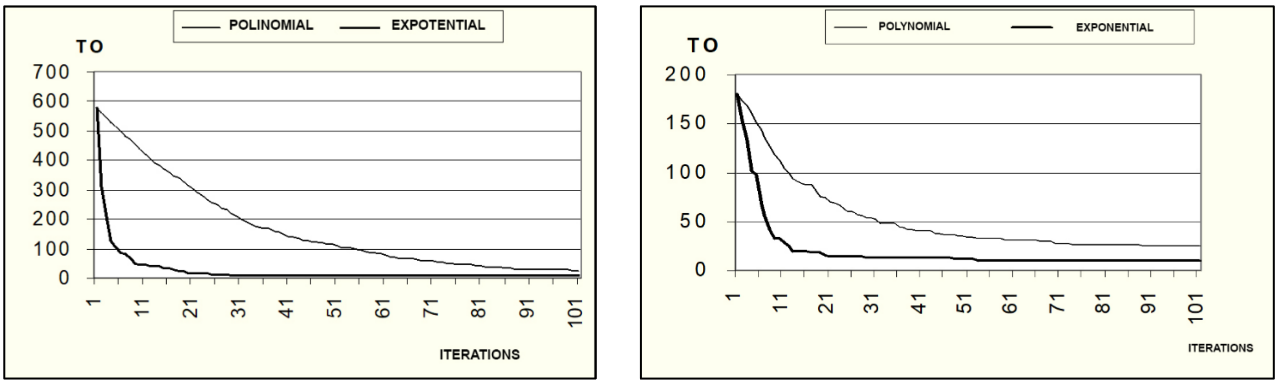
Initial TO	168	68	16
RI	21	10	13
MI	56	25	15
CI	39	18	16

Each of the three described algorithms for constructing an exponential neighborhood has its advantages and disadvantages. Table shows the results of the proposed algorithms. Three initial situations were considered — immediately after obtaining the initial solution (TO =168), after 50 iterations of the polynomial neighborhood search (TO = 68) and after 100 iterations of the polynomial neighborhood search (TO = 16, the situation is close to the local optimum). The table shows the values of the final TO after 100 steps in the exponential neighborhood for each of the algorithms. The experiment was performed for benchmarks proposed at [8], task 1, set 2.

As can be seen from the table above, the best results are obtained by the first of the proposed algorithms – RI, in which objects are removed randomly. In the future, it is proposed to use this algorithm as the most effective. However, it seems necessary to conduct further research in this direction.

Comparison of exponential and polynomial neighborhoods

Now let compares the search in polynomial neighborhood with swapping items [6–8] and in proposed here exponential neighborhood. We use only RI algorithm described above. The benchmarks are described at [9]. Now we show task 1 from test set 2–250 uniform items – at Fig. 1, *a*, and task 1 from set 6–120 triplet items at Fig. 1, *b*.



a *b*

Fig. 1 Search comparison in polynomial and exponential neighborhoods.
Task 1: *a* – set 2; *b* – set 6.

As can be seen from the diagrams above, when searching for an exponential neighborhood, there is a faster descent than when searching for polynomial neighborhoods. But the time costs searching in an exponential neighborhood is certainly higher than in a polynomial one, because polynomial search has quadratic complexity. In the following, it will be described how to use exponential neighborhood search most effectively as an integral part of the taboo search method.

CONCLUSION

At this paper an exponential neighborhood for BPP is proposed. Three algorithms of constructing exponential neighborhood are considered. The main advantage of this neighborhood is that search of optimal solution in it has polynomial complexity (cubic) despite on the cardinality of neighborhood is exponential. The results of the numerical experiment demonstrate the advantages of searching in an exponential neighborhood in comparison with polynomial ones. There are also certain disadvantages of this approach – the inability to change the number of items in the container and a fairly high calculation time. So, this neighborhood is recommended to use in combination with searching in polynomial neighborhoods, as well as in cases where a rapid descent to local (or global) optima is required.

СПИСОК ЛИТЕРАТУРЫ / REFERENCES

1. Гимади Э. Х., Глебов Н. И. Дискретные экстремальные задачи принятия решений. Новосибирск, 1991. С. 29-33. [[Gimadi E. Kh., Glebov N. I. Discrete Extremal Decision-Making Problems. Novosibirsk, 1991, pp. 29-33. (In Russian).]]
2. Лелякова Л. В., Харитонова А. Г., Чернышова Г. Д. Прикладные задачи о назначениях (модели, алгоритмы решения) // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. 2017. № 2. С. 22-27. EDN ZDWOKF. [[Lelyakova L. V., Kharitonova A. G., Chernyshova G. D. "Applied assignment problems (models, solution algorithms)" // Bulletin of Voronezh State University. Series: System analysis and information technologies. 2017. No. 2, pp. 22-27. EDN ZDWOKF. (In Russian).]]
3. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. М.: Мир, 1985. [[Papadimitriou H., Steiglitz K. Combinatorial Optimization. Moscow: Mir, 1985. (In Russian).]]
4. Романовский И. В. Алгоритмы решения экстремальных задач. М.: Наука, 1977. [[Romanovsky I. V. Algorithms for Solving Extremal Problems. Moscow: Nauka, 1977. (In Russian).]]
5. Gutin G. "Exponential neighborhood local search for the traveling salesman problem" // Comput. Oper. Res. 1999. Vol. 26. Pp. 313-320.
6. Usmanova A. R., Zemlyanov A. P. The local search algorithm in polynomial neighborhoods for the linear packing problem // CSIT'2016: Proceedings of the 18th International Workshop on Computer Science and Information Technologies. Czech Republic. Prague. Kunovice. Sept. 26-30, 2016. Vol. 1. Prague: Kunovice, 2016. Pp. 138-143. EDN XCLFXZ.
7. Усманова А. Р., Валиахметова Ю. И. Особенности метода поиска с запретами для задачи упаковки // СИИТ. 2022. Т. 4. № 2(9). С. 37-42. DOI 10.54708/26585014_2022_42937. EDN NASXQA. [[Usmanova A. R., Valiakhmetova Yu. I. "Features of the tabu search method for the packing problem" // SIIT. 2022. V. 4, No. 2(9), pp. 37-42. DOI 10.54708/26585014_2022_42937. EDN NASXQA. (In Russian).]]
8. Усманова А. Р., Валиахметова Ю. И. Особенности задач-триплетов в задаче упаковки // СИИТ. 2023. Т. 5. № 1(10). С. 34-40. DOI 10.54708/2658-5014-SIIT-2023-no1-p34. EDN VSYIOY. [[Usmanova A. R., Valiakhmetova Yu. I. "Features of triplet problems in the packing problem" // SIIT. 2023. V. 5, No. 1(10), pp. 34-40. DOI 10.54708/2658-5014-SIIT-2023-no1-p34. EDN VSYIOY. (In Russian).]]
9. Falkenauer E. "A hybrid grouping genetic algorithm for bin packing" // Journal of Heuristics. 1996. Vol. 2. Pp. 5-30.

Поступила в редакцию 8 июля 2024 г.

МЕТАДАННЫЕ / METADATA

Заглавие: Экспоненциальная окрестность для задачи упаковки контейнеров.

Аннотация: Статья посвящена известной задаче одномерной упаковки – Bin Packing Task (BPP). Проблема упаковки контейнеров широко распространена в различных отраслях промышленности и техники. BPP является NP-сложным, поэтому множество решений имеет экспоненциальную мощность по отношению к упакованным элементам. Авторы рассматривают модифицированную модель задачи – по сути, решают задачу планирования цеха. Цель состоит в том, чтобы устранить так называемое общее переполнение (ТО) – сумму разностей между емкостью корзины и весом совпадающих предметов в каждой корзине. Различные методы, использующие полиномиальные окрестности, требуют много времени. Авторы предлагают экспоненциальную окрестность, требующую полиномиального времени для поиска лучшего решения. Рассматривается линейная задача о назначениях для построения экспоненциальной окрестности. Несмотря на то, что есть n! решений, оптимальное решение можно найти за O(n3). Авторы рассматривают несколько алгоритмов построения экспоненциальной окрестности. Основная идея состоит в том, чтобы удалить по одному элементу из каждого контейнера в некотором осуществимом решении. И тогда нам следует такие распакованные предметы переназначить в использованные контейнеры, чтобы ТО было минимальным. Однако предлагаемый метод построения показательных решений не позволяет напрямую изменять количество предметов в корзине. Поэтому желательно сочетать поиск в экспоненциальной окрестности с некоторыми стратегиями, позволяющими изменять количество элементов, связанных с каждым контейнером. Результаты численного эксперимента сравнивают поиск в полиномиальных окрестностях и предложенный экспоненциальный.

Ключевые слова: проблема размещения контейнеров; экспоненциальная окрестность; оптимизация; локальный оптимум, задача о назначениях.

Язык статьи / Language: английский / English.

Об авторах / About the authors:

УСМАНОВА Анжелика Рашитовна
ФГБОУ ВО «Уфимский университет науки и технологий», Россия.
Доцент каф. вычислительной математики и кибернетики. Дипл. инж.-программист (Уфимск. гос. авиац. техн. ун-т, 1997). Канд. физ.-мат. наук (Башкирск. гос. ун-т, 2002). Иссл. в обл. эврист. алгоритмов задач дискр. оптимизации.
E-mail: kfmn2004@mail.ru
ORCID: https://orcid.org/0000-0002-9306-8826
URL: https://elibrary.ru/author_items.asp?authorid=132357

ВАЛИАХМЕТОВА Юлия Ильясовна
ФГБОУ ВО «Уфимский университет науки и технологий», Россия.
Доц. каф. вычислительной математики и кибернетики. Дипл. инж. (Уфимск. гос. авиац. техн. ун-т, 2004). Канд. техн. наук по мат. моделированию, числ. методам и комплексам программ (там же, 2008). Иссл. в обл. эврист. алгоритмов решения задач дискретной оптимизации.
E-mail: julikas@inbox.ru
ORCID: https://orcid.org/0000-0003-3897-118x
URL: https://elibrary.ru/author_items.asp?authorid=734538

USMANOVA Angelika Rashitovna
Ufa University of Science and Technology, Russia.
Ass. Prof., Dept. Computational Mathematics and Cybernetics. Dipl. Eng.-programmer (Ufa State Aviation Technical University, 1997). Candidate of Phys.-Math. Science (Bashkir State University, 2002). Research in discrete optimization.
E-mail: kfmn2004@mail.ru
ORCID: https://orcid.org/0000-0002-9306-8826
URL: https://elibrary.ru/author_items.asp?authorid=132357

VALIAKHMETOVA Juliya Ilyasovna
Ufa University of Science and Technology, Russia.
Ass. Prof., Dept. Computational Mathematics and Cybernetics. Dipl. Engineer (Ufa State Aviation Technical University, 2004). Candidate of Tech. Sci. (Ufa State Aviation Technical University, 2008). Research in the field of heuristic algorithms for discrete optimization.
E-mail: julikas@inbox.ru
ORCID: https://orcid.org/0000-0003-3897-118x
URL: https://elibrary.ru/author_items.asp?authorid=734538