

武汉理工大学

(申请工学硕士学位论文)

快速消费品制造企业 生产-配送集成调度研究

培养单位：机电工程学院

学科专业：工业工程

研究生：黄铭

指导教师：杜百岗 副教授

2023 年 5 月

分类号_____

密 级_____公开_____

UDC _____

学校代码_____10497_____

武汉理工大学

学 位 论 文

题 目_____快速消费品制造企业生产-配送集成调度研究_____

英 文 Research on integrated production and distribution scheduling

题 目_____for fast moving consumer goods manufacturing enterprise_____

研究生姓名_____黄 铭_____

指导老师 姓名_____杜百岗_____职称_____副教授_____学位_____博士_____

单位名称_____机电工程学院_____邮编_____430070_____

副指导老师 姓名_____职称_____学位_____

单位名称_____邮编_____

申请学位级别_____硕士_____学科专业名称_____工业工程_____

论文提交日期_____2023 年 3 月_____论文答辩日期_____2023 年 5 月_____

学位授予单位_____武汉理工大学_____学位授予日期_____2023 年 6 月_____

答辩委员会主席_____李益兵_____评阅人_____

2023 年 5 月

独创性声明

本人声明，所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得武汉理工大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签名： 黄铭 日期： 2023.5.20

学位论文使用授权书

本人完全了解武汉理工大学有关保留、使用学位论文的规定，即学校有权保留并向国家有关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。本人授权武汉理工大学可以将本学位论文的全部内容编入有关数据库进行检索，可以采用影印、缩印或其他复制手段保存或汇编本学位论文。同时授权经武汉理工大学认可的国家有关机构或论文数据库使用或收录本学位论文，并向社会公众提供信息服务。

(保密的论文在解密后应遵守此规定)

研究生（签名）： 黄铭 导师（签名）： 杨明 日期 2023.5.20

中文摘要

快速消费品制造业是生产包括饮料、包装食品及日用品在内的具有消费频率高、使用时限短等特征的制造产业总称。快速消费品制造企业大多具有流水线生产、基于托盘的仓储库存、以及自有车辆配送等特点。由于产品提前生产会产生库存积压，并且频繁的车辆配送也会带来较高的配送成本，导致生产和配送分别决策时，企业难以保证生产和配送的有效协调。因此，为了快速响应顾客需求、降低供应链的总成本，制定可以高效利用企业资源的生产-配送集成决策显得尤为重要。为此，本文通过结合快速消费品制造企业的实际特征：多产品分批生产、订单的软时间窗限制、数量受限的多行程异构配送车辆、以及不确定旅途时间等，开展了针对快速消费品制造企业生产-配送集成调度问题的研究，主要工作如下：

(1) 分析了快速消费品制造企业生产-配送集成调度流程，并结合企业实际特征，设计了面向快速消费品制造企业的生产-配送集成调度优化框架，该优化框架包含生产-配送协同机制设计以及车辆选择方案的迭代寻优。

(2) 针对考虑分批生产、软时间窗限制及数量受限的多行程异构车辆的快速消费品制造企业生产-配送集成调度问题，构建了以最大化总利润和最小化提前延期加权惩罚时间为双优化目标的混合整数非线性规划模型。首先，为了实现生产-配送协同机制，设计了巡回环划分算法、理想最佳出发时间的确定以及基于无等待调度的双向调度策略。然后，提出了基于问题特征和编码形式的自适应大邻域搜索策略改进的多目标进化算法求解该问题。最后，通过数值实验验证了所提算法的有效性。

(3) 针对考虑模糊旅途时间扰动的快速消费品制造企业生产-配送集成调度问题，构建了以生产-库存-配送总成本和提前延期加权惩罚时间为双优化目标的混合整数非线性规划模型。首先，定义了一种模糊加权自叠加理论进行模糊加权惩罚时间的计算。然后，提出了知识驱动协同进化算法解决该问题，其融合了基于不同更新策略的双子种群协同进化和知识驱动策略。最后，通过 CPLEX 求解器及数值实验验证了模型与算法的有效性，并针对车辆数量进行了敏感性分析，为企业决策者提供管理见解。

关键词：快速消费品制造企业；生产-配送集成调度；分批生产；软时间窗；模糊旅途时间

Abstract

Fast moving consumer goods (FMCG) manufacturing industry is the general term for manufacturing industries that produce beverages, packaged foods and daily necessities with high consumption frequency and short usage time frame. In this industry, most enterprises have the characteristics of pipeline production, pallet-based storage inventory, and own vehicles for distribution. Since the premature manufacturing of products will cause inventory backlog and the frequent vehicle delivery will incur high distribution cost, it is difficult for enterprises to ensure the effective coordination of production and distribution when production and distribution are decided separately. Therefore, in order to respond quickly to customer demand and reduce the total cost of the supply chain, it is particularly important to make integrated production-distribution decisions that can efficiently deploy enterprise resources. For this purpose, this paper conducts the research on the integrated production-distribution scheduling problem for the FMCG manufacturing enterprise by considering the real characteristics of the enterprise: multi-product batches production, the soft time window of orders, a limited number of multi-trip heterogeneous vehicles, and the uncertain travel time. The main work is as follows:

(1) The integrated production-distribution scheduling process of the FMCG manufacturing enterprise is analyzed. And an integrated production-distribution scheduling optimization framework for the FMCG manufacturing enterprise is designed by combining the real characteristics of the enterprise. The optimization framework includes the design of the production-distribution collaborative mechanism and the iterative optimization of the vehicle selection scheme.

(2) The integrated production-distribution scheduling problem of the FMCG manufacturing enterprise with batch production, soft time windows and a limited number of multi-trip heterogeneous vehicles is considered. A mixed integer nonlinear programming model with dual optimization objectives of maximizing total profit and minimizing the total early and tardy weighted penalty time is formulated. First, in order to realize the production-distribution collaborative mechanism, the tour division algorithm, the determination of the ideal optimal departure time of the tour, and the

bidirectional scheduling strategy based on no-wait schedule are designed. Then, a multi-objective evolutionary algorithm improved by adaptive large neighborhood search strategy based on the specific problem and coding form is proposed to solve this problem. Finally, the effectiveness of the algorithm is verified by the computational experiments.

(3) The integrated production-distribution scheduling problem of the FMCG manufacturing enterprise with fuzzy travel times is considered. A bi-objective mixed integer nonlinear programming model is established, which takes total production-inventory-distribution cost and total early and tardy weighted penalty time as optimization objectives. First, a fuzzy weighted self-superposition is defined for the calculation of fuzzy weighted penalty time. Then, a knowledge-driven co-evolutionary algorithm is proposed to solve this problem, which fuses a dual subpopulation co-evolution based on different update strategies and a knowledge-driven strategy. Finally, the correctness of the model and the effectiveness of the algorithm are verified by the CPLEX solver and computational experiments. And a sensitivity analysis is performed for the number of vehicles to provide management insights for enterprise decision makers.

Key words: Fast moving consumer goods manufacturing enterprises; Integrated production and distribution scheduling; Batch production; Soft time windows; Fuzzy travel times

目录

中文摘要	I
Abstract.....	II
第 1 章 绪论	1
1.1 研究背景及意义	1
1.2 国内外研究现状	2
1.2.1 确定环境下生产-配送集成调度研究现状	2
1.2.2 不确定扰动影响下生产-配送集成调度研究现状	4
1.3 课题来源及研究内容	5
1.3.1 课题来源	5
1.3.2 研究内容与组织架构	6
第 2 章 快速消费品制造企业生产-配送集成调度分析	8
2.1 生产-配送集成调度流程分析	8
2.2 生产-配送集成调度特征识别	9
2.2.1 流水线分批生产调度特征	9
2.2.2 车辆配送调度特征	9
2.2.3 不确定旅途时间扰动特征	10
2.3 生产-配送集成调度优化模式	11
2.3.1 生产-配送集成调度优化框架	11
2.3.2 生产-配送集成调度优化策略	12
2.4 本章小结	13
第 3 章 考虑分批生产及软时间窗限制的生产-配送集成调度	14
3.1 生产-配送集成调度模型	14
3.1.1 问题描述	14
3.1.2 符号定义及说明	16
3.1.3 混合整数非线性规划模型	17
3.2 基于自适应大邻域搜索策略改进的多目标进化算法（IMOEa）	19
3.3 IMOEa 中的协同机制	21
3.3.1 巡回环划分算法	21
3.3.2 基于 DPT 函数的理想最佳出发时间确定	22
3.3.3 基于无等待调度的双向调度策略的同时协调（BSS-NWS）	25

3.4 IMOEA 中的自适应大邻域搜索策略	28
3.4.1 自适应大邻域搜索策略的基本组成	28
3.4.2 基于问题特征的邻域结构设计	29
3.4.3 基于编码形式的邻域结构设计	31
3.5 数值实验与结果分析	32
3.5.1 测试实例的生成	32
3.5.2 对比算法的选取及多目标评价指标	34
3.5.3 算法参数设计	35
3.5.4 对比实验结果与分析	36
3.6 本章小结	40
第 4 章 考虑模糊旅途时间扰动的生产-配送集成调度	41
4.1 生产-配送集成调度模型	41
4.1.1 问题描述	41
4.1.2 符号定义及说明	42
4.1.3 混合整数非线性规划模型	43
4.2 模糊运算的处理	44
4.2.1 基础模糊运算	44
4.2.2 模糊加权自叠加理论及证明	45
4.3 知识驱动协同进化算法	47
4.3.1 基于多策略融合的算法框架	47
4.3.2 基于模糊旅途时间扰动的编码与解码设计	49
4.3.3 基础进化模式设计	51
4.3.4 基于不同更新策略的双子种群协同进化	52
4.3.5 知识驱动策略	53
4.4 数值实验与结果分析	56
4.4.1 测试实例的生成	56
4.4.2 对比算法选取及多目标评价指标	56
4.4.3 算法参数设计	56
4.4.4 MINLP 模型验证	57
4.4.5 对比实验结果与分析	58
4.4.6 敏感性分析	60
4.5 本章小结	61

第 5 章 总结与展望.....	63
5.1 本文总结	63
5.2 研究展望	63
致谢	65
参考文献	66
附录 A：攻读硕士学位期间发表的学术论文	71
附录 B：攻读硕士学位期间获得的科研成果.....	72

第1章 绪论

1.1 研究背景及意义

快速消费品制造企业的运营过程包括产品的生产制造、库存仓储以及配送交付，其中库存作为中间缓冲区将产品的生产和配送阶段相连接。此外快速消费品制造企业的多阶段优化工作相对复杂，生产与配送的管理仍面临着诸多的挑战^[1]。具体表现为生产计划与配送计划的制定更加依赖工作人员的经验，难以保证计划的合理性和协调性。往往使得生产计划具有盲目性、库存占用率偏高、配送车辆路径规划不合理以及资源利用率低等实际问题。因此，如何更好地建立一个合理、协调的生产-配送体系，成为了快速消费品制造企业面临的棘手问题。

传统的快速消费品制造企业的生产和配送往往是单独进行决策的。为保证客户的及时交货要求，往往通过产品的提前生产或者频繁的车辆配送的方式进行满足。然而，这两种方式将会导致产品库存时间过长或配送成本过高的问题，从而难以保证生产和配送的有效协调。与单独决策相比，集成决策具有降低整体成本和高效利用资源的优势^[2]。即集成决策可以权衡生产、库存和配送成本，更加注重供应链的整体效益。并且可以通过合理的设定生产和配送调度计划降低产品库存，从而实现企业资源的高效运作。此外，集成决策不仅可以消除生产中的盲目性以节省成本、提高效率^[3]，而且可以快速高效地响应客户需求，使企业能够为客户提供更灵活、更可靠的服务。因此，针对快速消费品制造企业进行高效的`生产-配送集成调度`（`Integrated production and distribution scheduling, IPDS`）是极为重要的。

快速消费品行业是生产包括饮料、包装食品及日用品在内的具有消费频率高、使用时限短等特征的制造产业总称。该行业是我国市场化程度高，竞争激烈的行业之一^[4]。首先，在实际场景下快速消费品往往采用高度集成化的节拍流水线进行生产。并且由于订单需求的实时波动，采用大量生产模式并不是一个明智的决定^[5]，因为它往往意味着大规模的库存。为了减少库存积压实现生产和配送的相互协调，以订单为导向的分批生产是许多生产和配送一体化企业的普遍做法。即将订单分组形成批次，然后针对批次进行生产调度。其次，随着商业竞争环境的日益激烈，更好的满足客户需求成为了企业所有业务的第一驱动力^[6]。其中订单的及时交付是衡量客户满意度的关键因素。由于订单的提前交付会导致

客户处的库存积压,而延迟交付则会产生缺货风险。所以订单被期望在规定的窗口内进行交付。然而,在实际场景中由于企业生产能力的限制和配送车辆数量和容量的限制,很难严格的要求所有订单都在时间窗口内完成交付,进而考虑允许提前和延迟交付但均会招致惩罚的软时间窗是合理的。最后,由于订单的配送受到道路交通状况、天气环境等诸多不确定因素的影响,所以实际的旅途时间往往在一定时间范围内波动^[7]。并且不确定的旅途时间会扰乱确定性的集成计划,甚至使其不可行^[8]。因此,在考虑分批生产、订单的交付时间限制、以及不确定扰动影响的现实背景下,为快速消费品制造企业设计一个协调、高效的生产-配送集成调度方案具有重要的意义。

1.2 国内外研究现状

IPDS 问题是供应链管理中的一个重要领域,受到学者们的持续关注。Potts^[9]的研究被大多数学者认为是该领域的开篇之作。Chen^[10]则分别从问题模型和方法论层面对 IPDS 问题进行了全面的回顾,使得该领域的研究逐渐系统化。近十年来,将配送阶段建模为车辆路径问题的 IPDS 问题得到了学者们越来越多的关注,也有部分学者将其称为集成生产调度和车辆路径问题(Integrated production scheduling and vehicle routing problem, IPSVRP)。本文所研究的问题归属于 IPSVRP (IPDS 的子领域)的研究范畴。Moons 等^[11]从生产模式、库存、车辆路由等方面对 IPSVRP 进行了详尽的综述。在本节中,将重点讨论典型的开创性文献和最新、最具代表性的 IPSVRP 文献,进而明确本研究的价值所在。

1.2.1 确定环境下生产-配送集成调度研究现状

IPSVRP 的生产模式可以分成两种类型,一种是按作业(项目)生产模式^[12-14](Manufacturing by job (item)),即将一个订单视为一个作业,生产阶段以作业为最小单元进行生产调度。另一种是分批生产模式^[7, 15-17](Manufacturing by batch),即将订单分组形成批次,然后以批次为最小单元进行生产调度。在执行分批生产的 IPSVRP 文献中,Gao 等^[15]针对时间敏感型药品以最小化的调度完成时间为优化目标执行分批生产,并在批次生产完成后立即交付,但并未考虑药品的库存以及交付时间的限制;Zou 等^[16]研究单一产品的分批生产,并采用数量受限的单行程同构车辆执行配送任务。提出了前向批次调度和后向批次调度两种方式来协调批次的生产和配送车辆的出发。前者是将订单按一定顺序排列后进

行连续生产,然后根据订单生产顺序和车辆容量进行分批配送。后者则是先对订单进行分批处理,然后进行批次连续生产以配合车辆的出发。吴瑶等^[7]研究在一条混流生产线上分批次生产多品种的易腐产品。并以同批次配送的订单应在同一批次生产为批次划分的依据。Guo 等^[17]考虑了多品种产品的分批生产模式,但是他们假定同一车间内批次之间仍然执行连续的生产。

在考虑交付时间限制的 IPSVRP 中,一些学者研究交付截止日期^[3, 18-19] (Delivery due date),即产品的交付仅受到一个最晚交付日期的限制。随着研究的深入,一些学者提出了具有严格的交付上下限的硬时间窗(Hard time window)限制。其中 Vahdani 等^[5]要求订单必须在规定的时间窗内进行交付。Fu 等^[13]要求订单即使提前到达,也要等到时间窗下限才能进行交付。然而,受到企业生产能力的限制,特别是配送阶段存在的诸多不确定性,在现实环境中很难保证所有的订单都能严格的在时间窗内进行交付。为了解决此问题,一些学者开始研究单边硬时间窗(Unilateral hard time window)。例如,订单可以被提前交付但不能延迟交付^[20, 21]。或者订单不能被提前交付但可以延迟交付^[22],即当车辆提前到达时,也需要等待到时间窗下限才能进行交付。在此基础上,Hou 等^[23]进一步将单边硬时间窗拓展为软时间窗(Soft time window)。他们以最小化提前延期加权惩罚时间和为优化目标,其中订单由单行程同构车辆进行配送,并且不考虑产品库存。此外,企业运营中需要兼顾客户满意度和经济收益。所以不仅需要考虑惩罚时间,还需要对供应链上的成本进行优化。于是 Liu 等^[24]研究了具有软时间窗限制的生产-配送集成调度模型,并以最小化配送成本、提前和延迟惩罚成本之和为单优化目标进行模型优化。他们考虑了灵活的车辆出发时间,并提出的一种顺序优化求解方法。即首先得到具有最小 Makespan 的车间调度方案,然后采用一种精确程序来获取该车间调度方案对应的最佳车辆出发时间,从而形成生产-配送集成调度方案。Mohammadi 等^[25]则制定了一个包含生产和配送的总成本、提前延期加权惩罚时间和的双目标优化模型。其中配送车辆只能被使用一次,即单行程车辆(Single-trip vehicles)。然而,在实际的快速消费品制造企业中,往往具有不同类型的配送车辆,并且车辆可以多次使用。即异构车辆(Heterogeneous vehicles)和多行程车辆(Muti-trip vehicles)。

综上所述,在现有的生产-配送集成调度研究中,很少有同时考虑分批生产模式、软时间窗限制以及多行程异构车辆等特征的研究。相关学者如有兴趣,可参考文献[11]和[26]对生产-配送集成调度问题进行全面的审查和回顾。然而,在快速消费品制造企业的运营过程中上述实际特征又难以忽视。因此,针对考虑了

分批生产、软时间窗、多行程异构车辆的快速消费品制造企业进行特征分析，并构建适宜的生产-配送集成调度优化模型是极为必要的。

1.2.2 不确定扰动影响下生产-配送集成调度研究现状

在考虑了不确定扰动影响下的 IPSVRP，大多数研究集中在处理订单需求的不确定性上。Adulyasak 等^[27]通过为每个需求情景分配一个概率从而描述需求的不确定性，并提出了一个 Benders 分解方法来解决所提问题。Hemmati Golsefidi 等^[28]考虑了多种需求的不确定性，包括每个零售商订单的需求量、从零售商处收集的缺陷产品的数量等。并提出了一种基于等价约束的鲁棒优化方法解决上述的不确定问题。Liu 等^[29]针对血液制品需求的不确定，构建了以最小化调度总成本为优化目标的数学模型，提出了一种鲁棒优化方法来处理该不确定需求。刘星等^[30]考虑了生产、加班成本及客户需求的不确定性，并建立了鲁棒优化模型来求解所提问题。Ghasemkhani 等^[31]则采用模糊规划的方式表征订单的不确定需求，构建了基于可信度的模糊机会约束规划模型用于处理模糊参数，并提出自适应差分进化算法解决所提问题。Biuki 等^[32]采用三角模糊数表征不确定的订单需求，并提出一种并行混合遗传和粒子群算法解决该问题。然而，受到动态时变交通路网的影响，车辆的旅途时间往往是不确定的。但是现有文献中涉及配送过程不确定性的研究较少。其中 Wang 等^[22]研究了一个具有单边硬时间窗和不确定旅途时间的生产-配送集成调度问题，其中订单由同构车辆配送。采用弹性 p -Robustness 来处理历史数据有限或不存在的旅途时间的变化，并开发了一种具有高效搜索策略的模因算法解决所提问题。Guo 等^[33]则采用正态概率分布来表征不确定旅途时间，并设计了一个采用两阶段启发式突变调度策略和局部禁忌搜索的改进遗传算法解决所提出的模型。值得注意的是，大部分企业很难获得大量可靠的旅途时间数据来拟合概率分布，获得可靠的概率分布是一项棘手的任务^[34]。另外，基于最坏可能情况的鲁棒优化处理方式往往又过于保守。因此，模糊规划可能是处理旅途时间不确定性的一种实用方法，但是相关的研究很少。

针对 IPSVRP 的求解算法，由于其属于 NP-hard^[11]，所以精确算法很难在合理的时间内找到最优解，（元）启发式算法成为求解该问题的主要方法。其中，一些研究采用基于问题特征的启发式求解方法。Qiu 等^[35]通过结合基于 Dantzig-Wolfe 分解的列生成公式来开发分支定界的变邻域搜索算法进行问题求解。Adulyasak 等^[36]开发了一种基于优化的自适应大邻域搜索策略。在该启发式中，

表示位置和路由决策的决策变量分别由枚举方案和上层搜索算子进行处理，与生产、库存和发货数量相关的连续变量通过解决网络流来优化。Belo-Filho 等^[37]针对易腐产品的集成调度问题，开发了自适应大邻域搜索策略。Absi 等^[38]也开发了一种两阶段迭代启发式算法，其中将问题分解为的批量处理问题和路由问题。后来 Chitsaz 等^[39]提出了类似于 Absi 等的三阶段迭代启发式。将第一个阶段再次进行拆分，即以近似总配送成本决定生产设置计划和确定生产、库存和交货数量。Solyali 等^[40]则设计了一个五阶段启发式算法。分别为建立客户访问序列、生成初始方案、求解各周期方案、修复不可行解决方案以及进一步的改进可行方案。Li 等^[41]也针对该问题进行了分层考虑，他们开发了基于数学规划的三级启发式算法来解决问题。包括获得的良好初始解、修复初始解中的潜在不可行解，并执行路线改进程序、采用迭代修复和优化过程以进一步改进解决方案。此外，由于进化算法在解决复杂的组合优化问题方面的卓越表现，许多研究采用以进化算法为基础的改进算法。例如，遗传算法^[18,28]，改进的遗传算法^[42-43]，模因算法^[19,22]，基于和声搜索改进的模因算法^[17]，基于蜜蜂算法改进的 MOEA/D^[44]；NSGA-II^[20]，基于遗传算法的混合多层次优化框架^[24]，自适应差分进化算法^[31]。受到上述研究启发，基于问题特征设计的启发式策略和进化类算法相结合的算法设计思路具有良好的发展前景。

综上所述，在现有的考虑不确定扰动影响的生产-配送集成调度研究中，很少有研究同时考虑不确定旅途时间和软时间窗限制的生产-配送集成调度，并且很少有使用模糊规划程序去处理旅途时间的不确定性。此外，采用基于问题特征改进的进化类算法的求解模式已被诸多研究验证了合理及优越性。因此，有必要针对快速消费品制造企业进行特征分析，并与进化类算法相结合设计出高效的优化算法以解决所研究的生产-配送集成调度问题。

1.3 课题来源及研究内容

1.3.1 课题来源

本文课题来源于以下基金项目和横向课题：

- (1) 国家自然科学基金项目：面向社会化协同的复杂产品系统制造服务供需配置博弈方法（项目编号：51705386）；
- (2) 横向课题：快速消费品中央调拨管理系统开发；

1.3.2 研究内容与组织架构

本文以快速消费品制造企业生产-配送集成调度为研究对象，通过分析企业的实际流程及特征，提出了快速消费品制造企业生产-配送集成调度优化框架及策略。并且提出了两种生产-配送集成调度模型，基于生产-配送集成调度优化框架及各自模型特点分别设计了算法进行模型求解。本文共分为 5 个章节，其组织架构见图 1-1 所示，具体的研究内容如下：

第 1 章：绪论。首先，阐述了快速消费品制造企业生产-配送集成调度的研究背景与意义。然后，分别从生产模式、交付时间限制、不确定扰动影响、求解算法等方面入手，有针对性的对生产-配送集成调度问题进行文献综述以阐明研究的新颖性。最后，介绍了本文的研究内容与组织架构。

第 2 章：快速消费品制造企业生产-配送集成调度分析。首先分析了所调查企业的生产-配送集成调度流程。然后通过识别企业所具备的实际特征：流水线分批生产调度、车辆配送调度、不确定旅途时间扰动等，提出了生产-配送集成调度优化框架及策略。并且该优化框架及策略为后文的算法设计提供了指导与依据。

第 3 章：考虑分批生产及软时间窗限制的生产-配送集成调度。首先构建了一个考虑了分批生产、软时间窗限制以及数量受限的多行程异构车辆等实际特征的双目标混合整数非线性规划模型。然后为求解该模型，开发了一种基于自适应大邻域搜索策略改进的多目标进化算法，其中包括协同机制的实现以及自适应大邻域搜索策略的设计。最后通过数值实验验证了模型和算法的有效性。

第 4 章：考虑模糊旅途时间扰动的生产-配送集成调度。首先，该章在第 3 章的基础上增加了对旅途时间不确定性的考虑。并通过模糊规划的方式表征不确定的旅途时间。构建了以总成本和提前延期加权惩罚时间作为双优化目标的混合整数非线性规划模型。其次，提出了模糊加权自叠加理论来处理模糊数的内部运算，并论证了该理论的合理性。然后，设计一种知识驱动协同进化算法来解决此问题。该算法融合了基于不同更新策略的双子种群协同进化和从问题特征中提取知识指导进化过程的知识驱动策略。然后，通过 CPLEX 求解器及数值实验验证了模型正确性及算法有效性。最后，通过配送车辆数量的敏感性分析，为快速消费品制造企业的决策者提供管理见解。

第 5 章：总结与展望。梳理并总结了所进行的研究工作，同时也对未来生产-配送集成调度的研究方向进行了展望。

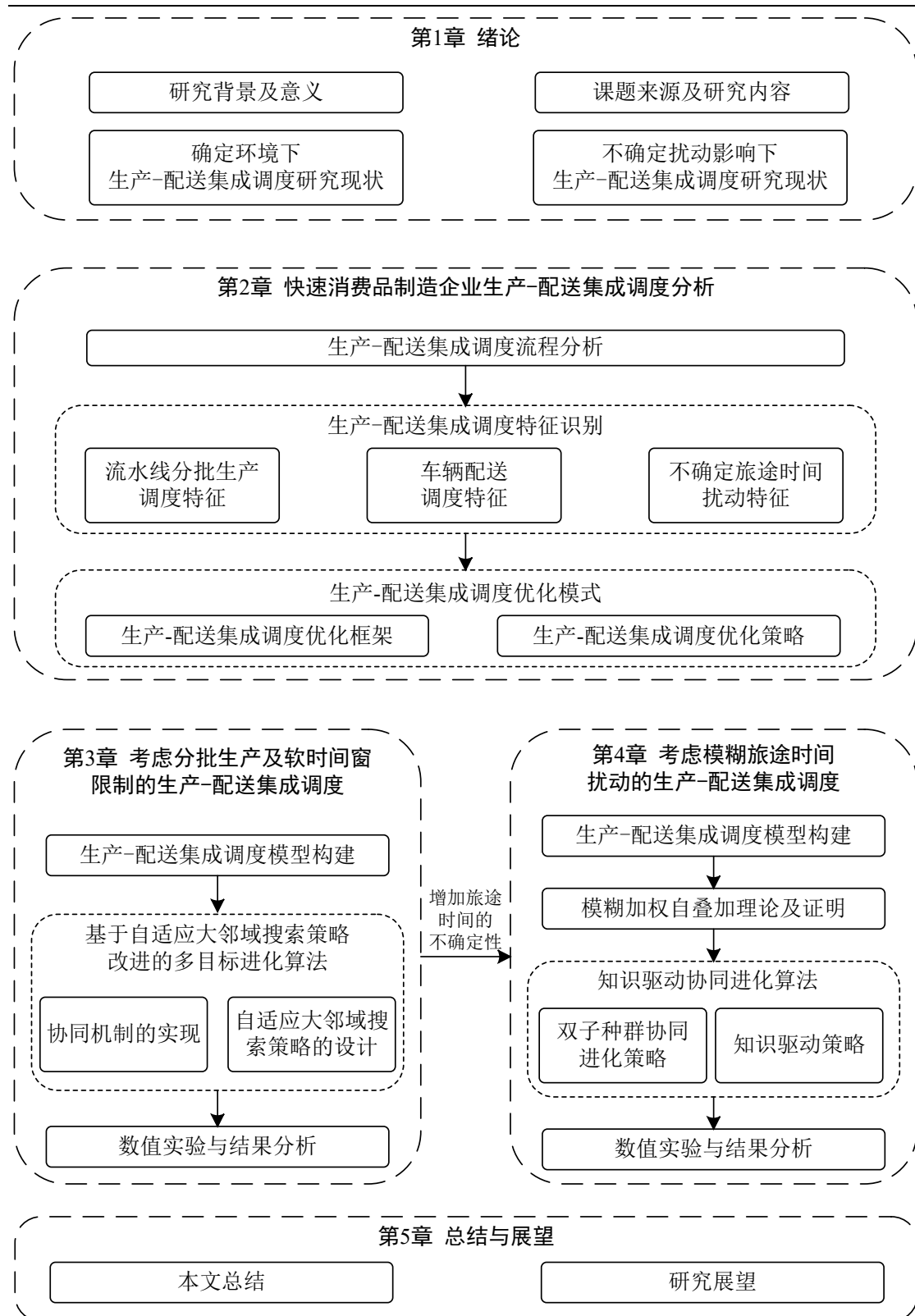


图 1-1 组织架构图

第2章 快速消费品制造企业生产-配送集成调度分析

2.1 生产-配送集成调度流程分析

快速消费品制造企业往往采用高度自动化、集成化的流水线形式进行生产，产品多为体积小，消费频繁的工业产品，例如罐装饮料生产企业提供的碳酸饮料、矿泉水、咖啡、运动饮料、苏打水等，细分种类达数十种。随着客户对产品种类和交付时效性的要求不断增高，一个高效协调的生产-配送集成调度流程至关重要。如图 2-1 所示，展示了所调查的快速消费品制造企业生产-配送集成调度流程。以客户作为生产-配送集成调度流程的起止点。首先，进行客户订单信息的录入工作，即将订单的交付时间窗、产品种类、产品数量与订单编号相关联。其次，企业的生产安排人员基于上述订单信息进行订单的分批，并合理调度批次形成分批生产调度计划。将该调度计划派发给生产操作人员执行生产，并对生产进度实时监控和反馈。然后，产成品将被打包装入标准托盘，通过使用堆垛机将托盘运入仓库进行储存。然后，依照制定的车辆配送计划，确定订单的配送车辆、出发时间以及交付顺序。其中通过叉车搬运、人工装车等方式实现产品由仓库到配送车辆的运送。最后，配送人员依照车辆配送计划进行客户订单的派发。

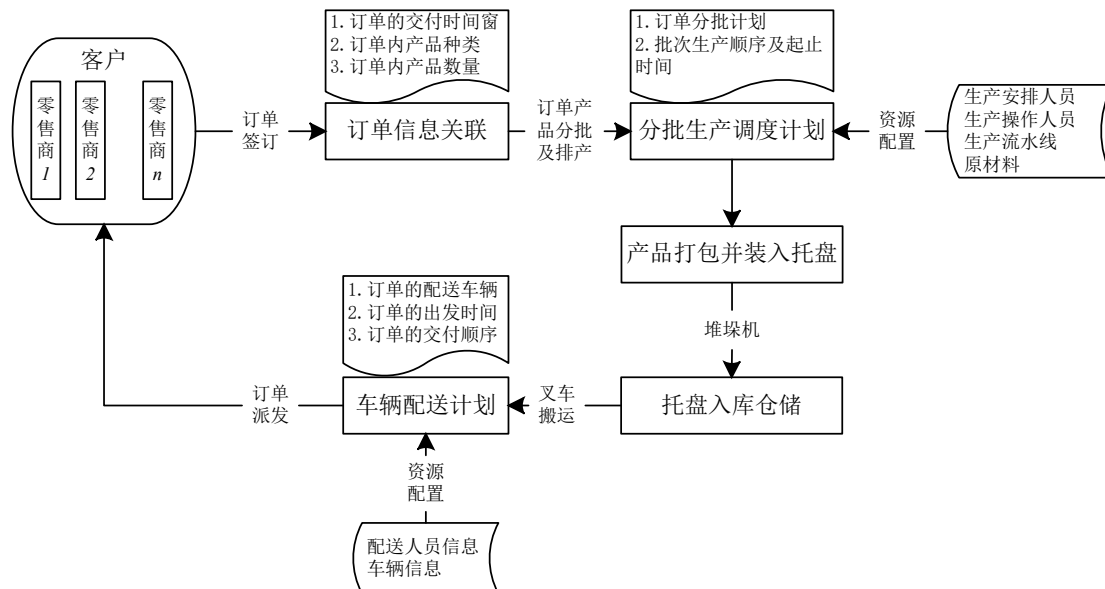


图 2-1 生产-配送集成调度流程

分批生产调度计划和车辆配送计划的制定是快速消费品制造企业运营过程

中的两个重要环节。其直接关系到订单产品的生产顺序、生产时间、储存时间、交付时间、配送车辆、配送路线，进而影响供应链的整体效益以及客户的满意程度。并且分批生产调度计划和车辆配送计划的制定具有多约束限制和多阶段组合的特征，因此如何将两个计划充分协调是生产-配送集成调度的关键及难点所在。

2.2 生产-配送集成调度特征识别

本节针对快速消费品制造企业生产-配送集成调度流程中存在的特征进行识别。进而明确的界定研究问题，为后续生产-配送集成调度优化模式的设计提供依据。

2.2.1 流水线分批生产调度特征

快速消费品制造企业以客户订单为驱动进行生产，未采用按照销售预期进行生产的模式，从而消除了生产的盲目性以及过量生产带来的产品库存问题。采用流水线分批生产的方式可以实现按需生产，降低企业库存，减少成品浪费。此外，快速消费品制造企业需要通过配送车辆向一组地理位置各异的零售商交付订单产品。由配送车辆一次运送的订单构成一个生产批次，从而将批次的生产和交付一一对应起来。在多条流水线上一个批次被视为一个生产单元，然后根据制定的批次生产顺序及时间执行生产调度。

2.2.2 车辆配送调度特征

快速消费品制造企业由于其产品特性及交付模式，往往采用自有车队执行订单的配送。从而企业需要制定相应的车辆配送计划来完成订单的交付。由于订单中的产品具有体积容量小、消费频繁、车辆容量可以满足多订单装载的特征，所以企业采用车辆路由的方式进行配送。其中将配送车辆由工厂出发依次交付一组零售商订单后返回工厂的车辆路线称为巡回环。如下所述，归纳了快速消费品制造企业车辆配送调度的具体特征：

(1) 具有软时间窗限制

由于快速消费品具有货架期短以及商品快速周转的特征，客户通常要求订单产品在某一时间区间内完成交付。因为提前的交付极易造成客户处的产品积

压,而延期交付则会导致客户处的产品短缺,所以提前和延期交付都不利于产品的快速周转。并且在市场竞争日益激烈的环境下,延期交付比提前交付更不能让人接受,因为延期交付直接影响的客户的收益。因此,在制定车辆配送计划时应充分考虑订单的软时间窗限制,从而尽可能的减少订单的提前和延期交付。

(2) 数量受限的异构车辆配送

快速消费品制造企业的自有车队由数量受限且容量不同的异构车辆组成。在制定车辆配送计划时,根据订单所占据的容量选择恰当的配送车辆可以提高车辆的负载率,从而有效节省企业的配送成本。

(3) 多行程配送车辆

快速消费品制造企业的配送车辆可以多次使用,即同一车辆可执行多个巡回环,但需要保证巡回环之间没有时间冲突。多行程的配送车辆可以有效提高车辆的使用频率,进而降低车队的车辆规模减少企业的车辆日常维护成本。

(4) 巡回环出发时间不固定

巡回环何时出发需要根据该巡回环内所有订单的软时间窗限制来具体确定。并且巡回环的出发也受到配送车辆可用性的影响,因此巡回环出发时间是不固定的。

(5) 与批次的生产、库存密切相关

快速消费品制造企业的产品进行车辆配送的前提是产品已完成生产。并且库存作为缓冲区连接着批次生产和配送。这一特征表明车辆配送阶段必然与批次的生产和库存阶段密切相关。因此,在制定车辆配送计划时应考虑企业生产能力限制及仓储库存时间的影响。

2.2.3 不确定旅途时间扰动特征

由于快速消费品制造企业的分批生产过程采用高度自动化流水线,所以产品加工环境较为确定,具有稳定的产品生产节拍。然而,车辆配送过程受道路交通状况、天气环境、突发情况等不确定因素的影响通常导致旅途时间的具有波动性。因此,考虑不确定旅途时间对快速消费品制造企业生产-配送集成调度产生的影响是极为必要的。

不确定旅途时间将直接影响车辆配送计划的制定。旅途时间的不确定直接导致巡回环执行时间的不确定,这不仅会导致巡回环出发时间、订单交付时间、仓储库存时间具有不确定性。而且会影响配送车辆在某一时刻的可用性。此外,

由于批次生产和配送密切相关，所以不确定的巡回环出发时间也将影响批次的生产调度。综上所述，不确定旅途时间将影响快速消费品制造企业的分批生产调度计划和车辆配送计划的制定。

目前，针对不确定性参数的处理方式主要包括：随机规划、模糊规划以及鲁棒优化。由于企业很难获得大量的可靠旅途时间数据来拟合概率分布从而实现基于概率分布的随机规划^[34]，并且基于最坏可能情况的鲁棒优化往往又过于保守^[28]。因此，本文选用模糊规划的方式进行不确定旅途时间的处理。

2.3 生产-配送集成调度优化模式

2.3.1 生产-配送集成调度优化框架

根据快速消费品制造企业生产-配送集成调度流程及特征，将生产-配送集成调度决策划分为四个子决策问题：

- (1) 订单的配送车辆选择决策；
- (2) 同一车辆交付的订单的巡回环划分决策（获取批次中的订单组成）；
- (3) 批次的生产顺序决策；
- (4) 批次生产和巡回环出发的协调调度决策；

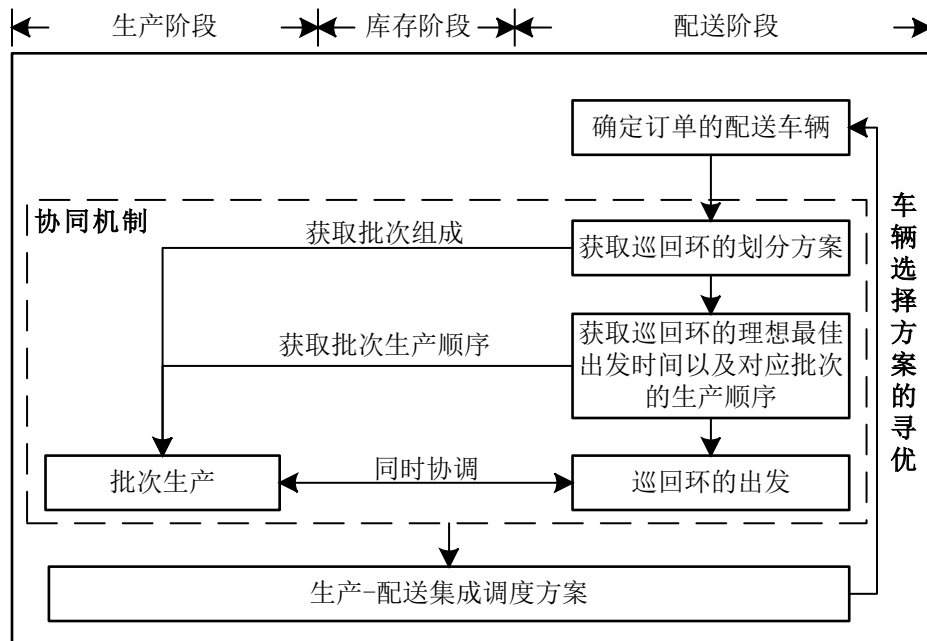


图 2-2 生产-配送集成调度优化框架

如图 2-2 所示，设计了生产-配送集成调度优化框架来解决所研究问题。该

优化框架采用内、外双层的优化模式。其中，外层优化为订单车辆选择方案的寻优，用于处理子决策问题(1)。内层优化为协同机制的实现，用于处理子决策(2)-(4)。即根据一个具体的车辆选择方案获得与之对应的最佳的生产-配送集成调度计划。协同机制具体流程如下：首先，根据订单的车辆选择方案，设计一个巡回环划分策略以获得每个批次中的订单组成。其次，根据订单的软时间窗限制，获取巡回环的理想最佳出发时间。基于该时间进行排序，获取与巡回环相对应的批次生产顺序。最后，同时协调批次生产和巡回环的出发，以确定最佳的生产-配送集成调度方案。综上所述，该生产-配送集成调度优化框架采用双层优化模式，将生产-配送集成调度中的四个子决策进行了有机结合，以实现分批生产调度计划和车辆配送计划的协调。

2.3.2 生产-配送集成调度优化策略

本节介绍解决生产-配送集成调度优化框架内的各子决策问题的优化策略，从而为优化框架的具体算法实现提供指导。以下是各子决策问题优化策略的选用分析。

子决策 1：订单的配送车辆选择决策。该问题可视为车辆选择方案的寻优问题，以车辆选择方案作为编码形式，以协同机制作为解码过程，通过采用迭代优化型算法策略获取最佳的车辆选择方案。其中迭代优化型算法策略可选取进化类算法、启发式算法、局部搜索算法、群智能优化算法中的一种或多种的组合。

子决策 2：同一车辆交付的订单的巡回环划分决策。该问题可视为根据订单的产品数量、交付时间限制以及配送车辆容量等条件制定巡回环划分方案。其优化策略可选用针对性较强的启发式算法。

子决策 3：批次的生产顺序决策。即根据巡回环中订单的交付顺序及交付时间限制获取巡回环的理想最佳出发时间，进而获取批次的生产顺序。由于巡回环的理想最佳出发时间的计算涉及到线性函数的求和及取极值等过程，故该问题的求解策略应基于数学规划程序进行。

子决策 4：批次生产和巡回环出发的协调调度决策。通过获取了批次的生产顺序、巡回环的理想最佳出发时间后，需要进行批次和巡回环的协调调度优化。其优化策略可选用基于特定规则的启发式算法、局部迭代优化算法。

2.4 本章小结

本章首先分析了快速消费品制造企业的生产-配送集成调度流程，并指出了分批生产调度计划和车辆配送计划的制定是该流程中的两个重要环节。然后，分别从流水线分批生产调度、车辆配送调度、不确定旅途时间扰动等方面对生产-配送集成调度流程中存在的特征进行识别分析。最后，提出了一个具有内、外双层优化模式的生产-配送集成调度优化框架来指导问题的解决。该优化框架包含生产-配送协同机制设计以及车辆选择方案的迭代寻优。最后，为具体实现该优化框架进行了优化策略的选用分析。本章为后续章节的研究提供了理论支持。

第3章 考虑分批生产及软时间窗限制的生产-配送集成调度

本章针对快速消费品制造企业的生产-配送集成调度系统，考虑了分批生产形式下的生产调度环境、订单的软时间窗限制以及通过数量受限的多行程异构车辆执行配送任务等实际特征。构建了以最大化总利润和最小化提前延期加权惩罚时间为双优化目标的混合整数非线性规划模型。然后，基于第2.3.1节的生产-配送集成调度优化框架，设计开发了一种基于自适应大邻域搜索策略改进的多目标进化算法进行模型求解。最后，通过数值实验验证了算法的有效性。

3.1 生产-配送集成调度模型

3.1.1 问题描述

本章所研究问题包括三个阶段：分批生产的生产阶段、库存阶段和具有软时间窗限制的配送阶段。如图3-1所示，生产阶段包括相互独立的流水生产线，每条生产线以不同的生产节拍生产一种类型的产品。每个批次包含多种类型的产品，其中 $B(I, P)$ 代表批次 $B(I)$ 的产品 P 。当同一生产线上相邻批次之间存在空隙时间时（例如： $B(1, I)$ 和 $B(2, I)$ ），生产线需要停机并重新启动从而造成生产启动成本。只有当批次内产品全部生产完成后，该批次才能执行交付。此外，如果批次中某一类型的产品生产完成后未立即配送则会产生库存。其中工厂有足够的库存容量空间，产品的库存和车辆容量以标准托盘为核算单位。最后，配送阶段被建模为车辆路径问题（Vehicle routing problem, VRP），执行配送任务的车辆路线称为巡回环。其被定义在整个图 $G=(N, A)$ 上，包含各种顶点 $N=\{0, 1, \dots, n\}$ ，弧集为 $A=\{(k, l): k, l \in N\}$ 。其中工厂位于顶点 0 ，零售商的集合被描述为 $R=N \setminus \{0\}$ 。通过数量受限的多行程异构车辆将订单交付给一组地理位置分散的零售商。其余的假设条件如下：

在每条生产线上批次内部的产品生产是连续的；

- 每辆车从工厂出发，依次访问巡回环上的零售商并返回工厂；
- 每个零售商的订单需求必须一次性被满足，不允许拆分交付；
- 允许未满载的车辆执行配送任务，同一车辆可执行多次巡回环，产品允许混

装；

- 巡回环执行期间，车辆连续交付不可等待、停滞；

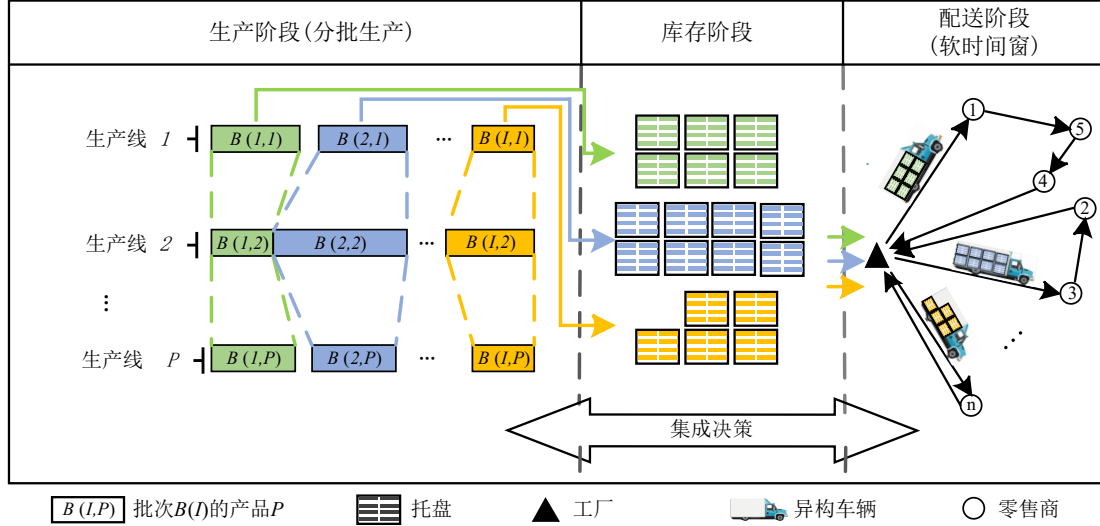


图 3-1 快速消费品制造企业生产-配送集成调度概要图

集成决策制定流程见图 3-2 所示。首先，为每一个订单选择配送车辆。然后，根据软时间窗和车辆容量限制，将分配给每辆车的订单划分形成不同的巡回环。同一巡回环上零售商订单的产品构成一个批次，巡回环和批次一一对应，并且每个巡回环都具有灵活的出发时间。根据计算得出的巡回环理想最佳出发时间获得批次的生产顺序。最后，通过同时协调具有生产顺序的批次和具有理想最佳出发时间的巡回环，得到生产-配送集成调度方案。

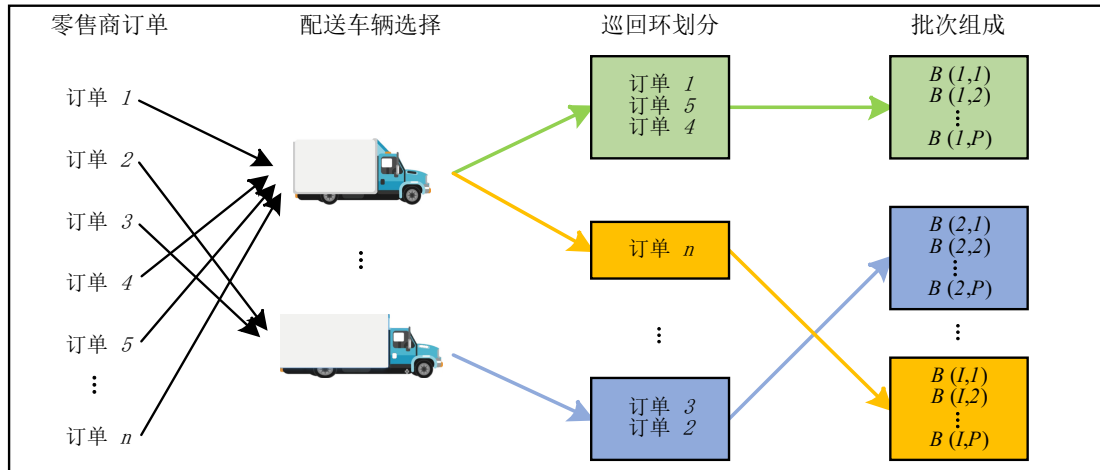


图 3-2 集成决策制定的流程示意图

3.1.2 符号定义及说明

针对所构建的生产-配送集成调度数学模型中使用的符号参数进行如下说明：

索引

p	产品类型索引（生产线）（ $p = (1, 2, \dots, P)$ ）
k, l	零售商索引（ $k, l = (1, 2, \dots, n)$ ）
$0, n+1$	工厂的起始和终止位置
i, j	批次生产顺序索引（ $i, j = (1, 2, \dots, n)$ ）
v	配送车辆索引（ $v = (1, 2, \dots, V)$ ）

常量

$D_{k,p}$	零售商 k 对产品 p 的需求量
Pr_p	单位产品 p 的平均售价
Pc_p	单位产品 p 的生产成本
t_p	产品 p 的生产节拍
Fs_p	生产线 p 的固定启动成本
O_p	单位产品 p 占据托盘的容量
Hc	单位托盘在单位时间的库存持有成本
C_v	配送车辆 v 的最大装载托盘数
Fc_v	配送车辆 v 的固定成本
Vc_v	配送车辆 v 的单位时间的可变成本
$T_{k,l}$	零售商 k 到零售商 l 的旅途时间
$[e_k, l_k]$	零售商 k 的配送时间窗
Rd	单位托盘产品的延期惩罚率
Re	单位托盘产品的提前惩罚率
M	足够大的数

辅助变量

$Tc_{p,B(i)}$	批次 $B(i)$ 中产品 p 的生产完成时间
a_k	零售商 k 订单的交付到达时间
$w_{p,B(i)}$	如果 $Ts_{p,B(i+1)} > Tc_{p,B(i)}$ 则为 1，否则为 0
$S_{B(i),B(j)}$	如果同一车辆配送批次 $B(i)$ 后紧接配送批次 $B(j)$ 则为 1，否则为 0

决策变量

$Ts_{p,B(i)}$	批次 $B(i)$ 中产品 p 的生产开始时间
---------------	---------------------------

$Td_{B(i)}$	批次 $B(i)$ 的所对应巡回环的出发时间
$E_{B(i)}$	如果批次 $B(i)$ 存在则为 1，否则为 0
$X_{k,l}$	如果零售商 k 被访问完成后紧接访问零售商 l 则为 1，否则为 0
$Y_{v,B(i)}$	如果配送车辆 v 进行批次 $B(i)$ 的配送则为 1，否则为 0
$G_{k,B(i)}$	如果零售商 k 的订单产品在批次 $B(i)$ 内生产则为 1，否则为 0

3.1.3 混合整数非线性规划模型

本节构建了快速消费品制造企业的生产-配送集成调度数学模型。其目标函数及约束条件具体描述如下：

(1) 总利润 (Total profit, TP)

总利润包括产品收入、生产启动成本、库存成本和配送成本。生产启动成本与连续批次间生产空隙的数量直接相关。库存成本与产品占用的托盘数量和存储时间有关。此外，配送成本由两部分组成。一部分是与车辆使用次数相关的固定成本，另一部分是与车辆路径时间相关的可变成本（燃料消耗成本）。

$$\begin{aligned}
 \text{Max } TP = & \sum_{p=1}^P \sum_{k=1}^n (Pr_p - Pc_p) D_{k,p} - \sum_{p=1}^P \sum_{i=1}^{n-1} W_{p,B(i)} Fs_p - \\
 & Hc \sum_{i=1}^n \sum_{p=1}^P \sum_{k=1}^n \left[G_{k,B(i)} D_{k,p} O_p (Td_{B(i)} - Tc_{p,B(i)}) \right] - \\
 & \sum_{v=1}^V \left\{ Fc_v \sum_{i=1}^n Y_{v,B(i)} + Vc_v \sum_{k=1}^n \sum_{i=1}^n \left[G_{k,B(i)} Y_{v,B(i)} \left(X_{0,k} T_{0,k} + \sum_{l=1}^{n+1} X_{k,l} T_{k,l} \right) \right] \right\}
 \end{aligned} \quad (3-1)$$

(2) 提前延期加权惩罚时间 (Early and tardy weighted penalty time, ETPT)

考虑到订单的交付时间限制，在软时间窗内的及时交付有利于提高客户的满意程度。故通过提前延期加权惩罚时间这一目标来衡量客户的满意程度。其由交付订单的提前或延迟时间、订单占用的托盘数量、提前或延迟惩罚率共同决定。

$$\text{Min } ETPT = \sum_{k=1}^n \left[(Re \max(e_k - a_k, 0) + Rd \max(a_k - l_k, 0)) \sum_{p=1}^P (D_{k,p} O_p) \right] \quad (3-2)$$

Subject to:

$$Ts_{p,B(i)} + t_p \sum_{k=1}^n G_{k,B(i)} D_{k,p} = Tc_{p,B(i)} \quad \forall p \in \{1, \dots, P\} \quad \forall i \in \{1, \dots, n\} \quad (3-3)$$

$$Tc_{p,B(i)} \leq Ts_{p,B(i+1)} + M(1 - E_{B(i+1)}) \quad \forall p \in \{1, \dots, P\} \quad \forall i \in \{1, \dots, n-1\} \quad (3-4)$$

$$Tc_{p,B(i)} \leq Td_{B(i)} \quad \forall p \in \{1, \dots, P\} \quad \forall i \in \{1, \dots, n\} \quad (3-5)$$

$$Td_{B(i)} = \sum_{k=1}^n (a_k - T_{0,k}) X_{0,k} G_{k,B(i)} \quad \forall i \in \{1, \dots, n\} \quad (3-6)$$

$$\sum_{l=1}^n X_{k,l} (a_k + T_{k,l} - a_l) = 0 \quad \forall k \in \{1, \dots, n\} \quad (3-7)$$

$$\sum_{l=1}^n (a_l + T_{l,n+1}) X_{l,n+1} G_{l,B(i)} \leq Td_{B(j)} + M(1 - S_{B(i),B(j)}) \quad (3-8)$$

$$\forall i \in \{1, \dots, n-1\}, \forall j \in \{i+1, \dots, n\}$$

$$\sum_{l=1}^n X_{0,l} = \sum_{k=1}^n X_{k,n+1} = \sum_{i=1}^n E_{B(i)} \quad (3-9)$$

$$\sum_{k=0}^n X_{k,l} = 1 \quad \forall l \in \{1, \dots, n\} \quad (3-10)$$

$$\sum_{l=1}^{n+1} X_{k,l} = 1 \quad \forall k \in \{1, \dots, n\} \quad (3-11)$$

$$e_k \leq e_l + M(1 - X_{k,l}) \quad \forall k \in \{1, \dots, n\}, \forall l \in \{1, \dots, n\} \quad (3-12)$$

$$l_k \leq l_l + M[1 + X_{k,l}(e_l - e_k - 1)] \quad \forall k \in \{1, \dots, n\}, \forall l \in \{1, \dots, n\} \quad (3-13)$$

$$\sum_{i=1}^n G_{k,B(i)} = 1 \quad \forall k \in \{1, \dots, n\} \quad (3-14)$$

$$\sum_{v=1}^V Y_{v,B(i)} \leq 1 \quad \forall i \in \{1, \dots, n\} \quad (3-15)$$

$$1 - M(1 - E_{B(i)}) \leq \sum_{k=1}^n G_{k,B(i)} \leq M E_{B(i)} \quad \forall i \in \{1, \dots, n\} \quad (3-16)$$

$$1 - M(1 - E_{B(i)}) \leq \sum_{v=1}^V Y_{v,B(i)} \leq M E_{B(i)} \quad \forall i \in \{1, \dots, n\} \quad (3-17)$$

$$G_{k,B(i)} \geq G_{l,B(i)} + X_{k,l} - 1 \quad \forall k \in \{1, \dots, n\}, \forall l \in \{1, \dots, n\}, \forall i \in \{1, \dots, n\} \quad (3-18)$$

$$G_{l,B(i)} \geq G_{k,B(i)} + X_{k,l} - 1 \quad \forall k \in \{1, \dots, n\}, \forall l \in \{1, \dots, n\}, \forall i \in \{1, \dots, n\} \quad (3-19)$$

$$E_{B(i+1)} \leq M E_{B(i)} \quad \forall i \in \{1, \dots, n-1\} \quad (3-20)$$

$$\sum_{p=1}^P \sum_{k=1}^n G_{k,B(i)} D_{k,p} O_p \leq C_v + M(1 - Y_{v,B(i)}) \quad \forall v \in \{1, \dots, V\}, \forall i \in \{1, \dots, n\} \quad (3-21)$$

$$E_{B(i)}, X_{k,l}, Y_{v,B(i)}, G_{k,B(i)}, S_{B(i),B(j)}, W_{p,B(i)} \in \{0, 1\} \quad (3-22)$$

$$Ts_{p,B(i)}, Tc_{p,B(i)}, Td_{B(i)}, a_k \geq 0 \quad (3-23)$$

其中, 约束(3-3)-(3-4)表示生产线上批次间的时间约束。约束(3-5)表示只有批次内的所有产品完成后才能执行该批次的交付。约束(3-6)-(3-13)用于描述路由的时间和顺序方面的约束。约束(3-6)表示从工厂到第一个零售商的时间约束。约束(3-7)表示巡回环上相邻零售商的交货顺序和时间的约束。约束(3-8)表示车辆在完成当前巡回环并返回工厂之前无法执行下一次巡回环。约束(3-9)表示巡回环将工厂作为开始和终止节点。约束(3-10)-(3-11)确保零售商只能被访问一次并且离开一次。约束(3-12)-(3-13)表示巡回环中零售商的访问顺序。约束(3-14)-(3-15)表示零售商的订单只能存在于一个批次中, 并且一个批次只能由一辆车执行交付。约束(3-16)表示只有当批次存在时, 批次才能包含订单产品。约束(3-17)表示如果批次存在, 则该批次必须由车辆执行交付。约束条件(3-18)-(3-19)表示在巡回环内零售商 k 紧随零售商 l 。如果零售商 k 由车辆 v 交付, 那么零售商 l 也将由车辆 v 交付, 反之亦然。约束(3-20)确保只有当前批次存在时才有可能存在下一批次。约束(3-21)表示批次所占用的容量不应超过配送车辆的容量。最后, 约束(3-22)-(3-23)定义了变量的域和性质。

3.2 基于自适应大邻域搜索策略改进的多目标进化算法 (IMOEa)

根据第 2.3.1 节的生产-配送集成调度优化框架, 提出了一种基于自适应大邻域搜索策略改进的多目标进化算法 (Multi-objective evolutionary algorithm improved by adaptive large neighborhood search, IMOEa)。该算法采用一维向量来表示编码个体, 每个个体代表一种车辆选择方案。个体由 n 个编码位置 (零售商数量) 组成, 每个编码位置存储交付该零售商订单的车辆序列号 ($1, \dots, V$)。IMOEa 包括协同机制和基于混合算法的优化模式。其中协同机制用于个体的适应度函数计算。即根据一个具体的车辆选择方案获取与之对应的最佳的生产配送集成调度计划 (个体的解码)。然后, 采用基于混合算法的优化模式执行订单的车辆选择决策。

IMOEa 的具体流程如图 3-3 所示:

步骤 1: 设定算法的参数, 并采用随机的方式生成初始种群, 即根据零售商订单所占据的托盘数, 为其随机选择可执行配送任务的车辆编号, 以保证初始种群的分散性和多样性。

步骤 2: 通过协同机制实现个体的解码。将由同一车辆配送的订单划分进入

不同的巡回环，见第 3.3.1 节。并提出了交付惩罚时间（Delivery penalty time , DPT）函数，以获取每个巡回环的理想最佳出发时间，见第 3.3.2 节。为解决批次生产和巡回环出发的协调问题，在第 3.3.3 节中设计了基于无等待调度的双向调度策略（Bidirectional scheduling strategy based on no-wait schedule , BSS-NWS）。从而根据所获取的生产-配送集成调度方案即可计算得出个体的适应度值。

步骤 3：采用基于特定问题和编码形式设计的自适应大邻域搜索策略（Adaptive large neighborhood search , ALNS）改进的多目标进化算法的优化模式。其中 MOEA 的遗传操作包括选择、交叉和变异操作。选择和交叉操作遵循 NSGA-II^[45] 中的二元锦标赛选择操作和标准的两点交叉操作。变异操作采用基于小概率的多点变异，旨在为部分零售商随机更换配送车辆，从而扩展算法的搜索范围。即随机选择多个编码位置为其更改车辆序列号，并且执行变异的编码位置数量也是随机的。通过 NSGA-II 中的非支配排序方法和拥挤距离计算方法获取 Pareto 前沿面。并在第 3.4 节中介绍了 ALNS 的具体执行流程。

步骤 4：基于精英保留策略进行种群的修剪，如果满足迭代终止条件，则算法结束，否则转至步骤 2。

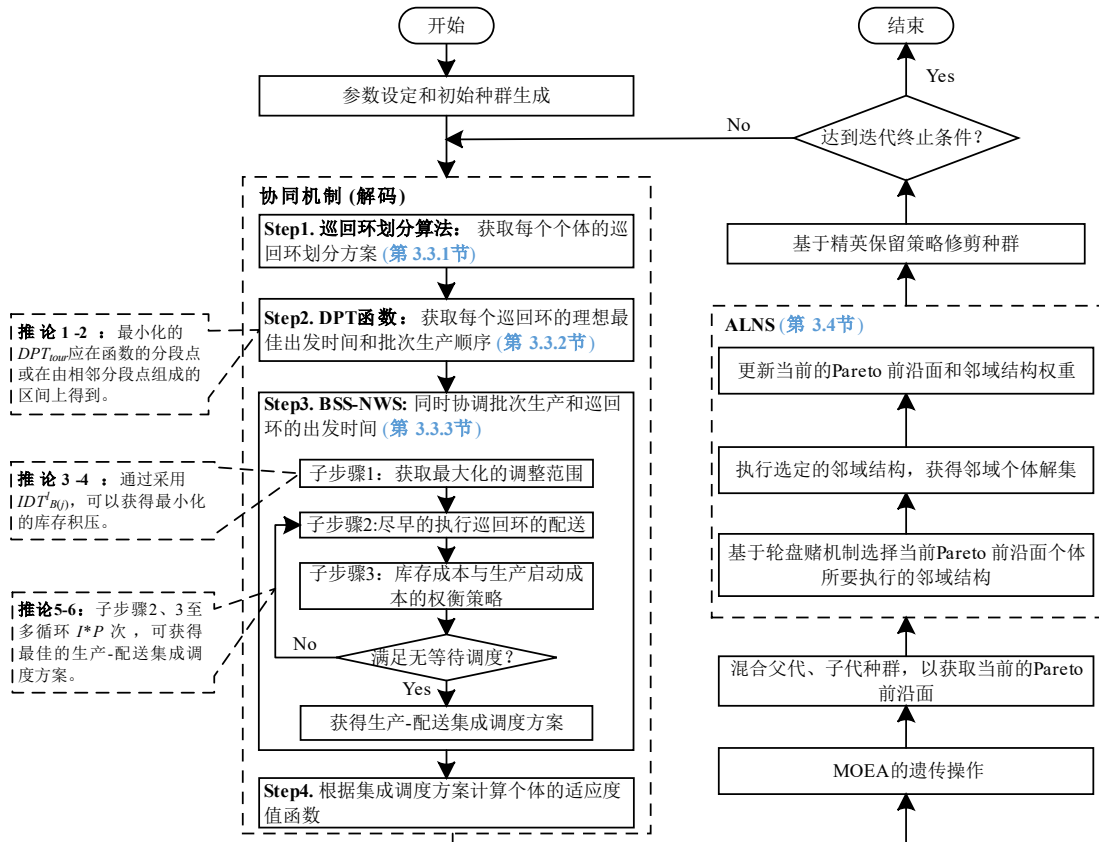


图 3-3 IMOE 算法流程图

3.3 IMOEa 中的协同机制

本节所提出的协同机制用于个体的解码，从而获得个体的适应度值函数。为了更加直观的展现此过程，如图 3-4 所示，进行了举例说明。零售商 1, 2, 4, 5, 6 的订单采用车辆 1，零售商 3, 7, 8 的订单采用车辆 2 进行交付。首先，利用巡回环划分算法获取巡回环构成。由于巡回环和批次一一对应，即获得批次的订单组成。然后，通过计算巡回环的理想最佳配送出发时间获取批次生产顺序。最后，采用 BSS-NWS 策略，进行批次生产和巡回环出发的同时协调，获得编码个体对应的生产-配送集成调度方案。

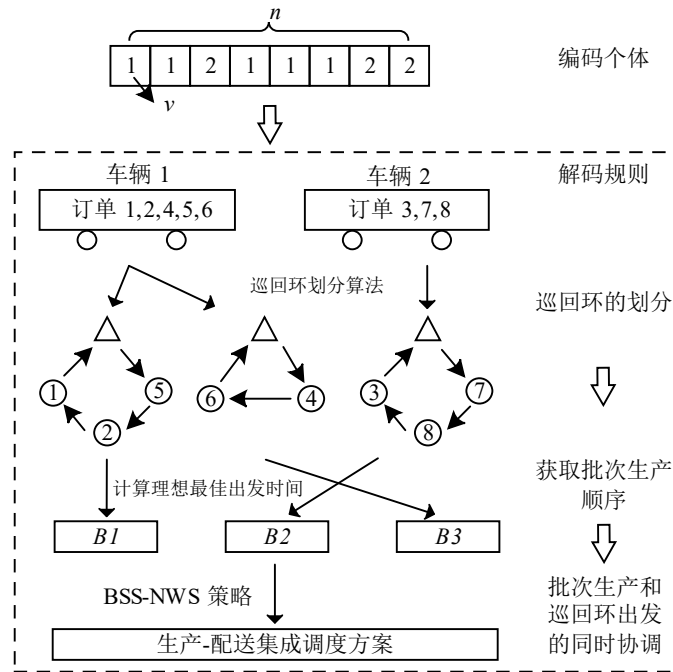


图 3-4 编解码示例图

3.3.1 巡回环划分算法

为了保证零售商订单的及时交付，该算法设计时考虑了软时间窗的影响。表 3-1 给出了基于车辆容量和软时间窗的巡回环划分算法的伪代码。根据车辆选择方案（个体编码），将由同一车辆配送的订单划分进入不同的巡回环。首先将由相同车辆配送的订单按其软时间窗下限由小到大排序，其中 RS_m^v 代表由车辆 v 配送的第 m 个订单。 N_v 代表由车辆 v 配送的订单数量， O_c 是当前巡回环 h 累计占据的托盘数。

表 3-1 巡回环划分算法

算法 1 巡回环划分算法

```

1. 输入:  $RS_m^v, N_v, O_c, h, C_v, [e_k, l_k], D_{k,p}, O_p, a_k$ ;
2. 输出:  $E_{B(i)}, X_{k,l}$ , 各巡回环的订单组成;
3. For  $v \in \{1, \dots, V\}$  do
4.    $O_c = 0; \quad h = 1;$ 
5.   For  $m \in \{1, \dots, N_v\}$  do
6.      $O_c = O_c + \sum_p^P (D_{RS_m^v, p} O_p);$ 
7.     If  $O_c > C_v$  then
8.       将  $RS_m^v$  放入下一巡回环; 更新  $O_c = \sum_p^P (D_{RS_m^v, p} O_p); \quad h = h + 1;$ 
9.     Else
10.      以当前巡回环  $h$  中的第一个订单的软时间窗上限作为起始时间。
11.       $a_{RS_m^v}^{curr}$  代表从起始时间开始计算, 将订单  $RS_m^v$  放入当前巡回环时,
12.       $RS_m^v$  的到达时间。
13.       $a_{RS_m^v}^{next}$  代表从起始时间开始计算, 将订单  $RS_m^v$  放入下一巡回环时, 车
14.      辆  $v$  返回工厂后再立即配送  $RS_m^v$  的到达时间。
15.      If  $a_{RS_m^v}^{curr} < e_{RS_m^v}$  and  $a_{RS_m^v}^{next} < l_{RS_m^v}$  and  $m \neq N_v$  then
16.        将  $RS_m^v$  放入下一巡回环; 更新  $O_c = \sum_p^P (D_{RS_m^v, p} O_p); \quad h = h + 1;$ 
17.      Else
18.        将  $RS_m^v$  放入当前巡回环;
19.      Endif
20.    Endif
21.  Endfor
22. Endfor

```

3.3.2 基于 DPT 函数的理想最佳出发时间确定

DPT 代表订单时超出时间窗的交付惩罚时间。对于某一零售商来说, 如果配送车辆提前或延迟到达时, DPT 将随着时间间隔的增加而线性增加。当在时间窗内到达时, DPT=0。因此, 如图 3-5 所示 DPT 函数是分段线性函数。

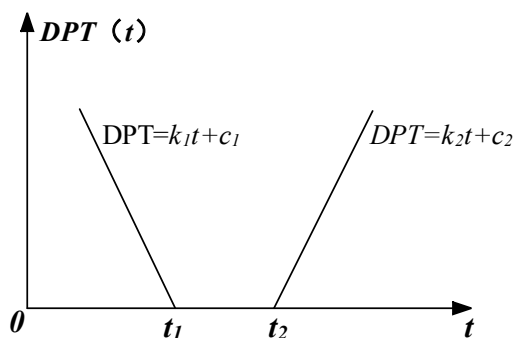


图 3-5 零售商的 DPT 函数

DPT 函数符号介绍如下：

- t 代表横坐标，表示巡回环的出发时间。
- $DPT(t)$ 代表纵坐标，每一个 t 出发对应一个 $DPT(t)$ 。
- t_1, t_2 是分段线性函数的分段点，即与零售商订单时间窗上下限所对应的巡回环出发时间。
- c_1, c_2 分别代表分段的 y -截距系数。
- k_1, k_2 分别代表该段的斜率系数，该斜率与零售商订单中的产品数量和提前或延期惩罚率有关。

由于一个巡回环通常包含一组零售商，故使用 DPT_{tour} 表示巡回环的交付惩罚时间。 DPT_{tour} 由该巡回环内所有零售商的 DPT 函数组成。并且对应最小化 DPT_{tour} 的出发时间 t 为巡回环的理想最佳出发时间。

推论 1: DPT_{tour} 为分段线性函数。

证明: 假设一个巡回环内包含 m 个零售商，根据零售商的 DPT 函数为分段线性函数的性质则 DPT_{tour} 至多可以获得 $2m$ 个分段点，从而将出发时间划分为 $2m+1$ 个分段区间。从而 $(t_i, t_{i+1}) \quad i \in \{0, \dots, 2m\} \quad t_0 = 0; t_{2m+1} = +\infty$ 是最小的分段区间，零售商的 DPT 函数在任意的 (t_i, t_{i+1}) 上都是线性的。根据线性函数经线性运算后仍为线性函数这一数学性质，即 DPT_{tour} 在每个区间 (t_i, t_{i+1}) 上都是线性函数，因此 DPT_{tour} 在 $t \in (0, +\infty)$ 上为分段线性函数。证明完毕。

推论 2: 与最小化 DPT_{tour} 相对应的出发时间 t 必在分段点处获得。如果相邻的分段点同时获得最小化 DPT_{tour} ，则在相邻分段点所组成的区间内均可获得最小化 DPT_{tour} 。

证明: 首先，最小化的 DPT_{tour} 必存在于区间 $[t_1, t_{2m}]$ 内，根据闭合区间上连续函数的性质，如果最小化的 DPT_{tour} 存在，则只能在端点、驻点及不可导点处获得。由于 DPT_{tour} 为分段线性函数，则不可导点一定是分段点。并且由于 DPT_{tour}

在每个区间 (t_i, t_{i+1}) $i \in \{1, \dots, 2m-1\}$ 上都是线性的，如果存在驻点也一定存在于以分段点为端点的区间上。因此，与最小化 DPT_{tour} 相对应的出发时间 t 必在分段点处获得。并且，如果相邻的分段点同时取得最小化的 DPT_{tour} ，那么根据 DPT_{tour} 在区间 (t_i, t_{i+1}) $i \in \{1, \dots, 2m-1\}$ 上属于线性函数的性质，最小化的 DPT_{tour} 可在区间 $[t_i, t_{i+1}]$ 内的任意处获得。证明完毕。

因此，仅需计算区间 $[t_1, t_{2m}]$ 上所有的分段点处的 DPT_{tour} 即可获得巡回环的理想最佳出发时间节点（或时间段）。从而将一个区间极值问题转化为一个有限节点极值问题，简化了理想最佳出发时间计算的复杂度。由于巡回环和批次一一对应，根据巡回环的理想最佳出发时间对巡回环进行由大到小排序，从而获取批次的生产顺序。表 3-2 描述了巡回环的理想最佳出发时间的求解过程。其中 N_{tour} 巡回环中零售商订单的个数， FRT 用于储存从工厂到零售商的路由时间， $Tour$ 代表巡回环内零售商的交付顺序。 SP 存储与 DPT_{tour} 分段点相对应的巡回环出发时间。

表 3-2 理想最佳出发时间的计算

算法 2 理想最佳出发时间的计算

1. **输入:** N_{tour}, FRT, SP ;
 2. **输出:** 理想最佳出发时间;
 3. **For** $k \in \{1, \dots, N_{tour}\}$ **do**
 4. **For** $i \in \{0, \dots, k-1\}$ **do**
 5. $FRT = FRT + T_{i,i+1}$;
 6. **End for**
 7. **If** $e_{Tour(k)} - FRT > 0$ **then**
 8. 将 $e_{Tour(k)} - FRT$ 放入 SP 中;
 9. **Else**
 10. 将 0 放入 SP 中;
 11. **End if**
 12. **If** $l_{Tour(k)} - FRT > 0$ **then**
 13. 将 $l_{Tour(k)} - FRT$ 放入 SP 中;
 14. **Else**
 15. 将 0 放入 SP 中;
 16. **End if**
 17. **End for**
 18. 删除 SP 中重复的时间节点, 然后针对 SP 中的每个时间节点计算 DPT_{tour} ;
 19. 与最小化 DPT_{tour} 相对应的时间节点（或时间区间）是理想最佳出发时间;
-

3.3.3 基于无等待调度的双向调度策略的同时协调（BSS-NWS）

当批次内的所有产品生产完成后立即交付，将其称为无等待调度。由于其不产生库存，所以这是一种理想的处理方式。但是受到生产能力和车辆数量限制等实际约束，很难保证每个批次均满足无等待调度。故提出了 BSS-NWS 来使得生产-配送集成调度方案尽可能地满足无等待调度。BSS-NWS 由三个子策略组成，其具体过程见图 3-3 中的对应部分。通过三个子策略的有效整合，实现了批次生产和巡回环出发的同时协调，并给出以下推论及证明来确立此概念。

(1) 子策略 1：获取最大化的调整范围策略

引入如下符号： $[IDT_{B(j)}^e, IDT_{B(j)}^l]$ 代表批次 $B(j)$ 的理想最佳出发时间区间（巡回环和批次一一对应）。 $TR_{v,B(i)}$ 代表车辆 v 交付完成批次 $B(j)$ 后的返厂时间。 $Td_{B(j)}^e$ 代表批次 $B(j)$ 可行的最早出发时间。

子策略 1 的执行步骤如下：首先，从时间节点 0 开始，按批次生产顺序连续排列各批次。从而获得各批次的可行的最早出发时间节点 $Td_{B(j)}^e$ 。然后，为了获得最大的调整范围以实现最小化的库存积压，根据 $IDT_{B(j)}^l$ 和 $Td_{B(j)}^e$ ，按批次生产顺序依次更新 $Td_{B(j)} = \max(Td_{B(j)}^e, IDT_{B(j)}^l)$ 。最后，根据无等待调度，按照逆向批次生产顺序执行批次的后向调度，从而更新 $Tc_{p,B(j)} = \min(Ts_{p,B(j+1)}, Td_{B(j)})$ 。

推论 3：如果批次 $B(j)$ 紧跟着批次 $B(i)$ 并由同一辆车执行交付任务，那么根据无等待调度， $Td_{B(j)}^e = \max(TR_{v,B(i)}, \max(Tc_{1,B(j)}, \dots, Tc_{p,B(j)}))$ 。

证明：批次的交付需要满足两个硬性条件。一是该批次内所有产品均生产完成。二是该批次的所选定的配送车辆在工厂内等待执行任务。如果批次 $B(i)$ 是车辆 v 的第一个巡回环，则 $TR_{v,B(i)} = 0$ ，否则 $TR_{v,B(i)} \neq 0$ 。为使生产-配送集成调度方案满足无等待调度，则 $Td_{B(j)}^e = \max(TR_{v,B(i)}, \max(Tc_{1,B(j)}, \dots, Tc_{p,B(j)}))$ 。证明完毕。

推论 4：通过使用理想出发时间的最晚时间节点 $IDT_{B(j)}^l$ ，以满足最大化的调整范围策略从而实现最小化的库存积压。

证明：对于巡回环来说，理想最佳出发时间可以在 $[IDT_{B(j)}^e, IDT_{B(j)}^l]$ 内的任一节点处获得，但是每个节点对于生产线来说却是不同的。受到生产线时间限制的影响，如果使用早于 $IDT_{B(j)}^l$ 的时间节点进行计算，则相应的批次出发时间 $Td_{B(j)}$ 也许可以提前。那么当按照逆向批次生产顺序执行批次的后向调度时， $Tc_{p,B(j)}$ 将会变小，即每个批次的后向调整范围将会变小。当 $Tc_{p,B(j)} < IDT_{B(j)}^e$ ，该批次将产生库存积压。因此，使用最晚时间节点 $IDT_{B(j)}^l$ 可以实现最小化的库存积压，并尽可能的满足无等待调度。证明完毕。

表 3-3 BSS-NWS 的举例数据

批次	生产时间		巡回环 执行时间	$[IDT_{B(j)}^e, IDT_{B(j)}^l]$
	生产线 1	生产线 2		
B1	30	21	36	[50,71]
B2	54	21	45	[100,130]
B3	50	30	20	[147,177]

如表 3-3 所示, 给出一个案例说明推论 4。其基于不同时间节点的对比甘特图见图 3-6 所示。子策略 1 是基于理想出发时间的最晚时间节点 $IDT_{B(j)}^l$ 。如果按照理想出发时间的最早时间节点 $IDT_{B(j)}^e$, 将会导致批次 $B(1,1)$ 和 $B(2,1)$ 的库存积压。显然, 时间节点设置越晚, 可调整范围越大, 造成库存积压的可能性就越小。

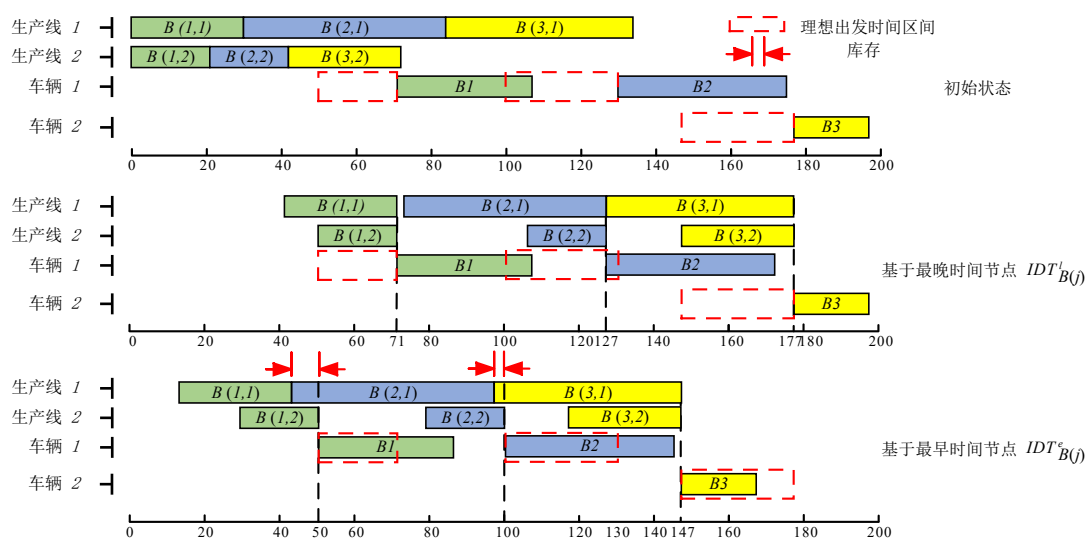


图 3-6 基于不同时间节点的对比甘特图

(2) 子策略 2: 最早的交付时间策略

由于子策略 1 使用 $IDT_{B(j)}^l$ 以获得最大化的调整范围, 所以生产-配送的总时间将会被拉长甚至可能超出规定的时间跨度, 不利于企业的产能节约以及高效生产的要求。因此, 在满足无等待调度以及不产生新的库存的基础上, 提出了如下的子策略 2 来提前批次的生产和交付。

子策略 2 的执行步骤: 按照生产顺序进行批次的前向调度。当生产线上相邻的两个批次之间存在空隙时间, 则意味着后一个批次的生产开始时间可以被提前, 并且提前后的批次生产完成时间可表示为: $Tc_{p,B(j)}^a = Tc_{p,B(j)} - (Ts_{p,B(j)} - Tc_{p,B(j-1)})$ 。因此, 批次的出发时间可以被提前 $Td_{B(j)}^a = \max(TR_{v,B(i)}, \max(Tc_{1,B(j)}^a, \dots, Tc_{P,B(j)}^a))$ 。通过与 $IDT_{B(j)}^e$ 比较后更新 $Td_{B(j)} = \max(Td_{B(j)}^a, IDT_{B(j)}^e)$ 。并根据无等待调度更新 $Tc_{p,B(j)} = \min(Ts_{p,B(j+1)}, Td_{B(j)})$ 。

推论 5: 在子策略 2 中, 每个批次的生产和交付时间节点只能向前调整或保持不变。因此, 生产-配送的总时间跨度将被缩短或保持不变。

证明: 根据批次生产顺序可得 $Ts_{p,B(j)} \geq Tc_{p,B(j-1)}$, $Tc_{p,B(j)}^a = Tc_{p,B(j)} - (Ts_{p,B(j)} - Tc_{p,B(j-1)})$ 即 $Tc_{p,B(j)}^a \leq Tc_{p,B(j)}$, 从而 $Td_{B(j)}^a = \max(TR_{v,B(i)}, \max(Tc_{1,B(j)}^a, \dots, Tc_{P,B(j)}^a)) \leq \max(TR_{v,B(i)}, \max(Tc_{1,B(j)}, \dots, Tc_{P,B(j)})) = Td_{B(j)}^e$ 。子策略 2 执行前, $Td_{B(j)} = \max\{Td_{B(j)}^e, IDT_{B(i)}^l\}$ 。执行后, $Td_{B(j)} = \max(Td_{B(j)}^a, IDT_{B(i)}^e)$ 。因此, $Tc_{p,B(j)}$ 和 $Td_{B(j)}$ 只能被向前调整或保持不变。证明完毕。

(3) 子策略 3: 库存成本与生产启动成本的权衡策略

在第一次执行完成子策略 2 后, 批次往往呈现不连续的生产。通过生产线上批次的提前生产可以消除相邻批次之间的空隙时间以降低生产启动成本, 但批次的提前生产有可能产生新的库存成本。故需要权衡库存成本与生产启动成本, 从而确定批次是否提前生产。另外, 当批次 $B(j)$ 进行提前生产时, $Tc_{p,B(j)}$ 与 $Td_{B(j)}$ 将不再对应。此时有必要去确定子策略 2 是否需要再次执行以更好地满足无等待调度。因此, 通过循环执行子策略 2 和 3, 直到没有批次可以提前生产, 即可得到最佳的生产-配送集成调度方案, 从而实现批次生产和巡回环出发的同时协调。

子策略 2 和 3 循环执行具体步骤如下: 按照批次生产顺序执行子策略 3 去判定生产线上的所有批次 $B(I, P)$ 是否进行提前生产。当消除空隙时间所产生的库存成本小于生产启动成本时, $B(I, P)$ 将提前生产。为了更好地满足无等待调度, 针对未被提前生产的批次执行子策略 2。最后, 通过循环执行子策略 2 和 3, 直到没有批量进行提前生产。推论 6 表明, 这个循环执行过程可以在多项式时间内完成。

本部分仍以表 3-3 所示案例说明上述执行过程。如图 3-7 所示, 通过执行子策略 2, 生产-配送集成调度方案进入阶段 2。然后, 按照批次生产顺序针对每一批次依次执行子策略 3。假设生产线 1 的生产启动成本大于批次 $B(1, 1)$ 提前生产所产生的库存成本, 则 $B(1, 1)$ 被提前生产。其余批次 $B(I, P)$ 均未提前进入阶段 3。值得注意的是, 在阶段 3 中批次 $B2$ 和 $B3$ 可以通过子策略 2 进一步优化, 从而批次 $B2$ 和 $B3$ 的生产完成及出发时间将分别提前至 100 和 150。然后再次执行子策略 3, 如此循环直至生产-配送集成调度方案不再改变。

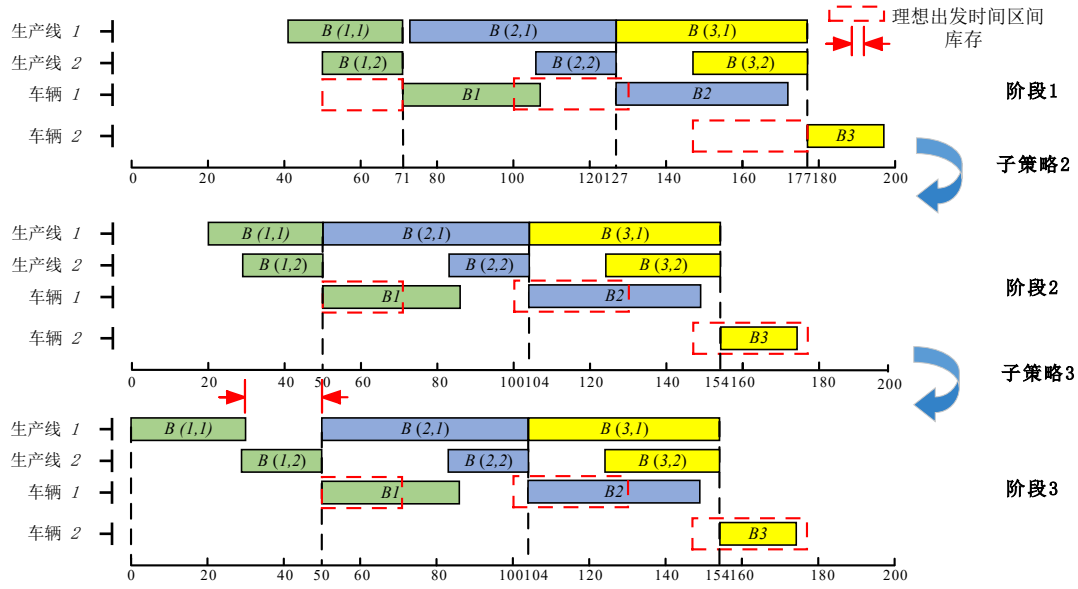


图 3-7 子策略 2、3 的执行

推论 6: 子策略 2、3 至多被循环执行 $I \cdot P$ 次后, 可以获得最终的生产-配送集成调度方案。其时间复杂度为 $O(I \cdot P)$ 。

证明: 生产阶段涉及 P 条生产线和 I 个批次, 采用子策略 3 按照批次生产顺序消除空隙时间。显然, 批次的时间节点将会向前调整或保持不变。根据推论 5 可知, 子策略 2 的执行也仅会使批次的时间节点将会向前调整或保持不变。并且在每次执行子策略 3 时, 首个提前生产的批次 $B(I, P)$ 的时间节点将不会再次发生改变。因此, 考虑最坏的可能, 如果每次执行子策略 3 时仅一个批次 $B(I, P)$ 被提前生产, 并且所有批次 $B(I, P)$ 均可被提前生产, 则子策略 3 至多被执行 $I \cdot P$ 次, 也即该循环最多执行 $I \cdot P$ 次即可获得最终的生产-配送集成调度方案。故其时间复杂度为 $O(I \cdot P)$, 可在多项式时间内获取。证明完毕。

3.4 IMOEa 中的自适应大邻域搜索策略

3.4.1 自适应大邻域搜索策略的基本组成

传统的 ALNS 具有多个破坏和修复算子, 相对独立的破坏和修复算子线性组合可以形成各种邻域结构, 其权重被分配给每个破坏和修复算子^[46]。由于传统的 ALNS 的破坏和修复算子的线性组合使得搜索过程具有随机性和盲目性, 从而可能减慢优化算法的运行速度。为了使搜索过程更具有针对性, 基于问题特征和编码形式设计了 4 个破坏-修复邻域结构, 并为每个邻域结构分配权重。其中

基于问题特征设计了 2 个邻域结构用于增强算法开发能力并提高局部搜索效率。基于编码形式设计了 2 个邻域结构用于增强算法探索能力并提高全局优化效果。

(1) 大邻域：基于不同规则设计的邻域结构将更改部分零售商订单的配送车辆序列号，从而生成新的编码个体。由于更改的零售商数量以及索引均不确定，所以邻域搜索的范围较大具备大邻域的特点。

(2) 自适应搜索机制：首先，为每个邻域结构分配一个权重，基于该权重通过轮盘赌选择机制进行邻域结构的选择。然后，在每次迭代中基于邻域结构的表现实时更新权重，进而实现自适应搜索机制。设 WNS_q 为安排给邻域结构 q 的权重。则选择执行该邻域结构的概率为 $WNS_q / \sum_q WNS_q$ 。

(3) 自适应权重调整。在算法启动时，为每个邻域结构分配相同的权重。然后在每次算法迭代中，基于 Pareto 前沿面上个体的更新比例进行自适应权重调整。将 Pareto 前沿面上新增的个体数量被设定为 Num_a ，去除的个体数量与原始总数的比值设定为 Rat 。根据公式(3-24)获取邻域结构的权重增加值，然后更新 WNS_q 。显然，当 $Rat > 0$ 时， Num_a 必大于 0。 Num_a 和 Rat 越大，说明所执行的邻域结构表现越好。因此， $\omega_1 < \omega_2 < \omega_3 < \omega_4 < \omega_5 < \omega_6$ 的设计是合理的。

$$\omega = \begin{cases} \omega_1, & Rat = 0 \text{ and } Num_a = 0 \\ \omega_2, & Rat = 0 \text{ and } Num_a > 0 \\ \omega_3, & 0 < Rat \leq 0.25 \\ \omega_4, & 0.25 < Rat \leq 0.5 \\ \omega_5, & 0.5 < Rat \leq 0.75 \\ \omega_6, & Rat > 0.75 \end{cases} \quad (3-24)$$

3.4.2 基于问题特征的邻域结构设计

为了说明邻域结构的执行过程，定义了以下符号：Pareto 前沿面个体集 $\xi = \{\xi_1, \dots, \xi_{num}\}$ 是自适应大邻域搜索的执行对象。 A_1, A_2 用于存储执行完邻域结构后获得的邻域个体解集。邻域搜索规模 ns 代表每个个体执行邻域搜索次数， NIS 代表邻域个体集合。

为了优化配送车辆的使用，避免同一车辆的连续执行交付任务。设计了邻域结构 1，见表 3-4 所示。针对 ξ 内的每个个体，如果相邻的批次由同一车辆执行交付，那么非首批次的订单容易产生库存积压和交付时间的延迟。因此，当相邻批次由同一车辆交付时，随机替换其中一个批次的配送车辆有很大的概率优化当前的解个体。其中使用了两种替换方法：整批替换和批次内订单的各自替换。

表 3-4 邻域结构 1：连续交付车辆的替换

邻域结构 1：连续交付车辆的替换

1. **输入：** $\xi = \{\xi_1, \dots, \xi_{num}\}$;
2. **输出：** 邻域个体集 NIS ;
3. $NIS = \emptyset$;
4. **For** $j \in \{1, \dots, num\}$ **do**
5. $A_1 = \emptyset, A_2 = \emptyset$;
6. **If** 个体 ξ_j 中存在相邻的批次由同一车辆交付 **then**
7. $A_1 \leftarrow$ 选择其中一个批次，并为该批次内的订单整体替换车辆;
8. $A_2 \leftarrow$ 选择其中一个批次，并为该批次内的每个订单分别替换车辆;
9. $NIS = [NIS, A_1, A_2]$;
10. **End if**
11. **End for**

表 3-5 邻域结构 2：更换车辆操作

邻域结构 2：更换车辆操作

1. **输入：** $\xi = \{\xi_1, \dots, \xi_{num}\}, C_v, BO_v^{B(i)}, BOR_{B(i)}, ns$;
2. **输出：** 邻域个体集 NIS ;
3. $C_v \leftarrow$ 车辆 v 的最大装载托盘数;
4. $BO_v^{B(i)} \leftarrow$ 由车辆 v 交付的批次 $B(i)$ 所占用的托盘数;
5. $BOR_{B(i)} \leftarrow$ 批次的实际占用率;
6. $NIS = \emptyset$;
7. **For** $j \in \{1, \dots, num\}$ **do**
8. 根据 ξ_j 获得所有批次 $B(i)$ 的订单组成;
9. $BOR_{B(i)} = BO_v^{B(i)} / C_v$; $k = 1$;
10. **While** $k \leq ns$ **do**
11. $A_1 = \emptyset, A_2 = \emptyset$;
12. 使用基于 $BOR_{B(i)}$ 的轮盘赌机制选择更换车辆操作的批次;
13. $A_1 \leftarrow$ 为所选批次 $B(i)$ 内的订单整体更换交付车辆;
14. $A_2 \leftarrow$ 为所选批次 $B(i)$ 内的每个订单分别更换交付车辆;
15. $NIS = [NIS, A_1, A_2]$;
16. $k = k + 1$;
17. **End while**
18. **End for**

由于采用一组数量受限的多行程异构车辆进行配送，所以根据批次订单容量大小选择合适的配送车辆至关重要。为了提高配送车辆的装载率、降低车辆配送成本设计了邻域结构 2，见表 3-5 所示。在所有的巡回环中，配送车辆的实际装载率越低的巡回环，该巡回环中零售商的订单被选中更换配送车辆的概率就越大。该邻域结构仍采用两种替换方法：整批更换和批次内订单的各自更换。

3.4.3 基于编码形式的邻域结构设计

为了提高当前 Pareto 前沿面个体跳出局部最优的可能性，设计了邻域结构 3，见表 3-6 所示。该邻域结构的主要步骤：随机选择个体的部分编码位置执行分离-凝聚操作，并针对 Pareto 前沿面个体集在所选的编码位置上找到出现频率最低和最高的车辆序列号。分离操作是更改所选编码位置为出现频率最低的车辆序列号。凝聚操作则是更改所选编码位置为出现频率最高的车辆序列号。如果车辆序列号出现的频率相同，则随机选择一个车辆序列号进行更改。

表 3-6 邻域结构 3：分离-凝聚操作

邻域结构 3：分离-凝聚操作	
1. 输入：	$\xi = \{\xi_1, \dots, \xi_{num}\}, ns$;
2. 输出：	邻域个体集 NIS ;
3.	$NIS = \emptyset$;
4. For	$j \in \{1, \dots, num\}$ do
5.	$k = 1$;
6.	While $k \leq ns$ do
7.	$A_1 = \emptyset, A_2 = \emptyset$;
8.	随机选择 ξ_j 的部分编码位置，用于更改车辆序列号;
9.	$A_1 \leftarrow$ 针对选择的编码位置，执行分离操作生成邻域个体;
10.	$A_2 \leftarrow$ 针对选择的编码位置，执行凝聚操作生成邻域个体;
11.	$NIS = [NIS, A_1, A_2]$;
12.	$k = k + 1$;
13.	End while
14.	End for

邻域结构 4 通过有意识地调整 Pareto 前沿面个体分布来扩展当前的 Pareto 前沿面。根据 Pareto 前沿面上个体的拥挤度距离，将当前 Pareto 前沿面上最拥挤的个体设为排斥个体 ξ_R (Repulsive individual)，最不拥挤的个体设为吸引个体

ξ_A (Attractive individual)。对于当前 Pareto 前沿面上的每个个体 ξ_j ，让其在接近 ξ_A 和远离 ξ_R 的方向上进行局部搜索。表 3-7 给出了邻域结构 4 的具体执行过程。

表 3-7 邻域结构 4：吸引-排斥操作

邻域结构 4：吸引-排斥操作	
1.	输入： $\xi = \{\xi_1, \dots, \xi_{num}\}, ns$;
2.	输出： 邻域个体集 NIS ;
3.	$NIS = \emptyset$
4.	For $j \in \{1, \dots, num\}$ do
5.	$k = 1$;
6.	While $k \leq ns$ do
7.	$A_1 = \emptyset, A_2 = \emptyset$;
8.	对照 ξ_j 和 ξ_A 寻找具有不同车辆序列号的编码位置；
9.	$A_1 \leftarrow$ 在所获得编码位置中，随机更改 ξ_j 的部分编码位置的车辆序列号，使其与 ξ_A 在该编码位置上的车辆序列号一致；
10.	对照 ξ_j 和 ξ_R 寻找具有相同车辆序列号的编码位置；
11.	$A_2 \leftarrow$ 在所获得编码位置中，随机更改 ξ_j 的部分编码位置的车辆序列号；
12.	$NIS = [NIS, A_1, A_2]$;
13.	$k = k + 1$;
14.	End while
15.	End for

3.5 数值实验与结果分析

仿真实验通过 MATLAB2020a 进行编程，并在具备 Intel Core i5-1035G1 - CPU 1.00GHz 和 16.0GB 的内存上运行算法。

3.5.1 测试实例的生成

由于本章所研究问题涉及生产和配送的集成，并且考虑了流水生产线及软时间窗交付限制等实际特征，目前还没有可以进行性能比较的基准数据集。因此，本章提出了一个包含 9 个小规模、9 个中等规模和 12 个大规模实例的数据集，以进行算法性能对比。测试实例(P - V - n - D)由生产线数量 $P \in \{3, 4, 5\}$ 、车辆数量 $V \in \{4, 5, 6\}$ 、零售商数量 $n \in \{20, 25, 30\}$ 、零售商产品需求 $D \in \{1, 2, 3\}$ 组合形成。

测试实例数据的具体生成方式如下：首先，一部分数据(例如： Pr_p , Pc_p ,

Fs_p , O_p , Hc , C_v , $T_{k,l}$)来自于一个实际的罐装饮料生产企业。该企业为超过 30 个零售商提供各种类型、规格的饮料产品。其工厂内包含 5 条生产线, 6 辆用于执行配送任务的异构车辆, 其车辆容量从 24~33 个托盘不等。如图 3-8 所示, 通过定位真实工厂和零售商站点的经纬度来计算每个站点之间的旅途距离, 然后根据配送车辆的平均速度得到旅途时间矩阵。此外, 考虑到企业数据的保密性, 另一部分数据($D_{k,p}$, t_p , $[e_k, l_k]$)通过仿真获得, 但这并不影响研究结论的正确性。如表 3-8 所示, 定义常量 $\alpha=1000$, 给出了测试实例关键参数生成的具体方法。零售商对不同产品的需求量 $D_{k,p}$ 遵循正态分布。生产节拍 t_p 与软时间窗下限 e_k 服从均匀分布。并给出了时间窗口宽度($l_k - e_k$)、单位产品占用托盘数量(O_p)及车辆的托盘装载数量(C_v)的离散取值范围。该测试实例数据集可以从 <https://www.huangm.cn/zip/IPSVRP-BM-STW-Dateset.zip> 中下载。

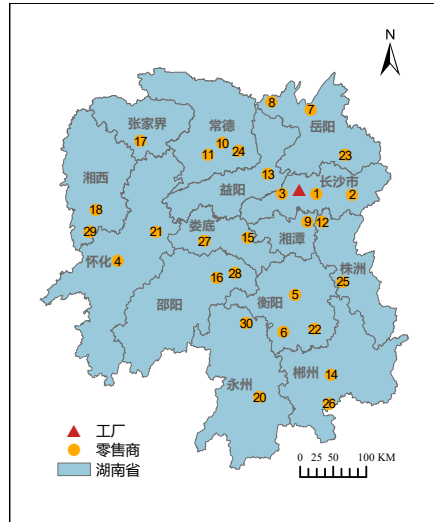


图 3-8 地域位置图

表 3-8 关键参数的生成方法

参数	生成方法
$D_{k,p}$	$\sim N(\alpha, \alpha^2)$
t_p	$\sim U(1/\alpha, 4/\alpha)$
$[e_k, l_k]$	$e_k \sim U(15, 100)$ $l_k - e_k \in \{0.5; 1; 1.5; 2; 2.5; 3\}$
O_p	$\in \{1/\alpha; 5/4\alpha; 5/3\alpha; 2/\alpha\}$
C_v	$\in \{24; 26; 28; 29; 31; 33\}$

3.5.2 对比算法的选取及多目标评价指标

在本章中，将 IMOEa 与该领域相关文献中提出的 NSGA-II^[20]、MODE^[47]、HGAPSO1^[48]、HGAPSO2^[32]进行算法比较。此外，通过本章所提出的 ALNS 对 MODE 进行改进形成 IMODE。为了公平对比，所有算法均采用本章所提出的编、解码规则。IMODE 与 IMOEa 的主要区别在于 IMODE 采用了差分进化中特有的交叉操作以及基于贪婪准则的种群个体选择方法。

为了全面的评价算法性能，本文分别从基数、扩散性、收敛性和分布性四个维度选取了 4 个评价指标，即错误率^[49](Error ratio, ER)、分布度量^[50](Distribution metric, DM)、反世代距离^[51](Inverted generational distance, IGD)、超体积^[52](Hypervolume, HV)，见公式(3-25)-(3-28)。其中 X_c , X_r 分别代表当前 Pareto 前沿面和参考 Pareto 前沿面， $|X_c|$ 表示 X_c 中解的个数。

指标 ER 评估算法的基数，其表征算法所求解集的错误率。ER 指标值越低，算法基数评价越好。指标 DM 用于评估扩散性和分布性，较小的 DM 表示算法所获得的 Pareto 前沿面上的个体是否更均匀、更广泛。指标 IGD 和 HV 采用了两种不同的方式评估算法的收敛性和分布性。IGD 需要真正的 Pareto 前沿面，其往往由所有算法获得的参考 Pareto 前沿面来表示。不同的是，HV 仅需要确定一个参考点 r ，将其设置为每个目标的最差点即可进行指标值计算。

$$ER(X_c) = \frac{|\{x \in X_c | x \notin X_r\}|}{|X_c|} \quad (3-25)$$

$$DM(X_c) = \frac{1}{|X_c|} \sum_{i=1}^m \left(\frac{\sigma_i}{\mu_i} \right) \left(\frac{|x_i^U - x_i^N|}{\max_{x \in X_c} x_i - \min_{x \in X_c} x_i} \right) \quad (3-26)$$

$$\sigma_i = \frac{1}{|X_c| - 2} \sum_{j=1}^{|X_c|-1} (d_i^j - \bar{d}_i)^2 \quad \mu_i = \frac{1}{|X_c| - 1} \sum_{j=1}^{|X_c|-1} d_i^j$$

$$IGD(X_c, X_r) = \frac{1}{|X_c|} \left(\sum_{x^2 \in X_r} \left(\min_{x^1 \in X_c} \|x^1 - x^2\| \right)^m \right)^{\frac{1}{m}} \quad (3-27)$$

$$HV(X_c; r) = \lambda \left(\bigcup_{x \in X_c} [x, r] \right) \quad (3-28)$$

3.5.3 算法参数设计

IMOEa 有四个重要参数：种群规模 N 、交叉概率 P_c 、突变概率 P_m 和邻域搜索规模 ns 。采用正交实验设计（Design of experiments, DOE）进行算法参数设计，其中各参数水平见表 3-9 所示。采用表 3-10 所示的正交表 $L_9(3^4)$ ，并对正交表中的每个组合独立运行算法 10 次，以 HV 的平均值作为评判结果。图 3-9 显示了参数主效应图，得到 IMOEa 的最佳参数值设置为： $N=200$ 、 $P_c=0.7$ 、 $P_m=0.1$ 、 $ns=8$ 。同理可得，对其余算法进行 DOE，从而使其参数设置为最佳。

表 3-9 参数水平表

参数水平	N	P_c	P_m	ns
1	200	0.7	0.1	5
2	250	0.8	0.2	8
3	300	0.9	0.3	10

表 3-10 正交表 $L_9(3^4)$

序号	参数水平				HV
	N	P_c	P_m	ns	
1	1	1	1	1	0.791886
2	1	2	2	2	0.785481
3	1	3	3	3	0.775823
4	2	1	2	3	0.786450
5	2	2	3	1	0.761023
6	2	3	1	2	0.794697
7	3	1	3	2	0.784757
8	3	2	1	3	0.791206
9	3	3	2	1	0.746733

主效应图

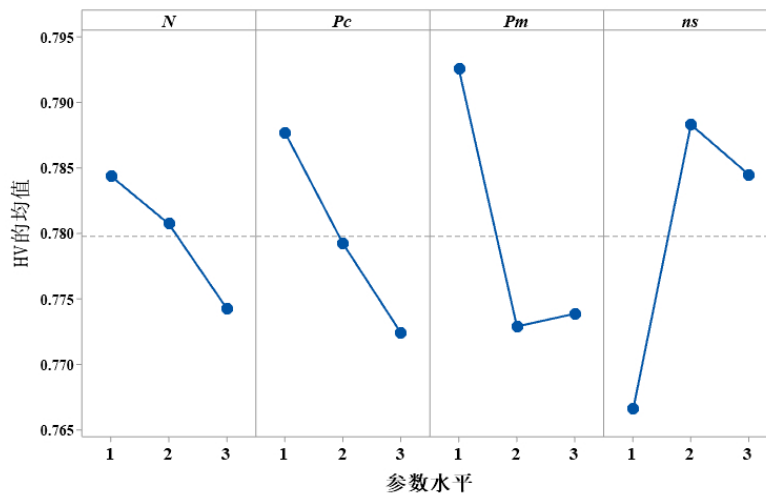


图 3-9 主效应图

3.5.4 对比实验结果与分析

为了验证 IMOEa 的算法性能, 在 30 个测试实例上进行了实验。为了避免偶然性, 每个算法均独立运行了 10 次。算法实验对比结果展示在表 3-11 中。对于 $f1$ 而言, 最佳值 (Best) 及平均值 (Avg) 越大越好, 对于 $f2$ 则正好相反。在获得最优表现的数量上 IMOEa 超过了十分之七, 其次是 IMODE, 其余算法获得的最优表现很少。可以得出 ALNS 可以增强算法的寻优能力。并且 IMOEa 在每个测试实例上获得了至少两个项目 (最佳值、平均值) 的最优表现, 故 IMOEa 在算法结果上对比其他算法具有明显优势。

表 3-11 算法实验对比结果

测试实例	目标	IMOEa		NSGA-II		MODE		IMODE		HGAPSO1		HGAPSO2	
		Best	Avg	Best	Avg	Best	Avg	Best	Avg	Best	Avg	Best	Avg
3-4-20-1	$f1$	29.6729	27.5221	29.1091	27.1434	28.5159	26.6016	29.6729	27.4502	28.9297	26.7667	29.6534	26.8900
	$f2$	0.3339	0.4883	0.3343	0.4826	0.3684	0.5065	0.3339	0.4813	0.3428	0.5063	0.3339	0.5027
3-4-20-2	$f1$	32.5643	31.2166	32.2198	31.0335	31.4319	30.3420	32.4362	31.1053	32.0945	30.6434	32.4633	31.1439
	$f2$	1.8943	2.1643	1.8943	2.1595	1.9903	2.1596	1.8943	2.1389	1.9268	2.1924	1.8943	2.1658
3-4-20-3	$f1$	25.1696	23.5569	24.9985	23.0484	24.1724	22.5124	25.1156	23.4790	24.7878	22.8749	25.1415	23.0306
	$f2$	0.5836	0.7700	0.5836	0.7471	0.6124	0.7654	0.5836	0.7788	0.5871	0.8061	0.5853	0.7571
4-4-20-1	$f1$	34.9078	33.6819	34.5783	33.2209	33.9352	32.7618	34.8316	33.5814	34.4258	33.0489	34.5838	33.3273
	$f2$	1.9725	2.2854	1.9725	2.2383	2.0229	2.2407	1.9725	2.2662	1.9898	2.2811	1.9725	2.2211
4-4-20-2	$f1$	35.2418	33.9473	34.9576	33.6765	34.3227	33.1747	35.2057	33.9644	34.8034	33.4932	34.9869	33.7916
	$f2$	2.0420	2.3224	2.0420	2.3053	2.0845	2.3327	2.0420	2.3264	2.0568	2.3525	2.0420	2.3101
4-4-20-3	$f1$	35.5526	34.4982	35.3585	34.3406	34.6286	33.6590	35.5155	34.4221	35.1420	33.8565	35.5362	34.3073
	$f2$	2.1063	2.3857	2.1119	2.4011	2.1457	2.3829	2.1063	2.3943	2.1139	2.4089	2.1063	2.3699
5-5-20-1	$f1$	34.5935	33.2449	34.1981	32.7233	33.5833	32.2086	34.4445	33.0348	34.0495	32.5119	34.5104	32.8047
	$f2$	2.0283	2.2753	2.0283	2.2393	2.0642	2.2713	2.0283	2.2498	2.0622	2.3196	2.0288	2.2380
5-5-20-2	$f1$	33.9991	32.8906	33.6327	32.4290	32.9764	31.7918	34.0312	32.7801	33.3907	32.1020	33.7944	32.3848
	$f2$	1.9178	2.1511	1.9267	2.1402	1.9685	2.1551	1.9205	2.1536	1.9467	2.2122	1.9191	2.1136
5-5-20-3	$f1$	31.3037	29.4783	30.9489	28.7944	30.0595	28.4277	31.3439	29.4224	30.5620	28.4085	30.8598	28.6934
	$f2$	0.0059	0.0847	0.0131	0.0876	0.0229	0.1081	0.0059	0.0870	0.0140	0.1035	0.0098	0.0814
3-5-25-1	$f1$	30.1163	28.1717	29.4742	27.3813	29.0096	26.9659	30.1246	28.0138	28.9846	27.0944	29.6254	27.5383
	$f2$	0.7945	0.9496	0.8405	0.9852	0.8601	0.9996	0.8328	0.9652	0.8274	0.9978	0.8278	0.9891
3-5-25-2	$f1$	39.7686	38.3174	39.4166	37.9894	38.6299	37.5677	39.8230	38.2905	39.1533	37.6218	39.7489	38.0734
	$f2$	3.1101	3.3848	3.1957	3.4807	3.3116	3.5187	3.0818	3.3922	3.2278	3.5282	3.1578	3.4162
3-5-25-3	$f1$	40.3849	38.7948	39.8260	38.4426	39.0711	37.9111	40.3705	38.8572	39.5312	38.0468	40.1770	38.5422
	$f2$	3.0864	3.3513	3.1365	3.3926	3.2588	3.4709	3.0961	3.3767	3.1915	3.4625	3.1063	3.4182
4-5-25-1	$f1$	44.0469	42.3952	43.4300	42.0737	42.6712	41.5031	44.0469	42.2953	43.0777	41.8443	43.7310	41.9022
	$f2$	3.5382	3.8276	3.6213	3.8480	3.7781	3.9728	3.5332	3.8158	3.6594	3.9870	3.5889	3.8545
4-5-25-2	$f1$	42.9189	41.3423	42.6204	41.0363	41.7669	40.4078	42.8997	41.2491	42.1210	40.7587	42.4529	40.9763
	$f2$	3.3168	3.6057	3.3855	3.6486	3.5680	3.7716	3.3257	3.6003	3.4330	3.7265	3.3952	3.6710
4-5-25-3	$f1$	42.8171	40.8576	42.3127	40.8418	41.5764	40.1901	42.7211	40.9102	42.0368	40.6100	42.5117	40.6675
	$f2$	3.2558	3.5203	3.3125	3.5503	3.4466	3.6554	3.2648	3.5077	3.3749	3.6308	3.2989	3.5793
5-5-25-1	$f1$	42.7096	41.0969	42.0447	40.7334	41.4395	40.2043	42.6211	41.0837	41.9860	40.4535	42.5561	40.6542
	$f2$	3.4645	3.7443	3.5877	3.8177	3.7092	3.9354	3.4984	3.7791	3.5461	3.8617	3.5062	3.8398
5-5-25-2	$f1$	43.5454	41.8226	43.3003	41.7989	42.2232	41.0252	43.5881	41.7592	42.9561	41.5468	43.5629	41.5828
	$f2$	3.7470	4.0052	3.8861	4.0999	3.9676	4.1916	3.7969	4.0194	3.8842	4.1942	3.8056	4.0845
5-5-25-3	$f1$	43.1884	41.5273	42.6723	41.3160	41.9015	40.8394	43.2125	41.5343	42.4696	41.1172	43.1191	41.1213
	$f2$	3.7210	3.9818	3.7761	4.0768	3.9622	4.1937	3.7311	4.0071	3.8735	4.1373	3.7702	4.0481
3-5-30-1	$f1$	43.5061	40.9025	42.9090	40.8042	41.8997	40.2908	43.4590	40.8288	42.5200	40.2801	43.0482	40.4234
	$f2$	2.4250	2.6519	2.4705	2.6879	2.6007	2.8645	2.4416	2.6574	2.4614	2.7791	2.4324	2.7498
3-5-30-2	$f1$	43.7864	41.1395	43.1416	41.1638	41.9769	40.3063	43.7545	41.0773	42.5164	40.3158	43.1267	40.6676
	$f2$	2.3478	2.5752	2.3687	2.6556	2.4363	2.7486	2.3616	2.5812	2.3945	2.6829	2.3721	2.6345
3-5-30-3	$f1$	42.9977	40.4843	42.4661	40.3720	41.0623	39.6057	43.0182	40.5757	41.9744	39.6933	42.2426	39.8379

	$f2$	2.2986	2.5090	2.3361	2.6051	2.4539	2.7045	2.3045	2.5450	2.3422	2.6353	2.3328	2.5676
4-6-30-1	$f1$	47.8534	45.2606	46.7782	44.8392	45.6284	44.1060	47.5255	45.2173	46.4822	44.1879	47.0085	44.5671
	$f2$	2.6741	2.8720	2.7088	2.8655	2.7910	3.0088	2.6741	2.8779	2.7207	2.9807	2.6962	2.8807
4-6-30-2	$f1$	47.5617	45.0799	46.7534	44.5923	45.4496	43.7100	47.5664	44.8650	45.9606	43.9255	46.7279	44.2242
	$f2$	2.6392	2.8433	2.6721	2.8721	2.7697	2.9841	2.6473	2.8690	2.6981	2.9484	2.6583	2.8671
4-6-30-3	$f1$	46.1832	43.8195	45.5572	42.8062	44.5334	42.7415	46.0415	43.6667	44.8325	42.5781	45.5599	42.9474
	$f2$	2.4691	2.6507	2.4996	2.6750	2.5783	2.8478	2.4735	2.6768	2.5254	2.7605	2.5176	2.6939
5-5-30-1	$f1$	47.0444	44.1049	46.5381	43.9812	45.4380	43.4158	46.8763	44.3385	45.9171	43.5402	46.3065	43.2833
	$f2$	2.6945	2.9072	2.7260	3.0181	2.8223	3.0964	2.7007	2.9568	2.7248	3.0539	2.7098	2.9683
5-5-30-2	$f1$	46.7432	44.0992	46.4529	43.7507	45.1902	42.9667	46.7192	44.1511	45.6571	43.2887	45.9791	43.1232
	$f2$	2.5520	2.7933	2.5838	2.8419	2.6833	2.9576	2.5694	2.8182	2.6160	2.9221	2.6019	2.8355
5-5-30-3	$f1$	46.7083	43.9373	45.8213	43.6349	44.9398	43.1219	46.5839	43.7823	45.6665	43.2142	46.0585	43.1493
	$f2$	2.6400	2.8737	2.6761	2.9328	2.8040	3.0658	2.6537	2.8917	2.7223	3.0166	2.6784	2.9315
5-6-30-1	$f1$	47.5447	44.8472	46.8446	44.4569	45.5618	43.7696	47.4755	45.1421	46.3904	43.7344	46.5109	43.8589
	$f2$	2.6283	2.8218	2.6503	2.8923	2.7596	3.0132	2.6359	2.8974	2.6874	2.9413	2.6638	2.8732
5-6-30-2	$f1$	48.0325	45.2735	47.4602	44.9852	45.8039	44.2493	47.7309	45.1991	46.6254	44.3017	47.1036	44.3795
	$f2$	2.8024	3.0008	2.8110	3.0628	2.9326	3.2114	2.8100	3.0339	2.8608	3.1599	2.8217	3.0715
5-6-30-3	$f1$	47.9307	45.1380	47.1306	44.7287	45.9534	44.1297	47.7149	45.2436	46.6101	44.3342	47.0278	44.4598
	$f2$	2.7291	2.9402	2.7536	3.0030	2.8766	3.1246	2.7391	2.9640	2.7962	3.1011	2.7340	2.9997

*阴影粗体字表示最优表现，表中数值均采用科学计数法($\times 10^4$)。

如图 3-10 所示，展现了 6 种算法在解决 4 个不同规模测试实例时获得的 Pareto 前沿面。可以看出 IMOEa 所获得的 Pareto 前沿面具有明显优势，其更加逼近理想的 Pareto 前沿面。表 3-12-表 3-13 展示了 6 种算法在指标 ER、DM、IGD 和 HV 上的比较结果。在 4 个指标上，IMOEa 的性能均优于其他 5 种算法。其中，当实例规模较小时 IMODE、HGAPSO1 和 HGAPSO2 有时还可以在某个指标上获得最优表现，但随着实例规模的增大，IMOEa 的优势越来越明显。

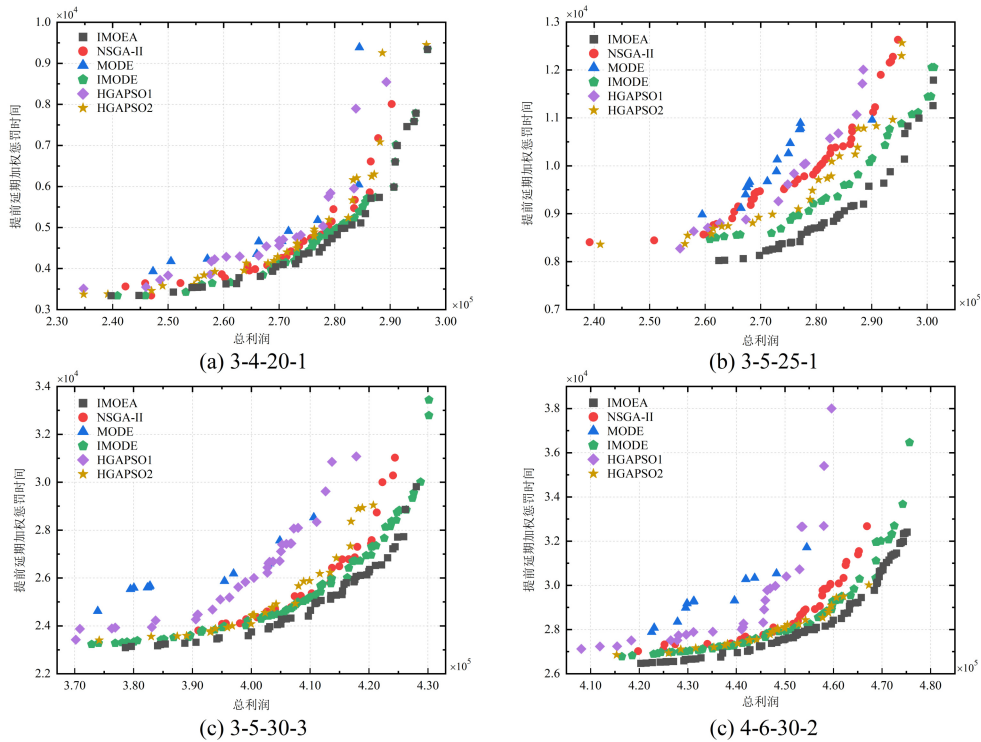


图 3-10 Pareto 前沿面对比图

表 3-12 各算法的指标结果（指标 ER、DM）

测试 实例	IMOEa		NSGA-II		MODE		IMODE		HGAPSO1		HGAPSO2	
	ER	DM	ER	DM	ER	DM	ER	DM	ER	DM	ER	DM
3-4-20-1	0.813	176.953	1.000	408.925	1.000	1012.409	0.890	235.798	1.000	478.656	1.000	277.259
3-4-20-2	0.753	155.178	0.961	481.209	1.000	1174.424	0.904	212.584	1.000	427.971	0.930	215.868
3-4-20-3	0.697	95.565	1.000	211.224	1.000	727.748	0.894	140.194	1.000	202.033	1.000	83.425
4-4-20-1	0.758	147.508	0.963	287.447	1.000	1625.489	0.872	240.508	1.000	289.350	0.976	234.979
4-4-20-2	0.796	304.594	1.000	502.316	1.000	1369.500	0.853	307.313	1.000	261.630	0.961	441.164
4-4-20-3	0.761	224.880	0.978	648.066	1.000	754.734	0.891	281.417	1.000	722.622	0.965	379.974
5-5-20-1	0.790	124.744	0.992	198.338	1.000	1138.073	0.972	130.921	1.000	330.818	0.993	237.228
5-5-20-2	0.704	193.492	1.000	322.972	1.000	1256.539	0.938	256.788	1.000	249.178	0.994	320.137
5-5-20-3	0.888	146.166	1.000	419.891	1.000	1545.250	0.903	185.117	1.000	548.521	1.000	573.111
3-5-25-1	0.879	111.588	1.000	314.359	1.000	1167.475	0.984	186.765	1.000	284.015	1.000	492.729
3-5-25-2	0.904	72.663	1.000	302.319	1.000	2380.844	0.925	98.211	1.000	686.965	1.000	254.911
3-5-25-3	0.885	88.100	1.000	434.573	1.000	1816.810	0.915	96.308	1.000	525.348	1.000	223.109
4-5-25-1	0.882	130.829	1.000	170.641	1.000	2371.371	0.927	58.451	1.000	587.368	1.000	264.839
4-5-25-2	0.907	104.273	1.000	260.247	1.000	2656.531	0.993	128.680	1.000	354.563	1.000	344.065
4-5-25-3	0.931	88.040	1.000	212.853	1.000	2226.408	0.922	104.706	1.000	461.341	1.000	349.414
5-5-25-1	0.811	106.613	1.000	297.388	1.000	2036.566	0.935	132.558	1.000	349.160	1.000	247.862
5-5-25-2	0.877	119.144	1.000	419.506	1.000	65535.000	0.940	100.982	1.000	340.005	1.000	325.167
5-5-25-3	0.925	130.610	1.000	220.867	1.000	2559.073	0.909	136.095	1.000	493.731	1.000	281.568
3-5-30-1	0.897	79.876	1.000	159.386	1.000	1237.406	0.948	93.312	1.000	167.392	0.998	303.170
3-5-30-2	0.906	48.233	1.000	263.667	1.000	2513.031	0.952	98.749	1.000	327.240	1.000	234.979
3-5-30-3	0.889	67.976	1.000	185.819	1.000	1510.680	0.955	103.358	1.000	393.756	1.000	221.944
4-6-30-1	0.908	57.929	1.000	232.422	1.000	2667.464	0.963	71.596	1.000	256.856	1.000	458.881
4-6-30-2	0.860	44.443	1.000	183.751	1.000	2361.161	0.996	294.517	1.000	283.086	1.000	355.926
4-6-30-3	0.887	55.202	1.000	139.925	1.000	1186.748	0.983	86.569	1.000	277.500	1.000	292.920
5-5-30-1	0.853	65.689	1.000	193.731	1.000	1863.349	0.977	111.173	1.000	313.825	1.000	320.307
5-5-30-2	0.891	49.909	1.000	195.271	1.000	2546.834	0.961	102.293	1.000	245.105	1.000	289.495
5-5-30-3	0.874	46.170	1.000	159.133	1.000	2680.882	0.968	87.768	1.000	258.836	1.000	274.963
5-6-30-1	0.876	47.744	1.000	160.720	1.000	2329.350	0.984	81.334	1.000	258.443	1.000	341.762
5-6-30-2	0.850	43.333	1.000	158.242	1.000	1508.620	0.996	98.344	1.000	339.296	1.000	339.869
5-6-30-3	0.888	44.628	1.000	166.338	1.000	1631.130	0.994	97.923	1.000	290.847	1.000	271.301

*阴影粗体字表示最优表现。

表 3-13 各算法的指标结果（指标 IGD、HV）

测试 实例	IMOEa		NSGA-II		MODE		IMODE		HGAPSO1		HGAPSO2	
	IGD	HV	IGD	HV	IGD	HV	IGD	HV	IGD	HV	IGD	HV
3-4-20-1	532.072	0.834	2204.650	0.743	4709.996	0.655	675.967	0.827	2659.055	0.721	1529.340	0.780
3-4-20-2	570.443	0.772	1733.405	0.711	5120.954	0.552	929.603	0.758	2737.517	0.646	987.322	0.748
3-4-20-3	418.533	0.817	1604.668	0.754	5068.178	0.618	740.428	0.803	2844.540	0.695	1169.140	0.778
4-4-20-1	555.542	0.783	1554.735	0.715	4905.358	0.565	730.764	0.772	2270.579	0.666	1213.191	0.738
4-4-20-2	815.066	0.788	2367.274	0.703	5582.110	0.565	1007.180	0.770	2757.711	0.673	1669.226	0.742
4-4-20-3	706.247	0.797	2046.905	0.738	5650.289	0.580	910.848	0.788	2609.101	0.687	1511.968	0.762
5-5-20-1	722.923	0.781	2019.750	0.688	5069.829	0.557	939.417	0.757	3171.879	0.622	1434.771	0.728
5-5-20-2	829.892	0.812	2373.508	0.723	6537.529	0.573	1187.822	0.788	4019.889	0.649	1693.064	0.759
5-5-20-3	763.913	0.827	1679.411	0.766	5209.076	0.666	775.504	0.828	2649.742	0.722	1770.440	0.772
3-5-25-1	1007.801	0.808	2765.153	0.674	6332.107	0.561	1700.845	0.759	3772.852	0.629	2342.822	0.707
3-5-25-2	841.104	0.782	3132.511	0.653	6829.452	0.511	1121.620	0.763	4551.726	0.587	1856.008	0.721
3-5-25-3	908.005	0.774	3187.893	0.644	6745.757	0.501	973.519	0.767	4767.538	0.568	2027.207	0.701
4-5-25-1	984.073	0.749	3138.001	0.646	8019.154	0.464	1055.831	0.754	5693.210	0.546	2474.548	0.664
4-5-25-2	1026.912	0.777	2849.824	0.676	7495.419	0.491	1312.834	0.760	4091.688	0.608	2526.806	0.689
4-5-25-3	952.843	0.768	2665.808	0.684	8236.468	0.511	1164.707	0.766	4186.207	0.619	2383.204	0.691
5-5-25-1	1080.870	0.755	3149.937	0.636	8128.502	0.461	1274.571	0.733	4307.272	0.584	2950.091	0.650
5-5-25-2	1043.166	0.780	3250.632	0.676	7875.435	0.492	1168.243	0.768	4494.356	0.608	2443.697	0.702
5-5-25-3	1116.070	0.769	3503.074	0.653	7184.169	0.478	1460.919	0.763	4042.472	0.595	2376.074	0.684
3-5-30-1	738.274	0.807	3023.203	0.715	10183.844	0.529	1047.356	0.787	5011.948	0.641	3317.072	0.695
3-5-30-2	660.972	0.808	2904.652	0.713	7705.875	0.559	1117.831	0.787	4507.999	0.650	2867.596	0.710
3-5-30-3	714.107	0.807	3173.211	0.707	8836.837	0.539	1190.720	0.796	5003.248	0.645	3007.602	0.705
4-6-30-1	713.236	0.827	3803.457	0.705	9646.208	0.556	1097.251	0.805	6049.145	0.645	3124.924	0.727
4-6-30-2	980.297	0.831	3533.873	0.716	10567.616	0.542	1857.136	0.786	6496.831	0.631	3359.326	0.720
4-6-30-3	689.569	0.829	2964.170	0.712	8890.875	0.548	1153.726	0.798	5414.265	0.634	2934.974	0.714
5-5-30-1	702.000	0.822	2942.195	0.727	7944.540	0.583	1382.595	0.796	4093.221	0.678	3235.909	0.708
5-5-30-2	590.214	0.827	2959.399	0.729	8529.088	0.567	1066.465	0.808	4941.014	0.663	3535.764	0.704
5-5-30-3	960.989	0.796	3012.551	0.703	8021.048	0.551	1469.603	0.764	4386.007	0.646	3435.849	0.682
5-6-30-1	696.718	0.812	3120.787	0.697	9212.530	0.521	1391.772	0.790	4966.893	0.618	3735.387	0.670
5-6-30-2	775.771	0.821	2925.011	0.725	9042.253	0.546	1362.648	0.781	5500.381	0.639	3631.297	0.691
5-6-30-3	966.183	0.821	3384.593	0.706	8405.218	0.549	1649.416	0.782	5194.568	0.639	3511.041	0.700

*阴影粗体字表示最优表现。

为了进一步说明算法之间的优势差异具有统计学意义。首先,进行 Anderson-Darling 测试检查各总体分布的正态性。由于大多数总体数据 P 值 <0.05 , 表明不满足正态性假设。其次, 针对 4 个评价指标进行多独立总体的 Kruskal-Wallis 检

验, 得到的 P 值 <0.0001 , 因此有足够的统计证据拒绝所有总体在统计上相似的零假设, 故算法之间存在统计学差异。其中平均秩值见表 3-14, 即在基于秩排序的比较中, IMOEa 明显优于其他算法。最后, 采用 Mann-Whitney 检验, 将显著性水平设置为 0.05 去判定 IMOEa 与各对比算法的差异是否具有显著性。相应的显著性对比结果列于表 3-15。结果表明, 在指标 ER、IGD 和 HV 上, IMOEa 与对比算法之间的 P 值 <0.05 , 即存在统计学上的显著差异。针对指标 DM, IMOEa、与 NSGA-II、IMODE、HGAPSO2 并不构成统计学上的显著差异。从而得出如下结论: 在算法对比中, IMOEa 与所有对比算法之间均存在统计学上的显著差异。因此, IMOEa 在基数、扩散、分布和收敛方面具有统计学上的显著优势。

表 3-14 Kruskal-Wallis 检验的平均秩

算法	ER	DM	IGD	HV
IMOEa	18.167	28.867	19.700	156.233
NSGA-II	116.833	88.000	98.067	84.900
MODE	126.000	165.500	163.800	18.233
IMODE	45.733	44.833	43.800	141.700
HGAPSO1	126.000	113.267	128.600	48.767
HGAPSO2	110.267	102.533	89.067	93.167

*阴影粗体字表示最优表现

表 3-15 Mann-Whitney 检验 P 值表

P 值 (\(:\)意味无法测试)	P 值 (ER)	P 值 (DM)	P 值 (IGD)	P 值 (HV)
IMOEa- NSGA-II	2.056E-12	2.731E-09	1.510E-11	2.039E-11
IMOEa-MODE	\	1.510E-11	1.510E-11	1.510E-11
IMOEa-IMODE	2.155E-08	1.034E-02	5.968E-07	1.751E-03
IMOEa-HGAPSO1	\	8.066E-11	1.510E-11	1.510E-11
IMOEa-HGAPSO2	4.380E-12	4.878E-10	2.488E-11	1.436E-10
IMODE-MODE	\	1.510E-11	1.510E-11	1.510E-11

结果讨论: IMOEa 和 IMODE 分别显著优于 NSGA-II 和 MODE。其中, IMOEa 中所采用的 MOEA 是基于 NSGA-II 设计的。因此, 所提出的 ALNS 可以有效地提高 NSGA-II 和 MODE 的算法性能。此外, 由于 IMOEa 的算法性能优于 IMODE, 可以得出 MOEA 和 ALNS 的结合使得 IMOEa 可以在全局探索和局部开发之间做出了更好的平衡。图 3-11 显示了各算法在 4 个指标上的箱线图。在这 4 个指标上 IMOEa 所获得中位数均处于最佳位置。值得注意的是, 在指标

DM、IGD、HV，IMOEa 的箱体宽度（四分位距(Interquartile range, IQR)）均小于其他 5 种对比算法，从而说明 IMOEa 所获结果具有低波动性和高稳定性的特点。因此，IMOEa 是解决所研究问题的优秀算法。

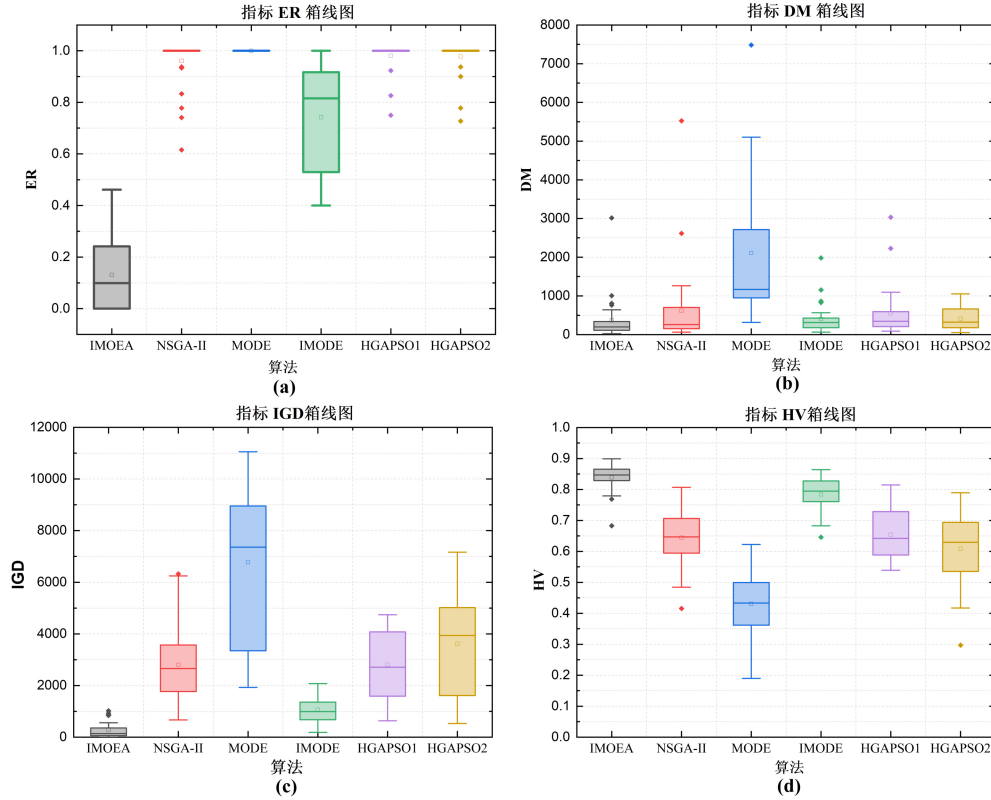


图 3-11 4 个指标的箱线图

3.6 本章小结

本章基于快速消费品制造企业的实际特征，提出了一个具有批次生产和软时间窗交付限制的生产-配送集成调度问题。构建了一个以总利润和提前延期加权惩罚时间的双目标混合整数非线性规划模型。基于所提出的生产-配送集成调度优化框架，设计开发了 IMOEa 进行模型求解。其中协同机制由巡回环划分算法、DPT 函数、BSS-NWS 组成，用于实现个体的解码。IMOEa 优化模式则是由基于问题特征和编码形式设计的 ALNS 改进 MOEA 形成的。然后，针对不同规模的 30 个测试实例进行了数值实验，说明了所提算法的有效性。实验结果表明，ALNS 和 MOEA 的高效配合使得 IMOEa 的探索 and 开发能力得到了很好的平衡。并且 IMOEa 在基数、收敛性和分布性上具有显著优势，IMOEa 是解决快速消费品制造企业生产-配送集成调度问题的一种极具竞争力的算法。

第4章 考虑模糊旅途时间扰动的生产-配送集成调度

快速消费品制造企业在制定生产-配送集成调度计划时，实际的配送环境受到道路交通状况、天气环境等诸多不确定因素的扰动往往导致旅途时间的不确定。不确定的旅途时间会扰乱确定性的生产-配送集成调度计划，甚至使其不可行。并且模糊规划是处理这种不确定性的一种实用方法^[31]。因此，本章具体研究了一个考虑软时间窗和模糊旅途时间的集成生产调度和车辆路径问题（Integrated production scheduling and vehicle routing problem with soft time windows and fuzzy travel times, IPSVRP-STW&FTT）。其中仍然执行分批生产以及采用数量受限的多行程异构车辆执行配送任务。

本章以经济效益和客户满意度为标准，建立了一个双目标混合整数非线性规划模型（Mixed integer nonlinear programming model, MINLP），该模型以总成本和提前延期加权惩罚时间作为优化目标。其次，由于同时考虑了软时间窗和模糊旅途时间，模糊加权惩罚时间的计算涉及到模糊数内的提前和延期部分的加权自叠加。而目前还缺乏处理模糊数内部运算的相关理论。故而提出了一个模糊加权自叠加理论来解决模糊处理问题。然后，仍然根据第2.3.1节的生产-配送集成调度优化框架，设计一种知识驱动协同进化算法（Knowledge-driven co-evolutionary algorithm, KDCEA）来解决这个问题。该算法融合了基于不同更新策略的双子种群协同进化和从问题特征中提取知识指导进化过程的知识驱动策略。最后，通过 CPLEX 求解器验证了 MINLP 模型的正确性，并通过数值实验说明了算法的有效性。并对配送车辆数量进行了敏感性分析，为快速消费品制造企业的决策者提供管理思路。

4.1 生产-配送集成调度模型

4.1.1 问题描述

本章考虑了由单个工厂、一组地理位置分散的零售商和数量受限的多行程异构车辆构成的 IPSVRP-STW&FTT。如图 4-1 所示，在生产调度阶段，每条生产线以不同的生产节拍生产一种类型的产品。订单产品执行分批生产，每个订单

可能包含多种类型的产品。执行订单交付的车辆路线被视为巡回环，一个巡回环内订单组成一个批次，即巡回环和批次一一对应，只有当批次内产品全部生产完毕后，该批次才能执行交付。此外，如果批次中某一类型的产品生产完成后未立即配送则会产生库存。其中工厂有足够的库存容量空间，产品的库存和车辆容量以标准托盘为核算单位。在车辆径阶段，其被定义在整个图 $G=(N, A)$ 上，其中包含各种顶点 $N=\{0, 1, \dots, n\}$ ，弧集为 $A=\{(k, l): k, l \in N\}$ 。其中工厂位于顶点 0 ，零售商的集合被描述为 $R=N \setminus \{0\}$ 。每个零售商订单具有一个软时间窗，提前或延期交付时会招致惩罚，并且车辆的旅途时间具有模糊性。其余假设条件如下：

- 在每条生产线上批次内部的产品生产是连续的；
- 每辆车从工厂出发，依次访问巡回环上的零售商并返回工厂；
- 每个零售商的订单需求必须一次性被满足，不允许拆分交付；
- 允许未满载的车辆执行配送任务，车辆可执行多个巡回环，产品允许混装；
- 巡回环执行期间，车辆连续交付不可等待、停滞；

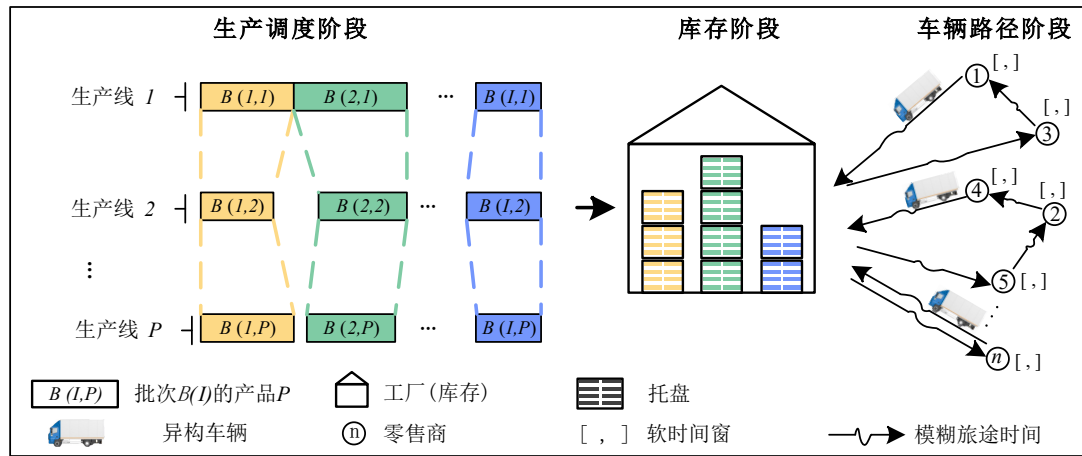


图 4-1 快速消费品制造企业生产-配送集成调度概要图

4.1.2 符号定义及说明

由于 IPSVRP-STW&FTT 中考虑了旅途时间的不确定性，所以执行批次交付的车辆返厂时间、车辆到达零售商的时间、车辆的出发时间以及零售商订单的提前延期均为模糊时间。即在第 3 章模型符号的基础上新增及更改的符号参数如下：

常量

$\widetilde{T}_{k,l}$	零售商 k 到零售商 l 的模糊旅途时间
$\widetilde{T}_{0,k}; \widetilde{T}_{k,n+1}$	零售商 k 到工厂的模糊旅途时间
辅助变量	
$\widetilde{Tr}_{B(i)}$	交付批次 $B(i)$ 的车辆模糊返厂时间
$\widetilde{a}_k$	零售商 k 订单的模糊交付到达时间
$\widetilde{wpt}_k$	零售商 k 订单的模糊提前延期加权惩罚时间
决策变量	
$\widetilde{Td}_{B(i)}$	批次 $B(i)$ 的所对应巡回环的模糊出发时间

4.1.3 混合整数非线性规划模型

本节构建了 IPSVRP-STW&FTT 的数学模型。其两个优化目标分别为总成本 (Total cost, TC) 和提前延期加权惩罚时间 (Early and tardy weighted penalty time, ETPT)。具体描述如下:

(1) TC

该优化目标为最小化 TC, 它反映了供应链上的整体经济效益。它由生产成本 (PC)、基于托盘数量和存储时间的库存成本 (IC) 以及涉及固定成本和可变成本的配送成本 (DC) 共同决定, 公式(4-1)展示了计算过程。

$$\begin{aligned} \text{Min } TC &= PC + IC + DC \\ &= \sum_{p=1}^P \left[PC_p \sum_{k=1}^n D_{k,p} \right] + Hc \sum_{i=1}^n \sum_{p=1}^P \sum_{k=1}^n G_{k,B(i)} D_{k,p} O_p \left(\widetilde{Td}_{B(i)} - Tc_{p,B(i)} \right) + \\ &\quad \sum_{v=1}^V \left\{ Fc_v \sum_{i=1}^n Y_{v,B(i)} + Vc_v \sum_{k=1}^n \sum_{i=1}^n \left[G_{k,B(i)} Y_{v,B(i)} \left(X_{0,k} \widetilde{T}_{0,k} + \sum_{l=1}^{n+1} X_{k,l} \widetilde{T}_{k,l} \right) \right] \right\} \end{aligned} \quad (4-1)$$

(2) ETPT

该优化目标为最小化 ETPT, 它是衡量客户满意度的关键指标。其由加权惩罚时间和订单占用的托盘数量共同决定, 公式(4-2)展示了计算过程。

$$\text{Min } ETPT = \sum_{k=1}^n \left[\widetilde{wpt}_k \sum_{p=1}^P \left(D_{k,p} O_p \right) \right] \quad (4-2)$$

受到模糊旅途时间的影响, 在第 3 章模型约束的基础上需要更改约束(3-5)-约束(3-8)及约束(3-22)-约束(3-23), 更改后的对应约束如下:

$$Tc_{p,B(i)} \leq \widetilde{Td}_{B(i)} \quad \forall i \in \{1, \dots, n\}, \forall p \in \{1, \dots, P\} \quad (4-3)$$

$$\widetilde{Td}_{B(i)} = \sum_{k=1}^n (\widetilde{a}_k - \widetilde{T}_{0,k}) X_{0,k} G_{k,B(i)} \quad \forall i \in \{1, \dots, n\} \quad (4-4)$$

$$\sum_{l=1}^n X_{k,l} (\widetilde{a}_k + \widetilde{T}_{k,l} - \widetilde{a}_l) = 0 \quad \forall k \in \{1, \dots, n\} \quad (4-5)$$

$$\widetilde{Tr}_{B(i)} \leq \widetilde{Td}_{B(j)} + M(1 - S_{B(i),B(j)}) \quad \forall i \in \{1, \dots, n-1\}, \forall j \in \{i+1, \dots, n\} \quad (4-6)$$

$$E_{B(i)}, X_{k,l}, Y_{v,B(i)}, G_{k,B(i)}, S_{B(i),B(j)} \in \{0, 1\} \quad (4-7)$$

$$Ts_{p,B(i)}, Tc_{p,B(i)}, \widetilde{Td}_{B(i)}, \widetilde{a}_k \geq 0 \quad (4-8)$$

其中，约束(4-3)表示只有批次内的所有产品完成后才能执行该批次的交付。约束(4-4)表示从工厂到第一个零售商的模糊时间约束。约(4-5)表示巡回环上相邻零售商的交货顺序和模糊时间约束。约束(4-6)表示车辆在完成当前巡回环并返回工厂之前无法执行下一次巡回环。约束(4-7)-(4-8)定义了变量的域和性质。

4.2 模糊运算的处理

鉴于三角模糊数(Triangle fuzzy number, TFN)被广泛应用在解决不确定性的车间调度^[53-54]和车辆路径问题中^[55-56]，故本章采用 TFN 表征旅途时间的不确定性。假设三角形模糊变量 $\tilde{T} = (T_1, T_2, T_3)$ 代表模糊旅途时间，其中 T_1, T_2, T_3 分别表示最短，最有可能，最长的旅途时间。

4.2.1 基础模糊运算

本章应用了 4 种基础模糊运算：加法(Add)、取大(Max)、期望(Expectation)、比较(Ranking)操作。其中由 Sakawa 等^[57]提出了加法和取大操作用于确定批次的模糊出发时间 $\widetilde{Td}_{B(i)}$ 以及模糊返厂时间 $\widetilde{Tr}_{B(i)}$ 和零售商订单的模糊到达时间 $\widetilde{a}_k$ 。期望操作用于确定 TFN 的期望值，由 Lei^[58]提出的排序操作用于评选最佳调度方案结果。设 $\tilde{x} = (x_1, x_2, x_3)$ ， $\tilde{y} = (y_1, y_2, y_3)$ ，4 种基础模糊运算描述如下：

加法操作： $\tilde{x} + \tilde{y} = (x_1 + y_1, x_2 + y_2, x_3 + y_3)$ ；

取大操作： $\tilde{x} \vee \tilde{y} = (x_1 \vee y_1, x_2 \vee y_2, x_3 \vee y_3)$ ，其中 $x_i \vee y_i = \max\{x_i, y_i\} \quad \forall i \in \{1, 2, 3\}$ ；

期望操作： $E(\tilde{x}) = (x_1 + 2x_2 + x_3)/4$ ；

比较操作： 依次利用三个规则去比较两个 TFN 的大小；

规则 1: 如果 $E(\tilde{x}) > E(\tilde{y})$ 则 $\tilde{x} > \tilde{y}$ ，反之 $\tilde{x} < \tilde{y}$ 。

规则 2: 如果 $E(\tilde{x}) = E(\tilde{y})$ 则比较 x_2 和 y_2 。如果 $x_2 > y_2$, 则 $\tilde{x} > \tilde{y}$, 反之 $\tilde{x} < \tilde{y}$ 。

规则 3: 如果 $x_2 = y_2$ 则比较 $x_3 - x_1$ 和 $y_3 - y_1$ 。如果 $x_3 - x_1 > y_3 - y_1$, 则 $\tilde{x} > \tilde{y}$, 反之 $\tilde{x} < \tilde{y}$ 。

4.2.2 模糊加权自叠加理论及证明

由于 IPSVRP-STW&FTT 中同时考虑了软时间窗和模糊旅途时间, 所以订单的提前延期加权惩罚时间具有模糊性。并且模糊提前延期加权惩罚时间 $\widetilde{wpt}_k$ 的计算涉及到 TFN 内部提前和延期两部分的加权自叠加, 从而导致真正的模糊集并非三角形, $\widetilde{wpt}_k$ 需要进行重构。然而, 目前缺乏处理模糊数内部运算的相关理论。为了达到根据 $\widetilde{a}_k$ 和 $[e_k, l_k]$ 获取 $\widetilde{wpt}_k$ 的目的, 本节定义了如下模糊加权自叠加理论去重构 $\widetilde{wpt}_k$:

首先, 根据零售商 k 订单的模糊到达时间 $\widetilde{a}_k = (a_1, a_2, a_3)$ 和软时间窗 $[e_k, l_k]$, 当配送车辆的到达时间节点为 a'_k 时, 此时违反软时间窗的惩罚时间 pt'_k 可由公式 (4-9) 表示。 $pt'_k < 0$ 表示提前到达, $pt'_k > 0$ 表示延期到达。由于 pt'_k, pt_k^2, pt_k^3 呈现递增顺序, 所以模糊惩罚时间表示为 $\widetilde{pt}_k = (pt_k^1, pt_k^2, pt_k^3)$ 。然后, 根据 $\widetilde{pt}_k$ 使用模糊加权自叠加运算获取 $\widetilde{wpt}_k$ 。由于单位托盘产品的提前和延期惩罚率分别为 Re 、 Rd , 则提前延期加权惩罚时间 wpt'_k 可通过公式 (4-10) 表示。即当 $pt'_k < 0$ 时 $\frac{|pt'_k| + pt'_k}{2} Rd$ 为 0, 反之 $\frac{|pt'_k| - pt'_k}{2} Re$ 为 0。从而可计算得出 $wpt_k^1, wpt_k^2, wpt_k^3$ 。由于存在取绝对值操作, 所有 $wpt_k^1, wpt_k^2, wpt_k^3$ 是无序的, $\widetilde{wpt}_k$ 的真实模糊集合范围并非三角形。公式 (4-11)-(4-12) 给出了两种 $\widetilde{wpt}_k$ 的重构方式。通过推论 7 的证明, 本章采用 $\widetilde{wpt}_{k,1}$ 表征 $\widetilde{wpt}_k$, 从而实现模糊加权自叠加。

$$pt'_k = a'_k - \min\{\max\{e_k, a'_k\}, l_k\} \quad (4-9)$$

$$wpt'_k = \frac{|pt'_k| - pt'_k}{2} Re + \frac{|pt'_k| + pt'_k}{2} Rd \quad (4-10)$$

$$\widetilde{wpt}_{k,1} = (\min\{wpt_k^1, wpt_k^2, wpt_k^3\}, wpt_k^2, \max\{wpt_k^1, wpt_k^2, wpt_k^3\}) \quad (4-11)$$

$$\widetilde{wpt}_{k,2} = (\min\{wpt_k^1, wpt_k^2, wpt_k^3\}, \text{mid}\{wpt_k^1, wpt_k^2, wpt_k^3\}, \max\{wpt_k^1, wpt_k^2, wpt_k^3\}) \quad (4-12)$$

推论 7: 模糊提前延期加权惩罚时间 $\widetilde{wpt}_{k,1}$ 比 $\widetilde{wpt}_{k,2}$ 覆盖更多的真实模糊集合范围。

证明: 首先分为如下两部分:

1) 当 $pt_k^1 > 0$ (或 $pt_k^3 < 0$), 由于 $pt_k^1 < pt_k^2 < pt_k^3$, 则 $wpt'_k = pt'_k Rd$ (或 $wpt'_k = |pt'_k| Re$), 所以 $\widetilde{wpt}_{k,1} = \widetilde{wpt}_{k,2} = \{wpt_k^1, wpt_k^2, wpt_k^3\}$ (或 $\{wpt_k^3, wpt_k^2, wpt_k^1\}$), 故 $\widetilde{wpt}_{k,1}$ 和 $\widetilde{wpt}_{k,2}$ 均

为真实的模糊集合范围。

2) 当 $pt_k^1 < 0 < pt_k^3$ ，此时真实的模糊集合范围不规则需要重构 TFN。为更加直观，分如下两种情况讨论：（注：图 4-2 中假设 $Re = Rd = 1$ ，实际上其取值不影响讨论结果。）

2.1) 当 $pt_k^1 < 0 < pt_k^2$ ，与 pt_k^1 对应的 wpt_k^1 分三种情况即 $wpt_k^1 < wpt_k^2$ ， $wpt_k^2 < wpt_k^1 < wpt_k^3$ ， $wpt_k^1 > wpt_k^3$ 。设 $\widetilde{pt}_k$ 的隶属度函数为 $\mu(t)$ ，如图 4-2 (a)-(c) 所示，将 $t < 0$ 的隶属度函数 $\mu(t)$ 翻折至 $t > 0$ 后形成 $\mu'(t)$ ，通过将 $\mu'(t)$ 与隶属度函数 $\mu(t)$ 在 $t > 0$ 的部分进行线性叠加，形成真实模糊集合范围（图中红色线条 $\mu_r(t)$ 与坐标轴包围区域）。

$\mu_r(t)$ 越大说明与之对应的 t 属于真实模糊集合范围的可能性越大。在这三种情况下， $t = wpt_k^2$ 均获得最大的 $\mu_r(t)$ ，所以 $(wpt_k^2, 1)$ 在真实模糊集合范围内，而 $(\text{mid}\{wpt_k^1, wpt_k^2, wpt_k^3\}, 1)$ 则可能在真实模糊集合范围外（例如：图 4-2 (b)-(c)）。因此， $\widetilde{wpt}_{k,1}$ 比 $\widetilde{wpt}_{k,2}$ 覆盖了更多的真实模糊集合范围。

2.2) 当 $pt_k^2 < 0 < pt_k^3$ ，如图 4-2 (d)-(f) 所示， wpt_k^3 分三种情况即 $wpt_k^3 < wpt_k^2$ ， $wpt_k^2 < wpt_k^3 < wpt_k^1$ ， $wpt_k^3 > wpt_k^1$ 。同理可得， $t = wpt_k^2$ 仍然获得最大的 $\mu_r(t)$ 。因此， $\widetilde{wpt}_{k,1}$ 比 $\widetilde{wpt}_{k,2}$ 覆盖了更多的真实模糊集合范围。

因此， $\widetilde{wpt}_{k,1}$ 比 $\widetilde{wpt}_{k,2}$ 覆盖更多的真实模糊集合范围，证明完毕。

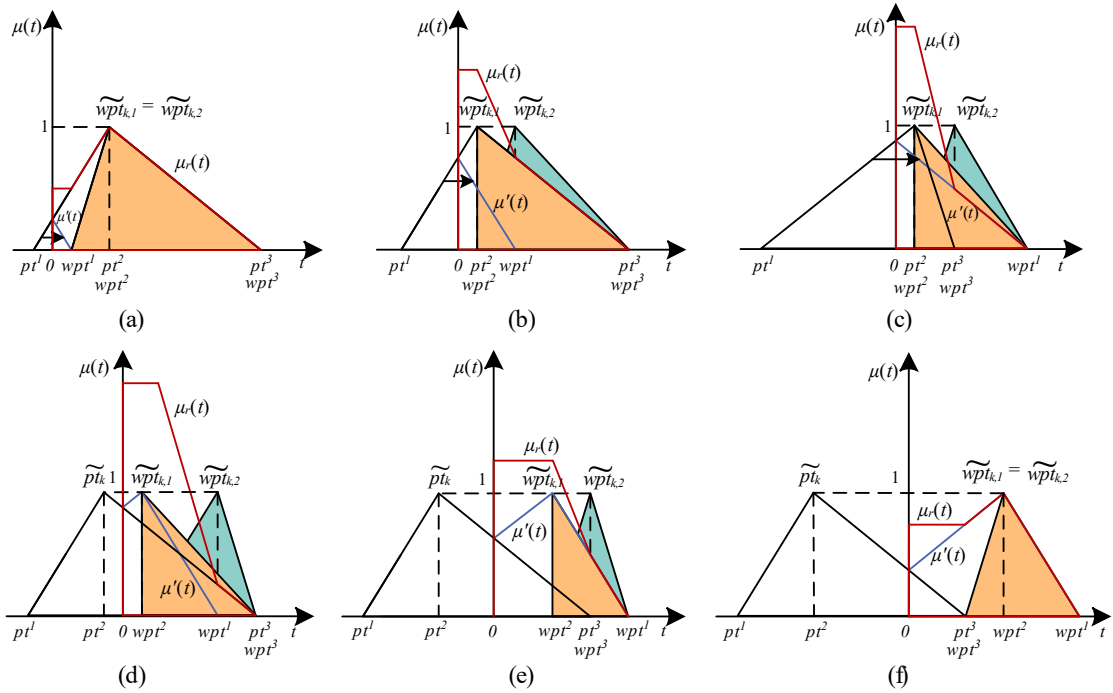


图 4-2 模糊提前延期加权惩罚时间的重构

举例说明：设零售商 k 订单的模糊到达时间 $\widetilde{a}_k = (3, 12, 14)$ ，软时间窗为 $[e_k, l_k] = [10, 11]$ ， $Re=1$ ， $Rd=3$ 。则 $\widetilde{pt}_k = (-7, 1, 3)$ ，从而 $wpt_k^1=7$ ， $wpt_k^2=3$ ， $wpt_k^3=9$ ， $\widetilde{wpt}_{k,1} = (3, 3, 9)$ ， $\widetilde{wpt}_{k,2} = (3, 7, 9)$ ，图 4-3 描述了整个模糊加权自叠加过程。

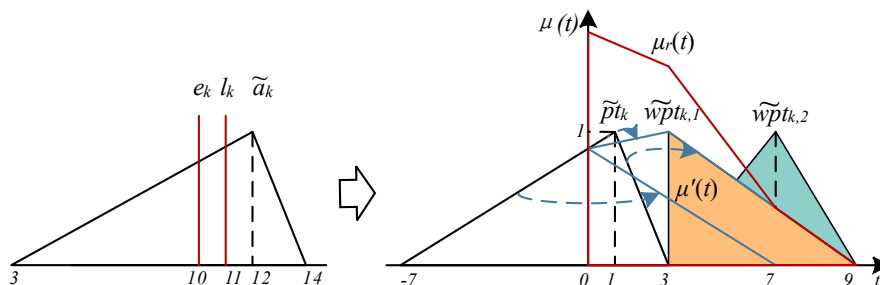


图 4-3 模糊加权自叠加示例

4.3 知识驱动协同进化算法

4.3.1 基于多策略融合的算法框架

在以往的相关研究中，大多数学者采用多样的搜索策略来提高所提算法的开发能力，使得算法具有更好的收敛性^[17]。然而，过度关注算法的开发能力将会导致算法的探索能力偏弱，从而导致解的多样性降低，影响 Pareto 解集的均匀性和扩散性。此问题就是经典的探索与开发困境^[59](Exploration versus exploitation dilemma, EvE)。

为了更好的解决 EvE 困境，使得算法在探索 and 开发之间取得适当的平衡，以获取均匀分散的 Pareto 解集。NSGA-II^[45]和 SPEA-II^[60]通过使用拥挤度距离来保持算法的探索能力。然而，利用这种非分解的 MOEA 生成均匀分散的 Pareto 解集并非易事^[61]。因此，将多目标优化问题分解为多个标量子问题的基于分解的多目标进化算法 (Multi-objective evolutionary algorithm based on decomposition, MOEA/D) 一经提出就受到了广泛的关注。其中均匀的权重向量和基于切比雪夫函数的更新策略为获取均匀分散的 Pareto 解集提供了保障。然后，Qi 等^[62]提出了具有自适应权重调整的 MOEA/D (MOEA/D with adaptive weight vector adjustment, MOEA/D-AWA)，用于解决均匀权重向量无法为不连续型的 Pareto 解集提供良好的多样性这一问题。其通过权重向量的 WS-转换和基于外部种群的自适应权重调整策略来弥补均匀权重向量的缺点。

基于上述讨论，本章设计了一种知识驱动协同进化算法 (Knowledge-driven co-evolutionary algorithm, KDCEA)。该算法以 MOEA/D-AWA 作为基本进化模

式，融合双子种群协同进化策略和知识驱动策略，分别用于增强算法的探索 and 开发能力。KDCEA 的框架如图 4-4 所示。

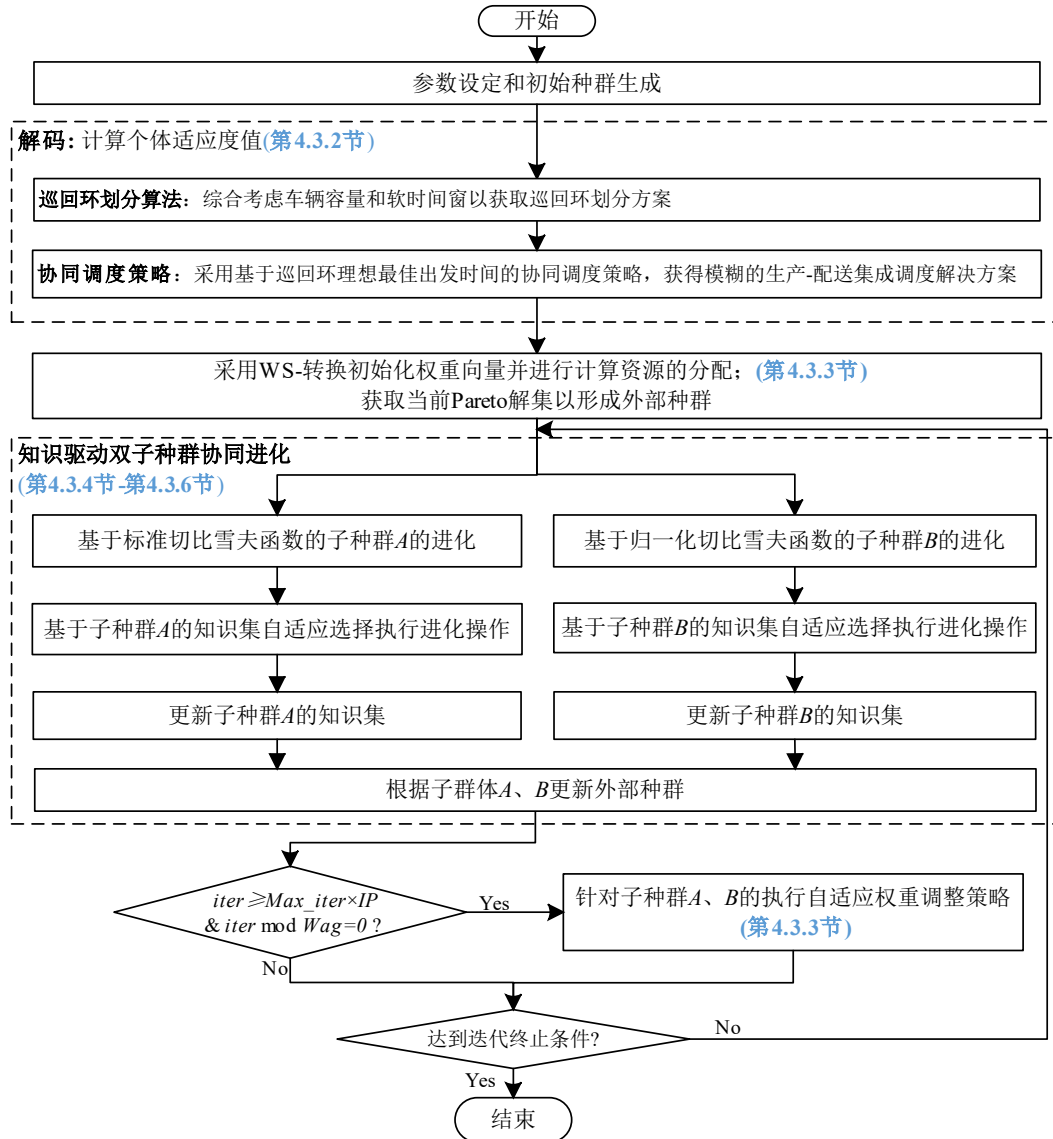


图 4-4 KDCEA 的框架流程

其具体流程如下：

步骤 1：设置算法参数，并采用随机方式生成初始种群以确保其多样性。

步骤 2：采用巡回环划分算法和协同调度策略进行个体的解码获取适应度值。

步骤 3：配置基础进化模式，采用 MOEA-D-AWA 中的 WS-转换进行权重向量的初始化，进行计算资源的分配，并获取当前 Pareto 解集以形成外部种群。

步骤 4：执行双子种群协同进化策略和知识驱动策略。首先，针对具有不同

更新策略的子种群 A、B，采用协同进化的思想将个体搜索引向更广泛的可行解区域。其次，针对每个子种群采用知识驱动策略去自适应的选择所要执行的进化操作和展开进化操作的设计工作。其中根据问题的特定知识，设计了四种具有不同搜索性能的进化操作，引导搜索进入更有前景的区域。然后，在每次迭代过程中，评估进化操作的表现并将其存储到知识集中。最后，基于子种群 A、B 更新外部种群。

步骤 5：根据执行自适应权重调整的迭代百分比（IP）、调整的间隔迭代数（Wag）、当前迭代数（iter）和最大迭代数（Max_iter），当 $iter \geq max_iter \times IP$ 并且 $iter \bmod Wag = 0$ 时，使用 MOEA/D-AWA 中的基于外部种群的自适应权重调整策略来改进 Pareto 解集的均匀性和扩散性。

步骤 6：保存当前 Pareto 解集（即外部种群），如果满足迭代终止条件，则算法结束，否则转到步骤 4。

4.3.2 基于模糊旅途时间扰动的编码与解码设计

KDCEA 的编码形式与第 3 章一致，仍然采用一维向量来表示编码个体，每个个体代表一个车辆选择方案。由于增加了对模糊旅途时间的考虑，所以解码部分存在变动，现介绍如下：

巡回环划分算法中零售商 k 订单的模糊交付到达时间 $\widetilde{a}_k$ 将会影响巡回环订单划分时的判定，即需将模糊数通过期望操作转化为单一数值后与软时间窗上、下限进行对比，表 4-1 给出了模糊旅途时间扰动下巡回环划分算法的伪代码。

在获取巡回环的订单组成后，计算巡回环的理想最佳出发时间是进行协同调度策略的前提。巡回环的理想最佳出发时间的获取流程如下：由于旅途时间 $\widetilde{T}_{k,l} = (T_{k,l}^1, T_{k,l}^2, T_{k,l}^3)$ 为模糊数，无法直接根据 DPT 函数计算巡回环的理想最佳出发时间。故本章采用期望操作将模糊旅途时间转化为旅途期望时间进行 DPT 函数的计算（DPT 函数的详细过程见第 3.3.2 节）。值得注意的是，在获取 DPT_{tour} 函数的所有分段点后，根据 $\widetilde{T}_{k,l}$ 和 $[e_k, l_k]$ 计算由各分段点处出发的巡回环模糊交付惩罚时间 $\widetilde{DPT}_{tour}$ ，然后基于模糊运算中的比较操作，获取最小化 $\widetilde{DPT}_{tour}$ 所对应的巡回环理想最佳出发时间。并且基于模糊运算的特点，巡回环的理想最佳出发时间往往为某一时间节点，而非时间区间。

由于巡回环和批次一一对应，根据巡回环理想最佳出发时间递增排序即可获得批次生产顺序。即获得决策变量 $E_{B(i)}$ 、 $X_{k,l}$ 、 $Y_{v,B(i)}$ 、 $G_{k,B(i)}$ 。然后执行协同

调度策略获得决策变量 $Ts_{p,B(i)}$ 、 $\widetilde{Td}_{B(i)}$ 。由于本章增加了对模糊旅途时间的考虑并且未对生产线生产启动成本进行研究，故而与第 3 章所提出的 BSS-NWS 策略具有明显不同。

表 4-1 模糊旅途时间扰动下巡回环划分算法

算法 3 模糊旅途时间扰动下巡回环划分算法

1. 输入: $Rs_m^v, N_v, O_c, h, C_v, [e_k, l_k], D_{k,p}, O_p, a_k$;
2. 输出: $E_{B(i)}, X_{k,l}$, 各巡回环的订单组成;
3. **For** $v \in \{1, \dots, V\}$ **do**
4. $O_c = 0; \quad h = 1;$
5. **For** $m \in \{1, \dots, N_v\}$ **do**
6. $O_c = O_c + \sum_p^P (D_{Rs_m^v, p} O_p);$
7. **If** $O_c > C_v$ **then**
8. 将 Rs_m^v 放入下一巡回环; 更新 $O_c = \sum_p^P (D_{Rs_m^v, p} O_p); \quad h = h + 1;$
9. **Else**
10. 以当前巡回环 h 中的第一个订单的软时间窗上限作为起始时间。
11. $\widetilde{a}_{Rs_m^v}^{curr}$ 代表从起始时间开始计算, 将订单 Rs_m^v 放入当前巡回环时,
12. Rs_m^v 的模糊到达时间。
13. $\widetilde{a}_{Rs_m^v}^{next}$ 代表从起始时间开始计算, 将订单 Rs_m^v 放入下一巡回环时,
14. 车辆 v 返回工厂后再立即配送 Rs_m^v 的模糊到达时间。
15. **If** $E(\widetilde{a}_{Rs_m^v}^{curr}) < e_{Rs_m^v}$ and $E(\widetilde{a}_{Rs_m^v}^{next}) < l_{Rs_m^v}$ and $m \neq N_v$ **then**
16. 将 Rs_m^v 放入下一巡回环; 更新 $O_c = \sum_p^P (D_{Rs_m^v, p} O_p); \quad h = h + 1;$
17. **Else**
18. 将 Rs_m^v 放入当前巡回环;
19. **Endif**
20. **Endif**
21. **Endfor**
22. **Endfor**

协同调度策略具体流程如下: 首先, 从时间节点 0 开始, 按批次生产顺序连续排列各批次。从而获得各批次的最早生产完成时间节点 $Tc_{p,B(j)}^e$ 。然后, 基于批次理想最佳出发时间 $Td_{B(j)}^{opt}$ (巡回环和批次一一对应) 和批次 $B(j)$ 的前一批次 $B(i)$ 的车辆模糊返厂时间 $\widetilde{Tr}_{B(i)}$, 采用模糊运算最大操作按批次生产顺序更新 $\widetilde{Td}_{B(j)}$,

见公式(4-13)。最后，按照逆向批次生产顺序进行批次和巡回环的协调，见公式(4-14)。从而获了模糊的生产-配送集成调度解决方案。

$$\widetilde{Td}_{B(j)} = \max \left\{ \max \left\{ Tc_{1,B(j)}^e, \dots, Tc_{P,B(j)}^e \right\}, Td_{B(j)}^{opt} \right\} \vee \widetilde{Tr}_{B(i)} \quad (4-13)$$

$$Tc_{p,B(j)} = \min \left\{ Ts_{p,B(j+1)}, \widetilde{Td}_{B(j)}(1) \right\} \quad (4-14)$$

4.3.3 基础进化模式设计

KDCEA 的基本进化模式沿用 MOEA/D-AWA^[62]，本节针对该算法具备的重要特点进行了说明。

标准的 MOEA/D 采用均匀的权重向量，其中每个权重向量 $\lambda' = (\lambda'_1, \dots, \lambda'_m)^T$ 满足 $\sum_{i=1}^m \lambda'_i = 1$ 和 $\lambda'_i \geq 0$ 。 m 表示优化目标的个数。MOEA/D-AWA 则引入了 WS-转换，见公式(4-15)。其将标量子问题的权重向量映射为解决映射向量，从而使得个体解的进化方向与权重向量相一致。另外，为每个个体（标量子问题）分配一个效用函数 $\pi^1, \pi^2, \dots, \pi^N$ 。并将每次迭代分为 5 个子迭代过程，每个子迭代采用基于效用函数的十元锦标赛选择法选择获得 $N/5$ 个个体执行进化。效用函数的更新规则如公式(4-16)所示，其中 Δ^i 表示个体解的改进率， U 表示改进基线。当 $\Delta^i \leq U$ 时，说明个体的改进效果较差，个体进化提升不明显。通过降低个体 i 的效用函数以影响锦标赛选择时选中概率。从而根据个体进化过程中的改进效果，将计算资源合理地分配给各个个体。

$$\lambda = WS(\lambda') = \left(\frac{\frac{1}{\lambda'_1}}{\sum_{i=1}^m \frac{1}{\lambda'_i}}, \frac{\frac{1}{\lambda'_2}}{\sum_{i=1}^m \frac{1}{\lambda'_i}}, \dots, \frac{\frac{1}{\lambda'_m}}{\sum_{i=1}^m \frac{1}{\lambda'_i}} \right) \quad (4-15)$$

$$\pi^i = \begin{cases} 1 & \text{if } \Delta^i > U \\ \left(0.95 + 0.05 \times \frac{\Delta^i}{U} \right) & \text{otherwise} \end{cases} \quad (4-16)$$

由于 IPSVRP-STW&FTT 属于组合优化问题，从而获得的 Pareto 前沿面往往具有不连续的性质。基于此特征，采用外部种群存储迭代期间获取的 Pareto 前沿面个体。并且当进化种群达到一定的收敛程度后，将使用自适应权重调整策略提高不连续型 Pareto 前沿面的均匀性。如图 4-5(a)所示，参考点 $z^* = (z_1^*, \dots, z_m^*)^T$ 通常由每个目标函数的当前最小值表示。当 Pareto 前沿面不连续时，在不连续区域的权重向量所对应的个体解将向最近的 Pareto 前沿端点处聚集，从而导致该

区域的个体解过于拥挤，而其他区域相对稀疏。自适应权重调整策略的执行，见图 4-5(b)所示。基于非支配排序方法中的邻近距离，移除拥挤区域的部分个体解^[63]。然后从外部种群中选择部分个体加入到 Pareto 前沿面中，该个体需要满足与当前 Pareto 前沿面个体的邻近距离较大这一特征，从而实现向稀疏区域加入个体解的目的。并根据公式(4-17)为添加的个体解生成新的最佳权重向量。其中 $F = (f_1, \dots, f_m)^T$ 表示个体解的目标函数值， ε 用于防止分母为零。

$$\lambda^{opt} = WS(F - z^*) = \left(\frac{1}{\sum_{i=1}^m \frac{1}{|f_i - z_i^*| + \varepsilon}}, \dots, \frac{1}{\sum_{i=1}^m \frac{1}{|f_i - z_i^*| + \varepsilon}} \right), \varepsilon > 0, \varepsilon \rightarrow 0 \quad (4-17)$$

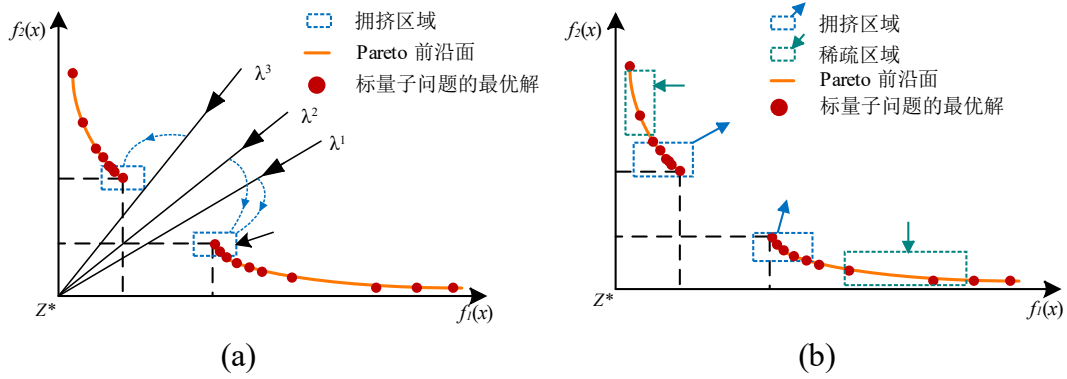


图 4-5 自适应权重向量调整策略

4.3.4 基于不同更新策略的双子种群协同进化

标准的切比雪夫函数（公式(4-18)）一种常用的更新策略，用于指导个体解沿权重向量所在方向进行更新和进化。其中 $\{\lambda^1, \dots, \lambda^N\}$ 代表 WS-转换后的权重向量，第 j 个标量子问题（个体解）的优化目标是 minimized $g^{te}(x | \lambda^j, z^*)$ 。 $z^* = (z_1^*, \dots, z_m^*)^T$ 为参考点，通常由每个优化目标的当前最佳值表示，并在迭代进化过程中持续更新。

$$g^{te}(x | \lambda^j, z^*) = \max_{1 \leq i \leq m} \left\{ \lambda_i^j |f_i(x) - z_i^*| \right\} \quad (4-18)$$

$$g^{te}(x | \lambda^j, z^*, z^{bad}) = \max_{1 \leq i \leq m} \left\{ \lambda_i^j \left| \frac{f_i(x) - z_i^*}{z_i^{bad} - z_i^* + \varepsilon} \right| \right\}, \varepsilon > 0, \varepsilon \rightarrow 0 \quad (4-19)$$

然而，不同的优化目标通常有不同的数量级和单位。所研究的模型目标分别为成本和时间，其在数量级及单位上均不一致。如果仅采用标准的切比雪夫函数

作为更新策略,则在计算 $g^{te}(x|\lambda^j, z^*)$ 时,数量级较大的目标将占主导地位,数量级较小的目标往往被忽略。因此,我们提出了归一化的切比雪夫函数(公式(4-19))实现对两个优化目标的平等对待。从而弥补较小数量级目标优化不完全的缺点。 $z^{bad} = (z_1^{bad}, \dots, z_m^{bad})^T$ 是一个参考点,由每个优化目标的当前最差值表示。

在 KDCEA 中,更新策略的执行步骤如下: Pop_A 表示子种群 A , NI 表示邻域个体的索引, NR 表示每次迭代替换个体的最大数量。首先,根据所获新个体 x' 和邻域个体集 $Pop_A(NI)$,找到所有满足 $g^{te}(x'|\lambda^j, z^*) < g^{te}(Pop_A(NI(u))|\lambda^j, z^*)$ 的可更新邻域个体集 $Pop_A(NI(u))$ 。然后,如果 $Pop_A(NI(u))$ 中邻域个体的数量小于 NR ,则使用 x' 更新 $Pop_A(NI(u))$ 。否则,在 $Pop_A(NI(u))$ 中随机选择 NR 个邻域个体,并使用 x' 进行更新替换。同理,子种群 B 使用类似的步骤,仅将可更新邻域个体满足的条件改为 $g^{te}(x'|\lambda^j, z^*, z^{bad}) < g^{te}(Pop_A(NI(u))|\lambda^j, z^*, z^{bad})$ 即可。

在双子种群协同进化策略中,子种群 A 、 B 之间不直接进行信息交流,而是在每次迭代中将获得的 Pareto 前沿面个体共享给外部种群,通过外部种群实现子种群 A 、 B 之间的信息交流。该方式可以有效的保护每个子种群的个体解特征,从而使得个体解集的范围变得更加广泛。此外,两个子种群的协同进化也能有效的避免单一种群容易陷入局部最优导致过早收敛的情况。

4.3.5 知识驱动策略

知识驱动策略用于执行进化操作的选择和设计。它包括动态和静态两部分:

(1) 基于知识集的自适应操作选择 (Adaptive operation selection, AOS),其中知识集随着算法迭代进行动态更新。

在 AOS 中,概率匹配法是一个常用的方法。即为每个操作分配一个持续更新的信用值,然后根据信用值采用轮盘赌机制进行操作选择。概率匹配法的一个弊端是它给平均表现最佳的进化操作匹配了高概率,该方式将会减缓算法的底层优化速度^[59]。为了在不同的迭代阶段根据实时性能有效地应用每个进化操作,提出了一个基于知识集的 AOS。其中知识集模拟了人脑决策行为,即基于现有的知识体系制定决策,早期不适宜的知识将被删除或更新。因此,知识集设定了一个固定的记录长度 (Record length, RL),并且记录的更新需要满足先进先出规则 (First-in-first-out, FIFO)。当知识集中记录存满时,新记录的插入意味着旧记录的删除。从而有效防止早期不适宜的知识影响当前的操进化操作选择。其次,每条记录将存储所执行的进化操作和执行该操作后获得的切比雪夫函数的改进

率 (Improvement rate of the tchebycheff function, TIR)。进化操作完成执行后, 个体由 x 改变为 x' , 则子种群 A 的 TIR 的计算见公式(4-20)所示 (子种群 B 同理)。显然, TIR 越大, 执行的进化操作就越有效。

知识驱动策略的具体过程如下: 当知识集中的记录未存满时, 则随机选择将要执行的进化操作。当知识集中的记录存满时, 将各进化操作的 TIR 之和作为信用值, 采用基于该信用值的轮盘赌机制进行 AOS。

$$TIR = \frac{g^{te}(x|\lambda^j, z^*) - g^{te}(x'|\lambda^j, z^*)}{g^{te}(x|\lambda^j, z^*)} \quad (4-20)$$

(2) 基于特定问题知识的进化操作设计, 此知识属于预先定义的静态知识。

常用的进化操作为通用型操作, 例如基于随机点位的交叉和变异操作。通用型操作的缺点是很难根据问题特征进行针对性优化。为了使算法的搜索过程更有针对性, 本节根据问题特征提取的特定知识设计了进化操作。Bandaru 等^[64]给出了相似的观点, 即对预优化获得的个体进行分析, 提取优秀个体的知识可以有有效的指导优化过程, 尽管这种方式具有一定的主观性。因此, 本节采用通用型操作 (标准的两点交叉、多点变异) 进行预优化。然后, 通过分析预优化获得的优秀个体, 提取得到如下的具体知识, 即优秀个体往往具有较高的车辆负载率和较低的提前延期加权惩罚时间, 基于此设计了进化操作 1、2。此外, 鉴于算法的编码形式和双子种群协同进化策略, 设计了进化操作 3、4。在图 4-6 中分别论述了进化操作 1-4, 其具体过程描述如下:

操作 1: 为了提高车辆的装载率, 采用基于排序的轮盘赌策略选择车辆装载率低的批次, 并随机替换所选批次中所有订单的车辆序列号。具体来说, 首先通过公式(4-21)计算每个批次的实际负载率, 其中 $LC_{B(i)}^v$ 表示车辆 v 在配送批次 $B(i)$ 的实际负载率。然后, 将所有的 $LR_{B(i)}$ 按由小到大排列, 令 $Rank_{B(i)}$ 表示批次 $B(i)$ 的排序值, 根据公式(4-22)计算各批次信用值 $CV_{B(i)}$ 。显然, $Rank_{B(i)}$ 的值越小, $CV_{B(i)}$, 越大, 从而增加了具有低负载率的批次的选择概率。最后, 依据 $CV_{B(i)}$ 进行轮盘赌获取执行订单替换的批次。

$$LR_{B(i)} = \frac{LC_{B(i)}^v}{C_v} \quad (4-21)$$

$$CV_{B(i)} = \frac{1}{LR_{B(i)} \times Rank_{B(i)}} \quad (4-22)$$

操作 2: 为了减少批次的提前延期加权惩罚时间, 针对个体 x 进行解码, 并

计算出各批次的模糊加权惩罚时间 $\widetilde{wpt}_{batch}$ 。通过模糊运算中的比较操作，获得最大化 $\widetilde{wpt}_{batch}$ 的批次。通过随机替换所选批次中所有订单的车辆序列号，得到一个新的个体 x' 。

操作 3：狭窄的搜索区域使得个体容易陷入局部最优。为了丰富个体的搜索区域，基于问题的编码形式设计了操作 3。首先，随机选择个体的部分编码位置。然后，在该个体的邻域个体集中计算所选编码位置的车辆序列号出现的次数。采用出现频率最低的车辆序列号替换所选编码位置的存储值，得到一个新的个体。

操作 4：考虑到双子种群协同进化的模式，操作 4 旨在通过外部种群引导和帮助两个子种群的进化。具体来说，从外部种群中随机选择一个精英个体，与当前个体进行标准的两点交叉，从而实现高质量编码段的转移。然后，进行随机多点变异，用于防止个体与精英个体过于相似而导致的提前收敛。

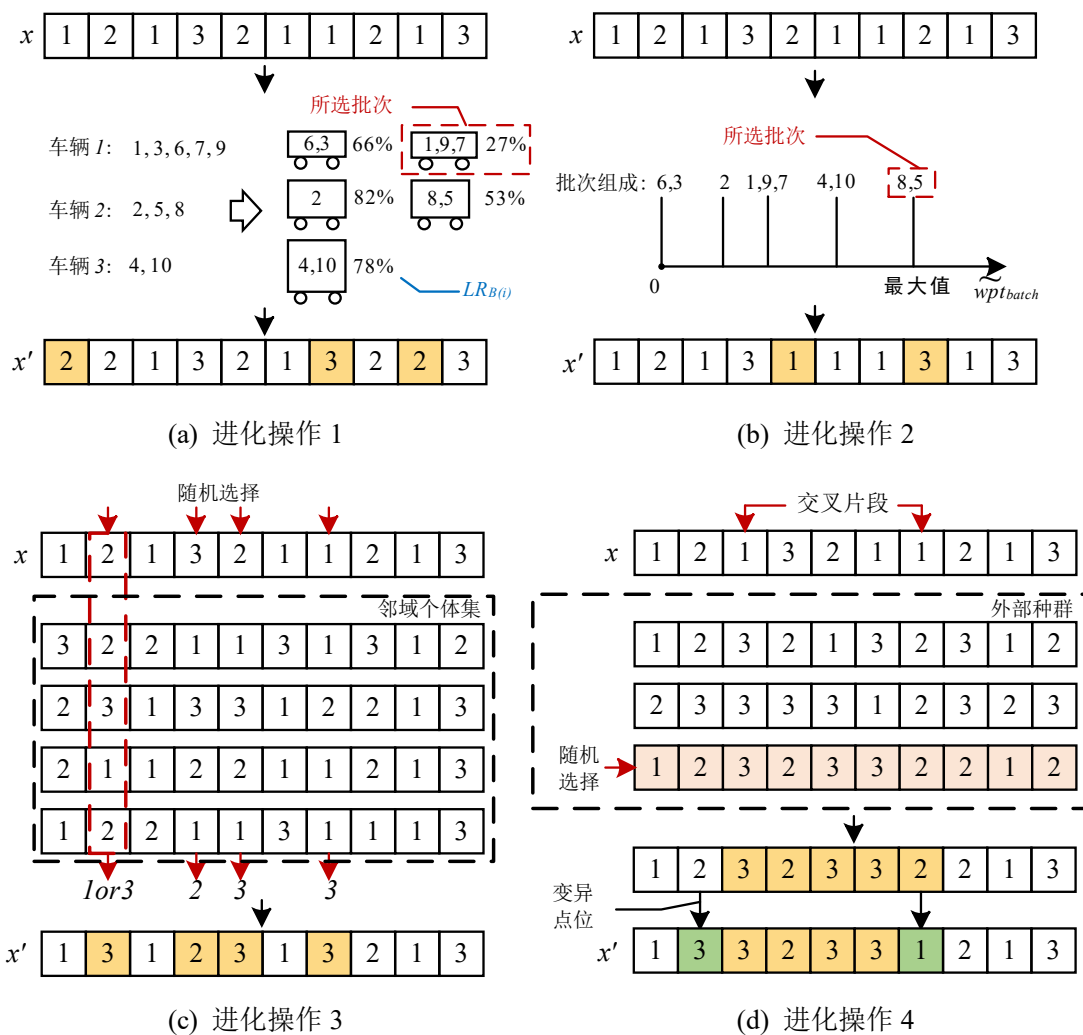


图 4-6 进化操作 1-4 示例

4.4 数值实验与结果分析

仿真实验通过 MATLAB2020a 进行编程，并在具备 Intel Core i5-1035G1 - CPU 1.00GHz 和 16.0GB 的内存上运行算法。

4.4.1 测试实例的生成

本节在第 3 章测试实例生成的基础上，提供了一个包含 9 个小规模、9 个中等规模和 9 个大规模实例的数据集，以进行算法性能对比。测试实例(P - V - n)是由生产线数量 $P \in \{3, 4, 5\}$ 、配送车辆数量 $V \in \{4, 5, 6\}$ 、零售商数量 $n \in \{20, 25, 30\}$ 组合形成。由于新增考虑了模糊旅途时间，故要设置 $\widetilde{T}_{k,l}$ 的生成方式： $\widetilde{T}_{k,l} = (T_{k,l}^1, T_{k,l}^2, T_{k,l}^3)$ ，令 $T_{k,l}^1 = (1 - 0.3\text{rand}())T_{k,l}^2$ ， $T_{k,l}^2 = T_{k,l}$ ， $T_{k,l}^3 = (1 + 0.3\text{rand}())T_{k,l}^2$ 。其余参数的详细设定方式见 表 3-8，在此不再赘述。该测试实例数据集可从 <https://www.huangm.cn/zip/IPSVRP-STW-FTT-Dateset.zip> 中下载。

4.4.2 对比算法选取及多目标评价指标

为了验证所提算法中策略的有效性，以 MOEA/D-AWA 为基础加入双子种群协同进化策略形成了协同进化算法 (Co-evolutionary algorithm, CEA)，知识驱动进化算法 (Knowledge-driven evolutionary algorithm, KDEA) 是融合了知识驱动策略的 MOEA/D-AWA。此外，比较算法中还包括众所周知的多目标进化算法：NSGA-II^[45]、MOEA/D-AWA^[62]以及解决类似模型的自适应差分进化^[31](Self-adaptive differential evolution, SADE)。为了进行公平的比较，对比算法均采用上述的编、解码。本章仍然使用错误率^[49] (ER)、分布度量^[50] (DM)、反世代距离^[51] (IGD)、超体积^[52] (HV) 作为算法的评价指标，其具体计算过程及含义详见第 3.5.2 节。

4.4.3 算法参数设计

KDCEA 有五个重要的参数：子种群规模 N ，效用函数的改进基线 U ，执行自适应权重调整的迭代百分比 IP ，每次迭代替换个体的最大数量 NR ，知识集的记录长度 RL 。采用正交实验设计以确定参数的最佳值。参数水平设置为： $N=60, 80, 100, 120$ 、 $U=0.001, 0.01, 0.05, 0.1$ 、 $IP=0.5, 0.6, 0.7, 0.8$ 、 $NR=1, 2, 3, 4$ 、 $RL=500, 1000, 2000, 3000$ 。通过正交表 $L_{16}(4^5)$ 进行正交实验设计。并对

正交表中的每个组合独立运行算法 10 次，以 IGD 的平均值作为评判结果的指标。图 4-7 显示了 IGD 的参数主效应图。KDCEA 的最佳参数值为： $N=120$ 、 $U=0.01$ 、 $IP=0.7$ 、 $NR=1$ 、 $RL=1000$ 。同理可得，对其余算法进行正交实验设计，从而使其参数设置为最佳。

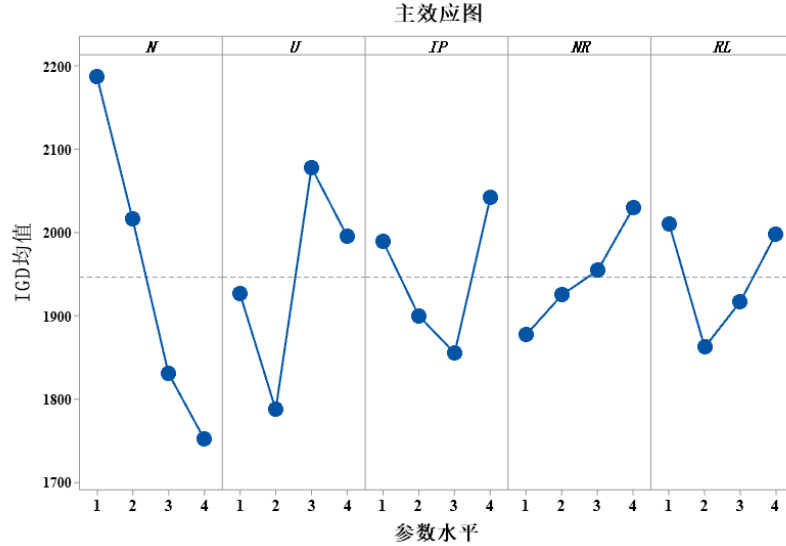


图 4-7 主效应图

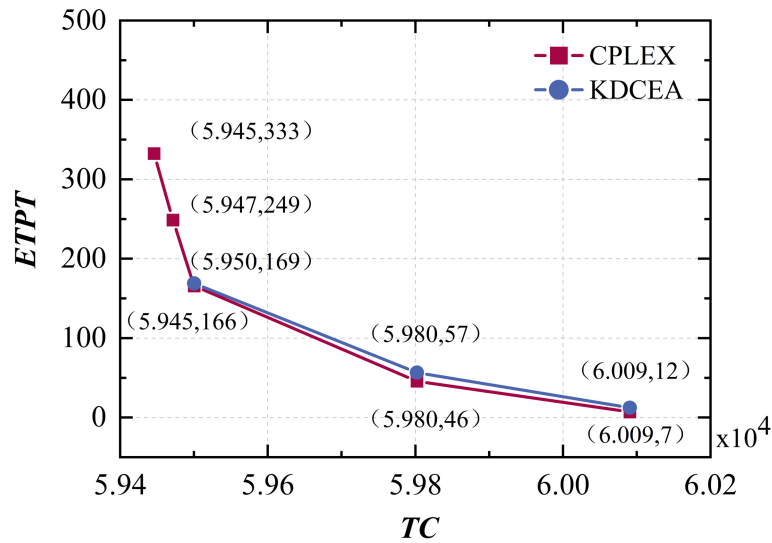


图 4-8 Pareto 前沿面对比

4.4.4 MINLP 模型验证

本节采用 CPLEX 12.9.0 求解器验证 MINLP 模型的正确性。首先，将 MINLP 模型中的非线性变量线性化。采用 ϵ -constraint 方法^[65]对线性化后的数学模型进行多目标优化。其中将网格点数设置为 4、eps 设置为 10^{-6} 以获得真实的 Pareto

前沿面。针对小规模实例 3-4-20, CPLEX 求解器无法在 2 个小时内找到真实的 Pareto 前沿面, 故生成极小规模实例 3-4-5 用于模型验证。图 4-8 显示了在实例 3-4-5 上 CPLEX 求解器和 KDCEA 的实验结果。结果显示, CPLEX 求解器可以获得真实 Pareto 前沿面验证了模型的正确性。并且 KDCEA 所获得的 Pareto 前沿面逼近真实的 Pareto 前沿面, 但是受到解码过程中的巡回环划分算法的影响, 算法优化解与最优解存在较小差距。

4.4.5 对比实验结果与分析

在 27 个测试实例上每个算法独立运行 10 次。为了避免偶然性, 采用 10 次运行的评价指标平均值作为表 4-2 中显示的结果。粗体字值代表在各测试实例上获取指标最佳值所对应的算法。对于指标 ER, KDCEA 在超过 20 个测试实例上的表现最佳, 显示出了优秀的基数性能。对于指标 DM, KDCEA 在 16 个实例上表现最佳, 其次 CEA 在不超过 6 个实例上表现最佳。对于指标 IGD 和 HV, 可以看出 KDCEA 在分布性和收敛性方面具有绝对优势。因此, KDCEA 相较于对比算法具有明显优势。

为了说明算法之间的差异具有统计学意义。首先, 使用 Anderson-Darling 测试进行正态性检验。由于大多数总体数据 P 值 < 0.05 , 表明不满足正态性假设。非参数检验是不可避免的。然后, 将显著性水平设置为 0.05, 采用 Mann-Whitney 检验去判定两两算法之间的显著性, 相应的结果列于表 4-3。结果显示, 除了在指标 DM 上 KDCEA 和 NSGA-II 之间的差异不显著外, 在其余指标上 KDCEA 显著优于其他算法。此外, 由于 KDEA、CEA 与 MOEA/D-AWA 在指标 DM 和 IGD 上的 P 值 < 0.05 , 说明双子种群协同进化策略和知识驱动策略分别显著提高了算法的扩散性和收敛性。

图 4-9 显示了所有算法在 3 个不同规模实例上的 Pareto 前沿面, 其中采用期望操作将 TFN 映射为单一值后完成 Pareto 前沿面的绘制。可以看出 KDCEA 更逼近于真实的 Pareto 前沿面。此外, 图 4-10 显示了实例 5-6-30 中具有最小化 TC 和最小化 ETPT 的个体对应的生产-配送集成调度方案 A、B 的甘特图。方案 A: 最小化 $TC = (3.373 \times 10^5, 3.401 \times 10^5, 3.437 \times 10^5)$ 和 $ETPT = (3470, 3678, 3875)$; 方案 B: $TC = (3.489 \times 10^5, 3.523 \times 10^5, 3.558 \times 10^5)$ 和最小化 $ETPT = (2485, 2623, 2815)$ 。显然, 两个优化目标 TC 和 ETPT 具有冲突关系。基于上述讨论可以得出, KDCEA 是解决 IPSVRP-STW&FTT 的优秀算法。

表 4-2 在 4 个评价指标上的算法结果

测试 实例	ER						DM					
	KDCEA	KDEA	CEA	MOEA/D- AWA	NSGA-II	SADE	KDCEA	KDEA	CEA	MOEA/D- AWA	NSGA-II	SADE
3-4-20	0.862	0.931	1.000	0.964	0.993	1.000	727.569	1996.638	702.379	1101.306	609.833	38513.848
3-5-20	0.956	0.988	0.967	0.961	0.952	0.972	683.358	1070.765	967.455	1314.914	1154.071	5733.524
3-6-20	0.899	0.986	0.986	0.981	1.000	0.958	785.208	2279.930	1240.489	1963.716	1401.187	17000.352
4-4-20	0.945	1.000	0.962	1.000	0.976	1.000	1532.001	808.451	466.620	2592.869	1659.239	7218.517
4-5-20	0.918	0.950	0.993	0.940	0.975	1.000	721.215	65535.000	892.468	1498.731	1292.955	15341.360
4-6-20	0.958	0.913	0.940	1.000	1.000	0.950	1370.335	21185.727	65535.000	2396.609	412.393	24286.135
5-4-20	0.850	1.000	1.000	1.000	0.960	1.000	380.399	296.029	299.985	5082.972	252.099	65535.000
5-5-20	0.845	1.000	0.983	0.950	0.975	1.000	388.891	849.229	388.780	412.393	455.223	1531.292
5-6-20	0.858	0.944	1.000	1.000	1.000	1.000	884.840	715.220	281.374	1144.478	389.476	1779.214
3-4-25	0.881	1.000	1.000	0.943	1.000	0.900	528.879	800.999	1349.568	65535.000	995.977	560.316
3-5-25	0.887	1.000	0.986	0.908	0.994	1.000	744.303	1879.597	610.173	1564.716	622.325	65535.000
3-6-25	0.936	0.973	0.988	0.978	0.975	1.000	583.303	990.167	463.147	2055.718	639.715	2121.251
4-4-25	0.941	0.967	1.000	1.000	0.967	0.983	457.275	28686.346	1612.903	5082.044	1130.498	4432.589
4-5-25	0.847	1.000	0.967	1.000	1.000	1.000	1675.178	708.010	780.827	1079.900	483.359	6585.271
4-6-25	0.933	0.971	1.000	0.958	1.000	1.000	820.549	1515.068	2706.821	2667.757	2376.460	84223.495
5-4-25	0.906	0.991	0.965	0.986	0.960	0.978	724.398	412.896	3006.938	1786.012	341.524	4318.223
5-5-25	0.921	0.973	0.993	0.883	0.986	1.000	493.565	609.940	695.576	1749.851	643.280	2881.242
5-6-25	0.913	1.000	0.992	0.866	0.983	1.000	231.923	742.559	528.252	1148.627	506.334	1551.517
3-4-30	0.952	0.896	0.900	0.896	0.970	1.000	707.801	963.370	1580.338	733.068	1597.269	65535.000
3-5-30	0.920	0.958	0.918	0.950	1.000	0.963	2414.319	2448.944	830.220	2345.296	1458.018	65535.000
3-6-30	0.907	0.973	0.913	0.967	0.982	0.967	1174.689	1193.598	1166.991	878.107	1344.154	2887.865
4-4-30	0.864	0.980	1.000	0.994	1.000	1.000	416.536	452.825	65535.000	65535.000	1516.978	1734.266
4-5-30	0.906	1.000	0.908	1.000	1.000	1.000	310.869	875.233	1731.693	662.596	720.106	65535.000
4-6-30	0.880	0.967	0.980	0.960	0.970	1.000	772.296	11119.875	906.623	2521.763	1212.000	5428.320
5-4-30	0.929	0.918	0.982	1.000	1.000	1.000	346.522	404.480	1474.522	788.138	506.680	915.781
5-5-30	0.908	0.980	1.000	0.971	1.000	1.000	224.505	422.457	263.420	873.387	1217.502	65535.000
5-6-30	0.882	1.000	0.992	0.989	0.994	1.000	296.126	728.349	791.133	462.777	222.996	65535.000

测试 实例	IGD						HV					
	KDCEA	KDEA	CEA	MOEA/D- AWA	NSGA-II	SADE	KDCEA	KDEA	CEA	MOEA/D- AWA	NSGA-II	SADE
3-4-20	451.559	512.973	905.200	518.024	1168.267	1838.265	0.914	0.913	0.856	0.906	0.838	0.804
3-5-20	412.044	670.457	659.904	675.699	786.757	1449.536	0.939	0.912	0.897	0.916	0.885	0.823
3-6-20	437.296	901.124	933.598	1151.629	1569.931	1869.648	0.944	0.891	0.902	0.888	0.866	0.867
4-4-20	767.573	848.598	845.759	1043.180	1148.924	1891.382	0.877	0.834	0.861	0.840	0.813	0.738
4-5-20	977.577	1235.503	1105.099	1273.945	1542.874	2653.409	0.833	0.800	0.810	0.805	0.789	0.713
4-6-20	849.460	1186.230	1389.458	1794.633	2180.185	2717.999	0.910	0.893	0.878	0.832	0.810	0.788
5-4-20	713.565	789.879	924.963	1104.613	791.109	1229.308	0.844	0.740	0.748	0.712	0.768	0.764
5-5-20	892.034	1176.938	922.030	1317.607	1380.090	1531.578	0.814	0.746	0.765	0.764	0.727	0.750
5-6-20	1019.477	1249.965	1238.887	1548.160	1553.918	2201.542	0.809	0.722	0.748	0.711	0.719	0.672
3-4-25	712.069	1276.357	1081.777	1203.901	1052.334	1014.925	0.803	0.702	0.705	0.587	0.785	0.788
3-5-25	1342.417	1921.549	1673.059	1864.908	2369.870	2575.147	0.782	0.698	0.740	0.729	0.670	0.656
3-6-25	884.022	1167.067	1326.763	1290.701	2946.850	2786.112	0.752	0.709	0.702	0.688	0.594	0.566
4-4-25	886.323	1125.174	1677.051	3027.064	1588.783	2909.041	0.935	0.918	0.883	0.827	0.903	0.857
4-5-25	849.782	1685.141	870.806	2167.329	3746.087	5547.947	0.862	0.792	0.849	0.736	0.690	0.649
4-6-25	1484.108	1788.838	1430.282	1541.816	3506.039	5808.489	0.832	0.799	0.824	0.820	0.696	0.602
5-4-25	808.347	930.322	1138.195	1341.010	829.013	1822.389	0.838	0.783	0.799	0.769	0.799	0.715
5-5-25	1352.395	1593.369	1628.500	2113.245	1714.469	2421.465	0.817	0.786	0.787	0.751	0.769	0.709
5-6-25	1627.409	1976.807	2050.613	2057.402	2314.960	4364.353	0.784	0.706	0.722	0.696	0.724	0.609
3-4-30	889.808	1268.485	1485.291	1309.035	1320.519	1555.827	0.769	0.708	0.656	0.658	0.700	0.669
3-5-30	1554.446	1889.773	1673.794	1662.765	2644.335	3538.478	0.752	0.722	0.683	0.695	0.697	0.639
3-6-30	1520.420	1614.761	1801.628	2088.209	3362.352	3587.667	0.786	0.757	0.741	0.737	0.638	0.620
4-4-30	1207.458	1287.783	1734.266	1787.075	2404.368	2524.122	0.785	0.740	0.673	0.635	0.678	0.676
4-5-30	823.400	1492.644	1231.818	2241.328	2307.321	4207.929	0.837	0.747	0.779	0.655	0.700	0.597
4-6-30	1802.885	1815.174	2010.082	2292.915	3101.016	4634.033	0.831	0.769	0.764	0.748	0.700	0.596
5-4-30	1300.022	1699.863	1995.496	2145.204	1734.822	2624.745	0.765	0.741	0.698	0.659	0.704	0.614
5-5-30	1036.194	1474.150	1442.206	1916.297	2142.186	3108.654	0.842	0.776	0.754	0.723	0.727	0.662
5-6-30	1608.052	2396.803	3692.457	2891.709	3864.148	8631.753	0.823	0.700	0.645	0.616	0.682	0.514

表 4-3 Mann-Whitney 检验 P 值表

算法对比	P 值 (ER)	P 值 (DM)	P 值 (IGD)	P 值 (HV)
KDCEA vs KDEA	0.000	0.005	0.006	0.001
KDCEA vs CEA	0.000	0.022	0.002	0.002
KDCEA vs MOEA/D-AWA	0.000	0.000	0.000	0.000
KDCEA vs NSGA-II	0.000	0.083	0.000	0.000
KDCEA vs SADE	0.000	0.000	0.000	0.000
KDEA vs MOEA/D-AWA	0.635	0.047	0.020	0.064
CEA vs MOEA/D-AWA	0.686	0.022	0.031	0.100

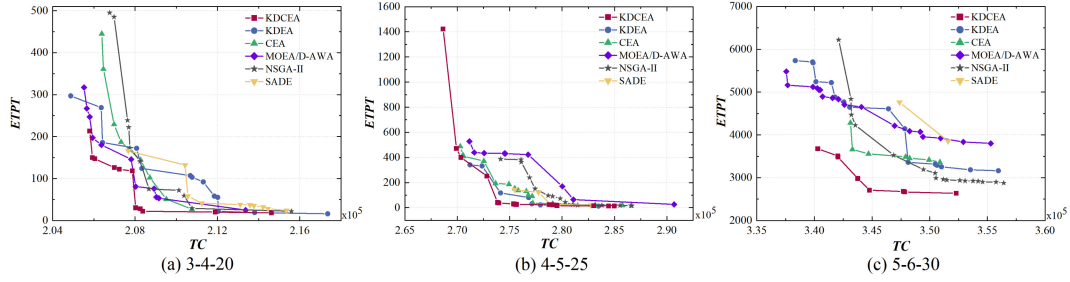
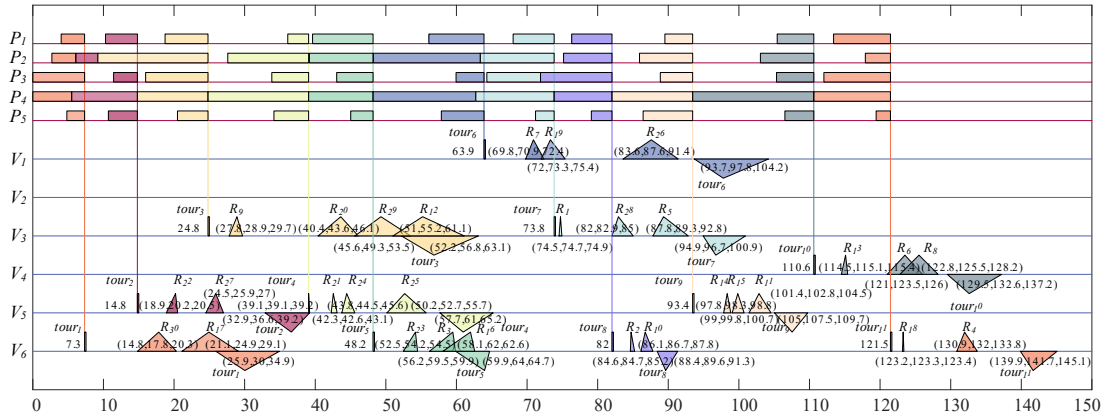
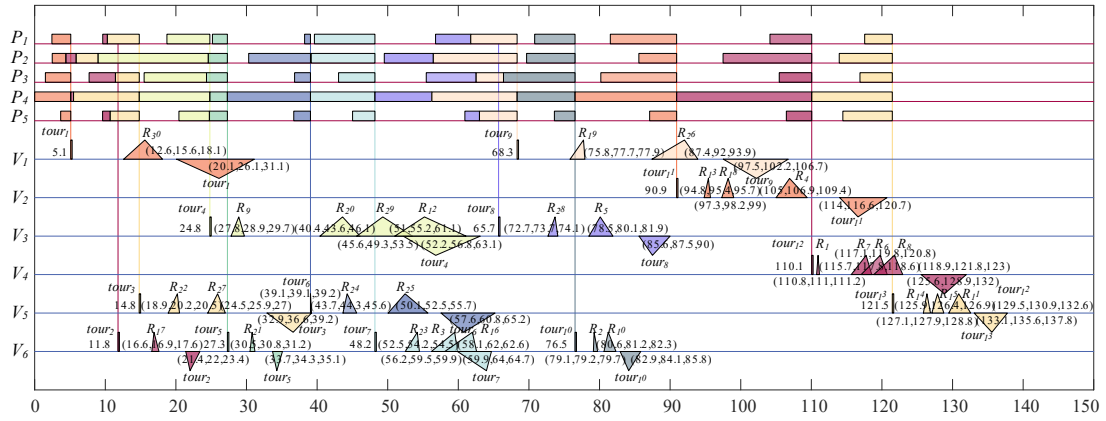


图 4-9 不同规模下算法的 Pareto 前沿面



(a) 在实例 5-6-30 中具有最佳 TC 的个体 A 的甘特图



(b) 在实例 5-6-30 中具有最佳 ETPT 的个体 B 的甘特图

图 4-10 在实例 5-6-30 中的甘特图示例

4.4.6 敏感性分析

受到软时间窗和模糊旅途时间的影响，在交付过程中配送车辆的数量至关重要，它与订单的及时性交付密切相关。为了研究车辆数量对所提问题的影响，在上述计算结果的基础上，进行了敏感性分析。如图 4-11 所示，每个点代表 KDCEA 在相应实例上得到的最佳目标函数值，每条线代表不同车辆数量影响下的目标升降趋势。结果表明，车辆数量在一定程度上的增加对目标函数 TC 的影

响并不明显。然而，随着车辆数量的增加，目标函数 ETPT 有下降趋势。并且随着实例规模的扩大，ETPT 的下降趋势逐渐明显。其原因是车辆数量的增加使决策者可以及时的交付订单，而不必等待正在执行任务的车辆。此外，随着实例规模的扩大，产品生产数量也不断增多，车辆数量成为限制决策过程的关键因素。从而使得车辆数量对 ETPT 的影响变得更加明显。

综上所述，快速消费品制造企业在制定生产-配送集成调度计划时，应当充分考虑模糊旅途时间扰动下对订单及时交付的影响。并且随着订单数量的增多（对应实例规模的扩大）快速消费品制造企业应当根据实际运营状况适量增加配送车辆数量，从而降低模糊旅途时间对企业运营造成的影响，增强客户的满意程度。

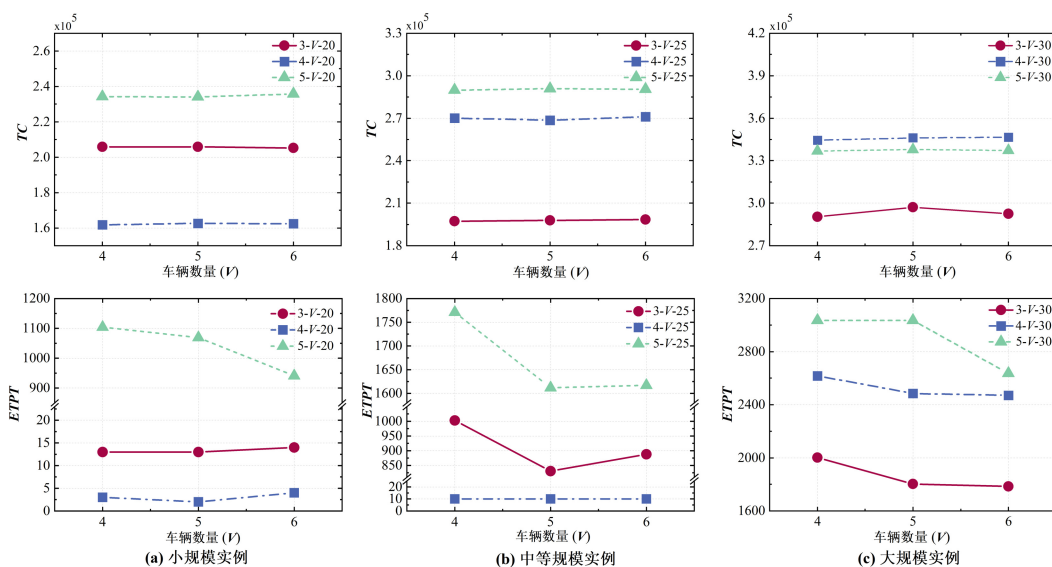


图 4-11 不同规模实例的敏感度分析

4.5 本章小结

本章针对考虑模糊旅途时间扰动的生产-配送集成调度问题进行了研究，具体考虑一个具有软时间窗和模糊旅途时间的集成生产调度和车辆路径问题（IPSVRP-STW&FTT）。其中借助模糊规划刻画了旅途时间的不确定性。并建立了一个以总成本和提前延期加权惩罚时间为双目标混合整数非线性规划模型。鉴于软时间窗和模糊旅途时间的同时考虑，提出了模糊加权自叠加理论来计算模糊加权惩罚时间。该理论可以有效地处理模糊数内部的加权自叠加。由于该问题为 NP-hard，故而开发了 KDCEA 来解决该问题。该算法融合了基于不同更新

策略的双子种群协同进化和知识驱动策略。然后采用 CPLEX 求解器验证了模型的正确性，并基于不同规模的测试实例对 KDCEA 的性能进行了评估。计算结果表明，双子种群协同进化策略和知识驱动策略可以有效提高 KDCEA 的探索 and 开发能力。并且 KDCEA 在基数、扩散性、收敛性、分布性上明显优于其他算法。最后针对车辆数量进行了敏感性分析，为决策者提供了管理见解。

第 5 章 总结与展望

5.1 本文总结

本文针对快速消费品制造企业生产-配送集成调度流程及特征进行了分析，设计了生产-配送集成调度优化框架。构建了考虑分批生产及软时间窗限制的生产-配送集成调度模型和考虑模糊旅途时间扰动的生产-配送集成调度模型，并分别设计了优化算法进行模型求解。本文的主要工作如下：

(1) 基于快速消费品制造企业的生产-配送集成调度流程，重点分析了企业的流水线分批生产调度特征、车辆配送调度特征、不确定旅途时间扰动特征。进而提出了生产-配送集成调度优化框架及策略，为后文的算法设计提供指导。

(2) 针对考虑分批生产及软时间窗限制的生产-配送集成调度问题，构建了一个考虑了分批生产、软时间窗限制以及数量受限的多行程异构车辆等实际特征的双目标混合整数非线性规划模型。然后，设计了一种基于自适应大邻域搜索策略（ALNS）改进的多目标进化算法（IMOEa），其中包括协同机制以及自适应大邻域搜索策略的实现。数值实验结果表明，ALNS 和 MOEA 的高效配合使得 IMOEa 的探索和开发能力得到了很好的平衡。IMOEa 是解决快速消费品制造企业生产-配送集成调度问题的一种极具竞争力的算法。

(3) 针对考虑了模糊旅途时间扰动的生产-配送集成调度问题，通过三角模糊数的方式刻画模糊旅途时间。首先，构建了以生产-库存-配送总成本和提前延期加权惩罚时间作为双优化目标的混合整数非线性规划模型。其次，提出了模糊加权自叠加理论来处理模糊数的内部运算，并论证了该理论的合理性。然后，设计一种融合了双子种群协同进化策略和知识驱动策略的知识驱动协同进化算法（KDCEA）来解决该问题。另外，通过 CPLEX 求解器验证了模型的正确性。数值实验结果表明，KDCEA 在基数、扩散性、收敛性、分布性上明显优于其他对比算法，算法性能表现优异。最后，针对车辆数量进行了敏感性分析，为快速消费品制造企业的决策者提供管理思路。

5.2 研究展望

本文主要考虑了多产品分批生产、订单的软时间窗限制、数量受限的多行程

异构配送车辆、以及不确定旅途时间等实际特征影响下的快速消费品制造企业生产-配送集成调度。并设计了相应的优化算法进行求解。但是考虑到快速消费品制造企业生产-配送的复杂运营流程，仍然存在需要进一步探索的研究方向，主要包括以下几个方面：

(1) 本文主要考虑了生产、库存、配送三阶段的集成调度，但是在现实中，企业生产所需的原材料通常也需要通过自有车辆进行运输。即增加对原材料供应阶段的考虑，其中原材料运输和产品交付的车辆可以混用。因此，一个考虑了原材料供应、生产、库存、配送的四阶段集成调度问题将在后续研究中进行讨论。

(2) 本文主要针对了单周期下的快速消费品制造企业生产-配送集成调度问题进行了研究。但是在现实中，客户具有周期性订货的特点，并且在生产中存在客户紧急订单的临时插入的情况。因此，在后续研究中将针对带订单紧急插入扰动的多周期生产-配送集成调度问题进行讨论。

(3) 由于企业的自有车辆需要进行日常维护，并且车辆数量越多维护成本越高，因此，确定客户数量和车辆数量间的最佳比例也将是值得研究的方向。

致谢

“我曾经以为日子是过不完的，未来是完全不一样的”。三年前，我背上书包，满怀憧憬的来到武汉求学。如今，我就待在自己的未来，要与这个朝夕相伴了三年的地方说再见。

回顾三年的研究生时光，我收获良多。我明白了人生中的每件事都有自己的节奏，耐心的去对待，烦躁和焦虑并不能解决问题。要相信北海虽赊，扶摇可接；东隅已逝，桑榆非晚。想要成为什么样的人，就要付出什么样的努力。

在秉持着努力就会有回报的信念一路向前时，才慢慢发现我所获得的一切都离不开身边人的帮助与支持。在此，谨向一路上给予我帮助、支持、鼓励和陪伴的人表示由衷的感谢！

感谢我的导师杜百岗老师和郭钧老师，从研究方向选取、模型算法设计、再到学术论文撰写，两位老师的悉心教导让我受益良多。并且在我面对挫折困难时，给予了莫大的鼓励和帮助。学生朽木，幸得雕琢。同时还要感谢课题组内的郭顺生老师、李益兵老师等在硕士期间给予我的帮助及指导。

感谢我的实验室同门和室友在科研及生活中给予的帮助和关照，以及一起在羽毛球场上挥洒汗水的球友们。是你们陪伴着我度过了一段愉快充实的时光。

感谢我的父母一直以来的支持和付出，是你们默默扛下了生活的重担，才让我得以安心求学。同时感谢我的姐姐黄焯，在我迷茫焦虑时对我的劝导和支持。因为你们的陪伴，让我倍感亲切和幸福。

最后感谢乐观的自己，当希望一个个落空，仍然选择沉着和坚持。感谢勇于试错的自己，在安静无声的日子不断沉淀。感谢自己站在人生的路口时，没有每次都选择那条看起来更容易走的路。从而才明白麻烦的事情里，隐藏着真正的乐趣。才明白人生的路从来都不是轻轻松松的，要打败很多很多的挫折与迷茫，懒惰与软弱。才明白只有抱着乐观积极的心态，持之以恒的努力，才能成为更好的自己。

行文至此，落笔为终。轻舟已过万重山，前路漫漫亦灿灿。

黄铭

2023年3月于武汉理工大学

参考文献

- [1] 刘星. 和声搜索算法优化快速消费品生产配送协调调度 [J]. 工业工程, 2016, 19: 14-17.
- [2] Chen W., Zhang Y., Zhou Y. Integrated scheduling of zone picking and vehicle routing problem with time windows in the front warehouse mode [J]. Computers & Industrial Engineering, 2022, 163: 107823.
- [3] Chevroton H., Kergosien Y., Berghman L., et al. Solving an integrated scheduling and routing problem with inventory, routing and penalty costs [J]. European Journal of Operational Research, 2021, 294(2): 571-589.
- [4] 吴艺杭. 我国快消品行业库存管理现状及问题分析 [J]. 物流工程与管理, 2019, 41: 60-62.
- [5] Vahdani B., Niaki S. T. A., Aslanzade S. Production-inventory-routing coordination with capacity and time window constraints for perishable products: Heuristic and meta-heuristic algorithms [J]. Journal of Cleaner Production, 2017, 161: 598-618.
- [6] Long J., Pardalos P. M., Li C. Level-based multi-objective particle swarm optimizer for integrated production scheduling and vehicle routing decision with inventory holding, delivery, and tardiness costs [J]. International Journal of Production Research, 2022, 60(11): 3319-3338.
- [7] 吴瑶, 马祖军. 时变路网下带时间窗的易腐食品生产-配送问题 [J]. 系统工程理论与实践, 2017, 37: 172-181.
- [8] Han J., Liu Y., Luo L., et al. Integrated production planning and scheduling under uncertainty: A fuzzy bi-level decision-making approach [J]. Knowledge-Based Systems, 2020: 201-202.
- [9] Potts C. N. Technical Note—Analysis of a Heuristic for One Machine Sequencing with Release Dates and Delivery Times [J]. Operations Research, 1980, 28(6): 1436-1441.
- [10] Chen Z.-L. Integrated Production and Outbound Distribution Scheduling: Review and Extensions [J]. Operations Research, 2010, 58(1): 130-148.
- [11] Moons S., Ramaekers K., Caris A., et al. Integrating production scheduling and vehicle routing decisions at the operational decision level: A review and discussion [J]. Computers & Industrial Engineering, 2017, 104: 224-245.
- [12] Kergosien Y., Gendreau M., Billaut J. C. A Benders decomposition-based heuristic for a production and outbound distribution scheduling problem with strict delivery constraints [J]. European Journal of Operational Research, 2017, 262(1): 287-298.
- [13] Fu L.-L., Aloulou M. A., Triki C. Integrated production scheduling and vehicle routing problem with job splitting and delivery time windows [J]. International Journal of Production Research, 2017, 55(20): 5942-5957.

-
- [14] Yagmur E., Kesen S. E. Bi-objective coordinated production and transportation scheduling problem with sustainability: formulation and solution approaches [J]. *International Journal of Production Research*, 2022: DOI: 10.1080/00207543.2020.2021.02017054.
 - [15] Gao S., Qi L., Lei L. Integrated batch production and distribution scheduling with limited vehicle capacity [J]. *Int J Production Economics*, 2015, 160: 13-25.
 - [16] Zou X., Liu L., Li K., et al. A coordinated algorithm for integrated production scheduling and vehicle routing problem [J]. *International Journal of Production Research*, 2018, 56(15): 5005-5024.
 - [17] Guo Z., Shi L., Chen L., et al. A harmony search-based memetic optimization model for integrated production and transportation scheduling in MTO manufacturing [J]. *Omega*, 2017, 66: 327-343.
 - [18] Devapriya P., Ferrell W., Geismar N. Integrated production and distribution scheduling with a perishable product [J]. *European Journal of Operational Research*, 2017, 259(3): 906-916.
 - [19] Yagmur E., Kesen S. E. Multi-trip heterogeneous vehicle routing problem coordinated with production scheduling: Memetic algorithm and simulated annealing approaches [J]. *Computers & Industrial Engineering*, 2021, 161: 107649.
 - [20] Ganji M., Kazemipoor H., Hadji Molana S. M., et al. A green multi-objective integrated scheduling of production and distribution with heterogeneous fleet vehicle routing and time windows [J]. *Journal of Cleaner Production*, 2020, 259: 120824.
 - [21] Park Y.-B., Hong S.-C. Integrated production and distribution planning for single-period inventory products [J]. *International Journal of Computer Integrated Manufacturing*, 2009, 22(5): 443-457.
 - [22] Wang D., Zhu J., Wei X., et al. Integrated production and multiple trips vehicle routing with time windows and uncertain travel times [J]. *Computers & Operations Research*, 2019, 103: 1-12.
 - [23] Hou Y., Fu Y., Gao K., et al. Modelling and optimization of integrated distributed flow shop scheduling and distribution problems with time windows [J]. *Expert Systems with Applications*, 2022, 187: 115827.
 - [24] Liu H. T., Guo Z. X., Zhang Z. Z. A hybrid multi-level optimisation framework for integrated production scheduling and vehicle routing with flexible departure time [J]. *International Journal of Production Research*, 2021, 59(21): 6615-6632.
 - [25] Mohammadi S., Al-e-Hashem S. M. J. M., Rekik Y. An integrated production scheduling and delivery route planning with multi-purpose machines: A case study from a furniture manufacturing company [J]. *International Journal of Production Economics*, 2020, 219: 347-359.

- [26] Kumar R., Ganapathy L., Gokhale R., et al. Quantitative approaches for the integration of production and distribution planning in the supply chain: a systematic literature review [J]. *International Journal of Production Research*, 2020, 58(11): 3527-3553.
- [27] Adulyasak Y., Cordeau J.-F., Jans R. Benders Decomposition for Production Routing Under Demand Uncertainty [J]. *Operations Research*, 2015, 63(4): 851-867.
- [28] Hemmati Golsefidi A., Akbari Jokar M. R. A robust optimization approach for the production-inventory-routing problem with simultaneous pickup and delivery [J]. *Computers & Industrial Engineering*, 2020, 143: 106388.
- [29] Liu P., Hendalianpour A., Razmi J., et al. A solution algorithm for integrated production-inventory-routing of perishable goods with transshipment and uncertain demand [J]. *Complex & Intelligent Systems*, 2021, 7(3): 1349-1365.
- [30] 刘星, 李波, 刘海. 不确定环境下快速消费品生产配送鲁棒优化模型 [J]. *运筹与管理*, 2016, 25: 57-62.
- [31] Ghasemkhani A., Tavakkoli-Moghaddam R., Rahimi Y., et al. Integrated production-inventory-routing problem for multi-perishable products under uncertainty by meta-heuristic algorithms [J]. *International Journal of Production Research*, 2022, 60(9): 2766-2786.
- [32] Biuki M., Kazemi A., Alinezhad A. An integrated location-routing-inventory model for sustainable design of a perishable products supply chain network [J]. *Journal of Cleaner Production*, 2020, 260: 120842.
- [33] Guo J., Liu W., Peng Z., et al. Integrated scheduling of distributed production and distribution in group manufacturing with uncertain travel time [J]. *Complex Intell Systems*, 2022: 1-19.
- [34] Fazel Zarandi M. H., Hemmati A., Davari S., et al. Capacitated location-routing problem with time windows under uncertainty [J]. *Knowledge-Based Systems*, 2013, 37: 480-489.
- [35] Qiu Y., Wang L., Xu X., et al. A variable neighborhood search heuristic algorithm for production routing problems [J]. *Applied Soft Computing*, 2018, 66: 311-318.
- [36] Adulyasak Y., Cordeau J.-F., Jans R. Optimization-Based Adaptive Large Neighborhood Search for the Production Routing Problem [J]. *Transportation Science*, 2014, 48(1): 20-45.
- [37] Belo-Filho M. A. F., Amorim P., Almada-Lobo B. An adaptive large neighbourhood search for the operational integrated production and distribution problem of perishable products [J]. *International Journal of Production Research*, 2015, 53(20): 6040-6058.
- [38] Absi N., Archetti C., Dautère-Pérès S., et al. A Two-Phase Iterative Heuristic Approach for the Production Routing Problem [J]. *Transportation Science*, 2014, 49(4): 784-795.
- [39] Chitsaz M., Cordeau J.-F., Jans R. A Unified Decomposition Matheuristic for Assembly, Production, and Inventory Routing [J]. *INFORMS Journal on Computing*, 2019, 31(1): 134-152.

- [40] Solyalı O., Süral H. A multi-phase heuristic for the production routing problem [J]. *Computers & Operations Research*, 2017, 87: 114-124.
- [41] Li Y., Chu F., Chu C., et al. An efficient three-level heuristic for the large-scaled multi-product production routing problem with outsourcing [J]. *European Journal of Operational Research*, 2019, 272(3): 914-927.
- [42] Sun X. T., Chung S. H., Chan F. T. S. Integrated scheduling of a multi-product multi-factory manufacturing system with maritime transport limits [J]. *Transportation Research Part E: Logistics and Transportation Review*, 2015, 79: 110-127.
- [43] Ma Z.-J., Wu Y., Dai Y. A combined order selection and time-dependent vehicle routing problem with time widows for perishable product delivery [J]. *Computers & Industrial Engineering*, 2017, 114: 101-113.
- [44] Gharaei A., Jolai F. A multi-agent approach to the integrated production scheduling and distribution problem in multi-factory supply chain [J]. *Applied Soft Computing*, 2018, 65: 577-589.
- [45] Deb K., Pratap A., Agarwal S., et al. A fast and elitist multiobjective genetic algorithm NSGA-II. [J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6: 182-197.
- [46] Aksen D., Kaya O., Sibel Salman F., et al. An adaptive large neighborhood search algorithm for a selective and periodic inventory routing problem [J]. *European Journal of Operational Research*, 2014, 239(2): 413-426.
- [47] Qu H., Ai X.-Y., Wang L. Optimizing an integrated inventory-routing system for multi-item joint replenishment and coordinated outbound delivery using differential evolution algorithm [J]. *Applied Soft Computing*, 2020, 86: 105863.
- [48] Soleimani H., Kannan G. A hybrid particle swarm optimization and genetic algorithm for closed-loop supply chain network design in large-scale networks [J]. *Applied Mathematical Modelling*, 2015, 39(14): 3990-4012.
- [49] Audet C., Bibeon J., Cartier D., et al. Performance indicators in multiobjective optimization [J]. *European Journal of Operational Research*, 2021, 292(2): 397-422.
- [50] Zheng K., Yang R.-J., Xu H., et al. A new distribution metric for comparing Pareto optimal solutions [J]. *Structural and Multidisciplinary Optimization*, 2016, 55(1): 53-62.
- [51] coello C. A. c., cort'es N. c. Solving Multiobjective Optimization Problems Using an Artificial Immune System [J]. *Genetic Programming and Evolvable Machines*, 2005, 6(2): 163-190.
- [52] Zitzler E., Deb K., Thiele L. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results [J]. *Evolutionary Computation*, 2000, 8(2): 173-195.
- [53] 郭钧, 钟精诚, 杜百岗, 等. 考虑模糊作业时间的再制造加工车间多目标调度方法 [J]. *控制与决策*, 2020, 35: 1497-1504.

- [54] Li R., Gong W., Lu C. A reinforcement learning based RMOEA/D for bi-objective fuzzy flexible job shop scheduling [J]. *Expert Systems with Applications*, 2022, 203: 117380.
- [55] 杨翔, 范厚明, 徐振林, 等. 模糊需求下多中心开放式车辆路径优化 [J]. *计算机集成制造系统*, 2019, 25: 469-479.
- [56] 李楠, 胡蓉, 钱斌, 等. 两阶段混合优化算法求解模糊需求下多时间窗车辆路径问题 [J]. *控制与决策*, 2022, 37: 1573-1582.
- [57] Sakawa M., Mori T. An efficient genetic algorithm for job-shop scheduling problems with fuzzy processing time and fuzzy due date [J]. *Computers & Industrial Engineering*, 1999, 36(2): 325-341.
- [58] Lei D. A genetic algorithm for flexible job shop scheduling with fuzzy processing time [J]. *International Journal of Production Research*, 2010, 48(10): 2995-3013.
- [59] Li K., Fialho A., Kwong S., et al. Adaptive Operator Selection With Bandits for a Multiobjective Evolutionary Algorithm Based on Decomposition [J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(1): 114-130.
- [60] Jensen M. T. Reducing the Run-Time Complexity of Multiobjective EAs: The NSGA-II and Other Algorithms [J]. *IEEE Transactions on Evolutionary Computation*, 2003, 7(5): 503-515.
- [61] Zhang Q., Li H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition [J]. *IEEE Transactions on Evolutionary Computation*, 2007, 11(6): 712-731.
- [62] Qi Y., Ma X., Liu F., et al. MOEA/D with adaptive weight adjustment [J]. *Evol Comput*, 2014, 22(2): 231-264.
- [63] Yang D. D., Jiao L. C., Gong M. G., et al. Adaptive ranks clone and k-nearest neighbor list-based immune multi-objective optimization [J]. *Computational Intelligence*, 2010, 26(4): 359-385.
- [64] Bandaru S., Ng A. H. C., Deb K. Data mining methods for knowledge discovery in multi-objective optimization: Part A - Survey [J]. *Expert Systems with Applications*, 2017, 70: 139-159.
- [65] Mavrotas G. Effective implementation of the ϵ -constraint method in Multi-Objective Mathematical Programming problems [J]. *Applied Mathematics and Computation*, 2009, 213(2): 455-465.

附录 A：攻读硕士学位期间发表的学术论文

- [1] 黄铭, 杜百岗*, 郭钧, 李益兵. 考虑软时间窗限制和模糊旅途时间的生产配送集成调度优化 [J]. 控制理论与应用, 2023, 40(x):1-10. DOI: 10.7641/CTA.2023.20920 (已录用, EI 检索, 本文第 4 章)
- [2] Huang, M., Du, B. G., Guo, J*. (2023). A hybrid collaborative framework for integrated production scheduling and vehicle routing problem with batch manufacturing and soft time windows. *Computers & Operations Research*. (二审中, JCR Q1, 本文第 3 章)
- [3] Huang, M., Du, B. G., Guo, J*, Li, Y, B. (2023). Integrated production scheduling and vehicle routing problem with soft time windows and fuzzy travel times based on fuzzy weighted self-superposition. *International Journal of Production Research*. (一审中, JCR Q1, 本文第 4 章)

附录 B：攻读硕士学位期间获得的科研成果

- [1] 发明专利：杜百岗，黄铭，郭钧，彭兆. 流水线车间分批生产与车辆配送集成调度方法及装置[P]: CN202210721173.7, 2022.(公开实审)
- [2] 软件著作权：杜百岗，黄铭. 生产与配送集成调度管理系统，计算机软件著作权：2022SR1106191, 2022 年 8 月.
- [3] 奖项：“华为杯”第十八届中国研究生数学建模竞赛，国家级三等奖：F2021300432, 2021 年 12 月.
- [4] 奖项：“华为杯”第十九届中国研究生数学建模竞赛，国家级二等奖：C2022200304, 2023 年 1 月.