



DOI:10.12404/j.issn.1671-1815.2403295

引用格式:陈皓宇,罗璟,杨灏泉,等.优化装箱填充率的改进鲸鱼模拟退火算法[J].科学技术与工程,2025,25(15):6477-6485.

Chen Haoyu, Luo Jing, Yang Haoquan, et al. Improved whale simulated annealing algorithm for optimizing the fill rate of bin packing[J]. Science Technology and Engineering, 2025, 25(15): 6477-6485.

交通运输

## 优化装箱填充率的改进鲸鱼模拟退火算法

陈皓宇<sup>1</sup>, 罗璟<sup>1\*</sup>, 杨灏泉<sup>2</sup>, 冯仁宇<sup>3</sup>, 袁锐波<sup>1</sup>, 甘雨<sup>4</sup>

(1. 昆明理工大学机电工程学院, 昆明 650500; 2. 云南柔控科技有限公司, 昆明 650031;  
3. 上海烟草集团有限责任公司, 上海 200090; 4. 昆明欧迈科技有限公司, 昆明 650106)

**摘要** 针对多约束、大规模的三维装箱问题,研究建立相应的数学模型,提出了改进鲸鱼算法与模拟退火算法相结合的混合元启发式算法。该算法对鲸鱼算法进行离散化处理,包括个体编码方式及更新机制等,利用模拟退火解决局部最优陷阱,并设计启发式装载规则进行解码,优化装箱方案。利用 Bischoff 和 Ratcliff 教授在 OR-Library 所提供的三维装箱标准算例和实际企业货物订单数据作为仿真案例,涵盖了从弱异构到强异构的货物类型。本研究所提算法同时兼顾了全局搜索和局部搜索能力,在面对多种箱子类型时,具有较高的填充率,其中弱异构型货物平均容器填充率为 92.24%,强异构型货物平均容器填充率为 88.78%,总体平均容器填充率达 91.29%,为三维装箱问题研究提供了一定的借鉴和参考价值。

**关键词** 三维装箱; 改进鲸鱼算法; 模拟退火算法; 启发式装载规则

**中图分类号** U169 TP18; **文献标志码** A

### Improved Whale Simulated Annealing Algorithm for Optimizing the Fill Rate of Bin Packing

CHEN Hao-yu<sup>1</sup>, LUO Jing<sup>1\*</sup>, YANG Hao-quan<sup>2</sup>, FENG Ren-yu<sup>3</sup>, YUAN Rui-bo<sup>1</sup>, GAN Yu<sup>4</sup>

(1. Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650500, China;  
2. Yunnan Soft Control Technology Co., Ltd., Kunming 650031, China; 3. Shanghai Tobacco Group Co., Ltd., Shanghai 200090, China;  
4. Kunming Oumai Technology Co., Ltd., Kunming 650106, China)

**[Abstract]** A mathematical model was established to address the multi-constraint, large-scale three-dimensional bin packing problem. A hybrid metaheuristic algorithm combining an improved whale algorithm with simulated annealing was proposed. The algorithm discretized the whale algorithm, including individual encoding and updating mechanisms, and utilized simulated annealing to overcome local optima traps. Moreover, a heuristic loading rule was designed for decoding and optimizing the packing solution. The algorithm was evaluated using standard packing instances from Bischoff and Ratcliff's OR-Library, as well as real-world cargo order data, covering a range of cargo types from weakly heterogeneous to strongly heterogeneous. The proposed algorithm achieved a balance between global and local search capabilities, resulting in high packing efficiency for various types of containers. Specifically, the average container filling rate were 92.24% for weakly heterogeneous cargo, 88.78% for strongly heterogeneous cargo, and an overall average of 91.29%. This result provides valuable insights and references for the study of three-dimensional bin packing problems.

**[Keywords]** 3D Bin packing; improved whale algorithm; simulated annealing algorithm; heuristic loading rules

随着全球化的进程、互联网的普及以及电子商务的蓬勃发展,物流产业的快速发展,进一步促进了物流运输的需求,加强了物流行业内的竞争和升级换代。目前绝大多数企业的物流装箱模式主要

以人工经验装载为主,这种方法虽然可以解决一定的问题,但局限性也很明显,主要表现在以下几个方面:首先,人工规划难以充分考虑到复杂的运输实际情况,特别是在运输量较大、需求较为复杂的

收稿日期:2024-05-06 修订日期:2024-08-01

基金项目:云南省科技厅重大专项(202202AC080003);云南省科技厅国际科技合作项目(GHJD-2022001)

第一作者:陈皓宇(1999—),男,汉族,重庆涪陵人,硕士研究生。研究方向:生产及制造系统工程。E-mail:674340205@qq.com。

\*通信作者:罗璟(1968—),女,壮族,云南文山山人,硕士,副教授。研究方向:生产及制造系统工程。E-mail:442540985@qq.com。

场景下,人工规划的效率和准确性都会受到很大的影响。其次,传统的装载规划方法难以适应多品种、多批次等多样化的运输需求,无法实现对不同物流需求的智能化处理。最后,传统的装载规划方法缺乏对装载方案的优化,无法做到在保证负荷情况下,最大化利用运输资源、提高装箱效率、减少运输成本。因此,为提高装箱效率,需要研究并设计更高效的装箱算法。

三维装箱问题是一个经典的 NP-hard 组合优化问题,通常描述为如何将不同大小和数量的货物尽可能有效地放入一个或多个三维容器中。该问题在制造业等领域中具有重要应用,可以帮助优化空间利用、减少运输成本等。在三维装箱问题中,通常需要考虑以下几个方面:三维容器的大小和数量;待装箱货物的数量、大小;装箱的约束条件,如是否允许货物重叠、是否需要考虑稳定性等。解决三维装箱问题的目标通常是 minimized 使用容器数量,或者最大化利用容器的空间。这涉及到如何确定货物的放置顺序、位置以及如何有效地填充容器空间等挑战。

国内外学者为了解决三维装箱问题,采用各种启发式算法、元启发式算法、精确算法等方法,如贪婪算法、遗传算法、模拟退火算法等,这些方法在实际应用中能够找到近似最优的解决方案,提高装箱效率,降低成本。George 等<sup>[1]</sup>首次提出了三空间法,在空间中放置一个货物会将剩余空间划分为 3 个新的空间。后来 Bortfeldt 等<sup>[2]</sup>扩展了这一方法,以考虑将空白空间表示为一组不相交空间的其他细分变体,一共有六种划分方式。Parreño 等<sup>[3]</sup>首次提出了最大剩余空间的表示方法。田鑫等<sup>[4]</sup>针对单品种货物的三维装车问题,基于分层的思想,将三维问题降至二维,提出了 9 种分层装车策略及两阶段优化算法;张长勇等<sup>[5]</sup>针对在线式货物码放问题,在考虑多种实际约束基础上,提出将在线极值点与模拟退火算法相结合的在线融合码放算法;张长勇等<sup>[6]</sup>针对多目标、大规模三维装箱模型提出了拟人式装箱算法;周丽等<sup>[7]</sup>针对多箱型多种类货物的三维装箱问题,设计了启发式模块算法,并用遗传算法对装载方案进行优化。尚正阳等<sup>[8]</sup>提出三维剩余空间优化算法,主要针对如何对剩余空间进行分割和货物如何选择最佳姿态两方面进行算法设计;姜凯文<sup>[9]</sup>基于鲸鱼算法对潜艇的多维磁场补偿进行研究,引入非线性权重来改进鲸鱼算法,提高其收敛速度,与标准鲸鱼算法有所提高;张强等<sup>[10]</sup>通过 Sigmoid 函数对鲸鱼算法进行离散化处理,利用基准函数验证其性

能;Zhang 等<sup>[11]</sup>提出了一种离散鲸鱼优化算法来解决以优化目标完工时间为目标的无等待流水线调度问题。

综上,为指导作业人员按科学的堆码方案装箱,提升装箱作业效率,降低运输成本,保护产品质量,并为未来的自动装车机器人做好技术储备,以最大化容器的填充率作为优化目标,创新性的提出一种改进鲸鱼模拟退火算法,解决了传统鲸鱼算法不能很好地处理离散化问题,通过引入精英座头鲸群和轮盘赌概率机制,提高了座头鲸选择的多样性,并相应地修改鲸鱼个体的更新方式,在此基础上加入模拟退火算法,对部分鲸鱼个体进行邻域变换搜索,避免算法陷入局部最优的情况。利用公开数据集和某企业货物订单信息验证了算法的适用性。

## 1 三维装箱数学模型

### 1.1 问题描述

三维装箱问题可以定义为:给定一个容器的基本数据,包含其长、宽、高,分别记为  $L, W, H$ , 一组货物数据  $B$ , 其中  $B$  包含  $n$  种货物, 记为  $B_1, B_2, \dots, B_n$ 。  $B_i$  为每种货物的数据, 包含其尺寸长宽高, 记为  $l_i, w_i, h_i$ , 数量  $\text{num}_i$ 。将这  $n$  种货物, 在满足一系列约束条件下, 尽可能多地放入容器内部, 使得容器的填充率最大, 优化装载方案。

### 1.2 问题假设

由于三维装箱问题较为复杂, 为了方便建模, 现做出如下假设:

(1) 容器和货物均为规则物体, 即长方体, 且货物的尺寸均小于容器尺寸。

(2) 货物间不存在属性冲突, 例如, 化学用品和食物不能放在一起, 本文中所考虑货物都能相邻放置。

(3) 货物都是刚性物体, 不考虑装载、运输过程中的所受的挤压变形。

(4) 容器内的所有空间都可以放置货物, 不存在空间限制。

(5) 上下堆叠货物时, 对货物不会造成损伤。

### 1.3 相关参数说明

以容器长为  $x$  轴、宽为  $y$  轴、高为  $z$  轴, 左后下角点为原点建立三维坐标系, 如图 1 所示。  $L, W, H$  表示容器的长、宽、高;  $l_i, w_i, h_i$  表示放入容器中的第  $i$  个货物的长、宽、高, 共有  $n$  个,  $i = 1, 2, \dots, n$ ;  $R_i$  为第  $i$  个货物所允许旋转的姿态;  $T_i$  为第  $i$  个货物与容器或者底部货物的接触面积;  $T_k$  为第  $i$  个货物的支撑货物  $k$  的上表面面积;  $q_i$  为第  $i$  个货物的重

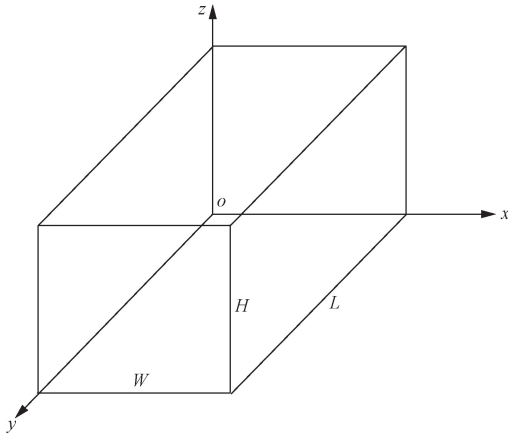


图1 容器三维坐标系

Fig. 1 Three-dimensional coordinate system of the container

量;  $Q$  为容器所能承受的最大重量;  $x_i, y_i, z_i$  为第  $i$  个货物的左后下角点在容器内  $x$  轴、 $y$  轴、 $z$  轴上的坐标值;  $\psi_{x_i}, \psi_{y_i}, \psi_{z_i}$  为第  $i$  个货物在  $xoy, yoz, xoz$  平面上的投影;  $\psi_{x_j}, \psi_{y_j}, \psi_{z_j}$  为第  $j$  个货物在  $xoy, yoz, xoz$  平面上的投影。

1.4 建立模型

(1) 目标函数: 最大化放入容器内部的货物数量, 使容器的填充率达到最大。表达式为

$$F = \max \frac{\sum_{i=1}^n l_i w_i h_i}{LWH} \quad (1)$$

(2) 约束条件: 本文中考虑的约束条件如下。

- ①方向性约束: 在进行放置时, 货物允许旋转的姿态最多有 6 种, 如图 2 所示;
- ②稳定性约束: 货物在放置时不能悬空, 即上方货物必须由下方货物的顶面完全支撑或者由容器底部完全支撑;
- ③重量约束: 货物的总重量不能超过容器所能承受的最大重量;
- ④空间约束: 货物在长宽高任一方向上都不能超出容器, 所有货物必须放置在容器内部;
- ⑤重叠约束: 两个货物在任一方向上不能交叉放置, 即在任一  $xoy, xoz$  和  $yoz$  3 个平面上的投影不能出现交叉情况;
- ⑥正交约束: 放置在容器中货物的每一个平面必须与容器的相对应平面平行 ( $\parallel$ ), 不允许斜放。

$$\begin{cases} l_i \parallel L, w_i \parallel W, h_i \parallel H, R_i = 1 \\ l_i \parallel L, h_i \parallel W, w_i \parallel H, R_i = 2 \\ w_i \parallel L, h_i \parallel W, l_i \parallel H, R_i = 3 \\ w_i \parallel L, l_i \parallel W, h_i \parallel H, R_i = 4 \\ h_i \parallel L, w_i \parallel W, l_i \parallel H, R_i = 5 \\ h_i \parallel L, l_i \parallel W, w_i \parallel H, R_i = 6 \end{cases} \quad (2)$$

$$\frac{T_i}{T_k} = 1 \text{ 或 } \frac{T_i}{LW} = 1 \quad (3)$$

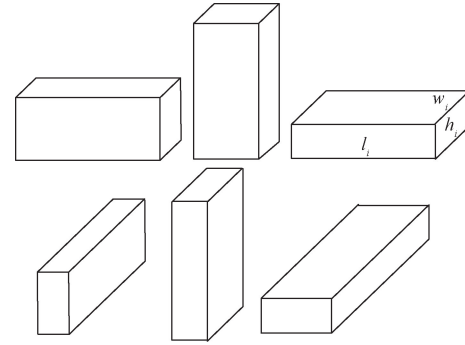


图2 货物允许旋转的姿态

Fig. 2 Cargo permitted rotational attitude

$$\sum_{i=1}^n q_i \leq Q \quad (4)$$

$$\begin{cases} x_i + l_i \leq L \\ y_i + w_i \leq W \\ z_i + h_i \leq H \end{cases} \quad (5)$$

$$\begin{cases} \Psi_{x_i} \cap \Psi_{x_j} = \emptyset \\ \Psi_{y_i} \cap \Psi_{y_j} = \emptyset \\ \Psi_{z_i} \cap \Psi_{z_j} = \emptyset \end{cases} \quad (6)$$

式(2)表示方向性约束, 同时也表示正交约束, 货物的每种姿态的每一个面均与容器对应面平行; 式(3)表示稳定性约束; 式(4)表示重量约束; 式(5)表示空间约束; 式(6)表示不重叠约束。

2 启发式装载规则设计

本文中采用启发式装载规则对改进鲸鱼模拟退火算法得出的结果进行解码操作, 主要包含 3 个环节: 货物如何选择容器内部空间, 即货物装载位置; 如何划分剩余空间; 如何进行空间合并。

2.1 选择空间

首先已知当前货物的装载顺序和装载姿态, 由改进鲸鱼模拟退火算法得出; 其次以货物的当前姿态遍历空间列表中所有空间, 货物放置在空间的左后下角处。将能放置当前货物的空间记录进可行空间列表; 最后计算可行空间列表中的空间体积, 选择体积最大的空间作为最佳放置空间, 因为可使分割出的空间更大, 为后续货物装载提供更大可能性。

2.2 划分剩余空间

每次成功放置货物后, 会自适应选择两种空间划分方式, 这两种划分方式都会将剩余空间分割为上侧空间、前侧空间和右侧空间。当前货物的当前姿态长宽高分别为  $l_i, w_i, h_i$ , 选择的空間长宽高分别为  $L_i, W_i, H_i$ , 令空间底部面积  $S_1 = L_i(W_i - w_i), S_2 = (L_i - l_i)W_i$  (图 3)。第一种分割方式: 当  $S_1 > S_2$  时, 那么前空间的底面积为  $S_1$ , 高为  $H_i$ , 右空间底面积

为 $(L_i - l_i)(W_i - w_i)$ , 高为 $H_i$ , 上空间为货物顶面的空间, 即上空间的底面积为货物顶面积, 高为 $H_i - h_i$  [图4(a)]; 第二种分割方式: 当 $S_1 < S_2$ 时, 那么前空间的底面积为 $l_i(W_i - w_i)$ , 高为 $H_i$ , 右空间底面积为 $S_2$ , 高为 $H_i$ , 上空间与第一种分割方式所产生的上空间相同[图4(b)]。

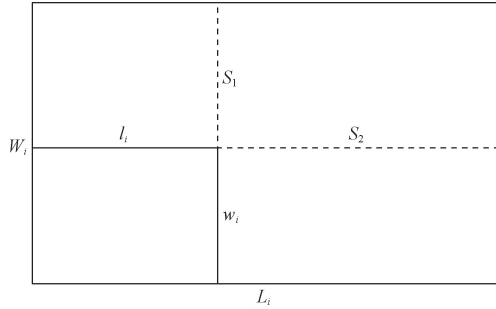
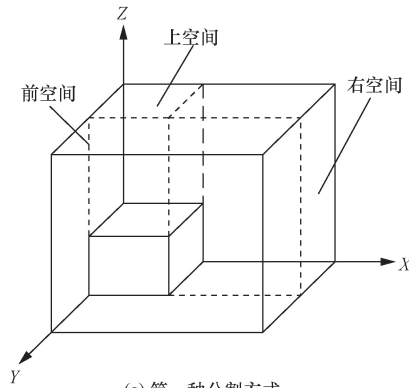
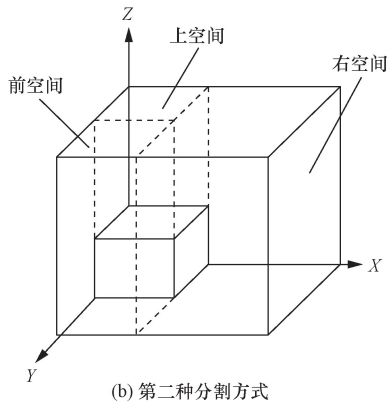


图3 空间底面积示意

Fig. 3 Schematic representation of space floor area



(a) 第一种分割方式



(b) 第二种分割方式

图4 两种空间分割方式

Fig. 4 Two types of space segmentation

### 2.3 合并剩余空间

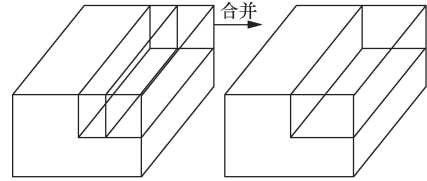
由于每次分割完空间后, 存在较多细小空间不能够放置任何一个货物, 所以需要空间合并来增加空间大小。本文中使用3种空间合并方式, 具体如下:

第一种合并方式: 判断左右相邻的空间, 如果

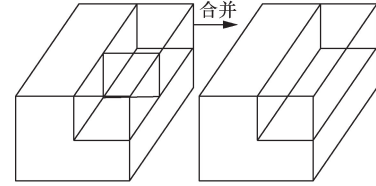
左侧空间的右面积完全等于右侧空间的左面积, 则这两个空间可以合并[图5(a)]。

第二种合并方式: 判断前后相邻的空间, 如果前侧空间的后面积完全等于后侧空间的前面积, 则这两个空间可以合并[图5(b)]。

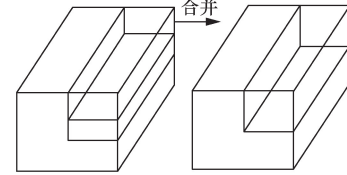
第三种合并方式: 判断上下相邻的空间, 如果上侧空间的下面积完全等于下侧空间的上面积, 则这两个空间可以合并[图5(c)]。



(a) 第一种合并方式



(b) 第二种合并方式



(c) 第三种合并方式

图5 三种空间合并方式

Fig. 5 Three types of space consolidation

## 3 改进鲸鱼模拟退火算法

### 3.1 改进鲸鱼模拟退火算法概述

鲸鱼算法<sup>[12]</sup>参数少、原理简单, 因此被广泛应用于各个领域。鲸鱼算法处理连续问题的解空间具有很强的优化空间, 可以在高维上快速收敛到最优解或更优解。但是, 三维装箱是一个离散的 NP-Hard 问题, 其变量为整数形式, 因此, 传统的鲸鱼算法无法有效求解三维装箱问题, 需要对算法进行改进。首先, 对鲸鱼算法进行离散化的处理, 采用离散化的二段式编码, 每头鲸鱼包含货物顺序和货物姿态的编码。其次, 根据所设计的启发式放置规则来进行解码, 计算出每头鲸鱼的适应度, 并对适应度进行排序, 按照一定比例选出精英座头鲸群, 精英座头鲸群包含适应度较高的鲸鱼个体。最后, 采用离散化后的鲸鱼种群更新方式来进行货物顺序和姿态的寻优, 利用模拟退火算法跳出局部最优, 来达到当前货物装载的最优解。

### 3.2 鲸鱼个体编码方式

改进鲸鱼算法的个体采用两段式编码,第一段包含货物的装载顺序,第二段为货物的装载姿态,假设一共有7个货物,个体编码方式如图6所示。

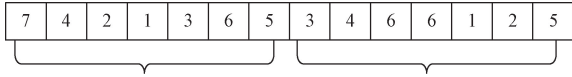


图6 鲸鱼个体编码方式

Fig. 6 How individual whales are coded

### 3.3 适应度函数

适应度函数是判断鲸鱼个体优劣与否的指标,关系到算法的收敛以及最优解的产生,以求解模型的目标函数作为适应度函数来评估鲸鱼个体,其数值越大,所得解的质量就越高,适应度函数定义为

$$\text{Fitness} = \frac{\sum_{i=1}^n l_i w_i h_i}{LWH} \quad (7)$$

### 3.4 鲸鱼个体更新方式判断

由于鲸鱼算法包括3种个体更新方式,在每个个体更新时需要判断采用何种方式。个体更新方式值的定义如下<sup>[12]</sup>。

$P$ :定义为区间(0,1)上均匀分布的随机数。

$A$ :平衡全局搜索和局部搜索的参数。 $A = 2ar - a$ ,其中  $a$  为收敛系数,在迭代过程中从2线性递减到0, $r$ 为定义在区间(0,1)上均匀分布的随机数。

具体判断方式如下。

当  $P < 0.5$  且  $|A| \geq 1$  时,采用随机搜索更新,因为在算法前期, $A$  值较大,处于探索阶段,随机搜索更新更容易发现更优解;当  $P < 0.5$  且  $|A| < 1$  时,采用包围捕食更新,算法后期  $A$  值较小,处于开发阶段,算法逐渐收敛,着重对局部更优解进行开发;当  $P \geq 0.5$  时,采用泡泡网捕食更新。

### 3.5 鲸鱼个体更新方式

鲸鱼算法的传统更新方式主要为连续变量所设计,针对此,提出以下3个离散化更新机制。

#### 3.5.1 符号说明

对鲸鱼个体更新所使用的符号进行如下说明:

$\times$ :符号后的鲸鱼个体将被截取,进行切片操作,截取的位置由符号前的系数给出;假设某一鲸鱼个体如图7所示。

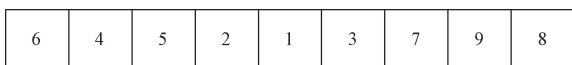


图7 随机鲸鱼个体

Fig. 7 Random individual whales

假设随机生成两个截取点位为“4”“7”,那么截取上述鲸鱼个体的第4位置到第7位置的片段,如图8所示。

$+$ :将符号后的片段拼接到符号前的鲸鱼个体的相应位置,并进行消除冲突操作。

采用部分映射方法消除冲突,即利用中间段的对应关系进行映射。消除冲突操作的具体步骤如下:①判断拼接后的个体其他位置是否有重复的元素出现,如出现,则删除其他位置重复的元素;②对该个体缺失的元素进行判断,并按照删除前在该个体元素出现的顺序依次填充空缺位置。消除冲突操作如图9所示。

$[\ ]$ :对符号内的数据进行四舍五入。

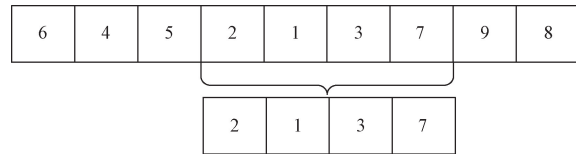


图8 截取出的片段

Fig. 8 Intercepted fragments

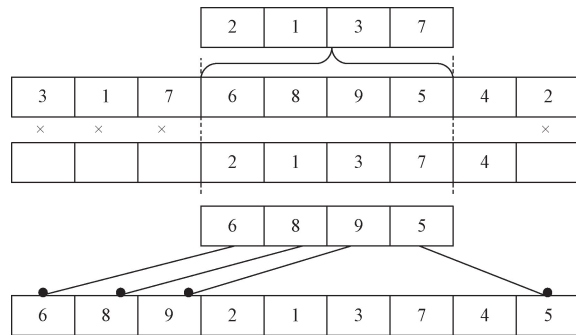


图9 消除冲突操作

Fig. 9 Deconfliction operation

#### 3.5.2 精英座头鲸群机制

传统鲸鱼算法的座头鲸为适应度值最优的个体,在包围捕食和泡泡网捕食更新时,都将从座头鲸的位置出发进行鲸鱼个体更新,但是此种方法的座头鲸选择较为单一,因此提出精英座头鲸群的方法,按一定比例划分精英座头鲸群和非精英群,利用轮盘赌概率从精英座头鲸群选择座头鲸来更新每个鲸鱼个体,提高了座头鲸选择的多样性。

#### 3.5.3 随机搜索更新

随机搜索更新为探索阶段,即全局搜索阶段。鲸鱼个体更新公式为

$$D = X(t) + C_1 X_{\text{rand}} \quad (8)$$

$$X(t+1) = X_{\text{rand}} + C_2 D \quad (9)$$

式(8)中: $X_{\text{rand}}$ 为从鲸鱼种群中随机选择的个体(不包含  $X(t)$ ); $X(t)$ 为当前种群的第  $t$  代; $C_1$ 和  $C_2$ 为系数,计算公式为

$$C_1 = [r_1 n] \quad (10)$$

$$C_2 = [r_2 n] \quad (11)$$

式中:  $r_1$  和  $r_2$  为在  $(0, 1)$  上均匀分布的随机数;  $n$  为货物数量。

式(8)表示由式(10)随机生成两个截取位置, 截取随机鲸鱼个体  $X_{rand}$ , 然后将截取出的片段拼接到  $X(t)$ , 并进行消除冲突操作, 来获得新个体  $D$ 。式(9)表示由式(11)随机生成两个截取位置, 截取由式(8)得出的  $D$ , 将截取出的片段拼接到  $X_{rand}$ , 并进行消除冲突操作, 作为  $X(t)$  的下一代个体。

### 3.5.4 包围捕食更新

包围捕食和泡泡网捕食更新均为开发阶段, 即局部搜索阶段。在包围捕食过程中, 鲸鱼个体更新公式为

$$D = X(t) + C_1 X^*(t) \quad (12)$$

$$X(t+1) = X^*(t) + C_2 D \quad (13)$$

式中:  $X^*(t)$  为利用轮盘赌从精英座头鲸群中选择出的座头鲸, 在每个鲸鱼个体更新时, 所选择的座头鲸可能会有所不同。式(12)表示由式(10)随机生成两个截取位置, 截取  $X^*(t)$ , 然后将截取出的片段拼接到  $X(t)$ , 并进行消除冲突操作来获得新个体  $D$ 。式(13)表示由式(11)随机生成的两个截取位置, 对式(12)得到的新个体  $D$  进行截取, 将截取出的片段拼接到最优个体  $X^*(t)$ , 并进行消除冲突操作, 作为  $X(t)$  的下一代个体。

### 3.5.5 泡泡网捕食更新

在泡泡网捕食更新过程中, 鲸鱼个体更新公式为

$$D = X^*(t) + C_1 X(t) \quad (14)$$

$$X(t+1) = X^*(t) + C_3 D \quad (15)$$

$$C_3 = [l e^{bl} \cos(2\pi l) n] \quad (16)$$

式中:  $b$  为螺旋系数;  $l$  表示均匀分布在区间  $[0, 1]$  内的随机数。

式(14)表示通过式(10)随机生成两个截取位置截取  $X(t)$ , 然后将截取出的片段拼接到选择出的座头鲸  $X^*(t)$ , 并进行消除冲突操作, 得到新的个体  $D$ 。因为鲸鱼算法被离散化, 为尽可能模仿泡泡网捕食行为, 由式(16)来生成截取点。式(15)表示通过式(16)生成两个截取位置, 对得到的新个体  $D$  进行截取, 然后将截取出的片段拼接到  $X^*(t)$ , 并进行消除冲突操作, 作为  $X(t)$  的下一代个体。

## 3.6 模拟退火策略

由于改进鲸鱼算法在进行局部搜索时, 总是以座头鲸对每个鲸鱼个体更新, 极容易陷入局部最优, 所以在改进鲸鱼算法基础上, 对部分鲸鱼个体<sup>[13]</sup>引入模拟退火操作, 进行领域变换搜索, 加入

随机交换、2-opt 和 3-opt 扰动机制, 使其跳出局部最优。

### 3.6.1 模拟退火步骤

在每次更新完鲸鱼个体后, 计算出每个鲸鱼个体的适应度值, 并按降序排列, 初始化模拟退火参数, 包括初始温度  $T_0$ , 终止温度  $T_{end}$ , 降温系数  $\alpha$ , 马尔科夫链长  $L$ 。对部分鲸鱼个体进行扰动, 对扰动后的新鲸鱼个体做出如下判断: 如果新解的适应度值优于旧解, 则接受该新解, 如果新解的适应度值低于旧解, 那么将以 Metropolis 准则接受新解, 对每个鲸鱼个体扰动  $L$  次, 在扰动完成后进行  $T = T_0 \alpha$  降温操作, 直至温度降至  $T_{end}$ , 输出最终的鲸鱼个体。

### 3.6.2 扰动机制

(1) 随机交换扰动机制。从鲸鱼个体随机选择两个点位, 交换其元素, 如图 10 所示。

(2) 2-opt 扰动机制。从鲸鱼个体随机选择两个点位, 交换两个点位之间的所有元素, 如图 11 所示。

(3) 3-opt 扰动机制。从鲸鱼个体随机选择 3 个点位, 按顺序交换着 3 个点位之间的所有元素, 如图 12 所示。

在选择扰动机制的时候, 将以相同的概率 1/3 进行判断。

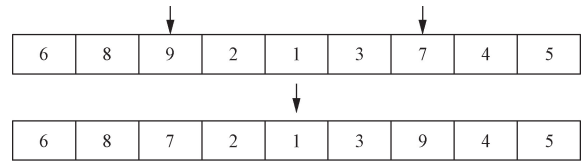


图 10 随机交换

Fig. 10 Random exchange

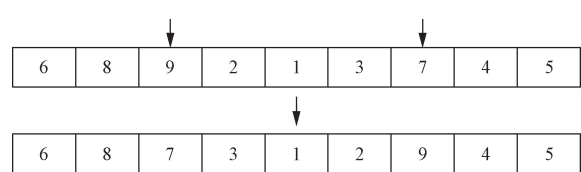


图 11 2-opt 扰动机制

Fig. 11 2-opt perturbation mechanism

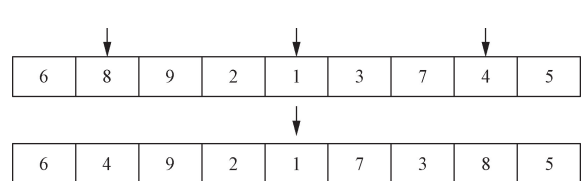


图 12 3-opt 扰动机制

Fig. 12 3-opt perturbation mechanism

## 3.7 改进鲸鱼模拟退火算法伪代码

---

算法 1 改进鲸鱼模拟退火算法  
初始化改进鲸鱼算法参数  
生成初始鲸鱼种群

---

```

通过启发式装载规则计算鲸鱼个体适应度值
根据适应度值按比例划分精英群和非精英群
while1 (t < Max_iter)
  for1 each whale
    更新 a, A, C, P, l
    If1 (P < 0.5)
      If2 (|A| ≥ 1)
        按照式(8)、式(9)更新鲸鱼个体.
      else if2 (|A| < 1)
        按照式(12)、式(13)更新鲸鱼个体.
      end if2
    else if1 (P ≥ 0.5)
      按照式(14)、式(15)更新鲸鱼个体.
    end if1
  end for1
  计算更新后的鲸鱼个体适应度值.
  初始化模拟退火算法参数.
  While2 (T0 > Tend)
    For2 i from 1 to L
      根据3种扰动机制对部分鲸鱼个体进行扰动.
      计算扰动后个体适应度值.
      If3 新解适应度值 > 旧解
        接受新解.
      else if3 新解适应度值 < 旧解
        以 Metropolis 准则接受新解.
      end if3
    end for2
    T0 = T0α.
  end while2
  根据当前种群的适应度值,划分精英群和非精英群.
  t = t + 1.
end while1
算法结束,返回适应度值最高的鲸鱼个体.

```

### 3.8 算法时间复杂度分析

本文算法主要是由初始鲸鱼种群、计算鲸鱼个体适应度、划分精英种群、鲸鱼个体更新和模拟退火策略组成。为了计算该算法的时间复杂度,需要详细分析伪代码中的每个部分的嵌套循环和操作。

(1)初始化部分。①初始化改进鲸鱼算法参数:这是常数时间操作,时间复杂度为 $O(1)$ ;②生成初始鲸鱼种群,假设种群大小为 $N$ ,时间复杂度为 $O(N)$ ;③通过启发式装载规则计算鲸鱼个体适应度值:其中启发式装载规则包含两个for循环,选择最佳放置空间的循环以及合并剩余空间的循环,假设每个循环的最大迭代次数为 $m$ ,则时间复杂度为 $O(Nm^2)$ 。

(2)主循环 while 1 的最大迭代次数为 Max\_iter,则时间复杂度为 $O(\text{Max\_iter})$ 。

(3)内部 for 1 循环①对每头鲸鱼个体进行操作,时间复杂度为 $O(N)$ ;②更新参数 $a, A, C, P, l$ ,时间复杂度为 $O(1)$ ;③条件 if 1、if 2 的两个分支,每个分支内的操作时间复杂度分别为 $O(1)$ 、 $O(N)$ ;④计算更新后的鲸鱼个体适应度值的时间复杂度

为 $O(Nm^2)$ 。

(4)模拟退火算法部分 while 2 的时间复杂度由初始温度 $T_0$ 、最终温度 $T_{\text{end}}$ 以及降温系数决定,假设为 $O(K)$ 。

(5)内部 for 2 循环。①对每个鲸鱼个体进行扰动操作,扰动次数为 $L$ ,时间复杂度为 $O(L)$ ;②计算扰动后的个体适应度值时间复杂度为 $O(m^2)$ ;③条件 if 3 每个分支内操作时间复杂度为 $O(1)$ ;④更新温度时间复杂度为 $O(1)$ 。

综上所述,主要影响时间复杂度的因素是主循环 while1 和内部的适应度值计算以及模拟退火算法的双重循环。经简化,得出本算法的时间复杂度近似为 $O[\text{Max\_iter}Nm^2(1 + KL)]$ 。

## 4 仿真实验与分析

通过3组仿真实验来证明改进鲸鱼模拟退火算法的适用性,分别是改进可行性实验、改进优越性实验和实例实验。本文提算法简称为 IWOA\_SA。利用 Matlab R2018b 编程实现,计算机处理器为 i7 @ 3.30 GHz,运行环境为 Windows11 的系统。算法涉及到的参数包括{鲸鱼种群大小 SearchAgents\_no:100;最大迭代次数 Max\_iter = 300;精英座头鲸群比例 Elite\_Whales = 0.25;模拟退火算法初始温度 $T_0 = 1$ ;终止温度 $T_{\text{end}} = 0.001$ ;降温系数 $\alpha = 0.9$ ;马尔科夫链长 $L = 40$ }。每类货物类型独立运行10次,取最终运行结果的平均值。

### 4.1 改进可行性实验

使用 Bischoff 等<sup>[14]</sup>所提供的三维装箱标准算例,其中有 BR1 ~ BR10 共10种类型的货物数据,每个类型包含100组货物数据,其中 BR1 ~ BR7 为弱异构型,BR8 ~ BR10 为强异构型,将未改进前的传统鲸鱼算法与改进鲸鱼模拟退火算法进行对比,具体结果如表1所示。

表1 算法改进前后对比

Table 1 Algorithm improvement before and after comparison

测试集	填充率/%	
	传统鲸鱼算法	IWOA_SA
BR1	88.27	92.43
BR2	85.89	93.68
BR3	87.09	92.71
BR4	81.54	92.84
BR5	82.44	91.33
BR6	81.82	90.80
BR7	81.47	90.22
BR8	78.54	90.61
BR9	78.33	88.21
BR10	76.87	87.53
平均	82.26	91.04
标准差	3.64	1.90

#### 4.2 改进优越性实验

使用的货物数据同 4.1 节,利用其他学者提出的三维装箱算法作为改进鲸鱼模拟退火算法的对比算法,验证算法的优越性,其具体结果如表 2 所示。

#### 4.3 实例实验

采用某企业的货物订单信息进行实例验证,货物数据如表 3 所示,其装载效果如图 13 所示,容器的填充率为 93.87%。

#### 4.4 实验结果分析

从 BR1 ~ BR10 三维装箱标准算例和企业实际装箱数据的结果来看,改进鲸鱼模拟退火算法都取得了不错的结果。改进后的平均容器填充率提高约 8.78%,标准差降低约 1.74,说明改进后算法性能明显提升,表现也更加稳定。在 BR1 ~ BR7 以及具体实例数据上,弱异构型货物的平均容器填充率达到了 92.24%,相较于对比算法平均提升了近 7%,在 BR8 ~ BR10 强异构型算例上填充率有所降

表 2 各算法对比结果

Table 2 Comparison results of each algorithm

测试集	填充率/%					
	3D_RSO <sup>[8]</sup>	MBL <sup>[15]</sup>	JHABC <sup>[16]</sup>	LNS <sup>[17]</sup>	GRASP <sup>[18]</sup>	IWOA_SA
BR1	84.38	89.92	84.42	92.2	89.07	92.43
BR2	84.29	90.64	83.30	88.4	90.43	93.68
BR3	84.63	90.67	86.84	88.8	90.86	92.71
BR4	83.23	91.46	85.26	87.0	90.42	92.84
BR5	82.87	91.70	84.17	84.5	89.67	91.33
BR6	81.82	90.87	89.33	78.3	89.71	90.80
BR7	80.73	90.31	85.87	72.7	88.05	90.22
BR8	79.19	90.07	87.66	62.1	86.13	90.61
BR9	77.23	89.87	86.93	61.1	85.08	88.21
BR10	74.68	89.45	87.29	56.3	84.21	87.53
平均	81.31	90.49	86.11	77.14	88.36	91.04

表 3 某企业的货物订单数据

Table 3 Cargo order data from a factory

烟箱类别	物料描述	长/mm	宽/mm	高/mm	数量	容器内尺寸/(m × m × m)
普通硬包	品牌 A(硬包)	455	243	570	50	9.6 × 2.45 × 2.8
普通软包	品牌 B(软包)	445	233	550	500	
软包硬化	品牌 B(软包)	445	230	543	250	
中支	品牌 B(中支)	475	345	306	75	
细支	品牌 A(细支)	528	262	287	100	
短支	品牌 B(短支)	398	235	538	100	
	品牌 B(软包)	482	260	278	200	
其他	品牌 B(双中支)	505	370	176	200	

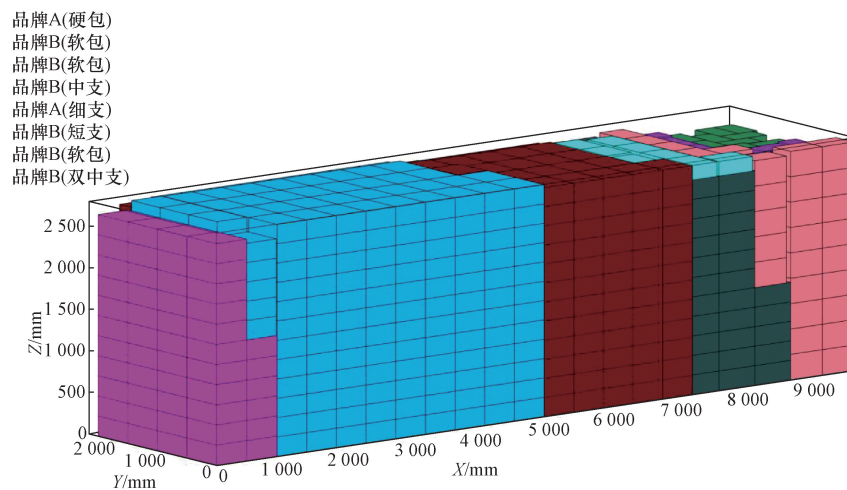


图 13 实例装载图

Fig. 13 Example loading diagram

低,但较之对比算法平均提升了近 11%,充分体现出该算法在解决三维装箱问题上的适用性和优越性。

## 5 结论

对于三维装箱问题的研究,本文提出了一种改进鲸鱼模拟退火算法。

(1) 针对传统鲸鱼算法容易陷入局部最优以及不能很好地处理离散化问题,本文中重新定义了鲸鱼个体的更新机制,包括座头鲸的选择和更新方式,利用模拟退火算法的全局搜索能力,引入随机交换、2-opt 和 3-opt 扰动机制,解决了局部最优陷阱。

(2) 通过 3 个仿真实验来验证算法的适用性,算法改进后的平均容器填充率提高了约 8.78%,标准差降低了约 1.74,相比于其他三维装箱算法,平均容器填充率也提高了 6~7 个百分点。结果表明,该算法无论是针对弱异构型或者强异构型货物数据,都取得了良好的装箱效果,可以为三维装箱问题的研究提供一定借鉴。

(3) 该算法涉及的参数较少,操作简单,易于实现,并有效地结合了鲸鱼优化算法的搜索策略和模拟退火算法的随机性,平衡了全局搜索和局部搜索能力,引入的精英座头鲸群机制提高了座头鲸选择的多样性。但其也存在一些不足,线性减小的收敛系数使得算法收敛较慢,而且模拟退火算法每次迭代需要额外计算温度和执行扰动,增加了计算复杂度。

(4) 未来可采用某些非线性收敛系数,加快算法的收敛,另外增加更多的模拟退火算法扰动机制,丰富寻找更优解的途径,从而达到更快地获得最优装箱方案。

## 参 考 文 献

- [1] George J A, Robinson D F. A heuristic for packing boxes into a container[J]. *Computers & Operations Research*, 1980, 7(3): 147-156.
- [2] Bortfeldt A, Gehring H, Mack D. A parallel tabu search algorithm for solving the container loading problem[J]. *Parallel Computing*, 2003, 29(5): 641-662.
- [3] Parreño F, Alvarez-Valdés R, Tamarit J M, et al. A maximal-space algorithm for the container loading problem[J]. *Inform Journal on Computing*, 2008, 20(3): 412-422.
- [4] 田鑫, 贺可太, 刘珮瑶, 等. 单品种货物装车问题的两阶段优化算法[J]. *科学技术与工程*, 2023, 23(11): 4725-4733.  
Tian Xin, He Ketai, Liu Peiyao, et al. Two-stage optimization algorithm for solving the single-variety cargo loading problem[J]. *Science Technology and Engineering*, 2023, 23(11): 4725-4733.
- [5] 张长勇, 刘佳瑜, 王艳芳. 求解货物在线装箱问题的融合算法[J]. *科学技术与工程*, 2021, 21(11): 4513-4518.  
Zhang Changyong, Liu Jiayu, Wang Yanfang. Fusion algorithm for solving the problem of online cargo packing[J]. *Science Technology and Engineering*, 2021, 21(11): 4513-4518.
- [6] 张长勇, 吴刚鑫. 基于分组策略的多目标三维装箱算法[J]. *包装工程*, 2023, 44(21): 204-213.  
Zhang Changyong, Wu Gangxin. Multi-objective three-dimensional crating algorithm based on grouping strategy[J]. *Packaging Engineering*, 2023, 44(21): 204-213.
- [7] 周丽, 杨江龙, 赵俊辉, 等. 基于混合遗传算法的多箱型三维装箱问题研究[J]. *包装工程*, 2022, 43(21): 213-223.  
Zhou Li, Yang Jianglong, Zhao Junhui, et al. Research on three-dimensional cartoning problem of multi-box type based on hybrid genetic algorithm[J]. *Packaging Engineering*, 2022, 43(21): 213-223.
- [8] 尚正阳, 顾寄南, 唐仕喜, 等. 高效求解三维装箱问题的剩余空间最优化算法[J]. *计算机工程与应用*, 2019, 55(5): 44-50.  
Shang Zhengyang, Gu Jinan, Tang Shixi, et al. Efficient residual-space-optimization algorithm for three dimensional container loading problem[J]. *Computer Engineering and Applications*, 2019, 55(5): 44-50.
- [9] 姜凯文. 基于改进鲸鱼算法的潜艇多维磁场补偿研究[D]. 武汉: 武汉理工大学, 2024.  
Jiang Kaiwen. Research on multi-dimensional magnetic field compensation of submarine based on improved whale algorithm[D]. Wuhan: Wuhan University of Technology, 2024.
- [10] 张强, 郭玉洁, 王颖, 等. 一种离散鲸鱼算法及其应用[J]. *电子科技大学学报*, 2020, 49(4): 622-630.  
Zhang Qiang, Gu Yujie, Wang Ying, et al. A discrete whale optimization algorithm and application[J]. *Journal of University of Electronic Science and Technology of China*, 2020, 49(4): 622-630.
- [11] Zhang S, Gu X. A discrete whale optimization algorithm for the no-wait flow shop scheduling problem[J]. *Measurement and Control*, 2023, 56(9/10): 1764-1779.
- [12] Mirjalili S, Lewis A. The whale optimization algorithm[J]. *Advances in Engineering Software*, 2016, 95: 51-67.
- [13] 马林, 秦阳, 秦嘉豪, 等. 大型星座混合模拟退火遗传算法测控任务规划[J]. *宇航学报*, 2023, 44(11): 1757-1766.  
Ma Lin, Qin Yang, Qin Jiahao, et al. Hybrid simulated annealing genetic algorithm measurement and control mission planning for large constellations[J]. *Journal of Astronautics*, 2023, 44(11): 1757-1766.
- [14] Bischoff E E, Ratcliff M S W. Issues in the development of approaches to container loading[J]. *Omega*, 1995, 23(4): 377-390.
- [15] 雷定猷, 洪舒华, 张英贵. 集装箱轻重货物混合平衡装载模型与算法[J]. *计算机工程与应用*, 2020, 56(8): 233-240.  
Lei Dingyou, Hong Shuhua, Zhang Yinggui. Model and algorithm for container mixed balanced loading of light and heavy cargo[J]. *Computer Engineering and Applications*, 2020, 56(8): 233-240.
- [16] Bayraktar T, Ersöz F, Kubat C. Effects of memory and genetic operators on artificial bee colony algorithm for single container loading problem[J]. *Applied Soft Computing*, 2021, 108: 107462.
- [17] Şafak Ö, Erdoğan G. A large neighbourhood search algorithm for solving container loading problems[J]. *Computers & Operations Research*, 2023, 154: 106199.
- [18] 于明正, 徐斌, 陈佳. 基于双层启发式遗传算法的三维装箱问题[J]. *科学技术与工程*, 2020, 20(5): 2042-2047.  
Yu Mingzheng, Xu Bin, Chen Jian. Research of 3D packing problem based on double-layer heuristic genetic algorithm[J]. *Science Technology and Engineering*, 2020, 20(5): 2042-2047.