

非支配排序遗传算法II求解越库配送车辆路径问题

李 雯

上海理工大学管理学院, 上海

收稿日期: 2024年12月9日; 录用日期: 2025年1月2日; 发布日期: 2025年1月13日

摘 要

越库是一种有效的物流策略, 然而, 物流费用高、客户满意度低仍是越库配送模式存在的主要问题。本文基于软时间窗约束, 研究了带软时间窗的越库配送车辆路径问题(VRPCDTW)。模型的目标是使经济成本之和最小化, 平均顾客满意度最大化。结合模型的特征, 提出了一种改进的非支配排序遗传算法II进行求解。最后, 实验数据表明, 通过与其他多目标进化算法进行比较, 该算法能在不同规模算例下有效求解模型, 算法综合性能更高。所提出的方法有效地降低了总配送成本, 提高了客户满意度。

关键词

越库配送, 车辆路径问题, NSGA-II算法

Undominated Sorting Genetic Algorithm II for Solving the Cross-Docking Distribution Vehicle Path Problem

Wen Li

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

Received: Dec. 9th, 2024; accepted: Jan. 2nd, 2025; published: Jan. 13th, 2025

Abstract

Cross-docking is an efficient logistics strategy; however, high logistics costs and low customer satisfaction remain key issues in cross-docking distribution. This paper investigates the Vehicle Routing Problem with Cross-Docking and Time Windows (VRPCDTW) under soft time window constraints. The objective of the model is to minimize total economic costs and maximize average

customer satisfaction. Based on the model's characteristics, an improved Non-dominated Sorting Genetic Algorithm II (NSGA-II) is proposed to solve the problem. Experimental results demonstrate that, compared with other multi-objective evolutionary algorithms, the proposed algorithm effectively solves the model across various instance scales and shows superior overall performance. The method successfully reduces total distribution costs and enhances customer satisfaction.

Keywords

Cross-Docking Distribution, Vehicle Routing Problem, NSGA-II 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

越库是一种物流配送模式，通过将供应商的货物直接转移到配送车辆上，从而减少仓储需求和中间库存。越库设施仅用作货物的中转和重新分配，而不进行长时间的储存，从而实现物流效率的提升和成本的降低。与传统配送模式相比，越库能够显著缩短配送时间，减少仓储成本，并降低整体碳排放。尤其在时效性要求高的配送场景中，越库具有明显的优势。

越库配送车辆路径问题(Vehicle routing problem with cross-docking, VRPCD)是在经典车辆路径问题(VRP)的基础上引入越库设施而形成的一类复杂优化问题，最早由 Lee 等[1]首次提出，通过对运输路径和越库转运过程的协同优化，实现了物流系统的总体成本最小化。Liao 等[2]证明了越库操作不仅能减少存储成本，还可以提高订单响应速度，使得产品能够更快到达客户手中。由于 VRPCD 属于 NP 难问题，传统的精确算法在处理大规模实例时计算效率低，因此各种启发式与元启发式算法得到了广泛应用。Mousavi 等[3]采用了基于粒子群优化(PSO)的算法，能够快速生成接近最优解的可行方案。Zhou 等[4]通过改进遗传算法(GA)求解 VRPCD，在保证求解精度的前提下提高了收敛速度。此外，Afshar-Bakeshlo 等[5]提出了混合算法，将蚁群优化(ACO)与模拟退火(SA)结合，用于求解包含多个越库的复杂路径问题。迭代局部搜索算法[6]、大邻域搜索算法[7]等也被用于求解该问题，有效改进了最优解的质量。Wen 等[8]在已有研究的基础上调整了约束条件的限制，建立了以总运输成本最小化的混合整数规划模型。Abad 等[9]考虑到越库中心的集成合并以及硬时间窗约束，尽可能在降低系统成本的同时缓解对环境的影响。然而，对时间约束的硬性要求往往伴随着高额成本。因此，在既考虑客户体验又考虑经济效益的情况下，软时间窗约束显然更适用于实际情况。

针对具有软时间窗的 VRPCD，Agustina 等[10]通过为客户设置时间限制，计算惩罚成本，在提升用户体验的同时尽量减少配送费用。刘虹等[11]同时考虑了协同到库、分拣和客户时间窗，以配送总成本最小为优化目标，建立越库配送路径多目标优化模型。然而，只关注交付环节的时间窗口是有所不足的。Acevedo-Chedid 等[12]则同时考虑拾取与交付的柔性时间约束，并基于易腐货物特性，在确保时间要求的前提下，优化取货、整合及发运步骤，以降低整体运输开销并提高上下游供应链服务。张政等[13]通过惩罚成本来衡量违反时间窗所造成的影响，并采用加入自适应机制的禁忌搜索算法优化总成本。但上述文献均将总成本最小作为唯一目标，没有基于客户角度衡量满意度，提高服务水平。

因此，本文基于越库配送模式，建立带软时间窗的越库配送车辆路径问题(VRPCDTW)模型，优化拾取交付路径，实现总配送成本最低，客户满意度最大化，在保证企业盈利能力的基础上使客户服务水平

更高,并提出一种改进的 NSGA-II 算法来求解该问题。

2. 越库配送问题的描述和模型

2.1. 问题描述

越库配送网络中存在一个越库中心、若干个供应商与零售商,如图 1 所示,越库中心首先安排车辆去拾取节点拾取货物,回到越库中心以后直接进行分拣分装,不进行仓储作业,之后再安排车辆去各个交付节点送货,最后返回越库中心。拾取节点和交付节点的供需关系是一一对应的。

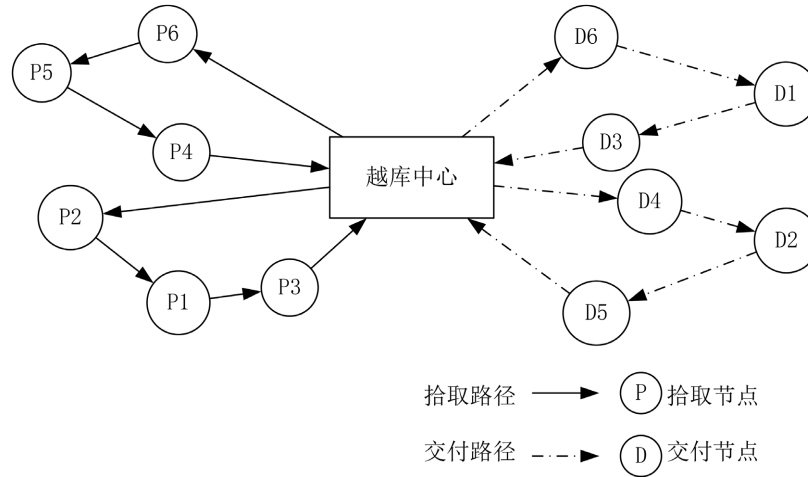


Figure 1. Cross-docking network
图 1. 越库网络示意图

2.2. 越库配送车辆路径问题模型

将 VRPCDTW 定义为一个有向图 $G=(N,E)$, 拾取节点的集合表示为 $P=\{1,\dots,n\}$, 交付节点的集合表示为 $D=\{n+1,\dots,2n\}$ 。其中, 越库网络中所有点的集合 $N=P\cup O\cup D$, 集合 E 表示网络中所有的可行弧。

在对模型参数的设置中, d_{ij} 表示弧 (i,j) 间的距离, η 表示车辆单位距离的运输成本, φ 表示每辆车的固定成本, v 表示车辆的行驶速度, b_i 表示货物量 ($i\in P\cup D$), Q_m 表示车辆最大载重量, α 表示在越库中心装卸货的固定时间, β 表示每托盘货物的单位装卸货时间。

在对模型变量的设置中, X_{ijk} 为 0-1 变量, 如果车辆 k 在弧 (i,j) 上行驶, 则等于 1, 否则为 0。 y_{ik} 为 0-1 变量, 表示车辆 k 是否访问节点 i 。 s_{ik} 为 0-1 变量, 表示车辆 k 是否在越库中心卸载节点 i 的货物。 h_{ik} 为 0-1 变量, 表示车辆 k 是否在越库中心装载节点 i 的货物。 t_{ijk} 为应变量, 表示车辆 k 在 i,j 两点间的行驶时间; L_{ik} 为应变量, 表示车辆 k 离开节点 i 的时间。 e_k 为应变量, 表示车辆 k 在越库中心完成卸载的时间。 r_k 为应变量, 表示车辆 k 在越库中心开始装载的时间。 d_i 为应变量, 表示节点 i 的货物在越库中心完成卸载的时间。 Ar_{ik} 为应变量, 表示车辆 k 到达节点 i 的时间。

建立多目标模型。式(1)的目标为系统总成本最低, 式(2)的目标为客户满意度最大化。

$$\min f_1 = \sum_{i\in N} \sum_{j\in N} \sum_{k\in K} X_{ijk} d_{ij} \eta + \sum_{j\in P} \sum_{k\in K} X_{ojk} \varphi \quad (1)$$

$$\max f_2 = \sum_{i\in N} ACS_i (L_{ik}) \quad (2)$$

本文的主要目的是提高企业经济效益的同时最大化客户满意度, 因此为了更加精准衡量客户满意度, 为每个供应商和零售商都定义了一个期望到达的时间窗口 $[e_i, l_i]$ 内和可以最大容忍到达的时间窗口 $[E_i, L_i]$ 。建立了客户满意度评价模型, 图 2 中描绘了客户 i 的满意度函数曲线。其中, 横坐标代表时间, 纵坐标代表客户满意度, 当车辆在 $[e_i, l_i]$ 内完成服务, 客户满意度为 100; 车辆在 $[E_i, e_i]$ 或 $[l_i, L_i]$ 服务客户时, 客户满意度有所降低, 随着与最佳服务时间段偏差的增大而降低。如果车辆在 E_i 之前或 L_i 之后向客户提供服务, 则客户满意度为 0。 L_{ik} 是车辆 k 离开客户 i 的时间。客户满意度的计算公式如式(3)所示。

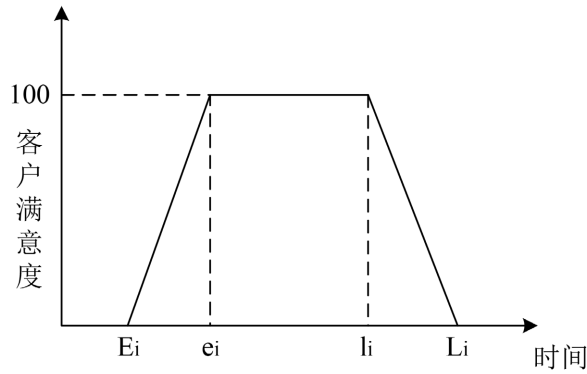


Figure 2. Customer satisfaction curve

图 2. 客户满意度曲线

$$ACS_i(L_{ik}) = \begin{cases} 0, L_{ik} \leq E_i \\ \left(\frac{L_{ik} - E_i}{e_i - E_i} \right) \times 100, E_i < L_{ik} \leq e_i \\ 100, e_i < L_{ik} \leq l_i \\ \left(\frac{L_i - L_{ik}}{L_i - l_i} \right) \times 100, l_i < L_{ik} \leq L_i \\ 0, L_i > L_{ik} \end{cases} \quad (3)$$

约束条件:

$$\sum_{j:(i,j) \in E} \sum_{k \in K} X_{ijk} = 1, \forall i \in P \cup U \quad (4)$$

$$\sum_{i \in P \cup D} \sum_{j:(i,j) \in E} b_i X_{ijk} \leq Q_m, \forall k \in K \quad (5)$$

$$\sum_{j:(j,i) \in E} X_{jik} = \sum_{j:(i,j) \in E} X_{ijk}, \forall i \in P \cup U \quad (6)$$

$$Ar_{jk} \geq Ar_{ik} + t_{ijk} - M(1 - X_{ijk}), \forall (i, j) \in E, k \in K \quad (7)$$

$$L_{ik} = Ar_{ik} + \beta b_i, \forall k \in K \quad (8)$$

$$E_i y_{ik} \leq Ar_{ik} \leq L_i y_{ik}, \forall i \in N, k \in K \quad (9)$$

$$s_{ik} - h_{ik} = \sum_{j \in P} X_{ijk} - \sum_{j \in D} X_{(i+n)jk}, \forall k \in K, i \in P \quad (10)$$

$$s_{ik} + h_{ik} \leq 1, \forall k \in K, i \in P \quad (11)$$

$$r_k \geq e_k, \forall k \in K \quad (12)$$

$$r_k \geq d_i - M(1 - h_{ik}), \forall i \in P, \forall k \in K \quad (13)$$

$$d_i \geq e_k - M(1 - s_{ik}), \forall i \in P, \forall k \in K \quad (14)$$

$$X_{ijk}, y_{ik}, s_{ik}, h_{ik} \in \{0, 1\} \quad (15)$$

$$t_{ijk}, L_{ik}, e_k, r_k, d_i, Ar_{ik} \geq 0 \quad (16)$$

约束(4)表示每个节点只能被一辆车服务一次。约束(5)表示每辆车的负载不能超过车辆最大负载量。约束(6)表示流量守恒约束。约束(7)表示车辆 k 到达节点 j 的时间。约束(8)计算车辆 k 离开节点 i 的时间。约束(9)表示车辆时间窗口的要求。约束(10)和(11)判断 k 是否在越库中心卸载或者装载节点 i 的货物。约束(12)和(14)表示车辆在完成卸载之前不能开始重新装载。约束(15)和(16)定义变量的范围。

3. 算法设计

由于 VRPCDTW 是 NP 难问题, 精确计算难以在短时间内得到结果, 因此需要通过启发式算法来求解。NSGA-II 算法是解决车辆路径问题有效的方法, 但仍存在陷入局部最优的问题。变邻域搜索算子通过对部分个体进行邻域优化, 增强局部搜索能力, 使解逐步收敛至 Pareto 前沿。可以减少对不良解的重新评估, 降低计算成本。将变邻域搜索算子融入 NSGA-II 中, 能在优化质量、种群多样性和计算效率间取得更好的平衡。具体流程会包括初始解生成、选择、交叉变异、非支配排序与拥挤距离计算以及变邻域算子搜索过程。

3.1. 选择

本文结合最优保护策略和轮盘赌选择方法来选择染色体。最优保护策略是一种保留优秀解的方法。在每一代中, 遗传算法会确保当前种群中的最优个体被直接传递到下一代, 而不进行交叉或变异操作。这可以防止优秀解在进化过程中被破坏, 从而加速收敛速度, 提高解的质量和稳定性。轮盘赌选择方法是一种基于概率的个体选择策略。每个个体的选择概率与其适应度值成正比, 适应度值越高的个体被选择的可能性越大。

3.2. 交叉和变异

在父代染色体 A 和 B 上随机选择两个交叉点, 整条染色体被分为前中后三个片段。将父代染色体 A 的中间片段移除, 插入至染色体末端位置, 本来在末尾的片段前移, 与最前面的片段连接, 各个片段内的基因顺序保持不变。将父代染色体 B 的中间片段移除, 插入至染色体最前端位置, 本来最前面的染色体片段后移, 与末尾的片段连接, 各个片段内的基因顺序保持不变。交叉后将所有重复的基因删除。变异操作则是在父代染色体上随机选择两个突变节点, 将这两个节点交换位置完成变异操作, 形成新的子代染色体。

3.3. 非支配排序与拥挤距离计算

计算群体中所有染色体的目标函数值, 并计算拥挤距离。计算公式如式(19)所示, 其中, $F_m^{\max}$ 和 F_m^{i+1} 分别是第 m 个目标函数的最大值和最小值, 而 F_m^{i+1} 和 F_m^{i-1} 是与第 i 个解相邻的两个解的第 m 个目标函数值。

$$CD_i = \sum_{m=1}^M \frac{F_m^{i+1} - F_m^{i-1}}{F_m^{\max} - F_m^{\min}} \quad (17)$$

根据 Pareto 最优前沿和拥挤距离选择新的种群。然后, 判断是否达到最大迭代次数。如果没有达到

最大迭代次数，则返回到遗传进化过程并继续迭代。当达到最大迭代次数时，终止算法并输出帕累托最优前沿。

3.4. 变邻域搜索算子

本文采用三种经典邻域搜索算子：2-opt 算子、单节点移动算子以及双节点移动算子。操作邻域搜索算子时，随机选择一个节点，计算它与所有其他节点之间的距离，并按顺序生成距离列表。以下是三种变邻域搜索算子的具体过程。

1) 2-opt 算子：从染色体中随机选择节点 i ，并从与节点 i 的距离列表里选择第一个节点 j 。2-opt 算子首先将节点 i 与其后面的节点断开，将节点 j 与其后面的节点也断开，再将中间断开的染色体片段翻转，使节点 i 与节点 j 连接，其余节点保持不变。

2) 单节点移动算子：从染色体中随机选择节点 i ，并从与节点 i 的距离列表里选择第一个节点 j 。单节点移动算子将节点 j 单独移除，并插入到节点 i 之后的位置，使节点 i 与节点 j 连接，其余节点保持不变。

3) 双节点移动算子：随机选择两个相邻的染色体节点，将其中的第一个作为节点 i ，从与节点 i 的距离列表里选择第一个节点 j 。将节点 i 与之相邻的节点单独移除，并插入到节点 j 之后的位置，使节点 i 与节点 j 连接，其余节点保持不变。

按照顺序进行邻域算子的搜索，如果找到更优的个体，则保留此操作，否则，继续选择距离列表里的下一个节点，重复该过程，直到找到更好的解或达到最大搜索次数。

4. 仿真实验

4.1. 算例设置

实验基于文献[8]公开的 VRPCD 标准算例进行修改加以生成不同规模的基准算例。其中：各个节点和越库中心的坐标、供应量、需求量、时间窗范围数据均选自 VRPCD 数据集。车辆单位距离的运输成本 η 设置为 7，每辆车的固定成本 φ 设置为 30。

算法编程使用 MATLAB R2018b，电脑操作系统为 Window10，运行内存为 8 G，CPU 为 Intel(R)Core(TM)i7-7700，主频为 3.60 GHz。具体算法参数设置如下：种群规模 $\text{pop_size} = 150$ ，最大迭代次数 $\text{Maxgen} = 150 \sim 300$ ，由于算例规模不同，越大规模的算例需要更大的迭代次数以接近真实 Pareto 解，因此为各算例设置不同的迭代次数。

4.2. 测试算法的性能

将本文设计的算法与 MOGE 和 NSGA-II 算法进行比较。算法收敛结果的比较如图 3 所示。该实验使用 100 个客户的数据集进行。图 3 的水平坐标表示算法的迭代次数，左边的垂直坐标表示总成本，右边的垂直坐标表示客户满意度。实线表示不同算法得到的总成本的最优值，虚线表示不同算法得到的顾客满意度的最优值。图 3 显示了这三种算法在收敛方面的差异。本文提出的算法在总成本和满意度方面相较其他两种算法收敛迅速，基本在 140 次完成收敛。而 MOGE 和 NSGA-II 算法在 200 次迭代内都没有收敛。因此，本文提出的算法具有更快的收敛速度，可以生成更好的目标函数值。

表 1 显示了三种算法得到的 Pareto 前沿中单个优化目标的最优值。可以看出改进的 NSGA-II 算法在小规模至大规模算例上都能得到最低的总成本与最高的客户满意度值。Pro1 与 Pro2 分别表示与 NSGA-II 和 MOGE 算法相比下的改进程度。与 NSGA-II 算法相比，总成本降低了 12.4%，客户满意度提高了 6.3%。与 MOGE 算法相比，总成本降低了 10.2%，客户满意度提高了 3.9%。此外，随着算例规模的扩

大，改进程度也更加显著。因此，本文提出的算法能够搜索到更高质量的解，算法的综合性能更好。

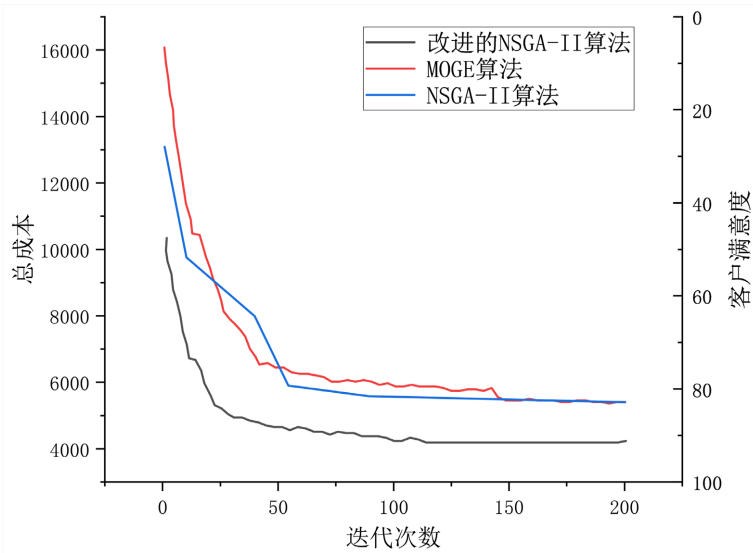


Figure 3. Number of iterations
图 3. 算法收敛性比较

Table 1. Comparative results of optimization objectives
表 1. 优化目标的比较结果

No.	改进后 NSGA-II		NSGA-II		MOGE		与 NSGA-II 比较		与 MOGE 比较	
	总成本	满意度	总成本	满意度	总成本	满意度	Pro1	Pro2	Pro1	Pro2
50	2200.74	96.43	2427.67	92.79	2376.96	93.76	10.3%	3.9%	8.0%	2.8%
100	5517.62	95.77	6205.91	89.64	6065.05	92.97	12.5%	6.8%	9.9%	3.0%
150	8026.43	93.29	9085.77	87.93	8945.43	89.45	13.2%	6.1%	11.4%	4.3%
200	11499.09	91.06	13043.41	84.12	12829.36	86.26	13.4%	8.3%	11.6%	5.6%
平均值	6810.97	94.14	7690.69	88.62	7554.21	90.61	12.4%	6.3%	10.2%	3.9%

4.3. 服务时效分析

对比 3 种不同的客户时间窗长度，其中，case1 为本文设定的时间窗长度；case2 为将所有客户时间窗的最早和最晚时间均缩小 30；case3 为将所有客户时间窗的最早和最晚时间均增加 30。表 2 为不同时间窗长度下求出的目标值，其中，Gap1、Gap2 分别为 case2 和 case3 求出的客户满意度与 case1 求出的满意度偏差。

Table 2. Analysis of the impact of time window length
表 2. 时间窗长度的影响分析

No.	Case1		Case2		Case3		Gap1	Gap2
	满意度	车辆数	满意度	车辆数	满意度	车辆数		
50	95.69	7	86.16	9	97.19	7	-11.1%	1.5%
100	93.56	14	84.22	15	96.93	12	-11.1%	3.5%
150	91.83	26	81.73	27	93.42	26	-12.4%	1.7%
200	89.37	28	80.38	30	92.82	26	-11.2%	3.7%

表 2 的数据表明当使用窄时间窗时, 客户满意度显著降低, 车辆的使用数量会增加或不变, 这可能是由于车辆能服务的时间范围收到限制, 需要通过增加车辆来满足窄时间窗的要求, 随着规模变大, 变化的幅度更加明显。当使用宽时间窗时, 客户满意度显著提升, 车辆的使用数量会减少或不变, 这可能是由于车辆能服务的时间范围变长, 面临违反时间窗的可能性降低, 可以不增加车辆以满足时间窗要求。从 case2 和 case3 与 case1 客户满意度的偏差可以看出时间窗长度的变化对客户满意度的影响较大。综上, 在不同规模算例下, 时间窗长度的变化对车辆使用数量以及客户满意度都有影响, 企业应根据不同的客户时间窗需求进行合理的车辆资源安排, 增加盈利能力, 提高客户满意度水平。

5. 结论

本文研究了带软时间窗的越库配送车辆路径问题, 建立了考虑运输成本、固定成本和客户满意度的数学模型, 目标是使物流总成本最小化, 客户满意度最大化。为了解决这个问题, 提出了结合变邻域搜索算子的非支配排序遗传算法 II。数值实验表明, 本文提出的算法相较于其他算法具有更好的搜索效率, 得到了帕累托最优前沿, 解的质量更高, 具有更快的收敛速度。VRPCDTW 模型能有效降低物流配送成本, 提高顾客满意度。此外, 服务时效分析得到时间窗长度的变化对客户满意度的影响较大, 企业应根据不同的时间窗需求进行合理的车辆资源安排, 提高客户满意度水平。未来的研究方向可以考虑结合碳排放相关因素, 促进企业在物流活动中节能减排; 也可以结合实际的应用场景, 不再将车辆速度设置为恒定不变的, 而是根据实际情况发生改变。最后, 未来的研究可以扩展到货运模式选择, 避开拥堵实现更加经济的方案。

参考文献

- [1] Lee, Y.H., Jung, J.W. and Lee, K.M. (2006) Vehicle Routing Scheduling for Cross-Docking in the Supply Chain. *Computers & Industrial Engineering*, **51**, 247-256. <https://doi.org/10.1016/j.cie.2006.02.006>
- [2] Liao, C., Lin, Y. and Shih, S.C. (2010) Vehicle Routing with Cross-Docking in the Supply Chain. *Expert Systems with Applications*, **37**, 6868-6873. <https://doi.org/10.1016/j.eswa.2010.03.035>
- [3] Mousavi, S.M., Torabi, S.A. and Tavakkoli-Moghaddam, R. (2014) Particle Swarm Optimization for a Multi-Product Multi-Period Cross-Docking Scheduling with Temporary Storage and Truck Allocation. *Computers & Industrial Engineering*, **71**, 115-128.
- [4] Zhou, Y., Zhang, J., Li, X. and Zhang, J. (2019) An Improved Genetic Algorithm for the Vehicle Routing Problem with Cross-Docking. *Applied Soft Computing*, **81**, Article 105506.
- [5] Afshar-Bakeshloo, M., Moattar Husseinzadeh, H. and Karimi, H. (2016) An Improved Ant Colony Optimization Algorithm for Cross-Docking Truck Scheduling Problem with Truck Substitutions. *Journal of Manufacturing Systems*, **40**, 46-54.
- [6] Morais, V.W.C., Mateus, G.R. and Noronha, T.F. (2014) Iterated Local Search Heuristics for the Vehicle Routing Problem with Cross-Docking. *Expert Systems with Applications*, **41**, 7495-7506. <https://doi.org/10.1016/j.eswa.2014.06.010>
- [7] Grangier, P., Gendreau, M., Lehuédé, F. and Rousseau, L. (2017) A Matheuristic Based on Large Neighborhood Search for the Vehicle Routing Problem with Cross-Docking. *Computers & Operations Research*, **84**, 116-126. <https://doi.org/10.1016/j.cor.2017.03.004>
- [8] Wen, M., Larsen, J., Clausen, J., Cordeau, J. and Laporte, G. (2009) Vehicle Routing with Cross-Docking. *Journal of the Operational Research Society*, **60**, 1708-1718. <https://doi.org/10.1057/jors.2008.108>
- [9] Kargari Esfand Abad, H., Vahdani, B., Sharifi, M. and Etebari, F. (2018) A Bi-Objective Model for Pickup and Delivery Pollution-Routing Problem with Integration and Consolidation Shipments in Cross-Docking System. *Journal of Cleaner Production*, **193**, 784-801. <https://doi.org/10.1016/j.jclepro.2018.05.046>
- [10] Agustina, D., Lee, C.K.M. and Piplani, R. (2014) Vehicle Scheduling and Routing at a Cross Docking Center for Food Supply Chains. *International Journal of Production Economics*, **152**, 29-41. <https://doi.org/10.1016/j.ijpe.2014.01.002>
- [11] 刘虹, 林楚玥. 考虑时间窗服务价值的越库车辆路径优化[J]. 西安电子科技大学学报: 社会科学版, 2019, 29(2): 12-23.
- [12] Acevedo-Chedid, J., Soto, M.C., Ospina-Mateus, H., Salas-Navarro, K. and Sana, S.S. (2023) An Optimization Model

for Routing—Location of Vehicles with Time Windows and Cross-Docking Structures in a Sustainable Supply Chain of Perishable Foods. *Operations Management Research*, **16**, 1742-1765. <https://doi.org/10.1007/s12063-023-00379-8>

- [13] 张政, 季彬. 考虑随机旅行时间与二维装载约束的越库配送车辆路径优化[J]. 控制与决策, 2023, 38(3): 769-778.