

**本科生毕业设计[论文]**

考虑网格仓间协同的生鲜配送车辆路径优化研究

|  |  |
| --- | --- |
| 院 系 | 管理学院 |
| 专业班级 | 管实1801 |
| 姓 名 | 戴蔓菲 |
| 学 号 | U201816168 |
| 指导老师 | 王 玥 |

2022年5月31日

**学位论文原创性声明**

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名： 年 月 日

**学位论文版权使用授权书**

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于 1、保密□，在 年解密后适用本授权书

2、不保密□。

（请在以上相应方框内打“√”）

作者签名： 年 月 日

导师签名： 年 月 日

# 摘 要

在移动互联网的日益普及、消费模式的不断升级、后疫情时代“宅经济”的迅速发展下，由美团优选、十荟团等社区团购平台发起的“中心仓—网格仓—消费者自提点”的三级生鲜配送网络悄然出现并逐渐渗透在广大消费者的日常生活中。在这三级生鲜配送网络里，网格仓位于关键节点，由网格仓发出的生鲜运输属于其中的关键链。然而，由于网格仓之间缺乏协调、生鲜消费随机性大、消费者自提点不确定性多、生鲜配送成本高、产品品质难以保障，网格仓生鲜配送的进一步发展举步维艰。此外，通过分析相关领域的国内外研究现状，本研究发现，现阶段研究成果还不足以为网格仓的生鲜配送运营提供全面且科学的指导法则。

为帮助网格仓改善生鲜配送经营现状、获取持续盈利，本研究提出了网格仓间协同开展生鲜配送服务的思想，并对其中涉及的配送车辆与订单任务的配置问题、配送车辆行驶路径的规划问题展开了探索，最终为网格仓的生鲜配送运营制定了一套科学合理且切实有效的分析和求解体系，即一个考虑网格仓间协同进行生鲜配送的车辆路径模型及一个基于变邻域搜索和自适应阈值接受算法的两阶段优化算法。此外，通过实验探究网格仓中的生鲜配送场景，本研究得到了如下结论：①道路交通状况会较大程度地影响订单任务执行顺序，并且整体不太通畅的道路交通状况还会诱导配送车辆暂时牺牲局部运输成本的优化而追求更大的客户满意度及订单交付的及时性；②运力资源充足时，适当增加配送车辆使用数量有助于降低总运营变动成本，并且整体上存在一个车辆的最佳使用数量，使得总运营变动成本达到最小；③单位品质正效益的增加仅在一定范围内对生鲜产品品质的提升有促进作用，在单位品质正效益高出该范围的情况下，想要获得生鲜产品品质的进一步提升，应考虑改变其他外部制约条件。

**关键词：**网格仓间协同生鲜配送；取货交付联合优化；时变路网；LIFO策略

# Abstract

With the increasing popularity of mobile Internet, the upgrading of consumption patterns and the rapid development of the "house economy" in the post-epidemic era, the three-tier fresh products distribution network of "central warehouse-grid warehouse-consumer pick-up point" cultivated by fresh products e-commerce platforms gradually affect the lives of consumers. In this three-tier fresh products distribution network, both the grid warehouses and the fresh products transport from the grid warehouses are very important. However, due to the lack of coordination between grid warehouses, the randomness of fresh products consumption, the uncertainty of consumer pick-up points, the high cost of fresh products distribution and the difficulty of guaranteeing product quality, it is hard to further develop fresh products distribution in grid warehouses. In addition, by analyzing the current research in this field, this study finds that present research results are not sufficient to provide comprehensive and scientific guidelines for the operation of fresh products distribution in grid warehouses.

In order to help grid warehouses improve their fresh products distribution operations and get sustainable profitability, this study proposes the idea of coordinating multiple grid warehouses to conduct fresh products distribution, and explores an order task allocation problem and a driving path planning problem, and finally develops an analysis and solution system for the fresh products distribution operations in grid warehouses, namely a vehicle routing model on the fresh products distribution under collaboration of grid warehouses and a two-stage algorithm based on variable neighborhood search and adaptive threshold acceptance algorithm. Besides, through the experimental investigation, the following conclusions were obtained: (1) Road congestion will induce vehicles to temporarily sacrifice local transportation cost optimization in pursuit of greater customer satisfaction and order delivery timeliness; (2)When the transportation resources are sufficient, an appropriate increase in the number of vehicles contributes to reduce the total operating cost; (3) The increase in the quality benefits only contributes to the improvement of fresh products quality within a certain range, and if the quality benefits are higher than that range, other external constraints should be changed in order to obtain further improvement of fresh products quality.

**Key Words：**collaboration of grid warehouses; joint optimization of pickup and delivery ; time-dependent networks ; LIFO

**目 录**

[摘 要 I](#_Toc105080485)

[Abstract II](#_Toc105080486)

[1 绪论 1](#_Toc105080487)

[1.1 研究背景 1](#_Toc105080488)

[1.2 研究目的 2](#_Toc105080489)

[1.3 研究意义 3](#_Toc105080490)

[1.4 研究内容与技术路线 4](#_Toc105080491)

[1.5 研究特色 6](#_Toc105080492)

[2 国内外研究现状 7](#_Toc105080493)

[2.1 生鲜配送问题研究 7](#_Toc105080494)

[2.2 取送货问题研究 9](#_Toc105080495)

[2.3 本章小结 10](#_Toc105080496)

[3 考虑网格仓间协同的生鲜配送优化模型 12](#_Toc105080497)

[3.1 问题描述 12](#_Toc105080498)

[3.2 模型构建 15](#_Toc105080499)

[3.3 本章小结 20](#_Toc105080500)

[4 基于变邻域搜索和自适应阈值接受算法的两阶段优化算法 21](#_Toc105080501)

[4.1 贪心算法生成初始解 21](#_Toc105080502)

[4.2 两阶段优化算法 23](#_Toc105080503)

[4.3 本章小结 33](#_Toc105080504)

[5 算例实验分析 34](#_Toc105080505)

[5.1 基本设置 34](#_Toc105080506)

[5.2 实验结果分析 37](#_Toc105080507)

[5.3 本章小结 49](#_Toc105080508)

[6 总结与展望 50](#_Toc105080509)

[6.1 研究总结 50](#_Toc105080510)

[6.2 未来展望 52](#_Toc105080511)

[致谢 53](#_Toc105080512)

[参考文献 54](#_Toc105080513)

[附录 58](#_Toc105080514)

# 

# 绪论

## 研究背景

传统的生鲜产品交易里，消费者线下购买产品后便自提回家，物流形式简单；而后，随着消费模式不断升级，一些生鲜企业开始推出“配送到家”的服务——消费者线下购买产品后，生鲜企业会安排专门的配送员将产品送至消费者的家中。如今，在移动互联网的日益普及、生鲜电商企业的迅速发展下，生鲜产品的交易再次突破了时空的壁垒，消费者在家通过手机APP下单即可购买并收到品质良好的水果蔬菜等生鲜。与此同时，伴随后疫情时代“宅经济”的逐渐盛行，线上下单线下收货的新兴生鲜交易模式开始受到越来越多消费者的青睐，解决生鲜产品的配送问题成为了众多生鲜电商企业寻求发展突破的关键。其中，以朴朴超市、叮咚买菜、每日优鲜等企业为代表，各大生鲜电商纷纷开始构建分布密度高且距离消费者近的前置仓，以便及时响应消费者的“一篮菜”需求。还有一些小型生鲜门店和大型商超则选择与京东到家、美团、饿了么等O2O平台建立合作关系，通过众包式生鲜配送，为消费者创造便利的服务体验。此外，在前置仓构建成本大、众包式生鲜配送难以有效监控生鲜品质的压力下，美团优选、十荟团等社区团购平台带领着“预售+次日达+消费者自提”的经营特色横空出世，使得以网格仓为核心的生鲜配送逐渐走进广大消费者的生活中。这种生鲜配送形式涉及一个生鲜产品从城市中心仓到网格仓再到消费者自提点（即拼团团长处）的三级网络结构。

网格仓属于整个生鲜配送网络的关键节点，网格仓仓长能否合理调度车辆并科学规划路径以确保生鲜产品可以安全、准时地抵达各个消费者自提点至关重要。然而就各大网格仓的经营现状来看，其生鲜配送还存在五大发展障碍。第一，网格仓之间缺乏协调。同一个网格仓仓长通常同时运营多个网格仓，而这些网格仓彼此割裂，运作缺乏整体有效的规划安排，难以实现全局最优。第二，生鲜产品的消费具有较大的随机性。波动幅度大的消费量可能使网格仓时而陷入配送服务疲于供给的困境，时而面临人力物力闲置空耗的状态。第三，消费者自提点具有不确定性。消费者自提点的不确定性表现在平台每天可能都有新入驻的团长，导致网格仓的服务末端可能经常发生变化，需要重新进行配送规划。第四，配送成本过高。当前冷链物流技术发展尚不完善，生鲜产品在运输途中难以得到妥善管理，产品存在大量损耗，引发较大的配送成本。第五，生鲜产品品质难以保障。生鲜产品保质期短且极易损坏，长时间在途运输或任何发生于此期间的不妥当管理行为都可能导致产品新鲜度下降，影响客户满意度。

上述五大发展障碍其实反映了一个本质问题：彼此之间缺乏联系的网格仓难以从一个整体视角实现资源的合理调度及分配（尤其是运力资源），生鲜配送路径的优化也迫在眉睫。并且，根据已有文献和资料，生鲜配送领域的相关研究大多停留在理论层面，而在与之相关的实际优化研究中，有些表现为现实的高度抽象，有些未能考虑联结多个配送点以开展整体优化。因此，本研究从全局优化的角度出发，针对原有网格仓独立进行配送任务的机制，提出网格仓间协同开展配送服务的思想——在整合多个网格仓配送任务及运力资源的前提下，实现运力资源的合理调度及配送路径的科学规划。

## 研究目的

在当前的运作机制下，一个网格仓通常向一个分区的多个自提点提供生鲜配送服务，各个分区之间的配送活动相对独立。由此带来的问题是运力资源与服务需求之间不能建立整体上的有效匹配，难以应对分区中随时可能发生的需求变故，如某一分区服务需求的剧增可能使该区全部运力资源都陷入响应瘫痪。为避免上述意外的发生，增强运力资源应对不确定性消费者需求的灵活性，本研究提出了网格仓间协同开展配送服务的思想。

单价高，保质期短是生鲜产品区别于其他商品最本质的特征，在其存储或搬运的任何环节出现差错，都会造成人为因素的糜烂和损坏。因此，消费者对生鲜产品的配送服务一般会存在时效性和品质方面的要求。而现实情况往往是，降低配送成本与提高客户满意度之间似乎难以取得平衡，许多生鲜企业在努力满足上述要求的同时，也产生了大量的成本。“优菜网”创始人丁景涛很早就意识到解决生鲜的物流问题是生鲜企业能够产生并持续盈利的核心思路，希望能“像送牛奶一样送菜”，开辟一条低成本生鲜物流的新道路。然而，“优菜网”还是没能如丁景涛期望的那样运营下去，过高的物流成本使其耗费了太多资本，最终不能支持经营。于是，为提高生鲜配送时效、降低生鲜配送成本，本研究从配送车辆调度以及行驶路径规划两方面展开了探索。

综上所述，本研究基于网格仓间协同开展生鲜配送服务的思想，对其内部的车辆路径问题进行了探索，希望在分析实际的基础上，通过科学建模与高效求解，为各个网格仓的生鲜配送经营提供决策支持，以进一步提高分区内运力资源应对不确定性消费者需求的能力、实现网格仓生鲜配送服务的整体优化，并使得“最小的成本收获最大的客户满意度”成为可能。

## 研究意义

本研究的意义可以从理论和现实两个层面分别阐述：

（1）理论层面

本研究系统地分析了各大生鲜电商企业所采取的生鲜配送模式、相关领域的国内外研究现状，并针对社区团购平台网格仓内生鲜配送活动的发展障碍，展开了对寻求有效且合理的解决方案的探讨。此项研究①从整体优化的视角出发，提出了网格仓间协同开展生鲜配送服务的思想；②综合考虑生鲜配送环节中的各类影响因素（如贮存温度与与运输时间对生鲜产品品质的影响、道路交通状况对配送效率的影响等），较为全面地刻画了网格仓中的生鲜配送场景；③创新的引入一种“后进先出（LIFO）”的交付策略，构建了全新的带取送货的车辆路径模型；④设计基于变邻域搜索（VNS）和自适应阈值接受算法（TA）的两阶段优化算法并实现编程运行，支撑了研究成果的落地性。

综上，本研究的内容将进一步丰富物流管理领域的理论和方法，为未来的相关研究提供参考思路。

（2）现实层面

近年来，我国经济发展迅速，人民群众对生鲜产品的消费迎来新高度。然而，在这片欣欣向荣的景象里，生鲜配送问题却日益暴露严重缺陷。据深圳商报称，2022年春节期间生鲜配送投诉占餐饮服务投诉的七成以上，投诉原因主要为未能如约完成生鲜产品的配送。不仅消费者群体对当前生鲜配送服务存在诸多抱怨，诸如美团优选这类生鲜商家也对其提供生鲜配送服务所带来的高额成本颇有怨词。自2022年初，由于无法承担持续亏损，美团优选已陆续在全国多所城市关停了社区团购业务，其中包括北京、甘肃、宁夏等地。

解决生鲜电商的生鲜配送问题显然已迫在眉睫。本研究应对当前我国冷链物流技术发展尚不完善的实际状况，希望通过优化改进各大社区团购平台生鲜配送中的关键环节（即生鲜产品从网格仓到消费者自提点处的物流活动），以帮助美团优选等生鲜电商改善经营现状，获取持续盈利。更进一步，研究成果有望促进全国生鲜物流的健康发展，减少整个国家的生鲜物流成本，从而推动更多的国民财富建设冷链物流核心技术，最终形成“财富推动技术建设，技术创造更大财富”的正反馈循环。

## 研究内容与技术路线

### 研究内容

本研究关注于网格仓间协同开展生鲜配送的问题，对其中所涉及的车辆从网格仓取货到交付至消费者自提点的完整业务流程进行了配送车辆的合理派遣、行驶路径的科学规划，以实现“用最小的成本收获最大的客户满意度”，即花费合理的开销，在客户期望的时间和地点交付符合客户期望的产品。

研究的具体内容包括以下部分：

（1）构建考虑网格仓间协同的生鲜配送优化模型。针对网格仓生鲜配送中的五大发展障碍——网格仓之间缺乏协调性、生鲜产品的消费表现出随机性、消费者自提点具有不确定性、生鲜配送成本过高、产品品质难以保障，同时结合对代表型企业“美团优选”运营特点的分析，本研究构建了一个综合考虑时变路网条件、后进先出交付策略、生鲜产品品质损耗的带取送货的车辆路径模型。

（2）设计求解算法逻辑并实现编程运行。根据优化模型，本研究设计了基于变邻域搜索和自适应阈值接受算法的两阶段优化算法用于计算最优配送方案。其中，算法阶段一使用改进的变邻域搜索算法先对贪心算法所得初始解进行车辆间订单集任务配置及部分路径规划；阶段二采用自适应阈值接受算法对阶段一所得解进行车辆内部行驶路径的深入规划。此外，本研究使用Java语言实现了算法求解过程的运行及求解结果的输出。

（3）测试数学模型的合理性、求解算法的有效性，并设计实验展开有关问题场景的探索。引入Solomon's benchmark数据集中的标准算例数据，代入上述优化模型和优化算法求解规划方案，验证了模型与算法的科学合理性。同时，对网格仓生鲜配送中的道路交通状况、配送车辆使用数量、生鲜产品品质效益进行了实验探索与论证分析，找到了与现实吻合的一般性规律和有利于改善现状的管理学启示。

### 技术路线

本研究的技术路线如图1-1所示：

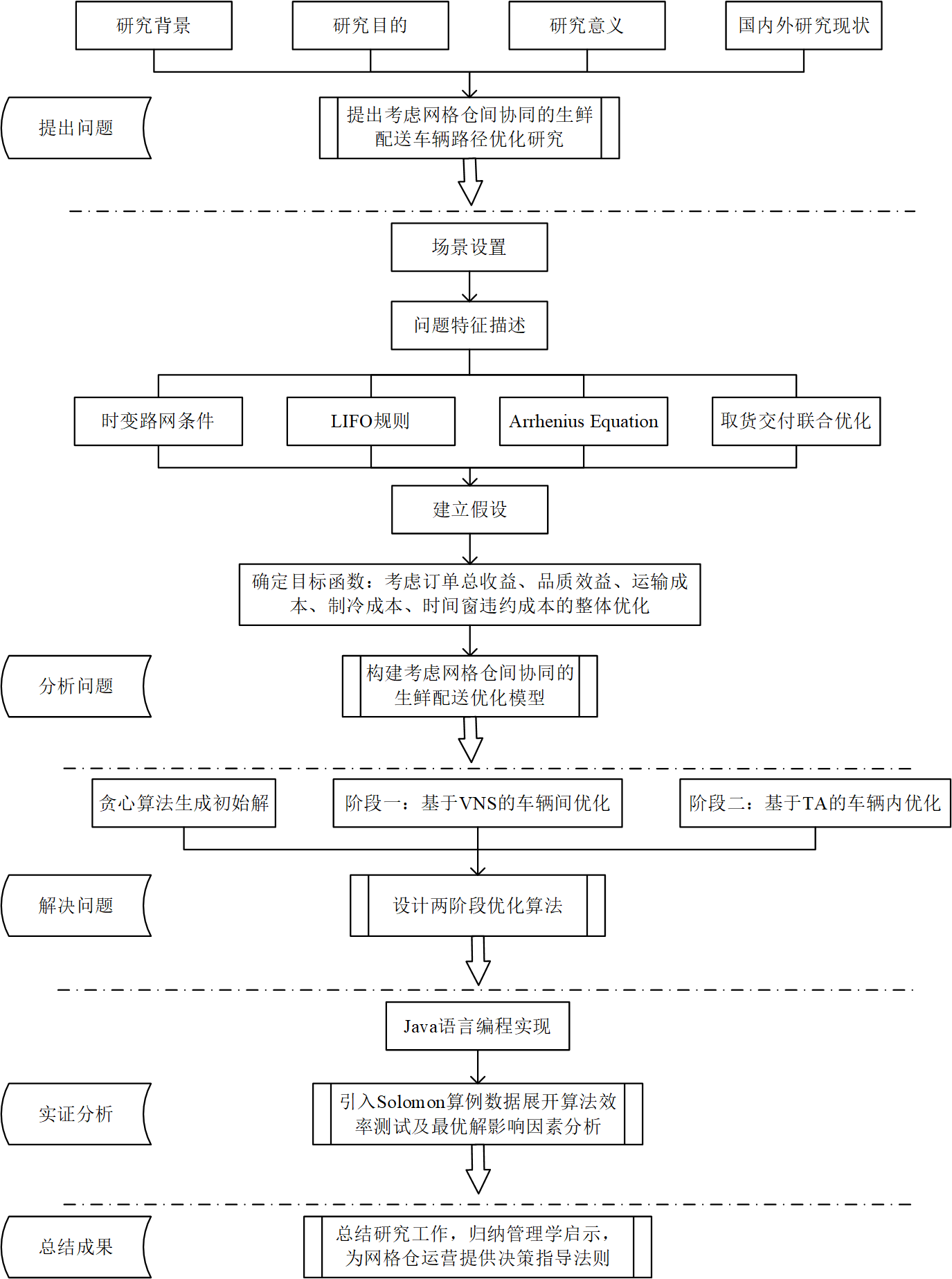


图1‑1 研究技术路线

## 研究特色

本研究基于网格仓间协同开展生鲜配送服务的思想，综合考虑时变路网条件、生鲜产品品质损耗两大分别影响生鲜配送时效及客户满意度的因素，设计了后进先出交付策略，对生鲜取货环节与交付环节的联合优化进行了探索。其中，

（1）取货交付联合优化：生鲜配送中，生鲜产品的取货环节与交付环节高度耦合。在原有的各网格仓独立进行生鲜配送服务的机制下，由于配送车辆的取货行为只发生于一个网格仓中，通常仅需展开对生鲜产品交付环节的规划设计。而在网格仓间协同进行生鲜配送服务的机制下，配送车辆的取货行为常常发生于多个网格仓中。因此，在这种情况下，想要达成全局最优，必须以一个整体的视角看待生鲜产品的取货与交付。

（2）时变路网条件：城市道路由于受到早晚出行高峰或节假日旅游高峰的影响，会发生一种与时间关联较大的交通拥堵，即在某一特定时间段内道路交通拥堵，而其他时间段内道路交通正常。相较于交通正常的时间段，在交通拥堵的时间段里，车辆平均行驶速度明显降低，车辆行驶一段相同的距离所需要的时间更长。正如前文所分析的，生鲜产品的配送对时效性要求高，任何影响其配送效率的因素都应该是问题研究的重点之一。因此，本研究引入时变路网条件以充分考虑道路交通状况对生鲜配送过程的影响。

（3）生鲜产品品质损耗：生鲜产品运输途中的品质损耗涉及一系列复杂的生化反应，但其主要与运输时间和贮存温度相关。本研究引入Arrhenius Equation刻画生鲜产品在途品质损耗，力求实际地改善生鲜配送中的品质损耗问题。

（4）后进先出交付策略：后进先出交付策略常用于脆弱物品的交通运输或配送容器口径较小的物流活动里。本研究在网格仓的生鲜配送中使用后进先出策略的意义在于，首先，后进先出策略可以显著降低翻箱率，在方便配送司机完成订单集交付的同时，减少团长接收订单物品的等待时间。其次，后进先出策略创造的交付时间优势可以极大地减少交付过程中产生的系列成本，如，车辆执行下个订单任务的时间占用成本、车辆与外界进行热交换所产生的能耗成本（考虑使用冷藏车执行任务）等。最后，后进先出策略可以有效降低交付过程中不可避免的人工操作所造成的生鲜产品品质损耗，如，因人工匹配订单与消费者自提点所造成的产品品质损耗、人工翻箱所造成的产品品质损耗等。

# 国内外研究现状

本研究基于网格仓间协同开展生鲜配送服务的思想，对网格仓中的生鲜配送问题进行了车辆派遣与路径规划探索，其研究本质在于一个取送货问题优化。本章将从“生鲜配送问题研究”及“取送货问题研究”两个维度对相关领域的国内外研究现状进行梳理和分析。

## 生鲜配送问题研究

在优化改进生鲜产品的配送问题上，国内外研究都从不同方面做出了诸多贡献，这些研究贡献大致可以分为关注配送场景的模型创新和关注求解效率的算法改进两大方向。

在模型构建方面，第一大类是比较经典的专注生鲜产品配送路径的规划研究，如许婷婷等（2021）以最小化成本为目标，对生鲜农产品的配送进行了路径规划；余海燕等（2021）考虑硬时间窗约束及动态订单的实时调度约束，对O2O生鲜外卖配送展开了优化研究。第二大类研究考虑生鲜配送问题中的多级目标，如最大化客户满意度、最小化配送成本等，尝试构建多目标优化模型以实现整体最优。类似的研究有Wang et al.（2016）针对生鲜产品的配送问题构建了一个最小化系统总成本和最大化生鲜产品新鲜度的多目标模型并设计了两阶段启发式算法求解模型；张倩等（2019）考虑了一种不确定的客户需求，对生鲜产品的配送问题建立了路径规划的多目标优化模型，主要优化的目标包括最大化客户满意度（保证生鲜产品品质）、最小化总成本（包括固定费用、运输费用及制冷费用）、最小化碳排放量。第三大类研究考虑订单交付的优先级或形式，如夏扬坤等（2021）考虑客户分级和客户订单可拆分，对配送中心的生鲜配送问题展开了研究；马艳芳等（2021）在考虑客户分类的条件下，建立了致力于最小化配送成本的两级容量有限车辆路径优化模型。第四大类研究思考了配送车辆类型的影响，如陆淼嘉等（2021）以城市社区网购生鲜无人车配送问题为研究对象，尝试构建仿真平台，以实现动静态订单分配策略设计和距离最近或时间最紧急配送路径规划；Song and Ko（2016）以最大化客户满意度为目标，分别对专用冷藏车和普通车关于生鲜产品配送的车辆路径问题进行了研究。第五大类研究致力于探讨道路状况对生鲜配送的影响，相关研究有Ma et al.（2017）综合考虑订单选择（有选择地接受客户订单，对于不接收的订单存在客户流失的机会成本）以及交通阻塞，对易逝品配送中的路径规划问题进行了研究；Wu et al.（2020）考虑现实中断意外（如交通障碍、交通阻塞、车辆故障等）对易逝品配送的影响，基于分批配送的思想构建了一个实时恢复中断配送模型；其他研究还有刘思远等（2021）、赵志学和李夏苗（2020）等。

算法设计上，大多学者提出的都是启发式算法，其中，方文婷等（2019）从节能减排的角度出发，考虑生鲜配送中车辆固定使用成本、绿色成本、制冷成本、货损成本以及时间窗违约成本的联合优化，提出了结合A-star算法的混合蚁群算法求解路径规划方案。侯宇超等（2018）设计了带精英策略的蚁群算法求解一个关注客户满意度、配送总成本及污染物排放量的多目标优化模型。Hu et al.（2017）针对生鲜产品的取货和配送问题，构建了两阶段分解算法进行求解，该研究将整个问题分为两个子问题，其一是最短路径问题，其二是出发时间调度问题，并采用反馈策略避免求解过程陷入局部最优的困境。Shukla and Jharkharia（2013）考虑带时间窗约束的生鲜农产品配送问题，提出了一种基于人工免疫系统的算法（AIS）用于解决问题，通过实例数据的测试，该研究对比分析了AIS、遗传算法和模拟退火算法的求解结果及求解效率，验证了AIS的性能优于另外两种算法。Osvald and Stirn（2008） 主要设计了一种基于禁忌搜索的启发式算法来优化生鲜产品的配送问题，通过算例验证，使用该算法可使变质的生鲜产品减少47%。Naso et al.（2007）考虑易逝品生产调度和配送路线的联合优化，设计了一种结合遗传算法（GA）和一组建设性启发程序（CHP）的优化算法解决问题。在该算法中，遗传算法作为主算法工具用于决策部分变量优先级；建设性启发程序则根据遗传算法的决策结果完善易逝品的生产调度和配送规划设计。Tarantilis and Kiranoudis （2002）研究了针对生鲜肉产品的开放式、多车场车辆路径问题，提出基于列表的阈值接受算法（LBTA），该算法每次都由一个阈值来判断是否接受解的移动，而阈值是通过一个会根据解空间实时更新和调整的列表来确定的。类似地，Tarantilis and Kiranoudis（2001）为解决新鲜牛奶配送问题，也提出了回溯自适应阈值接受算法（BATA），该算法的主要特征在于可以接受每一步移动所产生的新解，新解不一定要比当前解好，但也不能比当前解差太多（通常根据一个阈值来界定）。

## 取送货问题研究

许多学者都曾对取送货问题做过研究和探索，这些研究可以分为“一对一PDP研究”、“多对多PDP研究”、“一对多对一PDP研究”。

### 一对一PDP研究

在一对一取送货问题中，一个取货点对应一个交付点；通常假设配送车辆从一个公共停车场出发，逐一完成客户订单或客户请求对应的一组取货、交付任务，待所有任务完成后，配送车辆回到公共停车场。在这一问题的研究中，最经典的一类研究是单车辆取送货问题研究（SVPDP），即考虑仅使用一辆车完成所有取货和交付任务的研究。如，Bartolini et al.（2015）考虑客户请求具有在途运输时间限制，对配送车辆在有容量约束和无容量约束的两种情况下进行了讨论和分析，推导出了问题可行解的新下界，并针对问题的求解提出了两种精确算法。实验结果表明，该算法可以为多达50个客户请求的运输网络规划最优解决方案。还有一类研究考虑多车辆执行取送货的问题（MVPDP），如，Ropke and Pisinger（2006）在考虑时间窗约束和配送车辆容量约束的基础上，对使用多辆车完成一对一条件下的多点取货与多点交付问题展开了研究，同时提出了一种基于大邻域搜索算法的启发式算法求解最优规划方案。此外，在这两类问题的研究基础上，许多学者也开展了一些衍生问题的研究。如，Sahin et al.（2013）研究了一种考虑订单可拆分的多车辆、一对一取送货问题（MVPDPSL），并提出了结合禁忌搜索和模拟退火的启发式算法（TESA）求解问题。通过实验探索，该研究得出“允许订单拆分能有效降低运输成本”的结论。Marilène et al.（2015）针对带有时间窗约束和后进先出装载约束的取送货问题（PDPTWL），构建了车辆路径模型，并提出了三种精确的分支定价和切割算法求解模型。

### **一对多对一PDP研究**

在一对多对一取送货问题中，商品最初在一个中心仓库完成取货，随后被交付给不同的客户；而在交付商品的同时，通常又伴随其他物品的取货，从客户处取得的物品之后会被运往一个中心仓库。这类问题的研究背景通常是商品的分销和包装的回收，如，Prive et al.（2006）研究了魁北克一家公司的饮料分销和空瓶回收物流，为此构建了一个包含异构车队、时间窗约束、配送车辆载重和体积约束的混合整数规划模型，并针对模型的求解提出了三种启发式算法。Gribkovskaia et al.（2008）针对经常发生于挪威海上的从一个基地到多个石油平台的物质供给运输及从多个石油平台到一个基地的杂物回收运输，展开了路径优化研究，并设计了构造启发式算法和禁忌搜索算法用于求解最优方案。

### 多对多PDP研究

在多对多取送货问题中，商品可能在多个取货点中的一个位置被提取，随后可能被运往多个交付点中的一个位置。这类问题对应的现实场景一般是银行分支机构之间发生资金调拨、企业仓库之间发生物资调拨等。Hernández-Pérez and Salazar-González （2004）研究了运输车辆具有容量约束的单一商品、多对多取送货问题，该研究所提出的算法可以为多达260个客户的运输网络规划车辆行驶路径。Xu et al.（2014）在考虑运输车辆有容量限制的条件下，对树状网络中的物品从多个取货点到多个交付点的运输问题进行了最优路径规划研究，并设计了2-approximation算法求解此类问题。

## 本章小结

通过上述国内外研究现状的分析，可以发现：

（1）生鲜配送问题的优化研究通常是借助VRP及其衍生问题而展开的，虽然部分研究开始考虑一些与社会实际和社会发展接壤更为紧密的要素，如考虑道路交通状况对生鲜配送的影响、考虑订单配送的优先级、考虑生鲜配送过程中二氧化碳排放量对环境的影响等。但这些研究一般假设车辆从一个或多个配送中心开始逐一向末端消费者提供服务，未能建立起配送中心之间的协作关系，配送规划也仅针对生鲜产品的交付环节展开，对生鲜产品取货环节的规划问题考虑不足。

（2）取送货问题从问题所涉及的运输网络结构来看可以分为“一对一取送货问题”、“一对多对一取送货问题”和“多对多取送货问题”三类。这三类问题分别对应有不同的现实场景，如“一对一取送货问题”常见于电商企业的众包物流活动中，“一对多对一取送货问题”常发生于分销和回收的联合物流活动中，而“多对多取送货问题”通常存在于物资调拨活动或物品交换活动中。2.2节从这三个维度出发，细分出来的取送货问题研究不仅为研究本身所刻画的特有问题场景输出了最优解决方案，同时还为类似问题场景的研究提供了参考思路。

综上所述，在考虑网格仓之间协同开展生鲜配送服务的条件下，现阶段生鲜配送领域内的研究成果难以提供一套全面且科学的指导法则以帮助网格仓应对不可忽略的生鲜产品取货环节。因此，本研究从实际出发，在网格仓中的生鲜配送的大背景下，考虑了一个取送货问题，提出了网格仓间协同生鲜配送下的车辆派遣与路径规划探索，以期通过网格仓生鲜配送中取货环节与交付环节的整体联合规划，来达成改善网格仓生鲜配送经营状况、获取持续盈利的总目标。

# 考虑网格仓间协同的生鲜配送优化模型

本章将综合考虑时变路网条件、LIFO交付策略、生鲜产品品质损耗三大要素，对网格仓间协同开展生鲜配送的问题构建一个带取送货的车辆路径模型。

## 问题描述

本研究考虑一个发生于如图3-1所示的生鲜配送网络中的车辆路径问题。该网络由多个网格仓和多个消费者自提点组成；各网格仓每天都会接收由中心仓发出的生鲜产品，网格仓间需要相互协同运作以将生鲜产品安全运送至城市中的不同消费者自提点处。

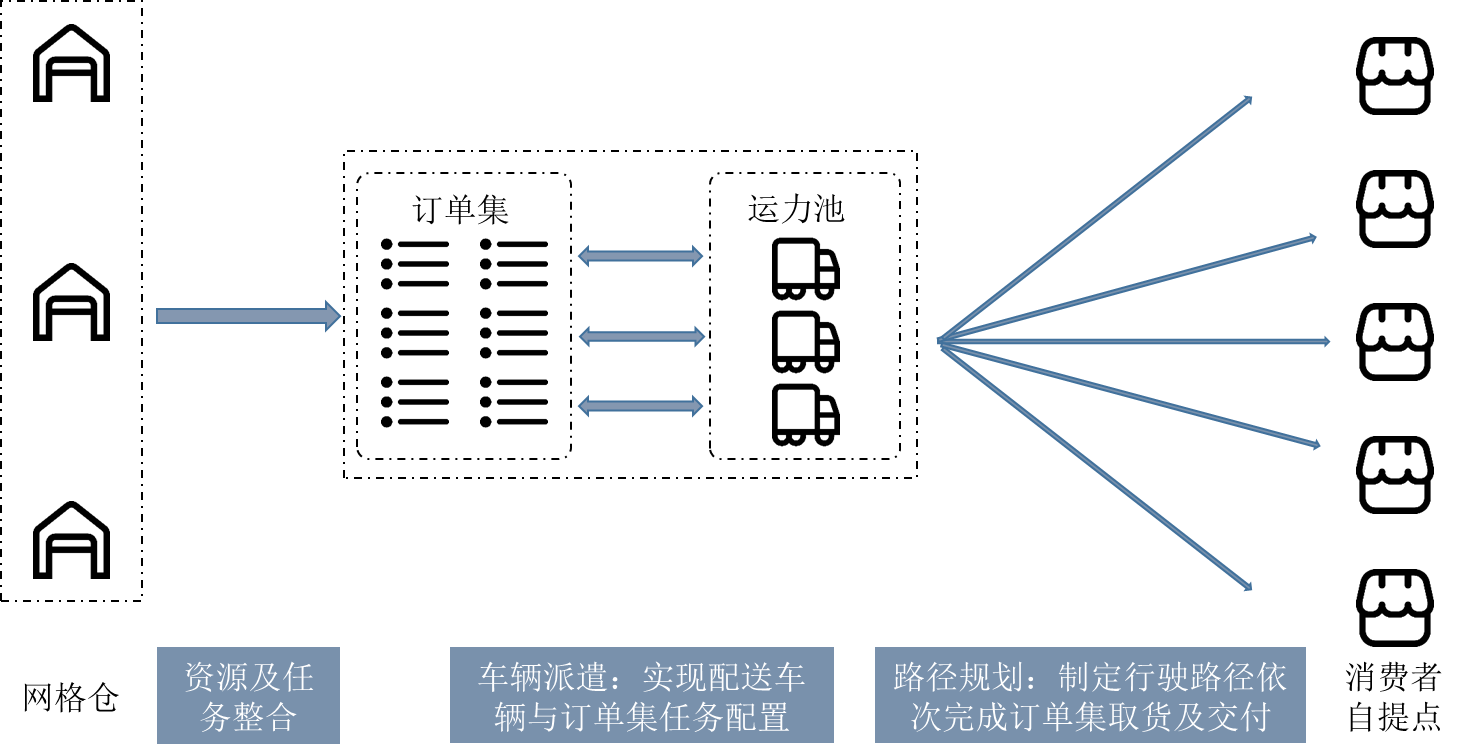


图3‑1 多网格仓协调开展生鲜配送服务

由于网格仓和消费者自提点通常都配备有良好的冷藏设施，生鲜产品在网格仓和消费者自提点处的品质损耗可以忽略不计，本研究也仅考虑生鲜产品发生于在途运输中的品质损耗。为减少该损耗，本研究假设使用冷藏车来完成生鲜配送任务，并且每个网格仓初始都停放有若干车型相同、容量具有限制的冷藏车。冷藏车在执行生鲜配送任务时，先是从初始网格仓出发，完成派遣路径上第一个订单集的取货（取货点可能位于初始网格仓，也可能位于其他网格仓），再依次按照顺序执行既定的订单集任务，待派遣路径上最后一个订单集成功交付至对应消费者自提点处，冷藏车返回初始网格仓。

由于该生鲜配送网络所涉及的客户订单体量较大，并且生鲜产品具有易损坏的特性，与之相关的装载和卸载都需轻拿轻放，因此，本研究考虑客户订单在网格仓的装载时间和在消费者自提点的卸载时间，同时假设两者正比于客户订单量。此外，拼团团长可能会对订单集的交付提出一些要求，本研究主要关注两大交付要求。其一是交付软时间窗要求。考虑到团长的特殊个人因素（如，自提点营业时间受限、个人空闲时间受限等），冷藏车进行订单集交付的时间应该在团长规定的时间窗内，任何早于或晚于时间窗的交付行为都可能造成额外的配送成本。其二是生鲜产品的最低新鲜度要求。团长出于减少消费者纠纷，提高竞争力的考虑，通常不会愿意接受品质较差的生鲜产品。

在上述问题特性的基础上，本研究所考虑的问题还具有“道路网络符合时变性”、“订单集交付遵循后进先出规则”、“生鲜产品在途运输品质损耗依赖于时间和温度”的三大特点。

### 时变路网条件

城市道路交通状况与时间的关联较大，具有一种时变性，具体表现为：早晚出行高峰时期或节假日旅游高峰时期，道路交通较为拥堵，车辆平均行驶速度缓慢；而其他时期，道路交通正常，车辆以正常的速度行驶（如图3-2所示）。由于生鲜配送对时效性要求很高，而道路交通状况会通过影响车辆行驶速度对生鲜产品的配送时效产生较大影响，因此，本研究引入时变路网条件以刻画道路交通状况所造成的可能影响，力求从更加现实的角度实现网格仓生鲜配送的最优规划。

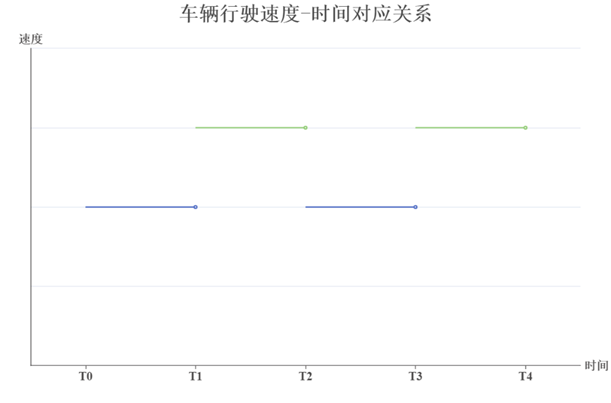


图3‑2 时变路网条件下，车辆行驶速度-时间对应关系

### 后进先出交付策略

采用后进先出交付策略要求每次进行交付的订单集必须是未交付订单集中最后一个完成取货的订单集。一条采用后进先出交付策略的行驶路径如图3-3所示，而图3-4则展示了一条未采用后进先出交付策略的路径。



图3‑3 采用后进先出交付策略的行驶路径



图3‑4 未采用后进先出交付策略的行驶路径

在网格仓的生鲜配送中使用后进先出交付策略具有诸多益处：i.减少翻箱率；ii.节约订单集交付的时间成本；iii.减少由于翻箱所造成的产品品质损耗。

### 生鲜产品品质损耗

易发生品质的自然损耗是生鲜产品不同于一般商品的显著特征之一，而生鲜产品的品质损耗主要受储存时间和储存温度的影响，其中涉及到一系列复杂的生化反应，本研究对此不做深入探讨。为构建生鲜产品品质损耗与储存时间及储存温度之间的关系，本研究引入经典的Arrhenius Equation，如式，该式也在Rong et al.（2011）、Zhang et al.（2003）等研究中有所运用。



其中，表示生鲜产品的初始品质；是一个与温度无关的反应速率常量，单位；是气体常数；是生鲜产品的贮存温度，单位；是生鲜产品的运输时间，单位；是活化能，表示生鲜产品温度指数依赖性的经验参数，单位。

## 模型构建

### 参数定义

本研究所用的主要参数如表3-1（部分参数及定义在后文给出）：

表3‑1 主要参数及定义

|  |  |
| --- | --- |
| **参数** | **定义** |
|  | 团长对订单交付的时间窗要求 |
|  | 团长对生鲜产品的最低新鲜度要求 |
|  | 对应于一个消费者自提点/团长的订单集，其中 |
|  | 订单集取货点构成的集合， |
|  | 订单集交付点构成的集合， |
|  | 冷藏车的集合， |
|  | 冷藏车k的起始位置点，，  冷藏车k的终止位置点，，  所有订单集初始存放点和交付点构成的集合  配送网络中所有位置点构成的集合， |
|  | 运输网络中各位置点间的连线， |
|  | 有向图，表示一张完整的配送网络，， |
|  | 有向图中顶点与顶点之间的距离，即的弧长 |
|  | 每辆冷藏车的最大载重量 |
|  | 冷藏车离开位置点的载重量 |
|  | 冷藏车到达位置点后的重量变化 |
|  | 冷藏车到达位置点的时间，其中 |
|  | 冷藏车离开位置点的时间 |
|  | 冷藏车经过弧的时间 |
|  | 订单装载/卸载速度 |
|  | 订单集所包含的生鲜产品的平均初始品质 |
|  | 冷藏车内的温度 |

### 决策变量

定义0-1决策变量，其表达式如式



一些说明：

①订单集取货点对应的实际位置为各个网格仓的实际地理位置，并且中有些元素的实际位置可能相同，即有些订单集所包含的生鲜产品初始存放于同一个网格仓中。

②订单集交付点对应的实际位置为各个消费者自提点的实际地理位置，并且订单集的每一个取货点都有一个交付点与之对应。

③对于一个取货点而言，对应于订单集所含生鲜产品的重量，同时满足。

④。

⑤；，其中，表示冷藏车在位置点的服务时间。由于订单集交付的最早时间不得早于团长规定时间窗的下限，因此可能会大于。

⑥有向图中各顶点间的距离满足三角不等式关系，即，恒成立。

### 目标函数

本研究的目标在于最小化网格仓生鲜配送服务的总成本，其总成本=运输成本+制冷成本+时间窗违约成本-品质效益。

（1）运输成本

设单位距离运输成本为，则完成全部订单集的配送所产生的运输成本为



（2）制冷成本

为较好地保存生鲜产品，冷藏车内的温度需维持在适宜水平上，冷藏车产生的能耗便形成了制冷成本。冷藏车的能耗来源于两部分，其一是进行订单集装卸货操作时，冷藏车与外界进行热交换会产生一定能耗；其二是运输途中为维持生鲜产品保存的适宜温度，冷藏车会消耗能量。这两种能耗均与时间相关，设前一种能耗的单位时间成本为，后一种能耗的单位时间成本为，则制冷成本为



其中，表示冷藏车与外界进行热交换所产生的能耗成本，与订单集装卸货的时间正相关；表示冷藏车为维持适宜生鲜产品保存的温度所产生的能耗成本，与冷藏车的完整行驶时间正相关。

（3）时间窗违约成本

假设由于等待团长接收订单集所形成的单位时间成本为，由于延迟交付订单集所产生的单位时间成本为，则时间窗违约成本可以表示为



（4）品质效益

品质效益主要源于团长对生鲜产品品质的满意度，并且分为正效益和负效益：当生鲜产品新鲜度高于团长的最低要求时，其超出的额度越大，团长的满意度越大，则正效益也越大；当生鲜产品新鲜度低于团长的最低要求时，其不足的额度越大，团长的满意度越小，造成的负效益越多。

每份完成交付的订单集所包含的生鲜产品剩余品质满足



则由此创造的品质效益为



其中，分别为满足和不满足团长对生鲜产品最低新鲜度要求下的单位产品品质所能创造的效益，。

综上，目标函数可以表达为



### 数学模型

根据上述分析，可构建数学模型如下：



Subject to

























其中，式是优化目标，即最小化网格仓生鲜配送总成本；式表示每一份订单集都必须提供配送服务；式表示同一份订单集的取货点与交付点是一一对应的，并且两者必须被安置于同一条派遣路径上；式表示每辆冷藏车初始从其停放的网格仓出发，执行完所有订单集任务后又回到终止点（在该条件的控制下，可能不必使用所有的冷藏车）；式表示流入流出平衡，即对于有向图中的任意节点 ，一旦发生冷藏车的到达行为就会伴随冷藏车的离开行为；式表示冷藏车从顶点到顶点的载重转换关系；式表示冷藏车的最大载重约束；式表示同一份订单的交付行为必须晚于取货行为。

式至表示后进先出交付策略。为解释该表达式的构建，首先引入两个总结于Cherkesly et al.(2015)的命题：

**命题1.**若，，且，之间不存在关于订单集和的任何交付，则发生于点的取货一定早于发生于点的取货。

**命题2.**对于任意两个订单集和的交付，若和可以依次交付，则和两点间不会发生已经取货但未完成交付的订单集，同时等式恒成立，否则由命题1可得。

通过上述命题，后进先出交付策略的数学表示可以分两种情况讨论：

1. 发生于取货点后的路径选择

若，则可以是与不重复的取货点，也可以是对应于的交付点（即），因此有对恒成立。

1. 发生于交付点后的路径选择

若，则可以是初始停放冷藏车的网格仓（即冷藏车完成所有订单交付后回到终止点），可以是一个除去之外的取货点，也可以是一个交付点。

考虑是仓库或取货点，则必须满足对成立。

考虑是交付点，则必须满足前提，因此不等式对且恒成立。

式描绘了冷藏车到达某一位置点的时间与其到达上一位置点的时间之间的关系。由于时变路网条件的限制，冷藏车行驶速度与时间的对应关系如图3-2所示。对于图中任一时间段而言，冷藏车在该时间段内的行驶速度恒定为。定义0-1变量：



冷藏车经过所需的时间取决于其离开点的时间和,两点间的距离，并且具有以下表达式：



## 本章小结

基于前文的分析，本章构建了时变路网条件以刻画道路交通状况对网格仓生鲜配送过程的影响；探讨发生于取货点与发生于交付点后的路径选择来制约冷藏车行驶路径上的订单集交付行为遵循LIFO规则；引入经典的Arrhenius Equation解释生鲜产品品质、运输时间、贮存温度三者之间的关系。在上述工作的基础上，本章还基于网格仓间协同运作的思想，搭建了一个带取送货的车辆路径模型（考虑网格仓间协同的生鲜配送优化模型），以探寻网格仓中生鲜配送问题的解决之道。

# 基于变邻域搜索和自适应阈值接受算法的两阶段优化算法

本研究的本质其实是一个考虑生鲜产品易损坏特性、多车辆、多仓库、带软时间窗约束及LIFO条件的取送货问题优化。对于这一问题的类似问题，许多研究都有过求解算法上的深入探索。其中，Cherkesly et al.（2015）和Alyasiry et al.（2019）分别提出了两种精确算法；Gao et al.（2011）提出了一种改进的变邻域搜索算法，特别地，该研究还使用了“树结构”的新概念来完美刻画一条采用LIFO策略的行驶路径。虽然上述算法都能高效解决传统的带取送货及LIFO条件的车辆路径问题，但由于本研究所考虑的应用场景较为复杂，模型中涉及大量与时间相关的限制约束或状态转换，这些都可能严重影响算法的收敛速度。因此，针对上一章节所构建的模型，本研究设计了一种基于变邻域搜索（VNS）和自适应阈值接受（TA）思想的两阶段优化算法进行求解。其中，阶段一主要是通过VNS算法实现不同车辆间订单集任务配置和部分路径规划；阶段二使用改进的自适应阈值接受算法完善同一车辆内行驶路径的规划。不同于Gao et al.（2011）所提出的改进式变邻域搜索算法，本算法简化了邻域搜索算子的设计，以缩小可行解空间，提高算法收敛速度；此外，阶段二的自适应阈值接受算法已经实验证明可以进一步优化阶段一所得解，并且收敛迅速，效果可观。

## 贪心算法生成初始解

定义集合表示一个解，其中，表示冷藏车k的派遣路径。对于每辆冷藏车的派遣路径，编码上使用包含2n个元素的List表示（n代表冷藏车需要响应的订单集个数），每个元素都包含3个要素，其一是订单集编号，其二是用于区分取货点和交付点的flag值（取货点标记为1，交付点标记为0，后文为便于读者理解，统将取货点的flag值表示为+，交付点flag值表示为-），其三是需求量（取货点的需求量为正，交付点需求量为负）。元素在List中的顺序即表示冷藏车采取对应行为的顺序，如，某一顺序上冷藏车访问了订单集的取货点（或交付点），则List中对应顺序所在元素的订单集编号为i，flag值为+（或-）。图4-1展示了上述编码思想的应用：该问题包含5个订单集任务的配置和路径规划，冷藏车1先后访问了订单集1和订单集2的取货点，随后访问了订单集2的交付点，后续分别完成了订单集4取货点和交付点的访问，最后实现订单集1交付点的访问；冷藏车2先进行了订单集5取货点和交付点的访问，随后进行了订单集3取货点和交付点的访问。

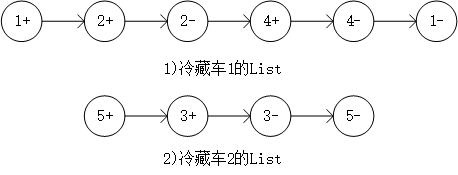


图4‑1 编码示意图

在使用贪心算法生成初始解时，定义一个待筛选订单集存放池.和一个筛选待定订单集存放池。初始设定，只要，就随机地从中拿出一个订单集放入冷藏车k的List序列中，每个订单集被取出的概率是。订单集放入冷藏车k的List序列时，遵循以下法则：以尽可能减少时间窗违约成本为前提，先根据交付时间窗安置订单集交付点的访问顺序，随后根据载重约束和LIFO规则确定订单集取货点的访问顺序，如果任何一种顺序的安排都会超过冷藏车的最大载重量，则将该订单集放入订单集存放池，重新在中选择其他订单集重复上述过程。待后，若，设置，，选择冷藏车k+1继续进行订单集交付点和取货点访问顺序的安置，直到所有的订单集存放池都为空。图4-2展现了本算法的主要思想：

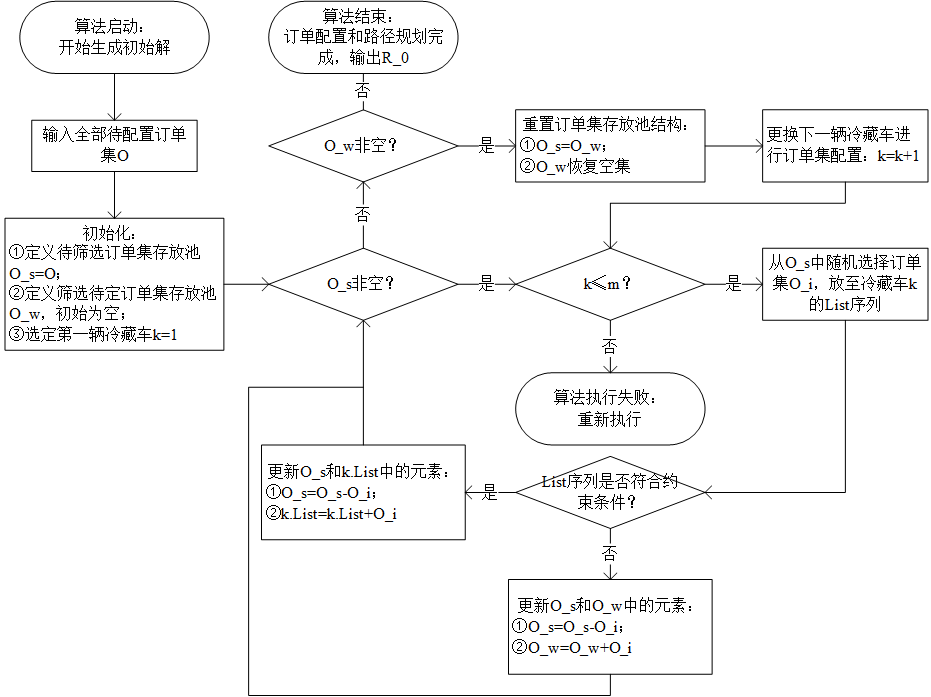


图4‑2 贪心算法生成初始解

## 两阶段优化算法

### 操作算子

为生成新的可行解，本算法引入两大类邻域搜索算子，分别应用于车辆路径间的邻域搜索（Inter-route Local Search）和车辆路径内的邻域搜索（Intra-route Local Search）。同时，为避免算法阶段一的求解陷入局部最优的困境，本研究提出了一个扰动算子。在具体介绍这些算子前，引入Carrabs et al.(2006)提到的两个概念：

Couple：记任意订单集的取货点和交付点构成的组合为Couple。Couple可视为一个仅含两个元素（订单编号相同，flag值不同）的无序集合，如，一个任意Couple可表示为。

Block：记冷藏车一条可行派遣路径中任意一条满足LIFO规则的子路径为Block。

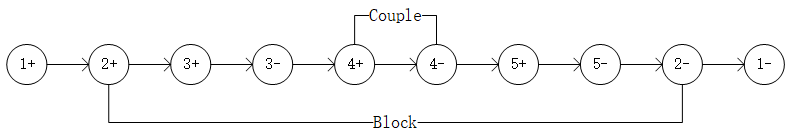


图4‑3 Couple与Block示意图

一个Couple可以形成一个最简单的Block，图4-3形象地解释了Couple和Block的概念。

以下展开对邻域搜索算子和扰动算子的解释。

（1）车辆路径间邻域搜索算子

路径间的邻域搜索操作于不同冷藏车的派遣路径上。通常有以下4种搜索算子：

**Inter-route Couple-exchange**

随机选择冷藏车派遣路径上的一个Couple与冷藏车派遣路径上的另一个Couple交换，如果考虑交换后各个Couple中的元素在派遣路径上的顺序，则可能采取的交换行为有：取货点与取货点交换，交付点与交付点交换；取货点被取货点与交付点同时替换；交付点被取货点和交付点同时替换。因而该算子的一次操作，能使一个原始解最多产生个新解。

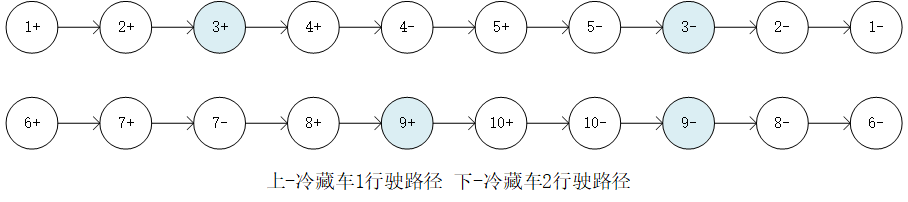


图4‑4 冷藏车1与冷藏车2原始路径

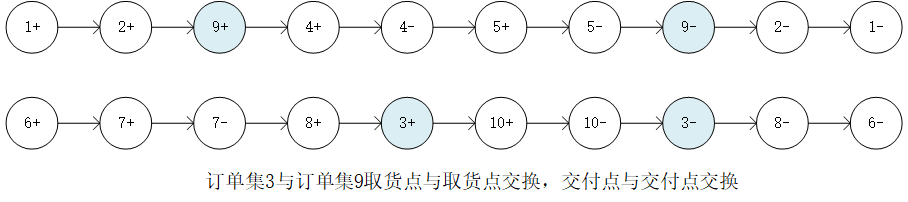


图4‑5 路径间couple交换方式1

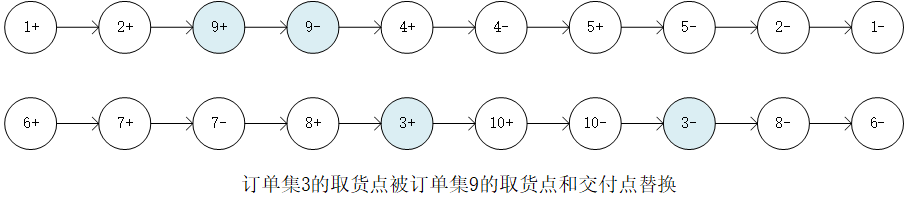


图4‑6 路径间couple交换方式2

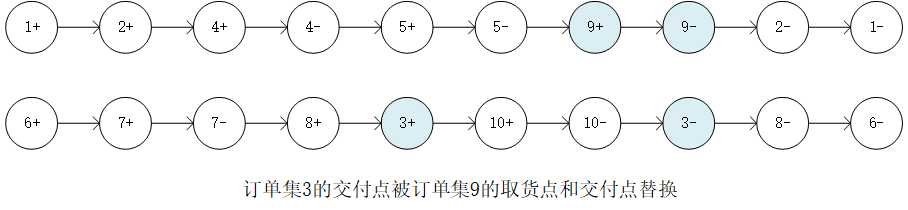


图4‑7 路径间couple交换方式3

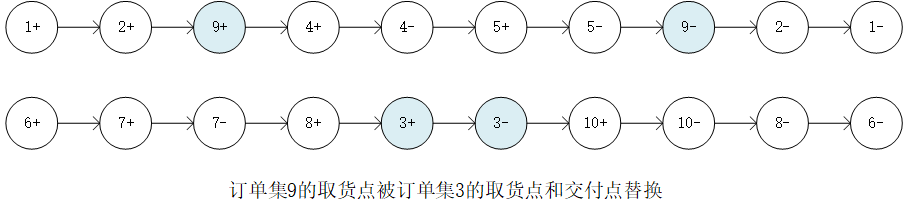


图4‑8 路径间couple交换方式4

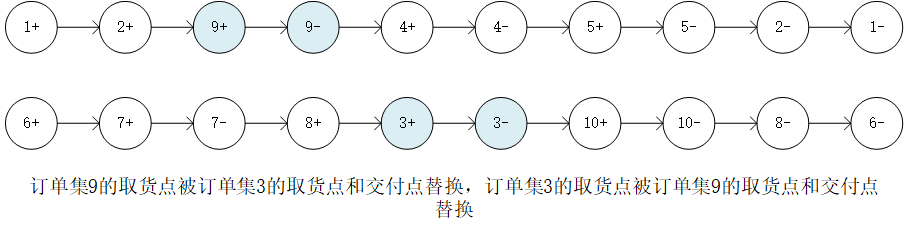


图4‑9 路径间couple交换方式5

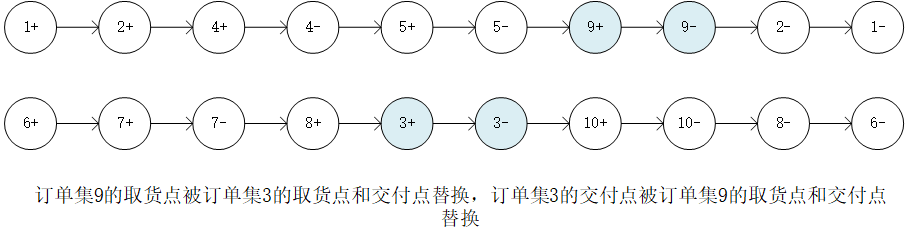


图4‑10 路径间couple交换方式6

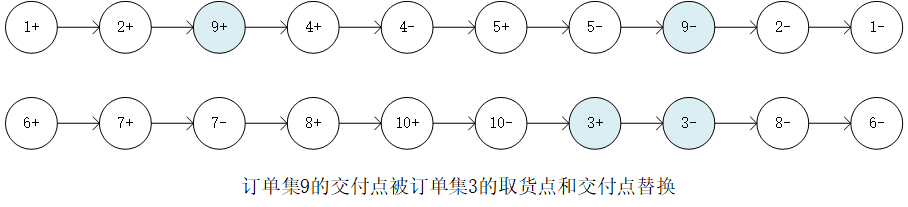


图4‑11 路径间couple交换方式7

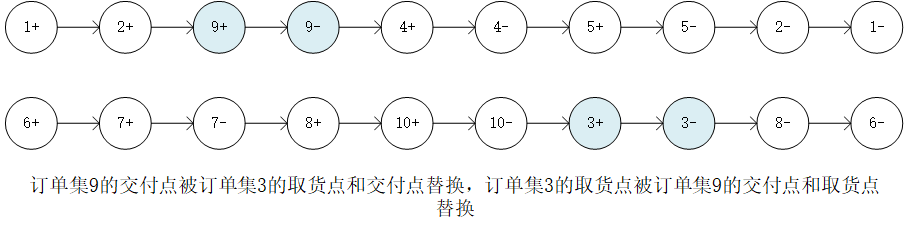


图 4‑12 路径间couple交换方式8

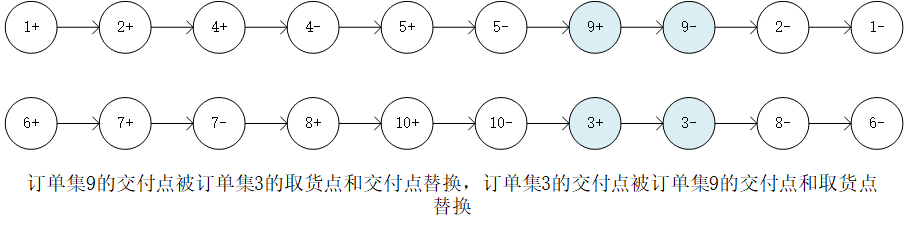


图4‑13 路径间couple交换方式9

**inter-route Block-exchange**

随机选择冷藏车派遣路径上的一个Block与冷藏车派遣路径上的另一个Block交换。该算子的一次操作能使一个原始解生成一个新解。

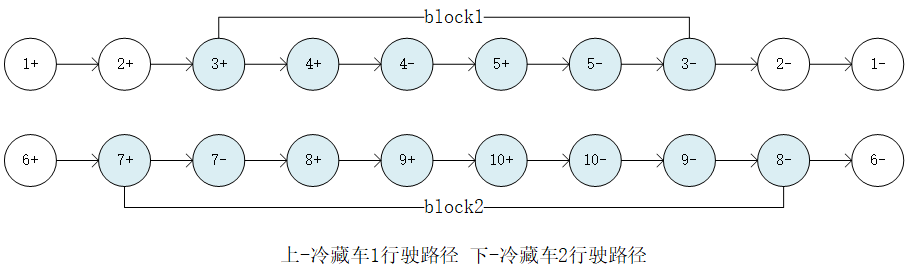


图4‑14 冷藏车1与冷藏车2原始路径

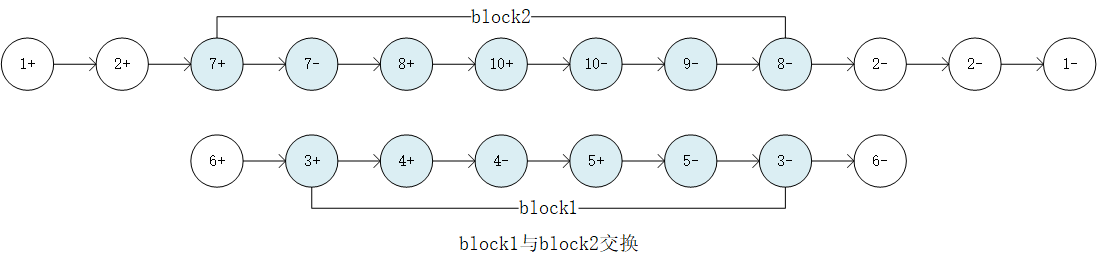


图4‑15 冷藏车1与冷藏车2发生block-exchange

**Inter-route Couple-relocate**

随机选择冷藏车派遣路径上的一个Couple，将其相关的取货点与交付点从的派遣路径中移除，并插入至冷藏车的派遣路径中。该算子的一次操作所带来的解空间变化较为复杂，与的原始派遣路径相关，但主要遵循以下原则：

先考虑Couple中取货点的放置顺序，根据该点的位置，可能的情况有：

①取货点放于两个取货点之间，则交付点可以置于取货点后面，可以置于前一个取货点所对应的交付点前面，或者后一个取货点所对应的交付点后面。

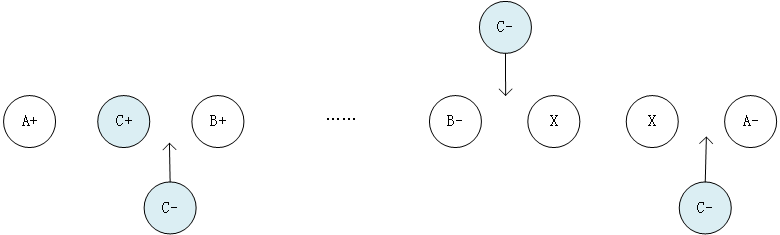


图4‑16 couple取货点位于两个取货点之间

②取货点放于两个交付点之间，则交付点必须置于取货点后面。

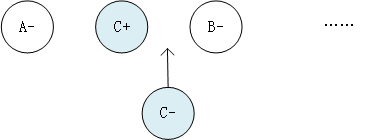


图4‑17 couple取货点位于两个交付点之间

③取货点放于一个取货点和一个交付点之间，则交付点必须置于取货点后面。

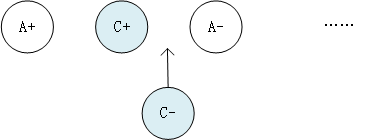


图4‑18 couple取货点位于一个取货点和一个交付点之间

④取货点放于一个交付点和一个取货点之间，则交付点可以置于取货点后面，也可以置于后一个取货点所对应的交付点后面。

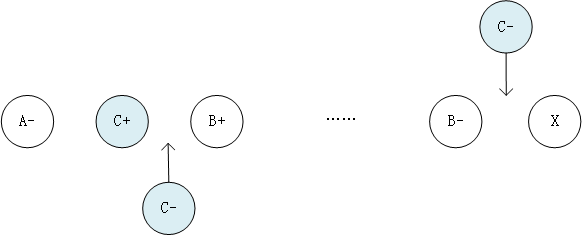


图4‑19 couple取货点位于一个交付点和一个取货点之间

⑤取货点放于行驶路径首位，则交付点可以置于取货点后面，也可以置于行驶路径最后。

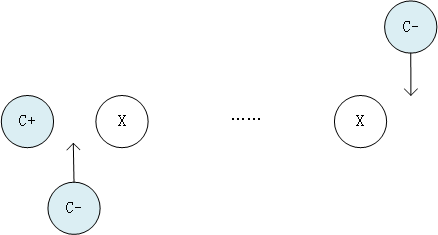


图4‑20 couple取货点位于行驶路径首位

⑥取货点放于行驶路径末位，则交付点必须置于取货点后面。

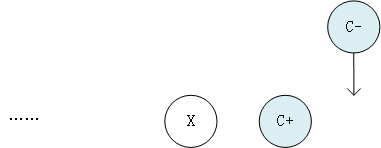


图4‑21 couple取货点位于行驶路径末位

**Inter-route Block-relocate**

随机选择冷藏车派遣路径上的一个Block，将其完整地从的派遣路径中移除，并插入至冷藏车的派遣路径中。该算子一次操作所带来的解空间变化与的原始派遣路径相关。理论上，可能产生的新解个数为原始派遣路径的长度加一。

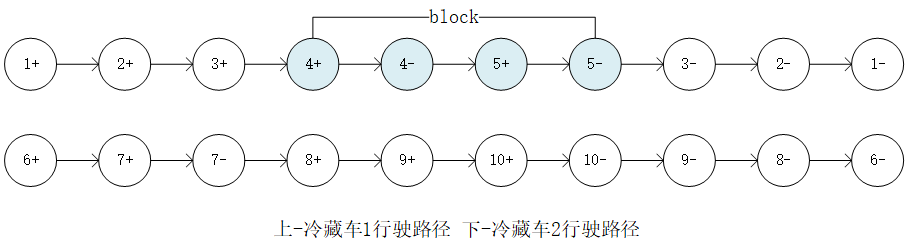


图4‑22 冷藏车1和冷藏车2原始行驶路径

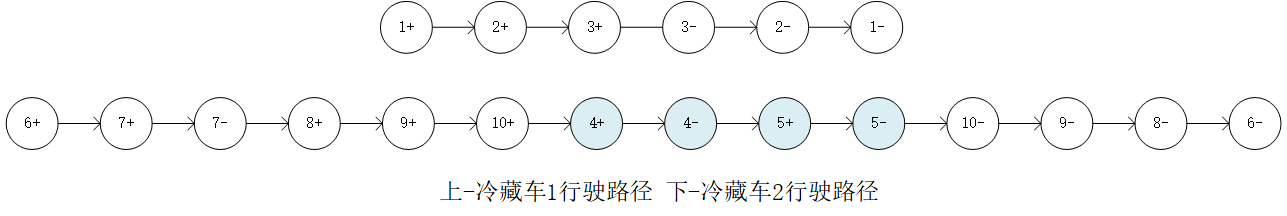


图4‑23 执行完block-relocate后

（2）车辆路径内邻域搜索算子

路径内的邻域搜索操作于同一辆车的派遣路径上，通常也有Intra-route Couple-exchange、Intra-route Block-exchange、Intra-route Couple-relocate、Intra-route Block-exchange4种算子，其作用方式与4种路径间邻域搜索算子类似，不再一一赘述。

（3）扰动算子

扰动算子主要包括以下步骤：

Step1. 随机选择两辆冷藏车和。如果，则转Step2；否则，转Step3。

Step2. 随机选择冷藏车上的两个Couple，执行Intra-route Couple-exchange算子。

Step3. 分别随机选择和上的两个Couple，从原始派遣路径中移除，并随机插入至对方的派遣路径中。

Step4. 上述步骤的操作可行且执行完毕，扰动运算终止，并成功生成一组新解；否则，返回Step1。

扰动运算的本质在于Couple的交换，但不同于Inter-route Couple-exchange运算的是，上述过程更具随机性，增加了解空间的丰富与多样化，使得算法的求解更易跳出局部最优的困境。

### 阶段一：基于VNS的车辆间优化

阶段一在优化配置不同冷藏车之间的订单集任务时，也会参与部分路径规划，算法核心在于变邻域搜索思想的运用：当前邻域找不到更好的解便跳到下一个邻域搜索，一旦找到更优的解则返回至第一个邻域搜索，其整体流程框架如图4-24：

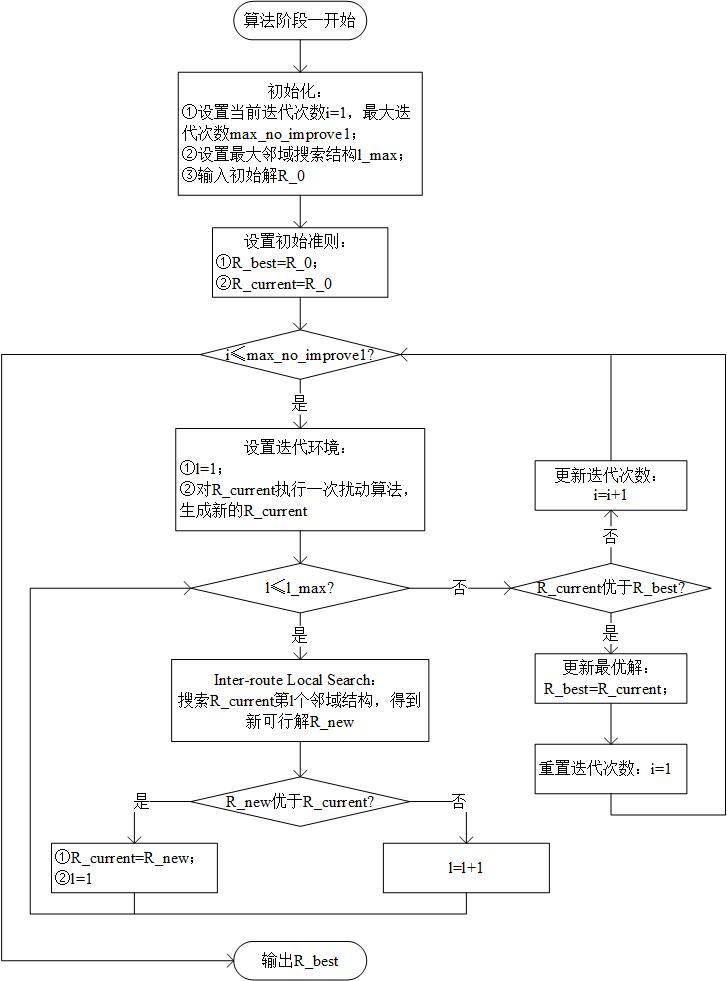


图4‑24 算法阶段一流程框架

### 阶段二：基于TA的车辆内优化

阶段二在阶段一所得解的基础上继续优化车辆内各个订单集任务的执行顺序，从而获得一个最佳的行驶路径。该部分的核心在于运用一个自适应接受阈值来判断是否接受算法每次迭代所产生的一个较差解，这也相当于对当前解实施了一次扰动算法，有助于求解过程跳出局部最优，其整体流程框架如图4-25：

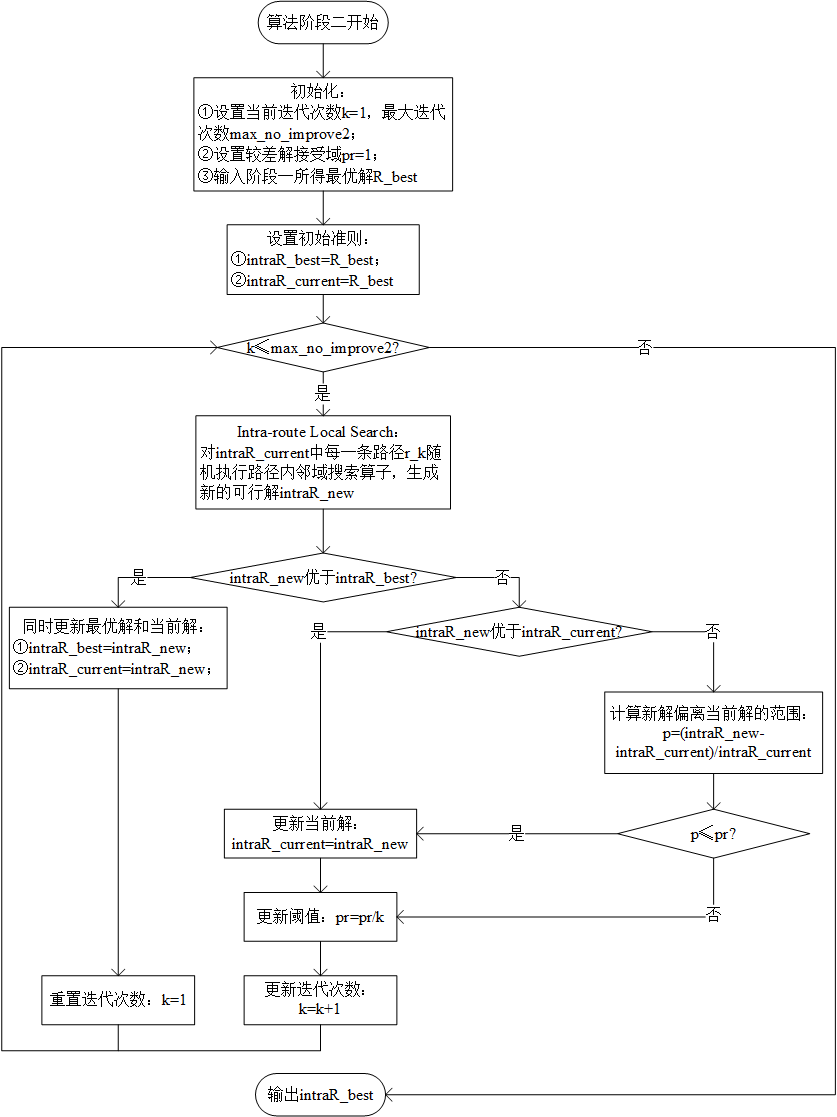


图4‑25 阶段二流程框架

## 本章小结

为求解上一章节所构建的数学模型，本章设计了两阶段优化算法，主要内容包括：优化运算前，使用贪心算法生成一组可行解；优化运算时，先基于变邻域搜索算法完成车辆间订单集任务配置和部分路径规划（阶段一），再运用自适应阈值接受算法进一步改进车辆内订单集任务执行顺序（阶段二）。为更好地支持算法阶段一和阶段二的运行，本章引入了4种车辆路径间邻域搜索算子和4种车辆路径内邻域搜索算子，同时，为帮助算法的求解突破局部最优的困境，算法阶段一还设计了一种可以增加解空间随机性和多样性的扰动算子，阶段二使用了一套自适应型较差解接受法则。

# 算例实验分析

本章节将引入算例数据，通过实验模拟一家社区团购平台网格仓中的生鲜配送实际运营状况及规划决策过程，一方面验证数学模型的合理性和求解算法的有效性；另一方面探索分析影响算法效率和求解结果的关键因素，从而进一步提升算法求解效度、找寻具有现实意义的管理学启示。

## 基本设置

### 算例数据预处理

实验场景设置需考虑消费者自提点的分布情况和消费者自提点的数量规模两个维度。其中，消费者自提点分布情况分为随机分布、聚类分布和混合分布三种类型，本研究以现实常见的混合分布网络作为研究对象；消费者自提点数量规模则以和作为小规模群体和大规模群体的典型代表。

根据上述分析，实验选取Solomon's benchmark数据集中需求规模分别为50和100所对应的标准算例RC205，RC206，RC207，RC208作为基础数据源。另外，为更贴合实验需要，做以下数据处理：

（1）添加网格仓定位

假设社区团购平台在城市中随机建造了5个网格仓，设置5个网格仓的位置数据如表5-1所示：

表5‑1 网格仓地理位置信息

|  |  |
| --- | --- |
| **网格仓编号** | **位置数据** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

初始阶段假设每个网格仓内都停放有1辆冷藏车。

（2）构建配送网络

对需求规模为的每个标准算例RC205，RC206，RC207，RC208，保持基础数据以形成配送网络中的交付点，并令每个交付点对应的取货点随机产生于5个网格仓中，由此构建了4个规模为50的配送网络，分别编号为“1-RC50”、“2-RC50”、“3-RC50”、“4-RC50”。

类似地，对需求规模为的每个标准算例RC205，RC206，RC207，RC208，保持其基础数据以形成配送网络中的交付点，令每个交付点对应的取货点随机产生于5个网格仓中，由此可以构建4个规模为100的配送网络，分别编号为“1-RC100”、“2-RC100”、“3-RC100”、“4-RC100”。

（3）转换时间范围

假设网格仓进行生鲜配送服务的时间范围为09:00-21:00，而算例所提供的时间计量范围是0-960，因此算例中的一单位时间相当于现实生活中的0.75分钟。对于算例中的任意时间，其对应于现实中的时间为时分，其中表示对向下取整。

（4）添加车辆行驶速度

假设冷藏车在正常时段的行驶速度为36km/h，在交通拥堵时段的行驶速度为18km/h。考虑早晚出行高峰的影响，一般09:00-10:00、16:00-18:00是交通拥堵较为严重的两个时间段。因此，对应于本实验，车辆行驶速度与时间段具有如表5-2所示的关系：

表5‑2 车辆行驶速度-时间段关系图

|  |  |
| --- | --- |
| **时间段** | **车辆行驶速度（单位距离/单位时间）** |
| 0-80（拥堵时段） | 0.4 |
| 80-560（正常时段） | 0.8 |
| 560-720（拥堵时段） | 0.4 |
| >720（正常时段） | 0.8 |

### 算法参数设置

本算法采用Java语言编程实现。算法涉及的相关参数设置如下：

（1）单位距离运输成本。考虑城市道路通常呈现“井”字型分布，运输距离的计算使用曼哈顿距离。如，位置点与位置点之间的距离为。

（2）订单集交付早于规定的时间窗所产生的单位时间等待成本为，订单集交付晚于规定的时间窗所产生的单位时间延误成本为。

（3）由于冷藏车与外界进行热交换所产生的能耗主要取决于订单集体量和装卸货速度，在本研究中，订单集体量和装卸货速度对于同一个算例而言属于常量，恒定不变，因此优化运算时可不做考虑。于是，制冷成本主要考虑冷藏车运输途中的能量消耗，设置单位时间成本为。

（4）设置团长对生鲜产品的最低新鲜度要求为。考虑到人们对负向服务体验总是比对正向服务体验更加敏感，设置高于此新鲜度要求产生的单位增值效益为，低于此新鲜度要求产生的单位增值效益为。生鲜产品新鲜度、贮存温度、运输时间三者之间的关系式，即式中的常数部分为。

（5）其他参数设置有：算法第一阶段4种路径间邻域搜索算子的最大迭代次数为；第一阶段算法的终止条件为，即连续600次内未见当前解发生改进便终止算法第一阶段。算法第二阶段4种路径内邻域搜索算子的最大迭代次数为；第二阶段算法的终止条件为最大迭代次数，即连续300次内未见当前解发生改进便终止算法第二阶段。

## 实验结果分析

### 算法测试

（1）算法收敛性

选取配送网络“1-RC100”进行实验，算法迭代过程数据如图5-1，5-2所示，实验结果数据如表5-3，5-4所示：

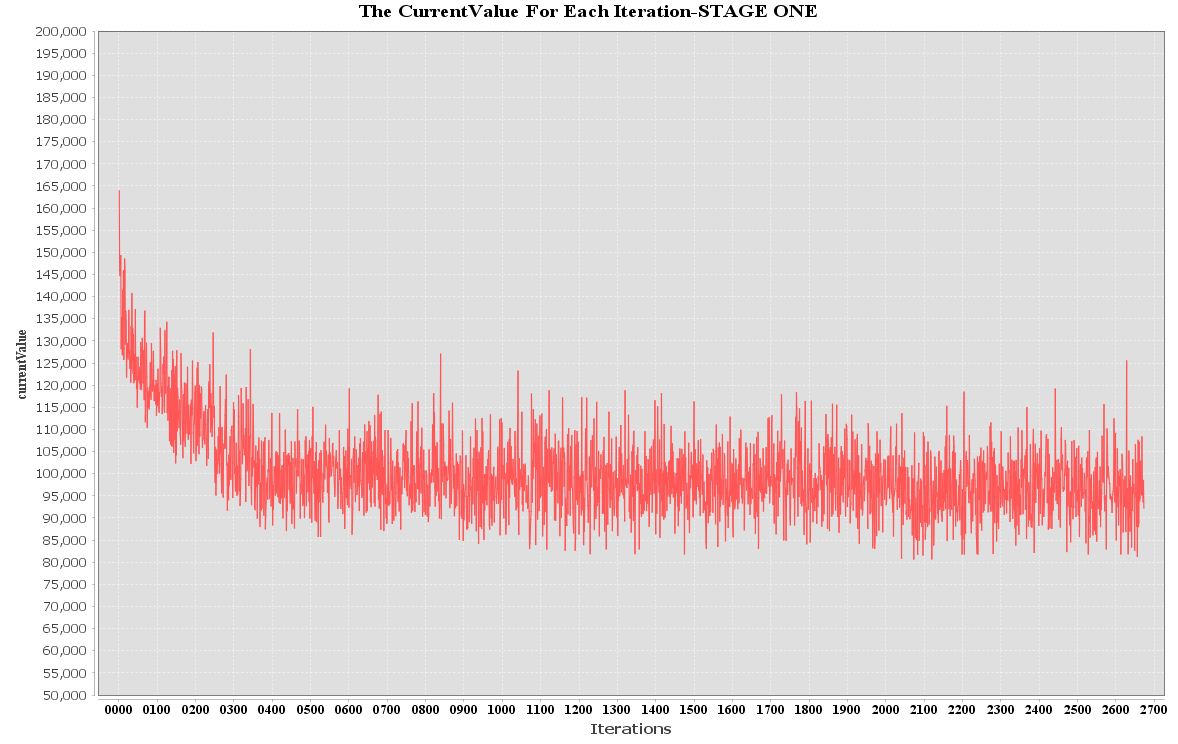


图5‑1 算法阶段一迭代过程图

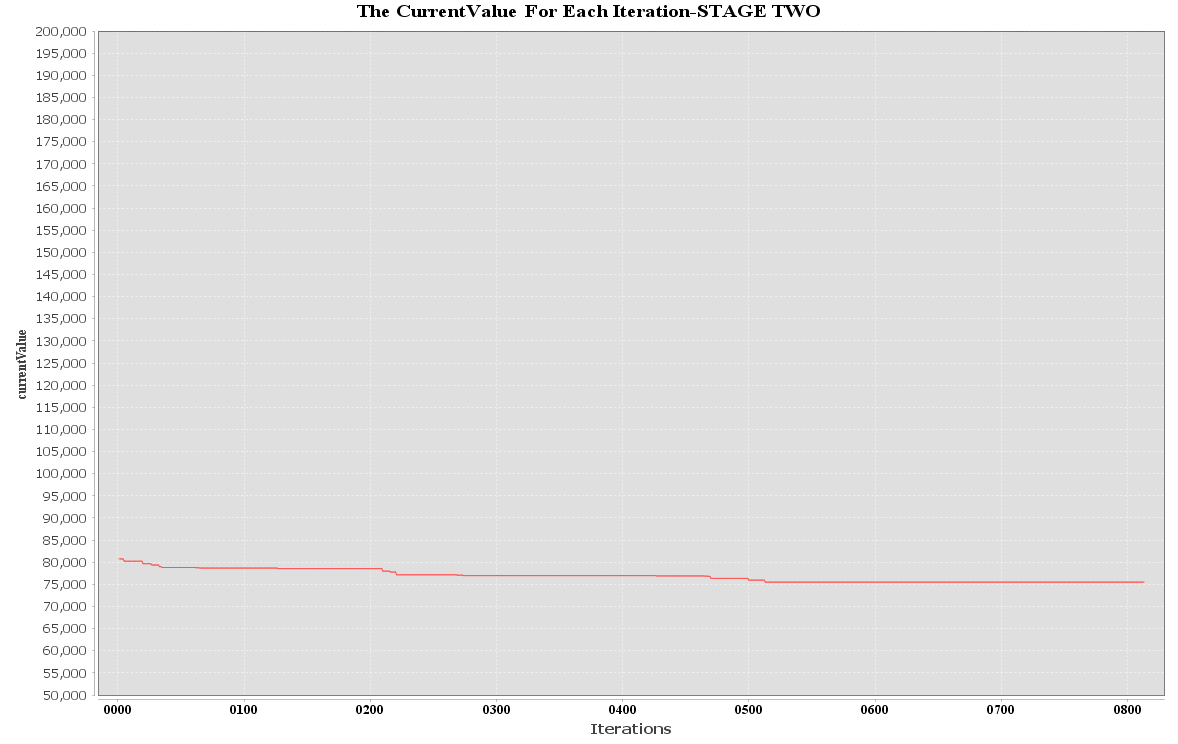


图5‑2 算法阶段二迭代过程图

阶段一在迭代至400次开始收敛，完整运行总时长大约为284.2s；阶段在迭代至500次开始收敛，完整运行总时长大约为1.0s。由此可见，本算法满足一定的收敛性，在实验设置的终止条件下，求解结果可以收敛于一个表现较优的解。

表5‑3 配送网络“1-RC100”算法求解结果

| 最优成本 | 运输成本 | 时间窗违约成本 | 制冷成本 | 品质效益 | 初始新鲜度 | 最优新鲜度 | 阶段一时长 | 阶段二时长 | 算法运行总时长 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 75494 | 26960 | 23995 | 40350 | 15811 | 0.27 | 0.74 | 284.2s | 1.0s | 285.2s |

表5‑4 冷藏车订单集任务执行顺序/路径规划

| 冷藏车编号 | 订单集任务执行顺序 |
| --- | --- |
| 1 | 76,63,163,176,7,8,5,6,106,105,108,107,37,36,38,138,136,137 |
| 2 | 51,85,50,27,26,28,29,31,34,32,30,33,62,67,167,162,133,130,132,134,131,129,128,126,127,150,185,151,97,81,96,94,194,196,181,75,77,177,175,197 |
| 3 | 25,18,19,64,164,119,118,125,43,42,72,78,73,15,47,14,12,11,13,9,109,113,111,112,114,147,115,173,178,172,142,40,140,143 |
| 4 | 35,87,86,88,188,186,187,70,54,71,69,53,52,152,153,169,4,3,1,2,79,61,161,179,102,101,103,104,48,23,21,49,20,22,122,120,149,121,123,148,93,84,184,193,171,154,170,135 |
| 5 | 95,92,192,195,100,82,90,98,99,83,183,199,198,190,16,17,117,116,182,74,57,65,165,157,174,41,44,45,46,60,10,59,159,110,160,146,145,144,141,39,139,24,124,91,191,55,155,200,58,56,66,80,180,166,156,89,189,158,68,168 |

（2）算法稳健性

保持实验条件不变，选取配送网络“1-RC100”模拟实验10次，记录每次实验结果如表5-5所示：

表5‑5 配送网络“1-RC100”十次模拟实验结果

| 实验次数 | 最优成本 | 偏差（%） | 阶段一运行时长(s) | 阶段二运行时长(s) | 算法运行总时长（s） |
| --- | --- | --- | --- | --- | --- |
| 1 | 75099 | 2.7 | 156.9 | 0.9 | 157.8 |
| 2 | 72866 | 0.3 | 333.6 | 1.1 | 334.7 |
| 3 | 76542 | 4.7 | 373.0 | 0.9 | 373.9 |
| 4 | 65952 | 9.8 | 408.0 | 0.9 | 408.9 |
| 5 | 73224 | 0.2 | 551.9 | 0.9 | 552.8 |
| 6 | 70108 | 4.1 | 275.2 | 0.6 | 275.8 |
| 7 | 76089 | 4.1 | 176.8 | 1.7 | 178.5 |
| 8 | 76013 | 4.0 | 438.1 | 1.3 | 439.4 |
| 9 | 69592 | 4.8 | 231.3 | 1.2 | 232.5 |
| 10 | 75494 | 3.3 | 284.2 | 1.0 | 285.2 |
| Average | 73098 | 0.0 | 322.9 | 1.1 | 324.0 |

10次实验最优成本的极值分别为76542和65958，平均值为73098。同时，计算各次实验最优成本的偏差均在平均值的10%以内，表明本算法稳健性良好。

（3）算法有效性

选取上述全部配送网络分别展开实验，记录每次实验结果如表5-6所示，实验结果表明无论是较大规模的配送网络还是较小规模的配送网络，本算法都可以在适当的时间内计算出一个良好的解决方案，验证了本算法设计的有效性。

表 5‑6 不同配送网络下的算法求解情况

| 配送网络 | 最优成本 | 运输成本 | 制冷成本 | 时间窗违约成本 | 品质效益 | 平均新鲜度 | 冷藏车使用数量 | 阶段一运行时长（s） | 阶段二运行时长（s） | 算法运行时长（s） |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1-RC50 | 48887 | 15360 | 23480 | 19500 | 9453 | 0.76 | 3 | 71.9 | 0.4 | 72.3 |
| 2-RC50 | 31651 | 14820 | 22710 | 8760 | 14639 | 0.82 | 3 | 116.7 | 0.8 | 117.4 |
| 3-RC50 | 28777 | 12300 | 19280 | 7580 | 10383 | 0.77 | 3 | 26.7 | 0.5 | 27.2 |
| 4-RC50 | 30862 | 13360 | 20190 | 6300 | 8988 | 0.75 | 3 | 26.6 | 0.7 | 27.3 |
| 1-RC100 | 75494 | 26960 | 40350 | 23995 | 15811 | 0.74 | 5 | 284.2 | 1.0 | 285.2 |
| 2-RC100 | 56883 | 27000 | 41780 | 22835 | 34732 | 0.85 | 5 | 966.4 | 2.1 | 968.6 |
| 3-RC100 | 70336 | 26500 | 39990 | 23950 | 20104 | 0.77 | 5 | 93.2 | 1.3 | 94.5 |
| 4-RC100 | 45365 | 23040 | 35350 | 12730 | 25755 | 0.80 | 5 | 140.3 | 2.4 | 141.6 |

### 算法对比分析

通过对实验结果的分析可以发现，本算法的核心在于第一阶段基于VNS算法实现车辆间订单集任务配置和部分路径规划，可以使贪心算法所得初始解获得较大程度的改进；而第二阶段则是在第一阶段所得解的基础上，利用自适应阈值接受算法的思想继续完善同一车辆内行驶路径规划，具有一定优化作用，但改进效果相比第一阶段改进效果对整体影响较小。因此，本节将通过设置相同的运行时间，对比两阶段完整优化（即考虑第一阶段和第二阶段的联合优化）的求解结果与一阶段优化（即仅考虑第一阶段的优化）的求解结果，探索算法第二阶段设计的必要性。

根据上一节第二部分的分析可知，算法阶段一的平均运行时长大约为322.9s，算法阶段二的平均运行时长大约为1.1s。因此，设置对比算法条件：

（1）两阶段优化：控制两阶段完整优化的最大运行时长为324.0s，其中，第一阶段的最大运行时长为322.9s，第二阶段的最大运行时长为1.1s。

（2）一阶段优化：控制一阶段优化的最大运行时长为324.0s。

对配送网络“1-RC100”，采用两阶段优化和一阶段优化分别模拟运行10次，记录每次实验最优成本如图5-3所示：

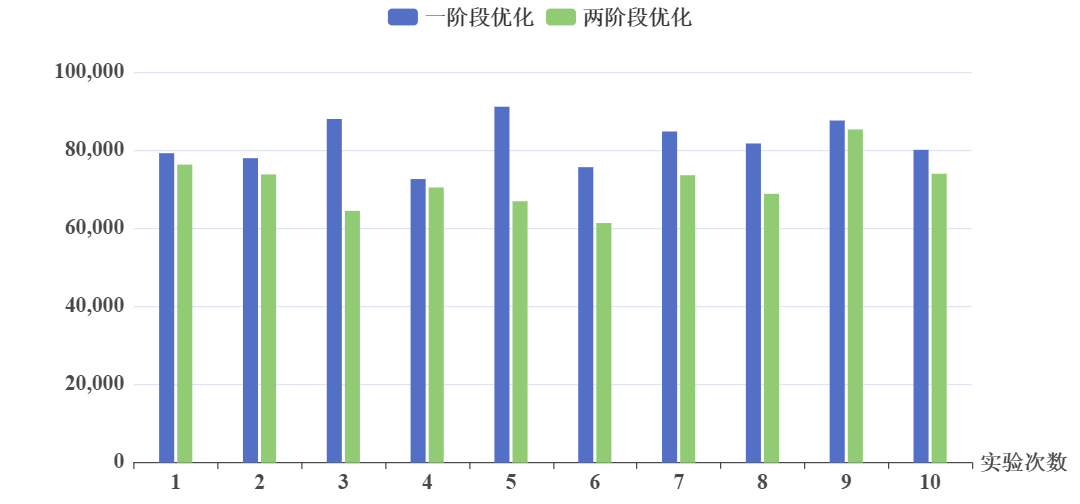


图5‑3 一阶段算法与两阶段算法求解结果对比图

10次实验结果表明两阶段优化算法比一阶段优化算法表现更佳。算法阶段一使用的4种路径间邻域搜索算子不仅丰富了订单集任务的配置形式，同时也间接影响了车辆对于每个位置点的访问顺序即行驶路径。理论上，在足够宽泛的终止条件下，算法阶段一可以使订单集任务配置和行驶路径规划的解放方案同时收敛至一个较优的解，但该解的质量不如阶段一和阶段二联合优化所得解的质量理想，主要原因在于算法阶段二所使用的4种路径内邻域搜索算子、随着迭代次数增加而逐渐减小概率地接受较差解，进一步地丰富了可行解的范围，使得算法可以较为迅速地收敛至一个更理想的解。因此，算法阶段二的设计是有必要的。

### 时变路网条件分析

本研究考虑城市道路在早晚出行高峰可能发生交通拥堵，引入了时变路网条件。本节将探讨不同时变程度的道路网络对生鲜配送最优解的影响。

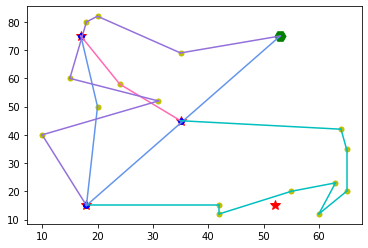
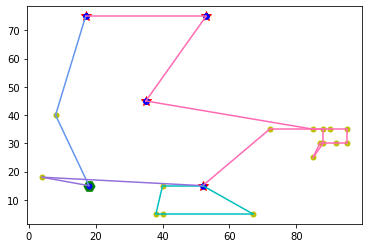
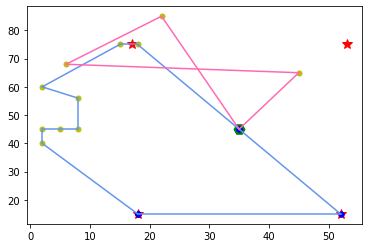
实验需用到的道路网络及各时段车辆行驶速度如表5-7所示：

表5‑7 不同路网下的车辆行驶速度-时间段对应关系

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **道路网络编号** | **0-80** | **80-560** | **560-720** | **>720** | **前两段平均速度** |
|  | 0.3 | 0.9 | 0.4 | 0.8 | 0.81 |
|  | 0.4 | 0.8 | 0.4 | 0.8 | 0.74 |
|  | 0.5 | 0.7 | 0.4 | 0.8 | 0.67 |
|  | 0.6 | 0.6 | 0.4 | 0.8 | 0.60 |
|  | 0.7 | 0.5 | 0.4 | 0.8 | 0.53 |
|  | 0.8 | 0.4 | 0.4 | 0.8 | 0.46 |
|  | 0.9 | 0.3 | 0.4 | 0.8 | 0.39 |

单位：距离/单位时间

对配送网络“1-RC100”使用上述道路网络，分别展开5次实验，都能得到类似的实验结果，记录视觉效果最好的一组实验结果如下：

* 使用道路网络，冷藏车行驶路径如图5-4所示，其中，断点个数为0。

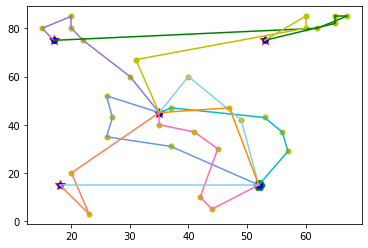
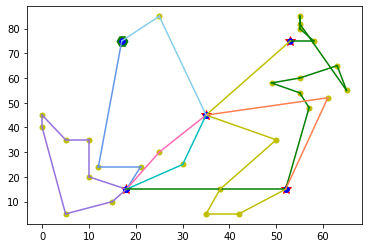
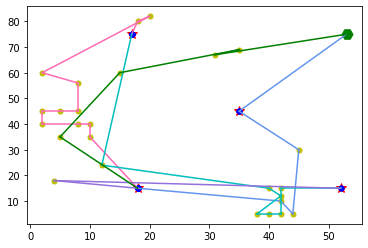
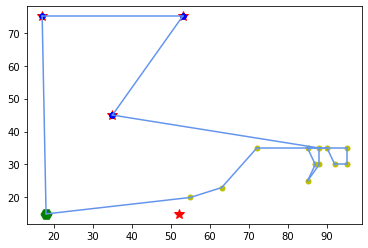
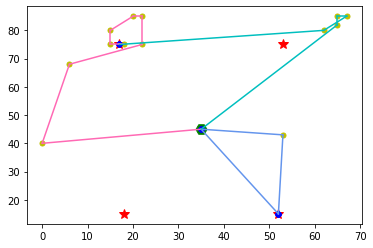


图5‑4 下的冷藏车行驶路径图

* 使用道路网络，冷藏车行驶路径如图5-5所示，其中，有两个不易观察出的断点。



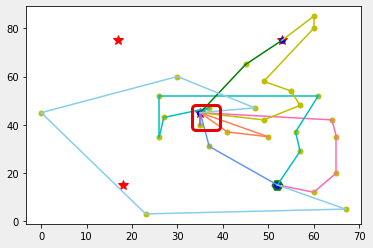
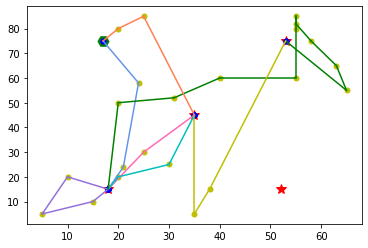
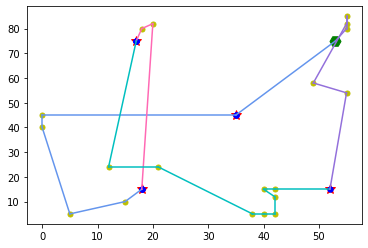
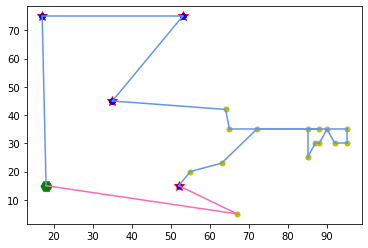
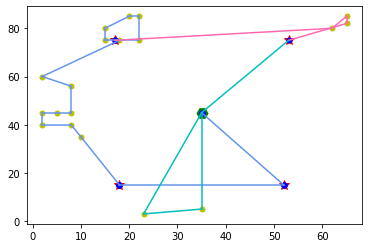


图5‑5 下的冷藏车行驶路径图

* 使用道路网络，冷藏车行驶路径如图5-6所示，其中，有1个关联路径较长且非常显著的断点。



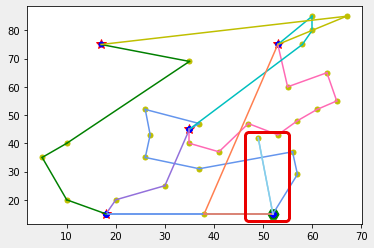
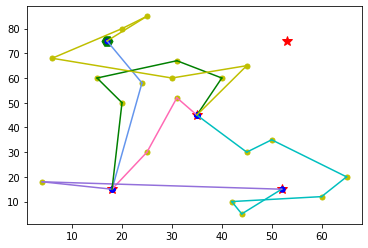
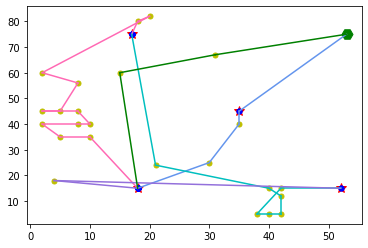
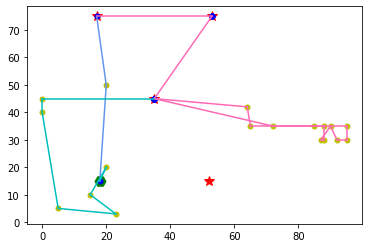
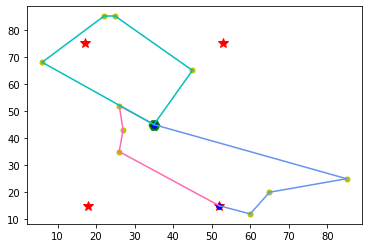


图5‑6 下的冷藏车行驶路径图

* 使用道路网络，冷藏车行驶路径如图5-7所示，其中，有2个关联路径较长的断点。



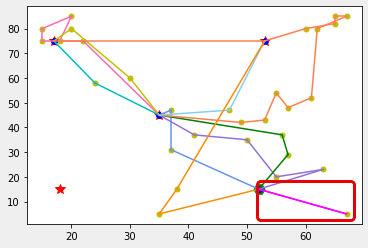
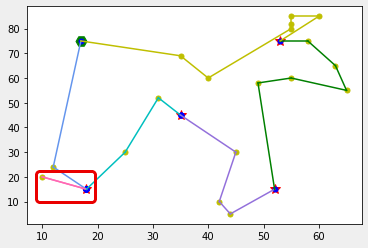
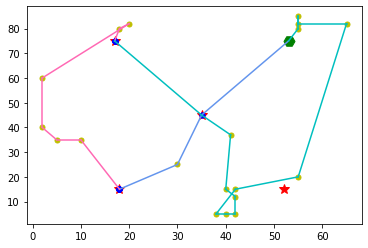
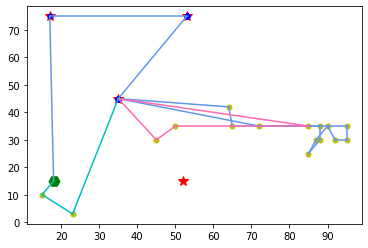
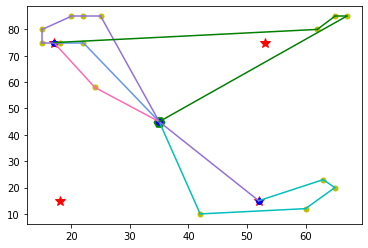


图5‑7 下的冷藏车行驶路径图

* 使用道路网络，冷藏车行驶路径如图5-8所示，其中，有2个关联路径较长的断点和1个关联路径较短的断点。



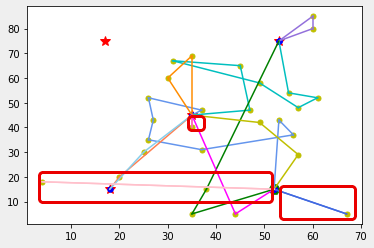
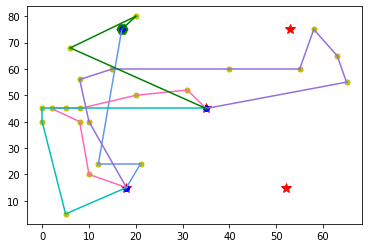
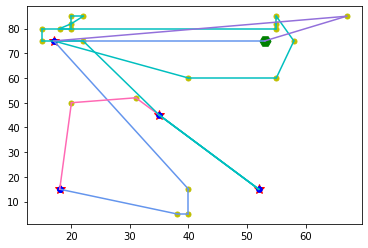
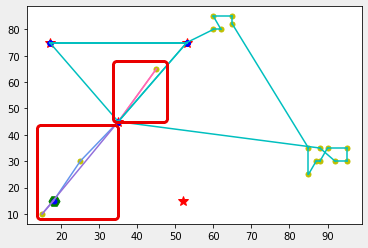
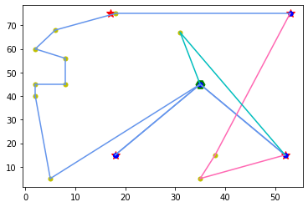


图5‑8 下的冷藏车行驶路径图

* 使用道路网络，冷藏车行驶路径如图5-9所示，其中，有3个关联路径较长的断点。



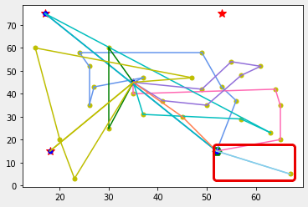
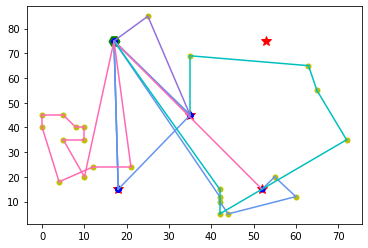
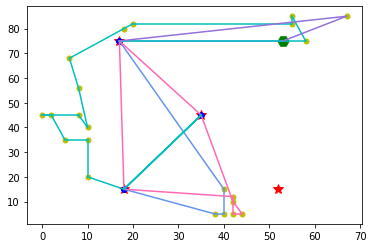
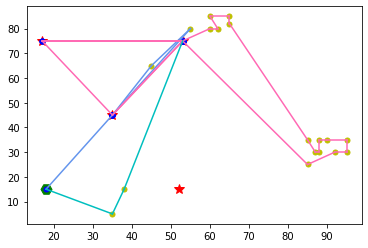
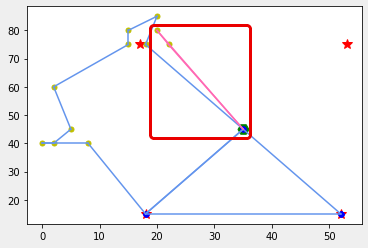


图5‑9 下冷藏车行驶路径

* 使用道路网络，冷藏车行驶路径如图5-10所示，其中，有4个关联路径很长的断点。



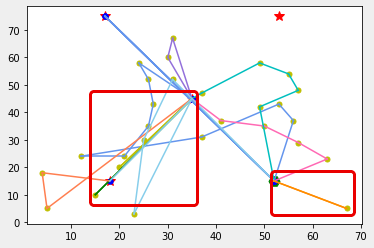
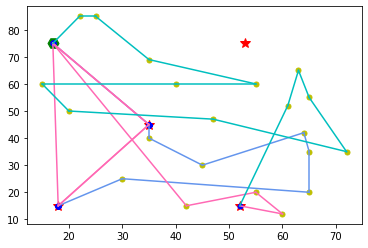


图5‑10 下的冷藏车行驶路径

关于实验结果图的一些说明：

（1）上图从左至右依次表示冷藏车1-5的行驶路径图。图中蓝色的点表示行驶路径中的取货点，黄色点表示交付点，红色五角星表示5个网格仓，绿色菱形点表示初始停放冷藏车的网格仓，冷藏车初始从该点出发，待执行完所有订单集任务后回到该点。

（2）连线连接了冷藏车相邻顺序的访问点（注意，由于本实验考虑的是曼哈顿距离，因此途中连线并不表示冷藏车的真实行驶路径），连线发生颜色变化表示上一个点是交付点而由该点引发了新的取货任务，因此一根颜色无变化的连线的实际起始点与实际终止点一定是取货点或网格仓。

（3）红色框圈住了某些仅与取货点建立单边联系的点（除其余取货点或网格仓位置点外），称之为“断点”。“断点”的形成是因为冷藏车在取货点完成取货，便立即出发访问对应交付点，在完成订单集交付后，又原路返回至取货点执行其他订单集的取货任务。

根据以上实验结果，可以发现，道路网络从变更至，断点状态呈现两个趋势：①数量有增无减；②关联路径在适当范围内变长。断点状态之所以会呈现这样的变化规律是因为，道路网络从变到的过程中，冷藏车的平均速度也在降低；为了确保部分订单集可以及时交付也为了保障生鲜产品品质，冷藏车宁愿“少取多送”，即每次在网格仓仅完成少量（甚至是一个）订单的取货任务，待这些订单实现交付后，再折回网格仓执行其他订单集的取货任务。此外，后进先出交付策略也一定程度地引导冷藏车牺牲局部运输成本换取更大的品质效益。

综上可得，道路交通状况会较大程度地影响订单集任务的执行顺序，并且糟糕的道路交通状况还会诱导冷藏车暂时牺牲局部运输成本的优化而追求更大的品质效益和订单交付的及时性。

### 冷藏车使用数量分析

目前为止的所有实验均是考虑使用5辆冷藏车执行订单集任务，而实验结果也表明5辆冷藏车恰能满足100个消费者自提点的服务需求。在这种条件下，运力资源得到充分利用的同时也引发了一个问题：冷藏车行驶路径过长，增加了部分生鲜产品的在途时间，导致其品质下降严重，影响客户满意度。因此，本节考虑增加冷藏车的使用数量、限制每辆冷藏车的行驶路径长度，探讨冷藏车的使用数量分别处于不同水平下，最优解对应的成本、收益大小。

使用配送网络“1-RC100”，对其分别在冷藏车使用数量为5、6、7、8、9、10、11、12的条件下模拟实验5次（考虑增加的冷藏车初始随机停放于5个网格仓中），记录每次实验的结果并绘制5次实验的平均结果如图5-11、5-12、5-13。

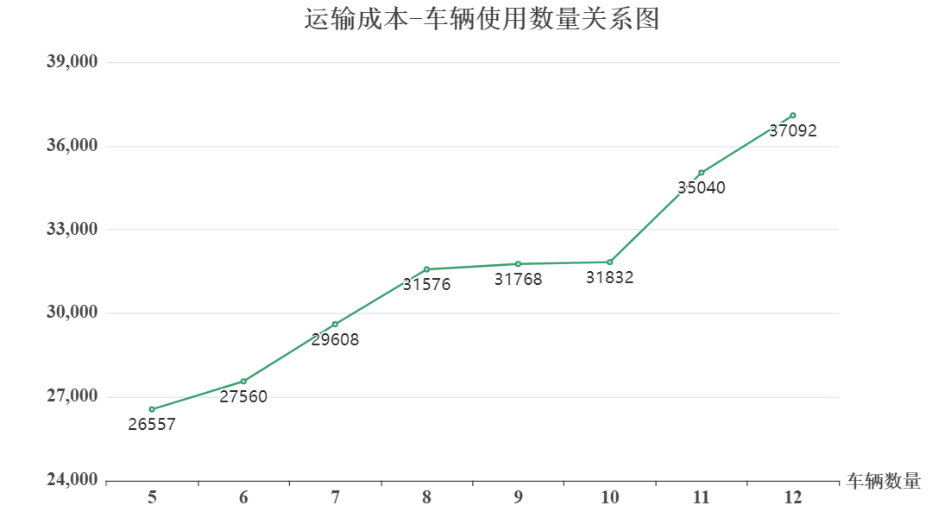


图5‑11 运输成本随冷藏车数量增加的变化关系图

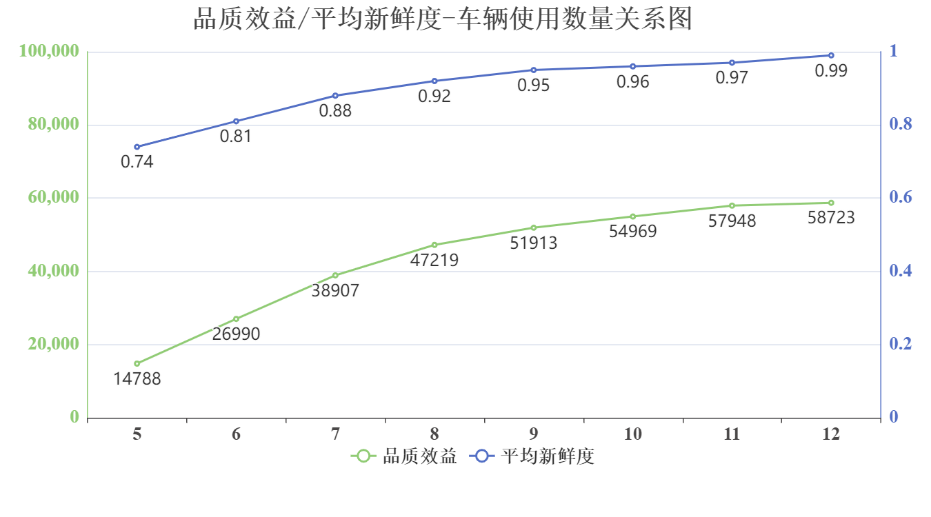


图5‑12 品质效益/新鲜度随车辆使用数量增加的变化关系图

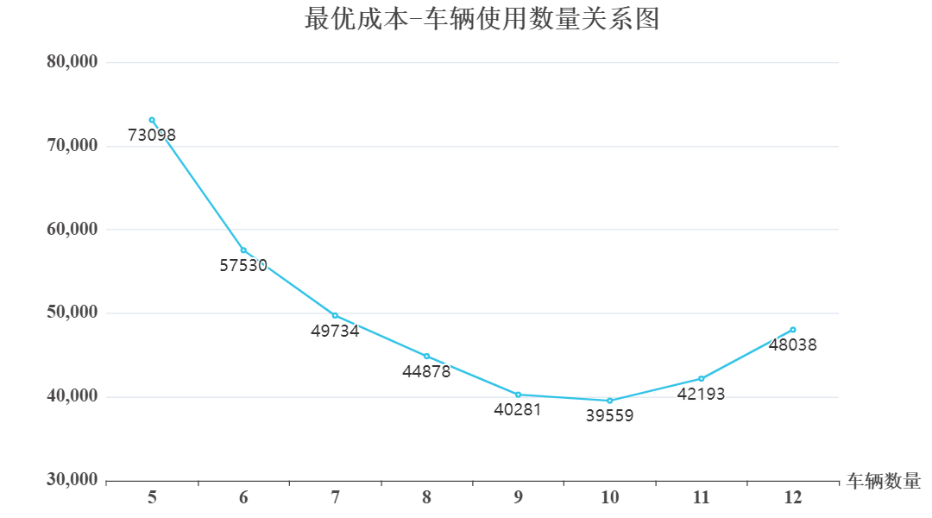


图5‑13 最优成本随车辆使用数量增加的变化关系图

根据上述实验结果可以发现：

（1）增加冷藏车的使用数量会带来运输成本的增加。如图5-11所示，随着冷藏车的数量从5增加至12，运输成本持续在增加。这主要是因为运输成本与冷藏车的行驶距离正相关，而行驶距离遵循三角不等关系，即使不考虑车辆使用的固定成本，派遣多辆车完成任务也会比派遣更少的车完成任务耗费的成本大。

（2）适当增加冷藏车的使用数量有助于提升生鲜产品的平均新鲜度，提高客户满意度。冷藏车使用数量从5增加至8的过程中，生鲜产品平均新鲜度提升明显；而随着冷藏车使用数量进一步增加，生鲜产品平均新鲜度的提升趋于平缓。这点也验证了实验前的理论分析：当冷藏车的可使用数量较少时，每辆冷藏车平均分配到的订单集任务数会偏多，导致部分生鲜产品在车上滞留的时间过长，最终影响其品质。

（3）适当增加冷藏车的使用数量有助于降低运营成本。在本算例实验中，整体冷藏车的使用数量为10时，运营成本达到最小。随着冷藏车的使用数量从10增加至12，增加趋于平缓的品质效益不足以弥补持续大幅度增加的运输成本，于是运营成本呈现不断增加的趋势。

### 品质效益分析

本节将展开生鲜产品平均新鲜度有关单位品质正效益的灵敏度分析，以探究高品质生鲜产品所创造的价值增加对于改善生鲜产品平均新鲜度的影响情况。

使用配送网络“2-RC100”，对其分别在从0开始以500的增速增加至7000的条件下模拟实验5次，记录每次实验的结果并绘制5次实验的平均结果如图5-14所示：

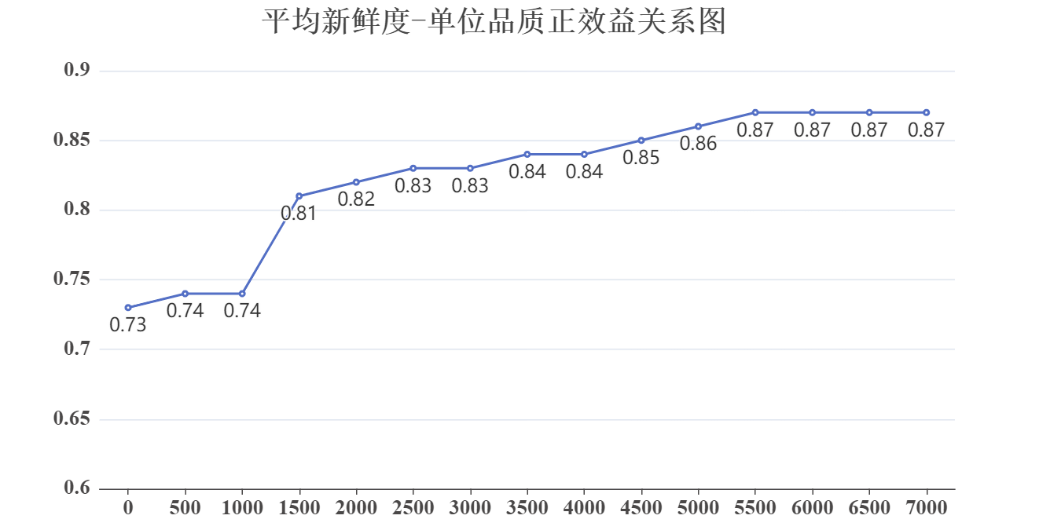


图5‑14 平均新鲜度随单位品质正效益增加的变化关系图

根据实验结果可知，单位品质正效益对生鲜产品平均新鲜度的影响近似呈现S型分布：

（1）时，的增加对生鲜产品平均新鲜度的影响较小，且当在此范围内波动时，最优解对应的平均新鲜度大约为0.74。

（2）时，生鲜产品平均新鲜度随的增加有较为明显的提升。

（3）时，随着进一步增加，生鲜产品平均新鲜度维持在0.87的稳定水平上。

本实验表明，在有限的范围内（本实验中是1000-5500），单位品质正效益增加能显著促进生鲜产品平均新鲜度的提升。低于该范围时，单位品质正效益在全局规划中的影响力较小，其增加对生鲜产品平均新鲜度提升的促进作用较弱。高于该范围时，受其他条件或约束的限制，生鲜产品平均新鲜度很难获得进一步提升。因此，在单位品质正效益较高的条件下，若想持续提升生鲜产品平均新鲜度，应考虑改变其他的外部条件，如增加冷藏车使用数量等。

## 本章小结

在算法效度的探索上，本章通过模拟运行多个算例数据，验证了两阶段优化算法的收敛性、稳健性以及有效性。并且，多次实验结果表明，在相同的有限运算时间里，两阶段优化算法的求解效率高于算法阶段一的求解效率，不仅证明了算法阶段二设计的必要性也证明了本算法设计的合理性。

在问题场景的探索上，本章探讨了道路网络呈现不同时变状态下的最优解情况，实验结果表明道路交通状况对冷藏车订单集任务的执行顺序影响较大，并且整体不太好的道路交通状况还会驱使冷藏车牺牲局部运输成本的优化以尽可能地及时交付订单集、收获更大的品质效益。通过探究冷藏车的使用数量对最优解的影响，本章发现增加冷藏车的使用数量会导致运输成本的增加，但能获得生鲜产品平均新鲜度和品质效益的提升，因此整体上存在一个车辆的最佳使用数量（在本实验中为10），使得运营成本达到最小。此外，本章还探究分析了单位品质正效益对于改善生鲜产品平均新鲜度的影响力，结果表明当单位品质正效益较小时（本实验为小于1000的范围），其增加对生鲜产品平均新鲜度提升的促进作用较为微弱；而当单位品质正效益达到某一界限时，其增加能带来生鲜产品平均新鲜度较为显著的提升；随着单位品质正效益进一步增加（本实验为5500），受其他约束和条件的限制，生鲜产品平均新鲜度不会持续提升，此时应考虑改变其它的外部条件以获得生鲜产品平均新鲜度的再次提升。

# 总结与展望

## 研究总结

“中心仓——网格仓——消费者自提点”的三级生鲜配送网络是伴随美团优选、十荟团等社区团购平台的兴起而逐渐发展起来的。网格仓是三级生鲜配送网络中的关键节点，由网格仓发出的生鲜产品运输活动也是该配送网络中的关键链。通过分析网格仓中的生鲜配送现状，本研究总结出其目前面临着①网格仓之间缺乏协调、②生鲜产品的消费具有较大随机性、③消费者自提点的不确定性较多、④生鲜配送成本过高、⑤生鲜产品品质难以保障的五大发展障碍。并且，进一步研究发现这五大发展障碍都指向了一个本质问题，即“彼此之间缺乏联系的网格仓难以从一个整体视角实现运力资源的合理调度，生鲜配送路径的优化也迫在眉睫”。为此，本研究提出了网格仓间协同开展生鲜配送服务的思想，并对其中涉及的配送车辆与订单任务的配置问题、配送车辆行驶路径的规划问题展开了探索，主要工作包括：

（1）基于对美团优选、十荟团等社区团购平台的网格仓中生鲜配送现状的分析，构建了考虑网格仓间协同的生鲜配送优化模型。该模型针对网格仓间协同开展生鲜配送服务的思想，建立了生鲜产品取货环节与交付环节的一体式优化，不仅综合考虑了道路交通状况、生鲜产品在途损耗两大分别影响生鲜配送效率和客户满意度的因素，还创新引入了一种“后进先出”的交付策略，以减少翻箱率、节省交付时间、降低人工操作所造成的品质损耗。

（2）结合上述模型特征，设计了基于变邻域搜索和自适应阈值接受算法的两阶段优化算法。其中，阶段一旨在实现车辆间订单集任务配置及车辆行驶路径的初步规划。此外，在该部分的算法设计中，为避免算法求解时陷入局部最优的困境，本研究还设计了一种可以增加解空间多样性的扰动算子。阶段二则是在阶段一的基础上，展开对车辆行驶路径的深入规划，以进一步提升最优解的表现情况。

（3）使用Java语言实现两阶段优化算法的求解运行，并带入Solomon算例进行算法测试及实验探究。多个算例的求解实验证明了两阶段优化算法的收敛性、稳健性及有效性；进一步，通过对比10次实验下，两阶段优化算法的求解结果和算法阶段一的求解结果，验证了算法阶段二设计的必要性及两阶段优化算法设计的合理性。此外，基于对网格仓中生鲜配送场景的实验探索，包括时变路网条件探索、冷藏车使用数量探索，以及单位品质正效益灵敏度分析，本研究归纳梳理了相应的管理学启示如下：

①道路交通状况对生鲜配送规划结果的主要影响体现在订单任务执行顺序上，整体不太通畅的道路交通状况会一定程度地驱使车辆采取一种“少取多送”的行为。“少取多送”行为表现为车辆一次仅完成少量订单的取货，待这些订单实现交付后又折回仓库继续执行其他订单的取货，虽然这种行为会增加局部运输成本，但能保障部分订单的交付及时性，同时可以收获更大的客户满意度。

②运力资源充足时，适当增加配送车辆的使用数量有助于降低总运营变动成本，并且整体上存在一个车辆的最佳使用数量，使得总运营变动成本达到最小。配送车辆使用数量的整体增加一方面带来了生鲜产品品质和品质效益的提升，另一方面也导致了运输成本的增加。因此，就整体而言，存在一个车辆使用数量的增加上界，在该界限左侧，产品品质和品质效益的提升大于运输成本的增加，总运营变动成本表现为减少；在该界限右侧，产品品质和品质效益的提升小于运输成本的增加，总运营变动成本表现为增多；在该界限处，总运营变动成本达到最小，这一界限即为车辆的最佳使用数量。

③在一定范围内，单位品质正效益的增加对生鲜产品品质的提升有较为显著的促进作用，但该促进作用的效果有限，若想获得生鲜产品品质的持续提升，应考虑改变其他的外部条件。单位品质正效益较小时，由于其在全局规划中的影响力较小，生鲜产品品质不会随之增加而产生显著变化；而单位品质正效益较大时，虽然其在全局规划中的影响力变大，但由于受到其他条件或约束的限制，生鲜产品品质很难获得进一步提升。因此，单位品质正效益的增加仅在一定范围内对生鲜产品品质的提升有促进作用，在单位品质正效益高于该范围的情况下想要获得生鲜产品品质的进一步提升，应考虑改变其他的外部制约条件或影响因素。

本研究的主要贡献在于，首先，为网格仓突破生鲜配送的发展障碍、获得持续盈利提供了一套分析和求解的方案体系，即上述的考虑网格仓间协同的生鲜配送优化模型、基于变邻域搜索和自适应阈值接受算法的两阶段优化算法。并且，研究通过实验探索得到的管理学启示值得类似机制学习和借鉴，并可能进一步运用至实际。其次，丰富了生鲜配送领域的理论及方法，为类似问题的研究提供了参考思路。

## 未来展望

此外，本研究还存在着以下局限性：①未能充分考虑冷藏车从网格仓出发的时间（本研究假设冷藏车都是在一个固定的时间点驶离网格仓）；②所考虑的道路交通状况过于单一，仅考虑了早晚出行高峰对道路状况的影响。

本研究未来将在以下方向上进一步尝试探索：

第一，展开对冷藏车发车时间的规划研究。第二，在考虑道路交通状况的基础上，展开应对交通意外的研究。该研究的现实意义在于，某些交通状况是难以预测的，因此，需要有一个能根据突发交通意外及时调整配送计划的反馈机制，以保证因意外造成的损失尽可能地小。第三，展开对动态订单分配策略及路径规划的研究。本研究是基于已知的静态订单池而展开的，而实际中，一些生鲜配送问题面对的是无法预知的动态订单需求，需要做到及时响应这些订单需求的同时，合理规划订单响应计划以降低总成本。

# 致谢

时光匆匆，转眼间，即将迎来本科四年生活的终止符，纵有万般不舍，却也只能无奈道别。四年间，我在这所学术氛围浓厚的校园里成长、收获，结识了许多重要的良师益友，在这里，我向他们表示最崇高的敬意与最诚挚的感谢。

首先，我想感谢的是我的导师王玥老师。本研究是在王玥老师悉心指导下完成，从前期的研究选题、资料搜集到中期的模型构建、算法设计、模拟实验，再到最后的论文撰写及修改，王玥老师对于我的每个问题始终不厌其烦地做出了解答并且提出了许多非常有价值的建议。研究推进过程中，她也时常鼓励我，让我能在繁重的科研工作中找到前进的动力和方向。

其次，我要感谢我的班主任祁超老师，给予了我许多学习和生活上的指导，让我在享受大学生活的同时还能保持一颗学习而又向上的心；感谢李建斌老师和关旭老师不仅让我感受到了学术研究的独特魅力，还一直在学术道路上指引着我勇往向前。此外，也非常感谢给我们上过课的所有老师，在各位老师的课堂上，我收获的不仅是知识更有迸发的灵感和人生的哲学！如果有机会，我还愿意回到那个奔波于各个课堂的时候，再次聆听一场学术和生活的启示录。

我还要感谢我的父母、同学和朋友。感谢我的父母无微不至的关怀，始终无条件地相信、支持着我，尤其在我遭遇困难时，总能耐心倾听我的诉说、努力帮助我解决困难；感谢管实1801班的所有同学和我的朋友们，非常幸运能和你们一起学习、一起成长，让我圆满度过了最美丽的大学时光。

回顾四年华科校园生活的点点滴滴，我体会过考试前夕在教学楼学到废寝忘食的专注与努力，也享受过考试结束后在宿舍一觉睡到自然醒的安逸与闲适。在这里，我怀揣着对未来的美好憧憬大步向前走着，虽然也曾有过迷茫和困惑，但始终保持着一种热情和期待；在这里，我总能感受到一种生生不息的活力和处惊不乱的从容；在这里，我想做的事情还有很多很多，总以为一切都来日方长，却未曾想过离别竟先一步来临。或许，时间就是从我迷迷糊糊拿着导航去往目的地到现在不知不觉也能走到目的地中匆匆流逝的吧！

我会永远记得在华科度过的最美好的四年时光。青山不改，绿水长流，期待在华科校园里的再度重逢。

# 参考文献

[1] Osvald A , Stirn L Z . A vehicle routing algorithm for the distribution of fresh vegetables and similar perishable food[J]. Journal of Food Engineering, 2008, 85(2):285-295.

[2] Hsiao Y H , Chen M C , Lu K Y , et al. Last-mile distribution planning for fruit-and-vegetable cold chains[J]. The International Journal of Logistics Management, 2018, 29(3):862-886.

[3] Wang X , Wang M , Ruan J , et al. The Multi-objective Optimization for Perishable Food Distribution Route Considering Temporal-spatial Distance[J]. Procedia Computer Science, 2016, 96:1211-1220.

[4] Chaudhuri A , Dukovska-Popovska I , Subramanian N , et al. Decision making in cold chain logistics using data analytics: a literature review[J]. The International Journal of Logistics Management, 2018, 29(3):839-861.

[5] Saif A , Elhedhli S . Cold supply chain design with environmental considerations: A simulation-optimization approach[J]. European Journal of Operational Research, 2016:274-287.

[6] Hsu C I , Hung S F , Li H C . Vehicle routing problem with time-windows for perishable food delivery[J]. Journal of Food Engineering, 2007, 80(2):465-475.

[7] 刘思远,陈天恩,陈栋,张驰,王聪.时变多车型下的生鲜农产品配送路径优化模型[J/OL].智慧农业(中英文):1-13[2021-11-28].

[8] 马艳芳,李保玉,杨屹夫,冯翠英.客户分类下生鲜配送两级路径问题与算法研究[J].计算机工程与应用,2021,57(20):287-298.

[9] 夏扬坤,邓永东,庞燕,王忠伟,高亮.带客户分级和需求可拆分的生鲜车辆路径问题[J].计算机集成制造系统,2021,27(04):1238-1248.

[10] 陆淼嘉,黄承媛,滕靖.基于多智能体的网购生鲜无人车配送调度仿真[J/OL].系统仿真学报:1-13[2021-11-28].

[11] 鲍媛媛. 新零售生鲜电商产品配送路径优化研究[D].江西财经大学,2021.

[12] 余海燕,唐婉倩,吴腾宇.带硬时间窗的O2O生鲜外卖即时配送路径优化[J].系统管理学报,2021,30(03):584-591.

[13] 许婷婷, 刘波, 孙茹,等. 生鲜农产品冷链物流路径优化[J]. 中国物流与采购, 2021(1):2.

[14] 余建军,程文琪,吴永忠.考虑顾客满意度的生鲜外卖路径规划[J].工业工程与管理,2021,26(04):158-167.

[15] 赵志学, 李夏苗. 时变交通下生鲜配送电动车辆路径优化方法[J]. 交通运输系统工程与信息, 2020, 20(5):9.

[16] 赵昊. O2O模式下多配送中心冷链运输路径优化研究[D]. 大连海事大学.

[17] 陈久梅,周楠,王勇.生鲜农产品多隔室冷链配送车辆路径优化[J].系统工程,2018,36(08):106-113.

[18] 张倩, 熊英, 何明珂,等. 不确定需求生鲜电商配送路径规划多目标模型[J]. 系统仿真学报, 2019, 31(8):9.

[19] Song B D , Ko Y D . A vehicle routing problem of both refrigerated- and general-type vehicles for perishable food products delivery[J]. Journal of Food Engineering, 2016, 169(JAN.):61-71.

[20] Ma Z J , Yao W , D Ying. A combined order selection and time-dependent vehicle routing problem with time widows for perishable product delivery[J]. Computers & Industrial Engineering, 2017, 114(DEC.):101-113.

[21] Wu Y , Zheng B , Zhou X . A Disruption Recovery Model for Time-Dependent Vehicle Routing Problem With Time Windows in Delivering Perishable Goods[J]. IEEE Access, 2020, 8:189614-189631.

[22] Amorim P , Almada-Lobo B . The impact of food perishability issues in the vehicle routing problem[J]. Computers & Industrial Engineering, 2014, 67(jan.):223-233.

[23] Hu H , Zhang Y , Zhen L . A two-stage decomposition method on fresh product distribution problem[J]. International Journal of Production Research, 2017:1-24.

[24] Chan F T S , Shekhar P , Tiwari M K . Dynamic scheduling of oil tankers with splitting of cargo at pickup and delivery locations: a Multi-objective Ant Colony-based approach[J]. International Journal of Production Research, 2014, 52(24):7436-7453.

[25] Naso D , Surico M , Turchiano B . Scheduling Production and Distribution of Rapidly Perishable Materials with Hybrid GA's[M]. Springer Berlin Heidelberg, 2007.

[26] Gribkovskaia I , Shlopak G L . A tabu search heuristic for a routing problem arising in servicing of offshore oil and gas platforms[J]. Journal of the Operational Research Society, 2008, 59(11):1449-1459.

[27] Hipólito Hernández-Pérez, Juan-José Salazar-González. A branch-and-cut algorithm for a traveling salesman problem with pickup and delivery[J]. Discrete Applied Mathematics, 2004, 145(1):126-139.

[28] 方文婷, 艾时钟, 王晴,等. 基于混合蚁群算法的冷链物流配送路径优化研究[J]. 中国管理科学, 2019, 27(11):9.

[29] 侯宇超,白艳萍,胡红萍,王鹏.基于精英蚁群算法的多目标生鲜配送路径优化研究[J].数学的实践与认识,2018,48(20):50-57.

[30] Tarantilis C, Kiranoudis C. A meta-heuristic algorithm for the efficient distribution of perishable foods. Journal of Food Engineering, 2001,50(1): 1–9.

[31] Tarantilis, C.D., Kiranoudis, C.T., 2002. Distribution of fresh meat. J. Food Eng. 51 (1),85-91.

[32] Sahin M , Cavuslar G , Oencan T , et al. An efficient heuristic for the Multi-vehicle One-to-one Pickup and Delivery Problem with Split Loads[J]. Transportation Research Part C, 2013, 27(Feb.):169-188.

[33] Zhang G , Habenicht W , Spiess W E L . Improving the structure of deep frozen and chilled food chain with tabu search procedure[J]. Journal of Food Engineering, 2003, 60(1):67-79.

[34] Zhou X , Lai X , Lim A , et al. An improved approximation algorithm for the capacitated TSP with pickup and delivery on a tree[J]. Networks, 2014, 63(2):179-195.

[35] Ropke S , Pisinger D . An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows[J]. Transportation ence, 2006, 40.

[36] Privé J, Renaud J, Boctor F, Laporte G (2006) Solving a vehicle-routing problem arising in soft-drink distribution. J Oper Res Soc 57:1045–1052.

[37] Cherkesly M , Desaulniers G , Laporte G . Branch-Price-and-Cut Algorithms for the Pickup and Delivery Problem with Time Windows and Last-in-First-Out Loading[J]. Transportation Science, 2015, 49(4):752-766.

[38] Bartolini E , Bodin L , Mingozzi A . The traveling salesman problem with pickup, delivery, and ride‐time constraints[J]. Networks, 2016, 67(2):95-110.

[39] Alyasiry A M , Forbes M , Bulmer M . An Exact Algorithm for the Pickup and Delivery Problem with Time Windows and Last-in-First-out Loading[J]. Transportation Science, 2019, 53(6):1695-1705.

[40] Marilène Cherkesly a d, D G D A , D G L B C . A population-based metaheuristic for the pickup and delivery problem with time windows and LIFO loading[J]. Computers & Operations Research, 2015, 62(C):23-35.

[41] Gao X , Lim A , Qin H , et al. Multiple Pickup and Delivery TSP with LIFO and Distance Constraints: A VNS Approach[C]// International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems. Springer, Berlin, Heidelberg, 2011.

[42] Shukla M , Jharkharia S . Artificial Immune System-based algorithm for vehicle routing problem with time window constraint for the delivery of agri-fresh produce[J]. Journal of Decision Systems, 2013, 22(3):224-247.

# 附录

以下附上部分关键代码：

//inter-exchange-couple

public List<List> exchangeCouple(Vehicle v1,int[][]stage1,int index1,Vehicle v2,int[][]stage2,int index2,int whichCase){

List<List> exchangeLater=new ArrayList<List>();

List<Order> route1=new ArrayList<Order>();

List<Order> route2=new ArrayList<Order>();

for(int i=0;i<v1.route.size();i++){

route1.add(v1.route.get(i));

}

for(int i=0;i<v2.route.size();i++){

route2.add(v2.route.get(i));

}

if(index1<=v1.route.size()&&index2<=v2.route.size()){

//System.out.println("v1.size: "+v1.route.size()+"index1: "+index1+"v2.size: "+v2.route.size()+"index2: "+index2);

int sign\_p1=0;int sign\_d1=0;

int sign\_p2=0;int sign\_d2=0;

Order pickup1=new Order();Order delivery1=new Order();

Order pickup2=new Order();Order delivery2=new Order();

if(findCoupleInRoute(v1,index1)<index1) {

pickup1=v1.route.get(findCoupleInRoute(v1,index1)-1);

delivery1=v1.route.get(index1-1);

sign\_p1=findCoupleInRoute(v1,index1);

sign\_d1=index1;

}

else{

pickup1=v1.route.get(index1-1);

delivery1=v1.route.get(findCoupleInRoute(v1,index1)-1);

sign\_p1=index1;

sign\_d1=findCoupleInRoute(v1,index1);

}

if(findCoupleInRoute(v2,index2)<index2) {

pickup2=v2.route.get(findCoupleInRoute(v2,index2)-1);

delivery2=v2.route.get(index2-1);

sign\_p2=findCoupleInRoute(v2,index2);

sign\_d2=index2;

}

else{

pickup2=v2.route.get(index2-1);

delivery2=v2.route.get(findCoupleInRoute(v2,index2)-1);

sign\_p2=index2;

sign\_d2=findCoupleInRoute(v2,index2);

}

int [][]cap=new int[2][];

cap[0]=stage1[2];cap[1]=stage2[2];

switch(whichCase){

case 1:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,2)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_p1-1, pickup2);

route1.add(sign\_p1, delivery2);

route2.add(sign\_p2-1, pickup1);

route2.add(sign\_d2-1,delivery1);

}

break;

case 2:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,3)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_d1-2, pickup2);

route1.add(sign\_d1-1, delivery2);

route2.add(sign\_p2-1, pickup1);

route2.add(sign\_d2-1,delivery1);

}

break;

case 3:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,1)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_p1-1, pickup2);

route1.add(sign\_d1-1,delivery2);

route2.add(sign\_p2-1, pickup1);

route2.add(sign\_d2-1,delivery1);

}

break;

case 4:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,4)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_p1-1, pickup2);

route1.add(sign\_p1,delivery2);

route2.add(sign\_p2-1, pickup1);

route2.add(sign\_p2,delivery1);

}

break;

case 5:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,7)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_d1-2, pickup2);

route1.add(sign\_d1-1,delivery2);

route2.add(sign\_p2-1, pickup1);

route2.add(sign\_p2,delivery1);

}

break;

case 6:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,8)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_p1-1, pickup2);

route1.add(sign\_d1-1,delivery2);

route2.add(sign\_p2-1, pickup1);

route2.add(sign\_p2,delivery1);

}

break;

case 7:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,5)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_p1-1, pickup2);

route1.add(sign\_p1,delivery2);

route2.add(sign\_d2-2, pickup1);

route2.add(sign\_d2-1,delivery1);

}

break;

case 8:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,6)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_d1-2, pickup2);

route1.add(sign\_d1-1,delivery2);

route2.add(sign\_d2-2, pickup1);

route2.add(sign\_d2-1,delivery1);

}

break;

case 9:

if(v1.capacityAfterExchangeCouple(route1,route2,cap,sign\_p1,sign\_d1,sign\_p2,sign\_d2,9)){

route1.remove(sign\_p1-1);

route1.remove(sign\_d1-2);

route2.remove(sign\_p2-1);

route2.remove(sign\_d2-2);

route1.add(sign\_p1-1, pickup2);

route1.add(sign\_d1-1,delivery2);

route2.add(sign\_d2-2, pickup1);

route2.add(sign\_d2-1,delivery1);

}

break;

default:

break;

}

}

exchangeLater.add(route1);

exchangeLater.add(route2);

return exchangeLater;

}



**本科毕业设计（论文）任务书**

题 目 考虑网格仓间协同的生鲜配送车辆路径优化研究

（任务起止日期：2021年11月 2 日～2022年6月5日）

院 系 管理学院

专业班级 管实1801

姓 名 戴蔓菲

学 号 U201816168

指导教师 王 玥

教研室（系、所）负责人 2021年 10月28日审查

院（系）负责人 2021年11月2日批准

|  |
| --- |
| 课题内容：  “中心仓——网格仓——消费者自提点”的三级生鲜配送网络是伴随美团优选、十荟团等社区团购平台的兴起而逐渐发展起来的。网格仓是三级生鲜配送网络中的关键节点，由网格仓发出的生鲜产品运输活动也是该配送网络中的关键链。通过分析网格仓中的生鲜配送现状，本研究总结出其目前面临着①网格仓之间缺乏协调、②生鲜产品的消费具有较大随机性、③消费者自提点的不确定性较多、④生鲜配送成本过高、⑤生鲜产品品质难以保障的五大发展障碍。并且，进一步研究发现这五大发展障碍都指向了一个本质问题，即“彼此之间缺乏联系的网格仓难以从一个整体视角实现运力资源的合理调度，生鲜配送路径的优化也迫在眉睫”。为此，本研究提出了网格仓间协同开展生鲜配送服务的思想，并对其中涉及的配送车辆与订单任务的配置问题、配送车辆行驶路径的规划问题展开了探索。研究的主要内容包括：系统分析国内外相关领域内的研究现状、构建考虑网格仓间协同的生鲜配送优化模型、设计收敛迅速，鲁棒性强的算法并实现编程运行输出解决方案、总结研究结论和启示。  本课题的研究内容旨在优化网格仓生鲜配送经营现状、减少生鲜配送总成本、提高客户满意度的同时，促进生鲜物流配送的健康发展。 |
| 课题任务要求：  1、对生鲜配送领域相关背景进行深入分析，总结归纳生鲜配送问题的国内外研究现状  2、完整描述考虑网格仓间协同的生鲜配送问题，并提出数学模型  3、选择合适的算法并编程实现数学模型的求解  4、使用算例对算法效率进行验证，并根据实验结果得到生鲜配送方面的管理学启示 |
| 主要参考文献（由指导教师选定）：  [1] 盛虎宜. 生鲜电商物流配送的车辆路径问题研究[D]. 电子科技大学, 2019.  [2] Ao A , Lzs B . A vehicle routing algorithm for the distribution of fresh vegetables and similar perishable food[J]. Journal of Food Engineering, 2008, 85( 2):285-295.  [3] Saif A , Elhedhli S . Cold supply chain design with environmental considerations: A simulation-optimization approach[J]. European Journal of Operational Research, 2016:274-287.  [4] Song B D , Ko Y D . A vehicle routing problem of both refrigerated- and general-type vehicles for perishable food products delivery[J]. Journal of Food Engineering, 2016, 169(JAN.):61-71.  [5] Hsu C I , Hung S F , Li H C . Vehicle routing problem with time-windows for perishable food delivery[J]. Journal of Food Engineering, 2007, 80(2):465-475. |
| 同组设计者：  无 |
| 指导教师签名：  年 月 日 |