基于贪心法以及遗传算法的工业装箱优化

陈意然，周正阳，吴思远，张烨钟

（绍兴文理学院，浙江 绍兴 312000）

摘　要　我国物流业发展迅速，网购已经成为大部分人的主流消费方式，随之带来的就是物流业的崛起，大量的订单打开了物流业在中国的市场。据数据统计，2022年中国的线上订单包裹已经超过了1000亿件，巨大的数量基数导致了如果在产品上做出稍微的优化，也会带来不菲的收益，于是如何优化包裹就成了一大值得探究的问题，本文就包装方案、尺寸优化、柔性因素三个问题展开讨论，旨在为相关人员提供参考。

关键词　贪心法；可变高度模型；遗传算法；自适应算法；模拟退火

中图分类号：F252；TP3　　**文献标识码**：A　　**文章编号**：1007-0745(2023)09-0001-03

1 研究背景

目前我国快递物流业已成为全球最大的快递市场。由于包裹的基数很大，所以适当节省包装材料就可以产生较大的经济效益。因此，我们需优化耗材方案，在装下包裹的同时节省材料与体积，并考虑货物与耗材之间的柔性问题，做出进一步优化。

2 问题重述

问题一：对订单数据给出包装方案。

问题二：优化每种耗材尺寸。

问题三：若考虑到耗材是柔性物体（长宽高的延伸比例不超过5%），重新探究问题一和问题二。

3 模型假设

1.所有需要装载的货物没有固定放置方向的要求。

2.待装物件都是由包装盒打包的规则长方体。

3.用袋子装物品时袋子的形状会因物件形状变化，在装如矩形物件后，袋子的边角处会存在一个不能放入任何物件的角，忽略此部分的体积。

4.不考虑货物与货物之间的挤压冗余，仅考虑耗材的形变。

4 模型的建立与求解

4.1 问题一：对订单数据给出包装方案

4.1.1 全部使用箱子作为耗材的方案

$$x_{(d,i)}=\begin{cases}1,\text{订单}d\text{使用}i\text{号箱}\\0,\text{订单}d\text{不使用}i\text{号箱}\end{cases}\tag{1}$$

其中i=1，2，3，4，5。

订单总数为Dn，令$num_{(d,i)}$为订单编号d需要的i号箱的数量，目标函数可以表示为：

$$minX=\sum_{d=1}^{Dn}\sum_{i=1}^{5}num_{(d,i)}*x_{(d,i)}\tag{2}$$

对于箱型选择和货物装载两个步骤，我们采用自适应随机算法，算法思想如下：

1.对任意订单d输入，首先规定物件在耗材中的摆放规则，在包装箱的左后方为原点建立空间直角坐标系，优先填满X轴［一维过程，记为step1］，在X轴达到最优后由X轴向Y轴延伸［二维过程，记为step2］，最后由XOY平面向Z轴顶端延伸［三维过程，记为step3］）。

2.随机选择箱型i，记Box_i（Li，Wi，Hi）为i号箱子的长宽高，goods(lj,wj,hj，d)为订单中第j件货物的长宽高。在订单d中随机选择首件货物，若能放下则更新箱内空间，以及装入后货物离坐标原点的最远距离点P（x,y,z），若不能则增大箱子型号，则结束，整体有约束：

$$\begin{cases}Li-x\geq 0\\Wi-y\geq 0\\Hi-z\geq 0\end{cases}\tag{3}$$

3.计算剩余空间是否能减小箱子的尺寸型号，若能则减小，重新进行第二步；不能则继续第四步。

4.根据重力式空间搜索策略[1]算法的摆放优先级选择货物最优尺寸，即在一维过程中需要在订单d中找到lj（wj或hj），直至装载完成。

$$\text{st}\quad\begin{cases}Li-x\geq lj\\Wi-y\geq wj\\Hi-z\geq hj\end{cases}\tag{4}$$

综上得到模型：

$$minX=\sum_{d=1}^{Dn}\sum_{i=1}^{5}num_{(d,i)}*x_{(d,i)}$$

$$minV=\sum_{d=1}^{Dn}\sum_{i=1}^{5}num_{(d,i)}*x_{(d,i)}*H_i*L_i*W_i$$

$$\begin{cases} Li-x\geq 0 \\ Wi-y\geq 0 \\ Hi-z\geq 0 \end{cases} \quad (5)$$

$$\begin{cases} Li-x\geq lj \\ Wi-y\geq wj \\ Hi-z\geq hj \end{cases}$$

H_i，L_i，$W_i, x, y, z, num_{(d,i)}, x_{(d,i)}\geq 0$

4.1.2 全部使用袋子作为耗材的方案

使用可变高度的袋子模型，配合贪心算法来解决问题[2]，首先用0-1规划选择袋型：

$$X_{\beta i}=\begin{cases} 1 第\beta 次装载时选择第i号袋子 \\ 0 第\beta 次装载时不选择第i号袋子 \end{cases}$$

i=1,2,3,4 β=1,2,3,4……

对EXCEL表格中的订单进行处理，把同一订单下的多种物品按照数量全部展开，a_{km}表示第k个订单的第m个物品，$a_{km}X_i$表示将第k个订单下的第m个物品装入第X_i号码袋子，V_{km}表示第k个订单的第m个物品的体积，显然有：$V_{km}=I_{km}*W_{km}*h_{km}$。

V_i表示使用第i个袋子的体积，显然有：$V_i=I_i*W_i*h_i$。

根据题目中所给h_i高度为1，进一步构造可变高度的袋子模型，袋子的高度h_i等于第一个装入的物品高度且可实时更新，有：$V_{im}=I_i*W_i*h_{km}$。

用贪心法将所有订单物品按底面积从大到小排列，并分别算出四个袋子的底面积，排除无法装下的物品，在符合装袋要求的物品中，优先将最大的物品装入满足底面积要求且剩余底面积最小的袋子，更新袋子剩余体积：$V_i'=V_{im}-V_{km}$。

选择满足装袋条件且次大的订单物品，继续进行装载，同时比较h_{km}与h_{k1m1}，倘若$h_{km}>h_{k1m1}$则不必更新袋子体积，否则再次更新袋子体积：$V_{im1}=I_i*W_i*h_{k1m1}$。

重复上述装载操作，在整个装载过程中，需同时满足下列装载条件：

1. 袋子约束：

袋子长+袋子高≥物品长+物品高；袋子宽+袋子高≥物品宽+物品高

对应写出数学约束：

$$\begin{cases} I_i+h_{k_\alpha m_\alpha}\geq I_{km}+h_{km} \\ W_i+h_{k_\alpha m_\alpha}\geq W_{km}+h_{km} \end{cases} \quad (6)$$

其中$h_{\beta ik_\alpha \beta im_\alpha}$表示第β次装载时向第i个袋子中放入第α次订单物品后，更新的袋子高度。

2. 方向约束。根据题意我们可以知道，袋子中的订单物品长宽高可以任意互换，因此我们建立方向约束：

$$\begin{cases} bI_{km}=\{0,1\} \\ bW_{km}=\{0,1\} \\ bh_{km}=\{0,1\} \end{cases} \quad (7)$$

bI_{km}，bW_{km}，bh_{km}为第k个订单下的第m个物品对应的边竖直放置作为高度，0表示可以放，1则表示不能放。

3. 设置双目标函数。目标函数要满足耗材总体积越小越好的条件，耗材数量少的条件，我们可以写出：

$$\min\beta$$

$$\min\sum_{n=1}^{\beta} I_{\beta i}*W_{\beta i}*h_{\beta ik_\alpha \beta im_\alpha} \quad (8)$$

4.1.3 箱子和袋子两种耗材同时使用的方案

由于袋子可塑性较强且比箱子更节省材料与空间，因此我们优先考虑使用袋子，当袋子装不下时，再考虑使用箱子[3]进行组装：

$$X_{\beta i}=\begin{cases} 1 第\beta 次装载时选择第i号袋子 \\ 0 第\beta 次装载时不选择第i号袋子 \end{cases}$$

i=1,2,3,4 β=1,2,3,4……

构造判断函数：

$$a_\beta=\begin{cases} 1 第\beta 次装载时所有袋子都不满足条件 \\ 0 第\beta 次装载时能选择袋子装配 \end{cases}$$

$$a_\beta X_{\beta\omega}=\begin{cases} 1 第\beta 次装载时第i号袋子装不下，选择第\omega 号箱子 \\ 0 第\beta 次装载时第i号袋子装不下，不选择第\omega 号箱子 \end{cases}$$

i=1,2,3,4 β=1,2,3,4……

ω=1,2,3,4,5

对于原来那些在上题情形超出1-4号袋子底面积范围的袋子，或者物品本身尺寸并不满足袋子自身的约束条件，尝试使用箱子进行装载，比较物品体积与每个箱子的体积，用$V_{\beta\omega}=I_{\beta\omega}*W_{\beta\omega}*h_{\beta\omega}$依次遍历五个箱子，比较$V_{\beta\omega}$与$V_{km}=I_{km}*W_{km}*h_{km}$的大小关系，找到满足条件且体积最小的箱子，进行装载，并更新箱子的剩余体积：$V_i''=V_{\beta\omega}-V_{km}$

再用V_i''遍历空箱子，若有多个订单物品的体积满足条件，则优先取体积最大的物品进行装箱。

目标函数需要综合考虑装袋与装箱部分：

$$\min\sum_{\beta=1}^{25837} I_{\beta i}*W_{\beta i}*h_{\beta ik_\alpha \beta im_\alpha}+a_\beta*\left(I_{\beta\omega}*W_{\beta\omega}*h_{\beta\omega}\right) \quad (9)$$

在这一部分我们还要加上箱子体积约束：$V_{km}\leq a_\beta V_{\beta\omega}$。

4.2 问题二：优化每种耗材尺寸

4.2.1 全部使用箱子时的耗材优化

采取遗传算法，新变量$I'_{\beta\omega}$表示更改后的第i种箱子的长度，$W'_{\beta\omega}$表示更改后的第i种箱子的宽度，$h'_{\beta\omega}$表示更改后的第i种箱子的高度：

$$\min\sum_{\beta=1}^{25837} V=I'_{\beta\omega}*W'_{\beta\omega}*h'_{\beta\omega}-I_{km}*W_{km}*h_{km}$$

同时我们需要保持耗材部分总体积最小化：

$$\min \sum_{\beta=1}^{25837} \mathrm{V} = \mathrm{I}'_{\beta\omega} * \mathrm{W}'_{\beta\omega} * h'_{\beta\omega}$$

着重考虑约束条件：装箱方向保持约束条件不变。

$$\begin{cases} b\mathrm{I}_{\mathrm{km}} = \{0,1\} \\ b\mathrm{W}_{\mathrm{km}} = \{0,1\} \\ bh_{\mathrm{km}} = \{0,1\} \end{cases}$$

使用遗传算法，根据每个箱子的既有尺寸在小范围内进行修改，利用遗传思路扩大尺寸变化范围，进一步寻找最优解。

4.2.2 全部使用袋子时的耗材优化

采取遗传算法[4]，更新后袋子依旧是可塑性的，高度为1，$\mathrm{I}'_{\beta i}$ 表示更改后的第i种袋子的长度，$\mathrm{W}'_{\beta i}$ 表示更改后的第i种袋子的宽度，目标是更新后的每个袋子的体积减去每个物品装袋后的剩余体积尽可能地小：

$$\min \sum_{\beta=1}^{25837} \mathrm{V} = \mathrm{I}'_{\beta i} * \mathrm{W}'_{\beta i} * h_{\beta \mathrm{ik}_{\alpha} \beta \mathrm{im}_{\alpha}} - \mathrm{I}_{\mathrm{km}} * \mathrm{W}_{\mathrm{km}} * h_{\mathrm{km}} \tag{10}$$

我们把假设的新袋装尺寸代入刚才的约束条件：

首先是袋子约束依然满足长宽条件：

$$\begin{cases} \mathrm{I}'_{\beta i} + h_{\mathrm{k}_{\alpha}\mathrm{m}_{\alpha}} \geq \mathrm{I}_{\mathrm{km}} + h_{\mathrm{km}} \\ \mathrm{W}'_{\beta i} + h_{\mathrm{k}_{\alpha}\mathrm{m}_{\alpha}} \geq \mathrm{W}_{\mathrm{km}} + h_{\mathrm{km}} \end{cases} \tag{11}$$

其次是对于装袋条件，如果尺寸过大且超出装袋范围则无法装袋：$\mathrm{I}'_{\beta i} \leq \mathrm{I}_{\mathrm{km}}$ 或者 $\mathrm{W}'_{\beta i} \leq \mathrm{W}_{\mathrm{km}}$，需在预处理时删除数据。

最后是保持上述装袋方向约束条件不变：

$$\begin{cases} b\mathrm{I}_{\mathrm{km}} = \{0,1\} \\ b\mathrm{W}_{\mathrm{km}} = \{0,1\} \\ bh_{\mathrm{km}} = \{0,1\} \end{cases} \tag{12}$$

用遗传算法[5]将新尺寸的装载物品数量作为适应度函数的数值，同时减去箱子剩余空间的大小作为罚项。

同时使用箱子和袋子问题，基本思路同4.1.3，装满袋子后考虑箱子。

4.3 问题三：若考虑到耗材是柔性物体，重新探究问题一和问题二

全部使用箱子作为耗材时的方案。在装载完成后，耗材长度上会有原长宽高1.05倍的空间约束条件：

$$\begin{cases} 1.05 * Li - x \geq 0 \\ 1.05 * Wi - y \geq 0 \\ 1.05 * Hi - z \geq 0 \end{cases}$$

在装载过程中，选取最佳货物时也要满足长度约束：

$$\begin{cases} 1.05Li - x \geq lj \\ 1.05Wi - y \geq wj \\ 1.05Hi - z \geq hj \end{cases}$$

在新条件约束下，

$$\begin{aligned} minX &= \sum_{d=1}^{Dn} \sum_{i=1}^{5} num_{(d,i)} * x_{(d,i)} \\ minV &= \sum_{d=1}^{Dn} \sum_{i=1}^{5} num_{(d,i)} * x_{(d,i)} * H_i * L_i * W_i \end{aligned} \tag{13}$$

求解目标函数：

利用遗传算法求解优化后的方案，设定目标函数：

$$\min \sum\nolimits_{\beta=1}^{25837} \mathrm{V} = \mathrm{I}'_{\beta\omega} * W'_{\beta\omega} * h'_{\beta\omega} - \mathrm{I}_{\mathrm{km}} * \mathrm{W}_{\mathrm{km}} * h_{\mathrm{km}} \tag{14}$$

同时需保持耗材部分总体积最小化：

$$\min \sum\nolimits_{\beta=1}^{25837} \mathrm{V} = \mathrm{I}'_{\beta\omega} * W'_{\beta\omega} * h'_{\beta\omega}$$

装箱方向保持约束条件不变：

$$\begin{cases} b\mathrm{I}_{\mathrm{km}} = \{0,1\} \\ b\mathrm{W}_{\mathrm{km}} = \{0,1\} \\ bh_{\mathrm{km}} = \{0,1\} \end{cases}$$

对于袋装和混装类型，只需要将参数做1.05倍处理，其余做法与前例相同。

5 应用前景

通过优化装箱方案，可以减少运输中的空间浪费，缩减运输次数和运输成本，节约包装材料的使用量，提高经济效益。

在只用箱子包装的情况下，本文使用的自适应随机算法得到的方案准确度较高，但是收敛速度慢。在考虑只用袋装时使用的贪心算法忽略了物体的三维特征，但是贪心算法在解决三维尺寸可变装箱问题且待装物件较少的时候，也能给出较为准确的结果，贪心算法的优势在于运算时间短、效率高，但在运行时可能会陷入局部最优解，因而在某些情况下得到的解准确度较低。

参考文献：

[1] 吴蓓，丁文英，杜彦华，等. 基于重力装载的自适应随机算法求解多箱型三维装箱问题[J]. 计算机集成制造系统，2020,26(11):3084-3093.

[2] 李孙寸，施心陵，张松海，等. 基于多元优化算法的三维装箱问题的研究[J]. 自动化学报，2018,44(01):106-115.

[3] 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(05):44-50.

[4] 曹先彬，刘克胜，王煦法. 基于免疫遗传算法的装箱问题求解[J]. 小型微型计算机系统，2000,21(04):361-363.

[5] 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.