

考虑易腐品的越库中心车辆调度研究

范春阳, 潘 飞*

上海理工大学管理学院, 上海

收稿日期: 2025年2月7日; 录用日期: 2025年2月28日; 发布日期: 2025年3月10日

摘 要

随着物流行业竞争加剧, 越库配送模式逐渐得到推广, 尤其在易腐品物流中展现出明显优势。易腐品因其易变质特性, 存储和运输难度大, 多次搬运和长时间存储容易加速变质, 增加损耗。越库模式通过减少商品的搬运次数和库存存储, 提升了物流效率并减少了运输成本。本文提出将越库模式应用于易腐品转运过程的车辆调度, 基于最小化总变质质量, 构建混合整数数学规划模型, 并设计基于车辆排序的遗传算法求解。通过数值分析对比最小化完工时间和最小化总变质质量模型, 分析两种不同模型在易腐品越库转运过程中的优劣, 为易腐品物流企业的越库实践提供指导, 具有重要的现实意义。

关键词

易腐品, 越库, 车辆调度, 遗传算法

Research on Truck Scheduling in Cross-Docking Centers for Perishable Goods

Chunyang Fan, Fei Pan*

Business School, University of Shanghai for Science and Technology, Shanghai

Received: Feb. 7th, 2025; accepted: Feb. 28th, 2025; published: Mar. 10th, 2025

Abstract

With the intensification of competition in the logistics industry, the cross-docking distribution model has gradually been promoted and has shown obvious advantages especially in the logistics of perishable products. Perishable products are difficult to store and transport due to their perishable nature. Multiple handlings and long-term storage are likely to accelerate deterioration and increase losses. The cross-docking model improves logistics efficiency and reduces transportation costs by reducing the number of times goods are handled and inventory storage. This paper proposes to

*通讯作者。

文章引用: 范春阳, 潘飞. 考虑易腐品的越库中心车辆调度研究[J]. 建模与仿真, 2025, 14(3): 52-65.
DOI: 10.12677/mos.2025.143202

apply the cross-docking model to truck scheduling in the transshipment process of perishable products. Based on minimizing the total deteriorated quality, a mixed integer mathematical programming model is constructed, and a genetic algorithm based on truck sequencing is designed for solution. Through numerical analysis, the models of minimizing the makespan and minimizing the total deteriorated quality are compared, and the advantages and disadvantages of the two different models in the cross-docking transshipment process of perishable products are analyzed, which provides guidance for the cross-docking practice of perishable goods logistics enterprises and has important practical significance.

Keywords

Perishable Goods, Cross-Docking, Truck Scheduling, Genetic Algorithm

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

物联网、带传感器的 5G 网络和互联网技术的快速创新催生了许多新的商业模式。亚马逊、京东、Postmate 等公司提供连接和匹配供需并提供有效中介服务的平台[1]。新技术使这些公司能够减少中间商数量并改变传统供应链服务运营。其中大多数公司设计了越库服务平台,以便在供应商和买家之间分配多种产品。虽然越库服务变得越来越重要,但由于其日益复杂且在无法实现完全自动化的情况下,还存在操作失误的风险,这也带来了一系列挑战。

现代多渠道零售商通常接收来自数十万供应商的产品并将其交付给数百万买家。在本研究中,我们研究易腐产品物流的越库服务平台优化。我们考虑易腐产品的一个独特运营特性——变质。易腐产品在供应链运营过程中会大幅变质,导致大量收入损失。根据中国物流与采购联合会冷链物流专业委员会的测算,2022 年我国食品冷链物流需求总量为 3.3 亿吨,其中 27.80%为蔬菜、23.50%为水果、19.60%为肉类、16.40%为水产品、7.10%为乳制品、5.50%为速冻食品。中国果蔬、肉类和水产品的变质损失分别控制在 15%、8%和 10%以下。尽管如此,与发达国家 5%以下的变质损失相比,仍存在差距[2]。虽然食品浪费是一个全球性问题,但某些物流环境中的高变质风险已成为全球可持续食品供应链的主要障碍。

在现有文献中,易腐产品的变质遵循具有恒定变质率(Deterioration Rate, DR)的动力学模型[3]。在此建模框架下,变质过程可定义为指数形式[4],且假定变质率恒定。我们认为该假设仅适用于稳定环境。对于大型复杂的越库服务平台,静态变质假设并不成立,因为一些操作环境往往不稳定。食品工程和化学工程文献也提供了证据,表明当处理环境和操作频繁转换时,易腐产品处于不稳定状态[5]。

这种不稳定状态是由随机操作环境变化引起的。操作失误是不稳定状态的另一个关键因素,包括不可预测的碰撞、过度振动或损伤。物理损伤(如搬运、碰撞、刮擦、摩擦等)通常比病原体生长更能降低产品质量[5]。例如,搬运过程中的振动会加速葡萄和草莓的变质[6]。在易腐产品供应链中,不稳定状态现象主要发生在非自动化或露天操作中,如越库中心的卸载、装载和转运操作,这些操作中温度/湿度变化和操作风险较高。

越库在许多行业中被广泛应用以提高产品流的效率[7]。当服务平台开始增加供应商、买家和产品类型的数量时,它变得至关重要。标准的越库中心操作流程包括以下步骤:(1) 卸载,(2) 识别,(3) 分拣,(4) 装载[8]。当入库卡车到达入库码头时,产品被卸载。卸载的产品在出库码头进行分拣和整合以便运

输。如果产品未立即装载到出库卡车上, 则会暂时储存, 直到预定的出库卡车到达出库码头。沃尔玛、UPS、丰田和亚马逊等公司的实际越库实施案例[9]表明, 它对于具有多供应商、多买家和多产品的复杂服务平台具有重要的竞争意义。

本文通过定义不同环境下易腐品的变质率对产品质量的影响来丰富易腐物流领域的研究。具体而言, 在越库服务平台的背景下, 我们考虑可能导致不稳定状态的三个主要因素: 产品批量、操作持续时间和环境不确定性。我们依靠物联网技术[10]提供实时数据来测量这些关键因素。这些因素的累积整合导致产品质量在不稳定状态下永久变化。为此, 我们首先引入可能在越库中心整个物流过程中发生的质量变质 (Quality Deterioration, QD)。质量变质变化描述了易腐产品在不稳定状态下质量的不可逆影响。其累积值 (Accumulated Quality Deterioration, AQD) 随后被用作构建混合整数规划模型的指标, 旨在优化易腐产品卡车调度问题。在数值分析中, 设计 9 个不同规模的实例来研究基于 AQD 的模型。我们还将其性能与传统的基于完工时间的模型进行比较, 结果表明基于 AQD 的模型在不稳定状态下对易腐产品质量保存具有显著优势。

2. 问题描述

本文考虑仅含有一个入库码头和一个出库码头的“1”字母型越库系统, 入库卡车与出库卡车一旦进入码头便不能离开, 直到其完成所有的卸/装货任务。如图 1 所示。

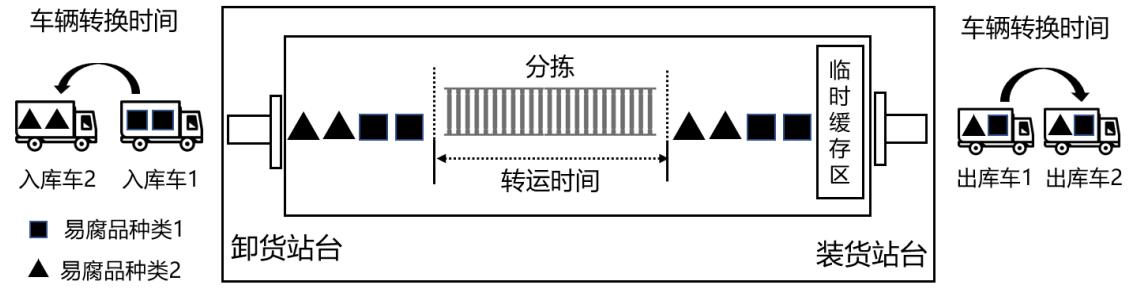


Figure 1. Flowchart of perishable product transshipment
图 1. 易腐品转运流程图

3. 符号说明与假设

3.1. 符号说明

参数和决策变量及其含义分别如表 1 和表 2 所示。

Table 1. Parameters and their meanings
表 1. 参数及其含义

参数	含义
I	入库车集合, $i \in \{1, 2, \dots, I\}$
J	出库车集合, $j \in \{1, 2, \dots, J\}$
K	易腐品种类集合, $k \in \{1, 2, \dots, K\}$
P	易腐品中转过程中经历的不同存储区域集合, $p = \{1, 2, \dots, P\}$
m	入库码头上按顺序排列的第 m 辆入货卡车, $m \in \{1, 2, \dots, I\}$
n	出库码头上按顺序排列的第 n 辆出货卡车, $n \in \{1, 2, \dots, J\}$

续表

r_{ik}	入库车 i 装载第 k 种易腐品的数量
s_{jk}	出库车 j 需要第 k 种易腐品的数量
D	车辆转换时间
V	易腐品从卸货站台到装货站台的传送时间
M	一个足够大的数
θ_{kp}	第 k 种易腐品 0 时刻的新鲜度
q_k^0	第 k 种易腐品在第 p 种存储区域的腐损率

Table 2. Algorithm decision variables and their meanings
表 2. 算法决策变量及其含义

决策变量	含义
e_m	入库车序列中位于第 m 的入库车的进入和离开码头的时
l_m	入库车序列中位于第 m 的入库车的进入和离开码头的时
E_n	出库车序列中位于第 n 的出库车的进入和离开码头的时
L_n	出库车序列中位于第 n 的出库车的进入和离开码头的时
X_{mnk}	入库车序列中位于第 m 的入库车转运至出库车序列中位于第 n 的出库车上第 k 种产品的数量
$p_{mi} \in \{0,1\}$	在入库车序列中，若入库车 i 在第 m 位置为 1，否则为 0
$q_{nj} \in \{0,1\}$	在出库车序列中，若出库车 j 在第 n 位置为 1，否则为 0
$W_{mn} \in \{0,1\}$	如果有易腐品从入库车序列中第 m 辆转运至出库车序列中第 n 辆上则为 1，否则为 0

3.2. 模型假设

我们考虑一个简化的越库中心，一侧有一个专用入库码头，另一侧有一个专用出库码头。我们做出以下假设：

- (1) 所有入库和出库卡车在时间 0 时均可使用；
- (2) 所有入库和出库车辆均已到达越库中心等待调度，且每辆入库车上的装载信息和出库车的需求信息已知；
- (3) 上一辆入/出库车离开站台至下一辆入/出库车驶入站台需计算车辆转换时间 D ，该值对于所有车辆是一致的；
- (4) 易腐品被卸载后，从卸货站台转运至装货站台的时间 V 是一致的；
- (5) 产品卸货顺序从 $k = 1$ 开始。装货顺序遵循先到先服务原则；
- (6) 易腐品在越库中转过程中会经历三个操作区域：入库车、越库中心和出库车，而三个操作区域变质率之间的大小关系为：越库中心 $>$ 入库车 $=$ 出库车；
- (7) 不同种类的易腐品有着不同的变质速率；
- (8) 越库中心中有一个临时存储区，其容量是无限的。如果卸载的易腐产品不是出境卡车急需的，则可以暂时存储在临时存储区中。

4. 不同操作区域下质量变质持续的时间

易腐品的变质通常呈指数性腐损[11]，主要是因为微生物繁殖、化学反应和环境因素(如温度、湿度、氧气等)会相互作用并加速腐烂过程。这些因素通常会使腐损过程在初期较为缓慢，但随着时间的推移，

腐蚀速度不断加快, 呈现指数性的特征。因此本文考虑导致不稳定状态的三个主要因素: 产品批量 X_{mnk} 、操作持续时间 t_p 和与操作环境相关的变质率 θ_{kp} 来定义易腐品的累计变质质量(AQD), 公式如下:

$$AQD = \sum_{m=1}^I \sum_{n=1}^J \sum_{k=1}^K \sum_{p=1}^P X_{mnk} \left(q^0 - q^0 e^{-\theta_{kp} t_p} \right) \quad (1)$$

由公式(1)可知, 计算累计变质质量的核心是计算易腐品处于三个操作区域下的变质时间 t_p 。时间范围由 e_m 、 l_m 、 E_n 和 L_n 表示, 对于某一批量货物 X_{mnk} , 都满足条件 $e_m + t_1(m, n, k) + t_2(m, n, k) + t_3(m, n, k) = L_n$, 考虑到实际在越库中心操作时对这一批量的货物是打包卸载和装载的, 我们以下将这一批货物当成一个整体, 其运输和卸载都需要花费 X_{mnk} 单位的时间, 以下是对于某一批量的产品 X_{mnk} 在不同操作区域下的变质时间:

入库车操作区域: 入库车操作区域的时间包括易腐品等待卸载的时间与产品卸载的时间, 对于某一批量的产品 X_{mnk} , 产品 X_{mnk} 的卸载等待时间由两部分组成: (1) 当第 m 辆预定入库卡车在时间 e_m 到达接收码头时, 它首先卸载类型 1 到 $(k-1)$ 的产品。由于我们定义了 $p_{mi} = 1$ 表示第 m 个预定位置被入库卡车 i 占用, r_{ik} 表示入库卡车 i 中装载的第 k 种产品的数量, 所以 $p_{mi} r_{ik}$ 的值表示第 m 个预定位置的入库车 i

中第 k 种产品的数量。所以从产品类型 1 到 $(k-1)$ 的卸载时间为 $\sum_{k'=1}^{k-1} p_{mi} r_{ik'}$; (2) 当它开始卸载第 k 种产品时, 这些产品应首先满足 1 到 $(n-1)$ 辆出境卡车的需求, 卸载时间为 $\sum_{j=1}^{n-1} X_{mjk}$ 。 X_{mnk} 表示卸载这一批货物

X_{mnk} 所花费的时间。则 $t_1(m, n, k) = \sum_{k'=1}^{k-1} p_{mi} r_{ik'} + \sum_{j=1}^{n-1} X_{mjk} + X_{mnk}$ 。

越库中心操作区域: 将 X_{mnk} 装载到第 n 辆预定出库卡车上, 首先需要满足两个条件。条件 1: 从第 m 辆预定入库卡车卸载的产品到达出库码头并准备装载到第 n 辆预定出库卡车上。在这种情况下, 时间等于 $e_m + \sum_{k'=1}^{k-1} p_{mi} r_{ik'} + \sum_{j=1}^{n-1} X_{mjk}$ 。当 X_{mnk} 的第一个产品卸载(需要 1 个单位时间)并经过转运时间 V 后, 产品

到达出库码头并准备装载。将这些值相加, 我们可以得到第一个时间条件: $e_m + \sum_{k'=1}^{k-1} p_{mi} r_{ik'} + \sum_{j=1}^{n-1} X_{mjk} + V + 1$ 。

条件 2: 第 n 辆预定出库卡车已进入出库码头并装载了其他分配的产品。在这种情况下, 在第 n 辆预定出库卡车在 E_n 进入装运码头后, 所有来自预定入库卡车序列的 1 到 $(m-1)$ 的产品将首先被装载, 这段时间等于 $E_n + \sum_{i=1}^{m-1} \sum_{k'=1}^K X_{ink'}$ 。当来自第 m 辆入库卡车的产品开始装载时, 其易腐品首先装载顺序是从类型 1 到

$(k-1)$, 在装载 X_{mnk} 之前的操作时间为 $\sum_{k'=1}^{k-1} X_{mnk'}$ 。一般来说, 批次产品 X_{mnk} 不能在

$\max \left(e_m + \sum_{k'=1}^{k-1} p_{mi} r_{ik'} + \sum_{j=1}^{n-1} X_{mjk} + V + 1, E_n + \sum_{i=1}^{m-1} \sum_{k'=1}^K X_{ink'} + \sum_{k'=1}^{k-1} X_{mnk'} \right)$ 之前装载。结合前面的推导, X_{mnk} 的第一个

产品在 $e_m + \sum_{k'=1}^{k-1} p_{mi} r_{ik'} + \sum_{j=1}^{n-1} X_{mjk} + 1$ 完成卸载过程。因此, 转运持续时间可以表示为这两个值之间的差值,

$$t_2(m, n, k) = \max \left(V, E_n + \sum_{i=1}^{m-1} \sum_{k'=1}^K X_{ink'} + \sum_{k'=1}^{k-1} X_{mnk'} - e_m - \sum_{k'=1}^{k-1} p_{mi} r_{ik'} - \sum_{j=1}^{n-1} X_{mjk} - 1 \right)。$$

出库车操作区域: 因为 $e_m + t_1(m, n, k) + t_2(m, n, k) + t_3(m, n, k) = L_n$, 所以 $t_3(m, n, k)$ 可以通过从 L_n 中减去 $t_1(m, n, k)$ 和 $t_2(m, n, k)$ 得到 $t_3(m, n, k) = L_n - e_m - t_1(m, n, k) - t_2(m, n, k)$ 。

考虑到出库车区域与入库车区域的变质速率相同, 因此把入库车区域与出库车区域的变质时间统一

表示 $G(m, n, k) = L_n - e_m - t_2(m, n, k) X_{mnk} W_{mn}$ 。

5. 数学模型

结合第 4 节的推导, 以下是最小化变质质量累积值(Accumulated Quality Deterioration, AQD)目标函数表达式:

$$AQD = \min \left(\sum_{m=1}^I \sum_{n=1}^J \sum_{k=1}^K X_{mnk} \left(q_k^0 - q_k^0 e^{-[\theta_{k1} G(m, n, k) + \theta_{k2} t_2(m, n, k)]} \right) \right) \quad (2)$$

约束条件:

$$\sum_{m=1}^I Y_{mi} = 1, \quad \forall i \in I \quad (3)$$

$$\sum_{i=1}^I Y_{mi} = 1, \quad \forall m \in I \quad (4)$$

$$\sum_{n=1}^J Z_{nj} = 1, \quad \forall j \in S \quad (5)$$

$$\sum_{m=1}^I X_{mnk} = \sum_{j=1}^J Z_{nj} s_{jk}, \quad \forall n \in J, k \in K \quad (6)$$

$$\sum_{n=1}^J X_{mnk} = \sum_{i=1}^I Y_{mi} r_{ik}, \quad \forall m \in I, k \in K \quad (7)$$

$$\sum_{m=1}^I X_{mnk} = \sum_{j=1}^J Z_{nj} s_{jk}, \quad \forall n \in J, k \in K \quad (8)$$

$$X_{mnk} \leq MW_{mn}, \quad \forall m \in I, n \in J, k \in K \quad (9)$$

$$l_m \geq e_m + \sum_{i=1}^I \sum_{k=1}^K Y_{mi} r_{ik}, \quad \forall m \in I \quad (10)$$

$$e_m \geq l_{m-1} + D, \quad \forall m > 1 \quad (11)$$

$$L_n \geq E_n + \sum_{j=1}^J \sum_{k=1}^K Z_{nj} s_{jk}, \quad \forall n \in J \quad (12)$$

$$E_n \geq L_{n-1} + D, \quad \forall n > 1 \quad (13)$$

$$E_1 \geq V + 1 \quad (14)$$

$$L_n \geq e_m + V + \sum_{k=1}^K X_{mnk} - M(1 - W_{mn}), \quad \forall m \in I, \forall n \in J \quad (15)$$

$$X_{mnk} \geq 0, \quad \forall m \in I, \forall n \in J, \forall k \in K \quad (16)$$

$$e_m \geq 0, \quad \forall m \in I \quad (17)$$

$$l_m \geq 0, \quad \forall m \in I \quad (18)$$

$$E_n \geq 0, \quad \forall n \in J \quad (19)$$

$$L_n \geq 0, \quad \forall n \in J \quad (20)$$

约束条件(3)和(4)指定了入库车及其位置之间的关系。换句话说, 这些约束确定了入卡车在接收码头的顺序。约束条件(5)和(6)确定了出库车在出库码头的顺序。约束条件(7)和(8)是等价的约束。约束条件(9)强制规定了 x_{mnk} 与 w_{mn} 之间的关系。约束条件(10)和(11)基于入库车的位置, 提供了其进出时间的有效顺序。约束条件(12)和(13)对出库车做了同样的规定。约束条件(14)规定了, 排在首位的出库车在第一批货物到达发货码头之前不能开始装载产品, 这个时间等于从入库码头到出库码头的运输时间加上卸货完成一个单位产品所需的时间。约束条件(15)规定, 如果某个产品从入库车 m 运输到出库车 n , 那么出库车最早的离开时间必须大于入库车到达时间加上该产品的运输时间和装载时间。显然, 约束条件(15)将入库和出库卡车的进出时间进行了关联。约束条件(16)到(20)定义了决策变量。

6. 自适应遗传算法

6.1. 算法编码

考虑到本文不同转运情况下, 入库车与出库车的排序不同, 根据问题描述, 本文针对性设计实数编码。本文采用两段式编码, 其中前 $\text{IMaskespan} \geq L_n$ 个表示入库车序列的染色体编码, 后 J 个表示出库车序列的染色体编码, 则有具体含义如下: 有 I 个入库车与 J 个出库车分别在入库码头与出库码头的排序。对上述编码方式举例说明, 假设有 5 辆入库车, 5 辆出库车进行转运, 如表 3 展示了一种可能的编码方案。

Table 3. Chromosome encoding method

表 3. 染色体编码方式

编码内容			序号		
入库车编码	4	3	5	1	2
出库车编码	5	4	3	2	1

6.2. 初始化种群

初始种群通过随机排列生成入库车与出库车序列。例如, 对于 5 辆入库车, 随机生成排列如[3, 1, 4, 2, 5], 确保每个染色体满足唯一性约束。

6.3. 适应度函数

在遗传算法中, 适应度值越高, 被选择的概率就越大, 好的基因就越容易传给下一代, 通常, 研究者采用目标函数的某种形式来计算适应度值。

本文研究的是易腐品越库车辆调度问题, 目的是通过调度入/出库车辆卸/装载顺序, 尽可能地降低易腐品在转运过程中的质量损耗。因此, 累计变质质量较小的染色体对应的适应度值就高, 相反, 累计变质质量较大的染色体对应较低的适应度值。

由于累计变质质量具有非负、连续且不易提前获知范围的特点, 本文提出采用反比例形式下的适应度函数, 如式(21)所示。

$$f_1 = \frac{1}{AQD} = \frac{1}{\sum_{m=1}^I \sum_{n=1}^J \sum_{k=1}^K x_{mnk} (q_k^0 - q_k^0 e^{-[\theta_{k1} G(m,n,k) + \theta_{k2} t_2(m,n,k)]})} \quad (21)$$

对于经典的完工时间(Maskespan)模型, 其目标函数值应不小于任何一辆出库卡车离开出口库码头的

时间, 即 $Makespan \geq L_n$, 其适应度函数如式(22)所示。

$$f_2 = \frac{1}{Makespan} = \frac{1}{L_n} \quad (22)$$

6.4. 选择操作

本文选择轮盘赌选择的方法, 根据个体的适应度从当前种群中随机选择父代个体。选择的概率与个体的适应度成正比, 适应度越高的个体被选择的概率越大。

6.5. 自适应交叉操作与变异操作

自适应遗传算法是通过对参数的自适应调整, 使得整个遗传算法不仅在收敛精度、收敛速度上得到了极大的提高, 还能不断保持种群的多样性, 其设计原则主要包括以下两点:

a: 在种群进化初期, 应尽量使种群在较大范围内进行搜索, 避免过早收敛;

b: 在种群进化后期, 当搜索结果趋近最优解时, 应通过进化策略, 使种群重点在局部范围内进行搜索。

自适应交叉概率: 针对每代种群的适应度值分布, 构造如下自适应交叉概率公式[12]。

$$p_c = \begin{cases} \frac{\sigma_1(f_{\max} - f)}{f_{\max} - f_{\text{avg}}}, & \text{如果 } f \geq f_{\text{avg}} \\ \sigma_2, & \text{如果 } f < f_{\text{avg}} \end{cases} \quad (23)$$

其中, $f_{\max}$ 指种群中最大的适应度值, f_{avg} 指种群中的平均适应度值, f 指要交叉的两个染色体中较大的适应度值, σ_1, σ_2 取 (0,1) 区间的值即可实现自适应调整。

交叉算子: 选用部分映射交叉, 其操作流程如图 2 所示。



Figure 2. Chromosome crossover process

图 2. 染色体交叉过程

变异算子: 选用交换变异算子, 即随机选取两个卸/装货位置进行交换, 若交换后的染色体适应度值高于交换前的, 则接受变异操作, 若交换后的适应度值低于交换前的, 则保留原先染色体。类似地, 自适应变异概率公式如下:

$$p_m = \begin{cases} \frac{\sigma_3(f_{\max} - f)}{f_{\max} - f_{avg}}, & \text{如果 } f \geq f_{avg} \\ \sigma_4, & \text{如果 } f < f_{avg} \end{cases} \quad (24)$$

其中, σ_3 , σ_4 取 (0,1) 区间的值即可实现自适应调整。其变异操作流程如图 3 所示。

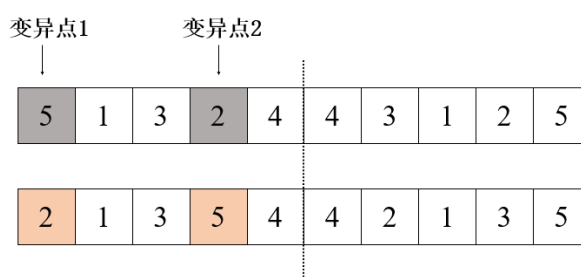


Figure 3. Chromosome mutation process
图 3. 染色体变异过程

6.6. 算法步骤

综合以上分析, 自适应遗传算法求解步骤如下:

- 步骤 1: 初始化算法参数。包含种群数量 N_p 、选择概率 P_s 、最大迭代次数 N_g 、自适应参数 σ_1 、 σ_2 、 σ_3 、 σ_4 等;
- 步骤 2: 生成初始解集, 计算其对应的总衰减量值;
- 步骤 3: 采用总衰减量值的倒数形式, 计算每条染色体的适应度值;
- 步骤 4: 采用轮盘赌策略选择出需要交叉的染色体, 并按照自适应的交叉概率, 执行部分映射交叉操作;
- 步骤 5: 将执行后交叉操作的染色体, 按自适应变异概率, 执行变异操作;
- 步骤 6: 计算执行完变异操作后每条染色体的适应度值;
- 步骤 7: 记录适应度值最高和最低的染色体序号, 并用具有最高适应度值的染色体替换掉最低适应度值的染色体, 同时, 对应的适应度值也被替换掉;
- 步骤 8: 判断是否满足终止条件, 满足则结束, 否则转入步骤 4。

7. 算例分析

由于本文研究的越库车辆调度问题在国际上尚无标准测试数据, 本文根据实际情况随机产生大量的实验数据来验证自适应算法与模型的有效性。根据产品数目、车量总数、产品种类数的不同将问题分为小、中、大 3 个规模, 见表 4, 每个算例进行 10 次实验。

场景方面为体现出不同种类易腐品之间的变质速率的差异, 本文算例设置前 50% 种类的易腐品为高变质率产品, 后 50% 种类的易腐品为低变质率产品。其不同环境下的变质速率见表 5。

参数取值方面, 取 $D = 60$, $V = 100$, $N_p = 1000$, $P_s = 0.9$, $\sigma_1 = \sigma_2 = 0.8$, $\sigma_3 = \sigma_4 = 0.05$, $N_g = 500$, $q^0 = 1$ 。

Table 4. Attribute values of the example
表 4. 算例各属性值

产品数规模	小			中			大		
车辆数规模	小	中	大	小	中	大	小	中	大
算例类别	1	2	3	4	5	6	7	8	9
产品总数	400	400	400	800	800	800	1200	1200	1200
产品种类数	2	2	2	4	4	4	6	6	6
入库车辆数	6	12	24	6	12	24	6	12	24
出库车辆数	4	8	16	4	8	16	4	8	16

Table 5. Setting of deterioration rate parameter values
表 5. 变质速率参数值设定

操作区域	低变质速率产品	高变质速率产品
入库车操作区域每单位时间变质量	0.000005	0.00005
出库车操作区域每单位时间变质量	0.000005	0.00005
越库中心每单位时间变质量	0.00001	0.0001

7.1. AQD 的模型

为了探索基于 AQD 的模型的性能，我们列出详细的评估过程如下：

步骤 1：使用上一节中提出的自适应遗传算法求解 AQD 模型。同时可以获得相应的卡车调度序列。

步骤 2：使用从第一步获得的卡车调度序列计算完工时间(Makespan)。

步骤 3：使用上一节中提出的自适应遗传算法求解基于完工时间的模型。用完工时间百分比偏差 (Makespan Percentage Deviation, MPD)测量差异率(步骤 2 的完工时间结果与步骤 3 的结果之间的差异)，MPD 由公式(24)计算：

$$MPD(\%) = \frac{Makespan(\text{步骤}2) - Makespan(\text{步骤}3)}{Makespan(\text{步骤}3)} \times 100\%$$

(25)

实验结果见表 6，我们得到了三个实例{1, 2, 4}，它们在第三步中产生了相同的完工时间(最优完工时间)。在 9 个实例中，最大 MPD 为 7.25%，平均 MPD 仅为 3.21%。这意味着基于 AQD 的模型不仅能产生最优的 AQD，还能提供令人满意的完工时间。

Table 6. The MPD of ADRV-based model
表 6. 基于 APD 模型下的 MPD

算例类别	Makespan (步骤 2)	Makespan (步骤 3)	MPD
1	730	730	0
2	1090	1090	0
3	1880	1810	3.87%
4	1285	1285	0
5	1570	1495	5.02%
6	2003	1886	6.20%
7	1789	1695	5.55%
8	2027	1890	7.25%
9	2714	2686	1.04%
平均值			3.21%

7.2. Makespan 的模型

同样, 我们对基于完工时间的模型使用上述三个步骤来探索其在 AQD 方面的性能。

步骤 1: 使用本文设计的自适应遗传算法求解基于完工时间的模型, 我们获得其相应的卡车调度序列。对于每个最优完工时间结果, 卡车调度序列不是唯一的, 这意味着求解完工时间模型可能会产生不同的卡车调度序列。在表 7 中, 我们以 9 个实例中的随机一个实例为例, 说明了最优解及其相关的可能序列。如果入库卡车按 2-1-3-4-6-5 的顺序卸载, 出库卡车按 4-3-2-1 的顺序装载, 则最优完工时间为 1695。如果我们为出库卡车选择另一个序列 4-3-1-2, 我们将获得相同的完工时间但不同的 AQD。在本文中, 我们记录了 9 个算例下每个最优完工时间相关的所有可能的调度序列。

步骤 2: 使用从第一步获得的卡车调度序列, 我们计算其相关的 AQD。我们计算每个实例的所有最优卡车调度序列的最小、最大和平均 AQD。

步骤 3: 求解基于 AQD 的模型。用 AQD 百分比偏差(APD)测量差异率(步骤 2 的 AQD 结果与步骤 3 的结果之间的差异)。APD 由公式(25)计算:

$$APD(\%) = \frac{APD(\text{步骤2}) - APD(\text{步骤3})}{APD(\text{步骤3})} \times 100\% \quad (26)$$

由步骤 3 得出的 AQD 最优解记录在表 8 的第一列, 基于步骤 1 获得的所有可能序列的最大、平均和最小 AQD 记录在第 3、4 和 5 列中, APD 总结在第 6、7 和 8 列中。9 个实例的 APD 的最大、平均和最小值分别为 50.05%、34.15%和 18.16%。换句话说, 基于 Makespan 的模型提供的解决方案会产生显著较大的 AQD。

Table 7. Different truck scheduling sequences of the Makespan-based model

表 7. 基于完工时间模型的不同卡车调度序列

算例类别	Makespan (最优值)	序列总数	入库车序列	出库车序列
7	1695	8	2-1-3-4-6-5	4-3-2-1
			2-1-3-4-6-5	4-3-1-2
			2-1-3-4-5-6	4-3-2-1
			2-1-3-4-5-6	4-3-1-2
			1-2-3-4-6-5	4-3-2-1
			1-2-3-4-6-5	4-3-1-2
			1-2-3-4-5-6	4-3-2-1
			1-2-3-4-5-6	4-3-1-2

Table 8. APDs of Makespan-based model

表 8. 基于 Makespan 模型下的 APDs

算例类别	最优 AQD	最大 AQD	平均 AQD	最小 AQD	最大 APD	平均 APD	最小 APD
1	3.36	4.61	3.99	3.36	37.20%	18.75%	0
2	3.35	6.04	4.70	3.35	80.30%	40.30%	0
3	4.57	7.77	7.35	6.93	70.02%	60.83%	51.64%
4	11.33	14.81	13.07	11.33	30.71%	15.36%	0

续表

5	12.66	16.13	15.91	15.68	27.41%	25.67%	23.85%
6	15.19	19.43	18.06	16.68	27.91%	18.89%	9.81%
7	29.70	47.26	42.40	37.54	59.12%	42.76%	26.40%
8	24.37	39.15	34.40	29.64	60.65%	41.16%	21.62%
9	30.06	47.22	43.17	39.12	57.09%	43.61%	30.14%
平均值					50.05%	34.15%	18.16%

7.3. APD 与 Makespan 模型性能对比

我们将表 6 中基于 AQD 模型下得出的 MPD 与表 8 中基于 Makespan 模型下的平均 APD 记录在表 9。由表 9 可知，基于 AQD 的模型的 MPD 非常小，而基于 Makespan 的模型的 APD 要大得多。这一结果意味着，如果越库中心以最小化 AQD 为目标安排卡车序列，完工时间将略高于或等于最优 Makespan。所有 9 个实例的平均 MPD 仅为 3.21%。然而，如果越库中心采用最小化 Makespan 为目标安排卡车序列，平均 APD 将增加到 34.15%，并且在每个实例下均表现出 MPD 要远远小于 APD。这些结果充分证明了基于 AQD 的模型在易腐产品卡车调度问题上相对于基于 Makespan 的模型的优越性。

Table 9. Comparison between MPD and APD
表 9. MPD 与 APD 对比

算例类别	MPD (AQD 模型)	APD (Makespan 模型)	差值(MPD 与 APD)
1	0	18.75%	18.75%
2	0	40.30%	40.30%
3	3.87%	60.83%	56.96%
4	0	15.36%	15.36%
5	5.02%	25.67%	20.65%
6	6.20%	18.89%	12.69%
7	5.55%	42.76%	37.21%
8	7.25%	41.16%	33.91%
9	1.04%	43.61%	42.57%
平均值	3.21%	34.15%	30.94%

7.4. 变质速率敏感性分析

为验证模型对变质速率参数的鲁棒性，选取算例 3 进行参数敏感性测试。设定以下两组高变质速率产品在越库中心的变质速率，低变质速率产品在不同操作区域下的变质速率与表 5 保持一致。

组 1：越库中心变质速率提高至 0.00012，出入库车辆速率保持不变(高变质场景)。

组 2：越库中心变质速率降低至 0.00008，出入库车辆速率保持不变(低变质场景)。

通过表 10 对比不同参数组下的 MPD 与 APD 变化，并以表 5 的数据作为基准组，结果表明在本文设置的情景下 MPD 或 APD 与高变质速率产品在越库中心的变质速率存在正相关关系。

Table 10. Sensitivity analysis results for deterioration rate
表 10. 变质速率敏感性分析结果

组别	MPD (AQD 模型)	APD (Makespan 模型)	变化值(MPD)	变化值(APD)
基准组	3.87%	60.83%		
1	4.14%	79.38%	0.27%	18.55%
2	1.66%	50.93%	-2.21%	-9.9%

8. 结论

本研究受到物联网和 5G 技术的最新创新和改进的推动, 这些技术催生了许多新的服务运营模式。现代服务平台连接和匹配数百万供应商和买家, 同时提供有效的中介服务。越库服务平台的效率和风险管理对公司至关重要。

在复杂的越库服务平台背景下, 我们在易腐库存管理领域引入了不稳定状态的新概念。我们研究了与不稳定状态相关的变质率变化风险的影响。在操作条件动态和操作过程转换频繁的情况下, 这种影响非常显著。我们在越库操作优化问题中研究了这种影响, 这是一个经典的物流问题, 其特点是一系列具有环境变化风险的处理操作过程。

我们基于自适应的遗传算法使用包含 9 个实例的数据集对模型进行测试, 并分析不同模型下的 MPD 与 APD。结果表明, 基于经典的 Makespan 的模型可能会导致巨大的 AQD, 而基于 AQD 的模型可以在增加一点 Makespan 的情况下获得最优的 AQD 值。这表明, 当越库操作中的不同易腐品在变质率上存在差异时, 基于 AQD 的模型表现更好。

对于未来的研究, 所提出的模型可以扩展到包括多个入库和出库码头。另一方面也可以探索新的高效启发式和元启发式方法来解决所提出的问题。

基金项目

上海市白玉兰人才计划浦江项目(23PJC074); 国家自然科学基金(72202137)。

参考文献

- [1] Taylor, T.A. (2018) On-demand Service Platforms. *Manufacturing & Service Operations Management*, **20**, 704-720. <https://doi.org/10.1287/msom.2017.0678>
- [2] 徐静, 朱玉, 戴盼倩, 等. 生鲜农产品冷链物流过程优化研究[J]. 中国农机化学报, 2024, 45(2): 115-121.
- [3] Taoukis, P.S. and Labuza, T.P. (1989) Applicability of Time-Temperature Indicators as Shelf Life Monitors of Food Products. *Journal of Food Science*, **54**, 783-788. <https://doi.org/10.1111/j.1365-2621.1989.tb07882.x>
- [4] Rong, A., Akkerman, R. and Grunow, M. (2011) An Optimization Approach for Managing Fresh Food Quality Throughout the Supply Chain. *International Journal of Production Economics*, **131**, 421-429. <https://doi.org/10.1016/j.ijpe.2009.11.026>
- [5] Zaerpour, N., Yu, Y. and deKoster, R.B.M. (2015) Storing Fresh Produce for Fast Retrieval in an Automated Compact Cross-dock System. *Production and Operations Management*, **24**, 1266-1284. <https://doi.org/10.1111/poms.12321>
- [6] Fischer, D., Craig, W., Watada, A., Douglas, W. and Ashby, B. (1992) Simulated In-Transit Vibration Damage to Packaged Fresh Market Grapes and Strawberries. *Applied Engineering in Agriculture*, **8**, 363-366. <https://doi.org/10.13031/2013.26078>
- [7] Mousavi, S.M. and Vahdani, B. (2017) A Robust Approach to Multiple Vehicle Location-Routing Problems with Time Windows for Optimization of Cross-Docking under Uncertainty. *Journal of Intelligent & Fuzzy Systems*, **32**, 49-62. <https://doi.org/10.3233/jifs-151050>
- [8] Vahdani, B. and Zandieh, M. (2010) Scheduling Trucks in Cross-Docking Systems: Robust Meta-Heuristics. *Computers & Industrial Engineering*, **58**, 12-24. <https://doi.org/10.1016/j.cie.2009.06.006>

-
- [9] Rahmanzadeh Tootkaleh, S., Fatemi Ghomi, S.M.T. and Sheikh Sajadieh, M. (2016) Cross Dock Scheduling with Fixed Outbound Trucks Departure Times under Substitution Condition. *Computers & Industrial Engineering*, **92**, 50-56. <https://doi.org/10.1016/j.cie.2015.12.005>
 - [10] Thibaud, M., Chi, H., Zhou, W. and Piramuthu, S. (2018) Internet of Things (IoT) in High-Risk Environment, Health and Safety (EHS) Industries: A Comprehensive Review. *Decision Support Systems*, **108**, 79-95. <https://doi.org/10.1016/j.dss.2018.02.005>
 - [11] Wang, X. and Li, D. (2012) A Dynamic Product Quality Evaluation Based Pricing Model for Perishable Food Supply Chains. *Omega*, **40**, 906-917. <https://doi.org/10.1016/j.omega.2012.02.001>
 - [12] Shojaedini, E., Majd, M. and Safabakhsh, R. (2019) Novel Adaptive Genetic Algorithm Sample Consensus. *Applied Soft Computing*, **77**, 635-642. <https://doi.org/10.1016/j.asoc.2019.01.052>