

1616843

# 北京交通大学

## 硕士学位论文

基于遗传算法的带时间窗车辆路径优化问题研究

**Study of Vehicle Routing Problem with Time Windows  
Based on Genetic Algorithm**

作者：蒋 波

导师：王喜富 教授

北京交通大学

2010年6月



## 学位论文版权使用授权书

本学位论文作者完全了解北京交通大学有关保留、使用学位论文的规定。特授权北京交通大学可以将学位论文的全部或部分内容编入有关数据库进行检索，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。

(保密的学位论文在解密后适用本授权说明)

学位论文作者签名：蒋波

签字日期：2010年6月12日

导师签名：

签字日期：2010年6月13日

中图分类号: U116.2  
UDC: 65.012.34

学校代码: 10004  
密级: 公开

# 北京交通大学

## 硕士学位论文

基于遗传算法的带时间窗车辆路径优化问题研究

Study of Vehicle Routing Problem with Time Windows Based on  
Genetic Algorithm

作者姓名: 蒋波

学 号: 08121266

导师姓名: 王喜富

职 称: 教授

学位类别: 工学

学位级别: 硕士

学科专业: 系统工程

研究方向: 信息系统工程

北京交通大学

2010年6月

## 致谢

本论文的工作是在我的导师王喜富教授的悉心指导下完成的，在论文的写作过程中，王老师倾注了大量的心血，从论文的选题、资料查询、开题、研究到最后的完稿，都提出了许多宝贵的意见。王老师严谨的治学态度和科学的工作方法给了我极大的帮助和影响，在此衷心感谢王老师对我的关心和指导。

王喜富教授悉心指导我们完成了实验室的科研工作，在学习上和生活上都给予了我很大的关心和帮助，他广博的学识、严谨的治学态度和善于从宏观把握、分析问题实质的方法，都激励我不断进步，值得我永远去学习，在此向王老师表示衷心的感谢。

另外，感谢多年来含辛茹苦、省吃俭用供我读书的家人致以我最崇高的敬意，感谢他们对我的信任、鼓励和支持，他们的理解和支持使我能够在学校专心完成我的学业。

最后，感谢审阅论文的各位专家和评委，谢谢各位老师。

## 中文摘要

**摘要：**车辆路径问题（Vehicle Routing Problem, VRP）是物流配送过程中的关键问题之一，随着物流配送行业竞争日益激烈和客户对物流配送时效性要求越来越高，对车辆路径问题的研究，尤其是对带时间窗车辆路径问题（Vehicle Routing Problem With Time Windows, VRPTW）的研究，不仅可以帮助运输企业提高服务水平，为客户提供快捷、准时、安全、舒适的服务，而且有助于企业节约运输成本，提高车辆利用效率，缩短生产周期，加速资金周转，实现资源的合理配置，汲取“第三利润源泉”的财富，因此研究带时间窗车辆路径问题具有重要的现实意义。

本文正是基于以上背景对带时间窗的车辆路径优化问题进行了相关研究。论文从旅行商问题出发，通过分析带时间窗车辆路径优化问题的基本理论，对可用于求解 VRPTW 的各种优化算法进行了对比，确定了遗传算法作为本文 VRPTW 求解算法。在此基础上，考虑配送距离、配送及时性以及配送车辆数对配送成本的影响，构建了以配送总成本最小化为目标的带有惩罚函数的 VRPTW 优化模型，并设计了适合于该模型求解的染色体编码方式以及遗传算子等。最后，应用算例进行了仿真试验，利用 MATLAB 软件分别计算出基于改进遗传算法和基本遗传算法的最优目标函数值与最优配送路径方案，通过对试验结果的对比分析，验证了本文所建模型及求解算法的合理性和有效性。

图 21 幅，表 8 个，参考文献 67 篇。

**关键词：**物流配送；时间窗；车辆路径问题；遗传算法

**分类号：**U116.2

## ABSTRACT

**ABSTRACT:** Vehicle Routing Problem(VRP)is one of the key processes of logistics distribution. As competitions among logistics distribution industry are getting more and more fiercer and customers' requirements for a time-effective logistics distribution are becoming more and more intense, studies on VRP, especially Vehicle Routing Problem with Time windows(VRPTW), are practically significant on every aspect of logistics distribution industry. Transportation companies can improve their services by providing their customers with fast, punctual, safe and comfortable services, their transportation costs can be cut down by improving vehicles utilization. Besides, production cycles are shortened and capital turnover is accelerated, therefore, rational allocations of resources can be realized and fortunes from "the third profit source" can as well be made. So the study of VRPTW is more practical significance.

In the view of this background, this paper do some research on VRPTW which can help the enterprises to improve their conditon.. Starting with the general account of TSP, the paper analyzed the basic theory of VRPTW, and then classified a variety of optimization algorithms of resolving vehicle routing problem with time windows, and ultimately determines the genetic algorithm as the solving algorithm of the model in this paper. On this basis, the mathematical model with time penalty to optimizing VRPTW was established under the consideration of transportation cost, degree of customs' satisfaction and vehieles' number. According to the characteristics of VRPTW, designed the chromsome encoding and special arithmetic operators to get the resolution. Finally, obtained the optimal objective function value and the arrangement respectively by the simple GA and the improved GA under a study case using MATLAB, and comparisons of computed results fully verified that the algorithm given in this paper can obtain an optimized solution effectively and to be a good scheme to solve VRPTW.

Twenty one figures, eight tables, sixty seven references.

**KEYWORDS:** Logistics Distribution; Time Windows; Vehicle Routing Problem; Genetic Algorithm

**CLASSNO:** U116.2

目录

中文摘要 ..... iii

ABSTRACT ..... iv

1 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 国外研究现状 ..... 2

        1.2.2 国内研究现状 ..... 3

    1.3 本文的研究方法及主要内容 ..... 4

    1.4 本章小结 ..... 6

2 车辆路径问题概述 ..... 7

    2.1 旅行商问题（TSP） ..... 7

    2.2 车辆路径问题（VRP） ..... 8

        2.2.1 车辆路径问题的描述 ..... 8

        2.2.2 车辆路径问题的组成要素 ..... 8

        2.2.3 车辆路径问题的数学模型 ..... 11

        2.2.4 车辆路径问题的分类 ..... 12

    2.3 带时间窗的车辆路径问题（VRPTW） ..... 13

        2.3.1 时间窗的概念及描述 ..... 14

        2.3.2 时间窗的分类 ..... 14

    2.4 本章小结 ..... 17

3 带时间窗车辆路径问题求解算法研究 ..... 18

    3.1 精确算法 ..... 18

    3.2 启发式算法 ..... 20

        3.2.1 传统启发式算法 ..... 20

        3.2.1 现代启发式算法 ..... 22

    3.3 遗传算法 ..... 25

        3.3.1 遗传算法基本思想 ..... 25

        3.3.2 遗传算法相关概念 ..... 26

        3.3.3 遗传算法工作流程 ..... 27

        3.3.4 遗传算法特点 ..... 28

3.4 常用算法概括与比较 .....	29
3.5 本章小结 .....	31
4 VRPTW 模型构建与算法设计 .....	32
4.1 VRPTW 模型构建 .....	32
4.1.1 问题描述 .....	32
4.1.2 基本假设 .....	33
4.1.3 惩罚函数 .....	33
4.1.4 参变量定义 .....	34
4.1.5 数学模型 .....	35
4.2 遗传算法设计 .....	36
4.2.1 编码与解码 .....	37
4.2.2 初始群体 .....	38
4.2.3 选择算子 .....	38
4.2.4 交叉算子 .....	41
4.2.5 变异算子 .....	42
4.2.6 适应度函数 .....	43
4.2.7 终止进化规则 .....	43
4.3 本章小结 .....	44
5 算例分析 .....	45
5.1 算例说明 .....	45
5.2 试验数据 .....	45
5.3 试验参数设定 .....	48
5.4 算法实现主要步骤 .....	48
5.5 试验结果分析 .....	53
5.6 本章小结 .....	59
6 总结及展望 .....	60
6.1 全文总结 .....	60
6.2 研究展望 .....	60
参考文献 .....	62
附录 A .....	65
作者简历 .....	68
独创性声明 .....	69
学位论文数据集 .....	70



## 1 绪论

### 1.1 研究背景及意义

在经济高速发展的今天,物流已经成为“第三个利润源”<sup>[1]</sup>,其重要性已被提升到一个战略高度,作为物流主要组成环节的物流配送,由于其配送成本在物流成本中的比重逐渐加大,选择有效的配送路线,减少配送成本,已经成为控制物流成本的主要措施。因此,优化物流配送运输,降低运输成本,是企业尤其是物流配送企业提高企业竞争力的重要途径。

车辆路径问题(Vehicle Routing Problem,缩写为VRP)<sup>[2]</sup>是物流配送过程中的关键环节,该问题以及其延伸问题一直受到学术界广泛的关注。VRP一般指对一系列的客户点,组织适当的行车路线,使车辆有序地通过它们,在满足一定的约束条件(如货物需求量、车辆容量限制等)下,达到一定的目标(如运距最短、费用最少等)。VRP作为物流配送的重要组成部分,是实现物流现代化的基础和前提条件。在满足约束条件下选取恰当的配送路径不仅有助于提高物流企业的经济效益,也有助于解决城市交通拥挤、能源短缺、大气污染等困扰人们的社会问题,实现效率、资源、环境和价值观念各方面的内在统一,促进物流业的进步和社会经济的可持续发展。

随着企业JIT(Just in time)战略的实施,零库存成为许多企业追求的目标,于是客户对货物的送到时间提出了更高的要求,一般都规定了一个希望被访问的时间窗,即要求在此时间窗内接受服务,这就产生了有时间窗的车辆路径问题(Vehicle Routing Problem With Time Windows,缩写为VRPTW)<sup>[3]</sup>。VRPTW是在VRP的基础上增加了客户要求访问的时间窗约束,VRPTW体现了客户对自身人力资源和库存成本的合理控制。对VRP的研究,尤其是对VRPTW的研究,不但可以帮助运输企业提高服务水平,为客户提供快捷、准时、安全、舒适的服务,而且有助于企业节约运输成本,提高车辆利用效率,缩短生产周期,加速资金周转,实现资源的合理配置,汲取“第三利润源泉”的财富,因此更加具有实际意义。

由于VRPTW本身的复杂性和许多现有求解方法的先天性不足,使得求解VRPTW变得比较困难,经常无法得到令人满意的优化结果,制约了一些物流配送企业经济效益和服务质量的提高。因此,建立符合实际情况的数学模型,构造高效、合理的求解方法显得愈发迫切。VRPTW是复杂的、综合性强的组合优化问题,

而解决优化方面的问题,具有隐式并行性和全局搜索性的遗传算法具有无可比拟的优势。针对传统的遗传算法存在“早熟收敛”、易陷入局部极值点等不足,设计寻优性能优越的改进遗传算法解决带时间窗车辆路径问题,可为实际物流配送提供切实可行的路径优化手段,以达到降低物流成本,提高企业利润和配送服务质量的目的,本研究具有十分重要的现实意义。

## 1.2 国内外研究现状

车辆路径问题(VRP)最早由著名学者 Dantzig 和 Ramser 于 1959 年提出<sup>[2]</sup>,由于该问题将运筹学理论研究和实际生产生活紧密联系起来,因此自提出后即显现出旺盛的生命力,引起了运筹学、应用数学、组合数学、网络分析、图论、计算机应用等学科的专家和运输计划制定者和管理者的极大重视,成为运筹学和组合优化领域的前沿和热点问题。

有时间窗的 VRP (VRPTW)是对经典 VRP 加上时间窗约束(即加上客户要求访问的时间窗口),可以看作是 VRP 的一个特殊类。从集合论的观点,VRPTW 是 VRP 的一个真子集。1985 年, Savelsbergh 证明了 VRPTW 是一个 NP (Non-linear Problem, 非线性规划) 疑难问题,很难求得问题的最优解<sup>[4]</sup>。目前,国内外专家和学者对其求解主要集中在启发式算法上,即求得问题的近似最优解(可行解)。

### 1.2.1 国外研究现状

国外关于车辆路径问题的研究较早,最初由 Dantzig 和 Ramser<sup>[2]</sup>于 1959 年首先提出后,众多学者对其进行了理论研究及试验分析,取得了很大进展。最早研究带时间约束的路径问题是 1981 年 Christofides 在一篇技术报告中对带时间约束的旅行售货员问题(TSPTW)的优化算法研究,而最早发表的研究文献则是 1983 年 Baker 对 TSPTW 问题提出一种最优化算法<sup>[5]</sup>,最早对 VRPTW 问题的算法研究是 1986 年 Solomon<sup>[6]</sup>的启发式算法研究。

在国外,大批学者对带时间窗车辆路径问题进行了深入的研究。如 Clark 和 Wright<sup>[7]</sup>提出了求解该问题的 Clark-Wright 算法(成本节约法), Paessens<sup>[8]</sup>等人通过使用适当的数据结构,降低了它的复杂度,Clark-Wright 法随后成为许多专家和学者研究 VRPTW 的基础,但该算法求出的解是较优的可行解,不一定是最优解。Olatz Arblaitz<sup>[9]</sup>等人提出了解决 VRPTW 的系统设计与分析,并把搜索限制在小范围可行解之内,同时把 META 启发方法和路线规划启发方法结合起来。Jorg Homberger<sup>[10]</sup>等人运用两种进化策略来解决车辆路径问题,并充分利用了 IOTA 启

发技术。Hoong Chuin Lau<sup>[11]</sup>等人提出了两阶段方法来解决车辆路径问题,在第一阶段中运用了新的路径规划启发式算法来获得初始解,第二阶段用禁忌算法来改进解。Wee-Kit<sup>[12]</sup>重点研究了禁忌算法和遗传算法混合的启发式算法,并将两种优势结合起来寻求近似最优解。1981年Fisher<sup>[13]</sup>等人针对带能力约束、时间窗口以及无停留时间的VRP,提出了三下标车辆流方程。Wark<sup>[14]</sup>等人提出了重复匹配的方法,该算法在其模型里同时考虑了时间约束和能力约束,因此适用于这类具有强约束的VRP,另外,重复匹配算法也可求解较大规模的VRP。Gambardella<sup>[15]</sup>等人提出一种MACS(Multiple Ant Colony System)系统用来解决VRPTW,其以蚁群算法为基础,主要设计用来解决两个目标函数的VRPTW,一个是使车辆数最小,另一个是使时间最短。Ombuki等人<sup>[16]</sup>于2002年提出了一种基于遗传算法和禁忌搜索的混合遗传算法,其用遗传算法优化车辆使用数,用禁忌搜索算法优化车辆行驶距离,仿真结果优于各单一算法的求解。Brasy<sup>[17]</sup>于2003年基于可变邻域搜索提出了一个四阶段启发式算法求解了VRPTW。Russell<sup>[18]</sup>等人于2006年利用散射搜索算法求解VRPTW问题;Calvete<sup>[19]</sup>等人于2007年提出了一种目标规划方法用于求解带软时间窗的VRPTW问题。Azi<sup>[20]</sup>等人于2007年基于最短路径方法提出一种两阶段算法求解了VRPTW问题。

### 1.2.2 国内研究现状

我国对于旅行商问题的理论研究较多<sup>[21][22]</sup>,但对于车辆路径问题的研究在90年代以后才逐渐兴起,比国外相对落后。随着客户物质需求的多样性和小规模性以及经济全球化趋势的发展,运输规划的重性日益显著,近年来我国理论界逐渐开始关注车辆路径问题的解决方法,取得了较为显著的成果。但总的来说,我国目前对车辆路径问题的理论研究仍相对薄弱,有待进步的研究。

在带时间窗车辆路径方面,国内学者也做了大量的研究工作。如李大卫<sup>[23]</sup>等人于1998年对适用于TSP的最近距离搜索启发式算法进行修正,构造出评价函数,并据此提出一个求解VRPTW的启发式算法。谢秉磊<sup>[24]</sup>等人于2000年将货运量约束和时间窗约束转化为目标约束,设计了基于自然数编码的可同时处理软、硬时间窗约束的遗传算法,实验分析获得了较好的结果。周贤伟和李光远<sup>[25]</sup>于2001年根据车辆装载GPS设备的特性,建立了货物运输VRPTW的数学模型,并设计了求解的遗传算法。张丽萍<sup>[26]</sup>等人于2002年通过引入新颖交叉算子,构造了一种改进遗传算法。该算法摆脱了对群体多样性的要求,不存在传统遗传算法常见的“早熟收敛”问题,可用于解决VRPTW。宋厚冰和蔡远利<sup>[27]</sup>于2003年针对VRPTW,在标准遗传算法的基础上,将分组信息与每一个染色体结合,并辅以补

交换局部搜索技术,构造了一种改进遗传算法,使得求解结果更接近最优解。宾松和符卓<sup>[28]</sup>于 2003 年通过引用一种新的编码方法及交叉和变异概率的自适应机制,构造了改进遗传算法来求解带软时间窗的 VRPTW。刘小兰<sup>[29]</sup>等人于 2004 年为了克服原有大规模邻域搜索算法不能有效求解时间窗较宽 VRP 的缺陷,介绍了 VRPTW 的通用数学模型。通过分析各主要变量之间的关系,构造了一种简单、快速的确定性初始算法,并通过引入“短路径优先策略”,构造了改进的大规模邻域搜索算法。刘诚<sup>[30]</sup>等人于 2005 年针对遗传算法在求解 VRPTW 时初始种群的单一性,提出一种并行遗传算法,该算法对不同的种群用不同的初始化方法,效果比较理想。杨宇栋<sup>[31]</sup>等 2006 年引入客户直接排列的解的表示方法,改进了模拟退火算法,提高了求解 VRPTW 的效率。张丽艳<sup>[32]</sup>等于 2006 年利用粒子群算法和模拟退火算法的混合算法求解了 VRPTW 问题。张海刚<sup>[33]</sup>等于 2007 年研究了基于改进免疫遗传算法的带硬时间窗车辆调度问题的实现,并对算法的信息熵计算加以改进,显著提高了算法的全局收敛可靠性和收敛速度。

纵观国内外相关研究现状,可以得出以下几个结论:

1) 国内外对带时间窗车辆路径问题的模型研究方面已有不少成果,但还不够完善。不同研究人员思考问题角度不同,决定了建立模型时考虑因素有差异,如何构建更贴切现实的理论模型已成为研究者们关注的焦点。

2) 配送不但要满足客户的实物需求还要满足客户的时间需求,特别是在客户要求尽量减少库存的前提下,及时配送变得越来越重要,因此考虑满足客户的时间窗口约束的车辆路径问题需引起足够重视。

3) VRPTW 的复杂性决定了其相应求解算法研究的重要性,当前国内外学者的研究多集中在算法创新,即如何构造运算简单、寻优性能优异的求解方法。其中遗传算法作为解决大规模问题的现代启发式算法,具有搜索能力强、鲁棒性强以及计算效率高等特点和优势,如何进一步提高遗传算法的寻优性能成为研究的重要方向。

### 1.3 本文的研究方法及主要内容

本文以带时间窗车辆路径问题的相关理论为基础,设立一定的假设条件并结合配送中心的条件及客户需求等实际情况,将配送距离、配送及时性以及配送车辆数作为目标函数的主要考虑因素,建立带时间窗车辆路径问题的数学模型,并为模型的求解提供切实可行的遗传算法设计,最后通过算例验证模型和算法的合理性及有效性。

本文的技术路线如图 1-1 所示。

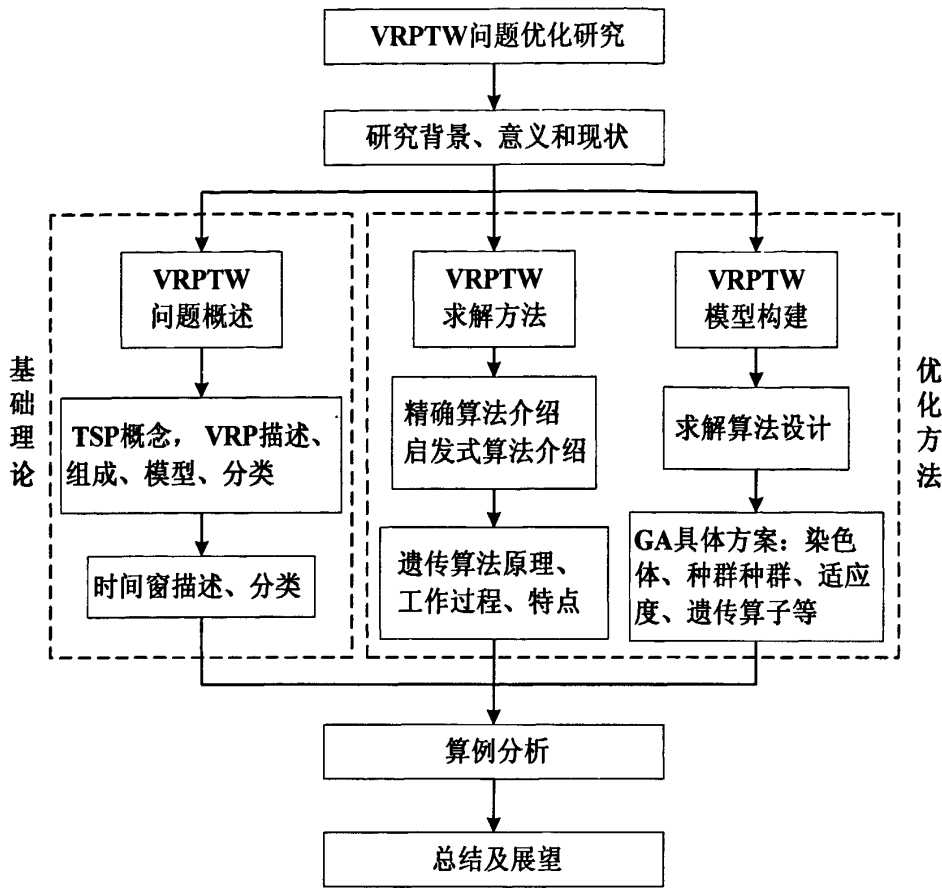


图 1-1 论文技术路线图

Fig 1-1 The Technology Roadmap of Thesis

论文各章的具体内容安排如下：

第一章：绪论。本章介绍论文的研究背景与意义，国内外关于 VRPTW 的研究现状以及论文研究的主要内容。

第二章：车辆路径问题概述。本章首先引入旅行商问题，然后介绍辆路径问题（VRP）的基本描述、组成以及分类，进而引出带时间窗的车辆路径问题（VRPTW）的概念、时间窗分类等，为论文整体思路的建立打下坚实的理论基础。

第三章：带时间窗车辆路径问题的求解算法研究。本章首先对求解带时间窗车辆路径问题的各种优化算法进行综述，并特别对全局搜索能力较强的遗传算法进行详细介绍，包括其原理及构成要素、工作流程、特点等，最后对各种优化算法进行了归纳和比较，为本文模型求解算法的选择提供参考和依据。

第四章：VRPTW 模型构建与算法设计。本章在对问题目标和约束条件合理分析的基础上，设定表示服务时间窗的惩罚函数，建立了考虑配送距离成本、车辆启动成本及惩罚成本的 VRPTW 数学模型，并设计了求解 VRPTW 的遗传算法。

第五章：算例分析。本章选取算例对本文研究的问题进行试验，分别求出基本遗传算法和改进遗传算法下的 VRPTW 解决方案，并通过对试验结果的对比分析，验证本文所建模型及算法设计的可行性和有效性。

第六章：总结及展望。通过对全文内容的归纳和总结，提出 VRPTW 进一步研究的方向。

## 1.4 本章小结

本章首先阐述了论文的研究背景及意义，然后对带时间窗的车辆路径问题（VRPTW）的国内外研究现状进行了综述，结合车辆路径问题的实际情况和研究现状引出论文的研究对象，最后概括说明了论文的主要内容与论文框架。

## 2 车辆路径问题概述

车辆路径问题 (VRP) 是从旅行商问题 (TSP) 演化而来, 旅行商问题是车辆路径问题的一个特例, 它只包括一条路径问题, 且没有能力约束的。而带有时间窗约束的车辆路径问题 (VRPTW) 是在车辆路径问题基础上根据学术研究和实际应用延伸和变化的一种型态, 它们三者之间是一脉相承的、相互联系的。下面将分别予以介绍。

### 2.1 旅行商问题 (TSP)

旅行商问题 (Traveling salesman Problem, 简称 TSP)<sup>[35]</sup> 是最著名的同时也是最简单的车辆路径问题, 旅行商问题是 VRP 问题的一个特例 (即当 VRP 问题中只包括一条路径, 且没有能力约束的时候, 就成为旅行商问题)。TSP 问题是运筹学、图论及组合优化中的著名难题, 由于其有广泛的应用背景, 引起了人们的极大兴趣, 现已归入 NP 问题类。TSP 本身可以直接用于解决类似 TSP 的最优巡回线路等问题。

TSP 一般描述为: 旅行商从驻地出发, 经每个目标城市至少一次后返回原地, 应如何安排其旅行线路, 才能使总的旅行距离 (或时间、费用等) 最少。一个典型的 TSP 问题示意图如图 2-1 所示。

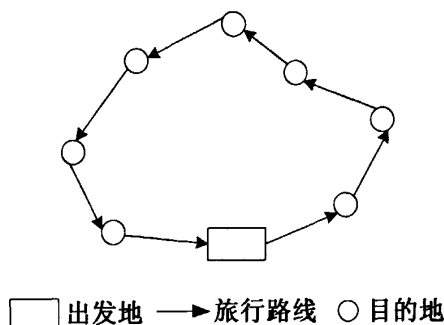


图 2-1 TSP 示意图

Fig 2-1 Schematic Diagram of TSP

旅行商问题 (TSP) 通过扩展旅行商的数目进而可以得到多旅行商问题 (m-TSP), m 个旅行商要遍历所有城市且每个城市只被一个旅行商访问。在 m-TSP 问题中, 给每个旅行商 (车辆) 加上容量约束就得到了一般车辆路径问题 (VRP), 然后继续将各种约束条件加入到问题的实际模型中, 就可以得到各种车辆路径问题。

## 2.2 车辆路径问题 (VRP)

### 2.2.1 车辆路径问题的描述

车辆路径问题 (Vehicle Routing Problem, 简称 VRP) 这一名词最早是在 20 世纪 50 年代末期由线性规划大师 Dantzig 和 Ramser 提出<sup>[2]</sup>, 它主要解决的是在已知客户位置及货物需求量的情况下, 如何派车, 派多少辆车, 每辆车走什么样的路线才能既保证满足客户要求, 又能使成本最低的问题。车辆路径问题是组合优化领域中著名的 NP-hard 问题之一, 与众多的实际问题都有相似性, 如铁路运输、公交调度、水道航线、路由选择等, VRP 研究具有相当大的实际意义。

车辆路径问题 (VRP) 的一般定义为: 对一系列给定的客户 (送货点或取货点), 确定适当的配送车辆行驶路线, 使其从配送中心出发, 有序地通过它们, 最后返回配送中心, 并在满足一定的约束条件下 (如货物需求量、发送量、车辆容量限制、行驶里程限制、时间限制等), 达到一定目标 (如路程最短、费用最少、时间尽量少、使用车辆数尽量少等)<sup>[21]</sup>。图 2-2 为单一配送中心的 VRP 图示。

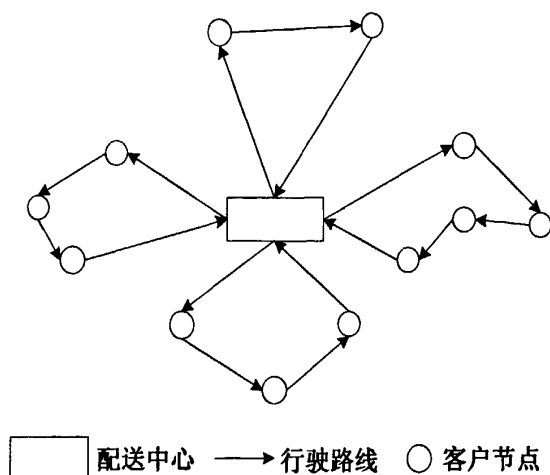


图 2-2 VRP 示意图

Fig 2-2 Schematic Diagram of VRP

### 2.2.2 车辆路径问题的组成要素

车辆路径问题主要包括货物、车辆、配送中心、客户、运输网络、约束条件和目标函数等要素<sup>[36]</sup>。

#### (1) 货物

货物是配送的对象。可将每个客户需求 (或供应) 的货物看成一批货物。每



批货物都包括品名、包装、重量、体积、要求送到（或取走）的时间和地点、能否分批配送等属性。货物的品名和包装，是选用配送车辆的类型以及决定该批货物能否与其它货物装在同一车辆内的依据。货物的重量和体积是进行车辆装载决策的依据。当某个客户需求（或供应）货物的重量或体积超过配送车辆的最大装载重量或容积时，则该客户将需要多台车辆进行配送。货物的送到（或取走）时间和地点是制定车辆的出行时间和配送路线的依据。允许货物分批配送是指某个客户的需求（或供应）的货物可以用多辆车分批送到（或取走），即使其需求（或供应）量小于一辆车的最大装载量。

## （2）车辆

车辆是货物的运载工具。其主要属性包括车辆的类型、装载量，一次配送的最大行驶距离、配送前的停放位置及完成任务后的停放位置等。车辆的类型有通用车辆和专用车辆之分，通用车辆适于装运大多数普通货物，专用车辆适于装运一些性质特殊的货物车辆的装载量是指车辆的最大装载重量和最大装载容积，是进行车辆装载决策的依据。在某配送系统中，车辆的装载量可以相同，也可以不同。对每台车辆一次配送的行驶距离的要求可分为以下几种情况：

- 1) 无距离限制；
- 2) 有距离限制；
- 3) 有距离限制，但可以不遵守，只是不遵守时需另付加班费。

车辆在配送前的停放位置可以在某个停车场、物流中心或客户所在地。车辆完成配送任务后，对其停放位置的要求可分为以下几种情况：

- 1) 必须返回出发点；
- 2) 必须返回某停车场；
- 3) 可返回到任意停车场；
- 4) 可停放在任何停车场、物流中心或客户所在地。

## （3）配送中心

配送中心是指进行集货、分货、配货、配装、送货作业的地点。在某配送系统中，配送中心的数量可以只有一个，也可以有一个以上配送中心的位置可以是确定的，也可以是不确定的。对于某个配送中心中心，其供应的货物可能有一种，也可能有多种，其供应的货物数量可能能够满足全部客户的需求，也可能仅能满足部分客户的需求。

## （4）客户

客户也称为用户，包括分仓库、零售商等。客户的属性包括需求（或供应）货物的数量、需求（或供应）货物的时间、需求(或供应)货物的次数及需求（或供应）货物的满足程度等。在某个配送系统中，某个客户的需求（或供应）货物的

数量可能大于车辆的最大装载量,也可能小于车辆的最大装载量;某客户的需求(或供应)货物的时间是指要求货物送到(或取走)的时间,对配送时间的要求可分为以下几种情况:

- 1) 无时间限制;
- 2) 要求在指定的时间区间内完成运输任务;
- 3) 有时间限制,但可以不遵守,只是不遵守时要给予一定的惩罚。。

某客户对需求(或供应)货物的满足程度的要求可分为两种情况:

- 1) 要求全部满足;
- 2) 可以部分满足,但不满足时要受到惩罚。

#### (5) 运输网络

运输网络是由顶点(指配送中心、客户、停车场)、无向边和有向弧组成的。边、弧的属性包括方向、权值和交通流量限制等。运输网络中边或弧的权值可以表示距离、时间或费用。边或弧的权值变化分为以下几种情况:

- 1) 固定,即不随时间和车辆的不同而变化;
- 2) 随时间不同而变化;
- 3) 随车辆的不同而变化;
- 4) 既随时间不同而变化,又随车辆不同而变化。

对运输网络权值间的关系可以要求其满足三角不等式,即两边之和大于第三边,也可以不加限制。对运输网络中顶点、边或弧的交通流量要求分为以下几种情况:

- 1) 无流量限制;
- 2) 边、弧限制,即每条边、弧上同时行驶的车辆数有限;
- 3) 顶点限制,即每个顶点上同时装、卸货的车辆数有限;
- 4) 边、弧、顶点都有限制。

#### (6) 约束条件

车辆路径优化问题应满足的约束条件主要包括:

- 1) 满足所有客户对货物品种、规格、数量的要求。
- 2) 满足客户对货物发到时间范围的要求。
- 3) 在允许通行的时间进行配送(如有时规定白天不能通行货车等)。
- 4) 车辆在配送过程中的实际载货量不得超过车辆的最大允许装载。

#### (7) 目标函数

车辆路径问题经常选用的目标函数主要有:

1) 配送总里程最短。配送里程与配送车辆的耗油量、磨损程度以及司机疲劳程度等直接相关,它直接决定运输的成本,对配送业务的经济效益有很大影响。

2) 配送车辆的吨位公里数最少。该目标将配送距离与车辆的载重量结合起来考虑,即以所有配送车辆的吨位数(最大载重吨数)与其行驶距离的乘积的总和最少为目标。

3) 综合费用最低。降低综合费用是实现配送业务经济效益的基本要求。在物流配送中,与取送货有关费用包括:车辆维护和行驶费用、车队管理费用、货物装卸费用有关人员工资费用等。

4) 准时性最高。由于客户对交货时间有较严格的要求,为提高配送服务质量,有时需要将准时性最高作为确定配送路线的目标。

5) 运力利用最合理。该目标要求使用较少的车辆完成配送任务,并使车辆的满载率最高,以充分利用车辆的装载能力。

6) 劳动消耗最低。即以司机人数最少、司机工作时间最短为目标。

### 2.2.3 车辆路径问题的数学模型

一般车辆路径问题在满足以下约束条件下,求解 $m$ 辆车的总行驶成本最小:

- (1) 每条路径的起点和终点都是供货点;
- (2) 每个客户只应该被车辆服务一次;
- (3) 每条路径中的所有客户需求的总和不能超过车辆的载重量 $Q$ 。

车辆路径问题模型用数学语言可以描述为<sup>[21]</sup>:

$$y_{ki} = \begin{cases} 1, \text{点}i\text{的任务由车辆}k\text{完成} \\ 0, \text{否则} \end{cases} \quad (i = 0, 1, \dots, n; k = 1, 2, \dots, K)$$

$$x_{ijk} = \begin{cases} 1, \text{车辆}k\text{从点}i\text{行驶到}j\text{完成} \\ 0, \text{否则} \end{cases} \quad (i, j = 0, 1, \dots, n; k = 1, 2, \dots, K)$$

以最小化系统运行费用为目标,建立如下车辆路径问题模型:

$$\text{Min } f = \sum_i \sum_j \sum_k c_{ij} x_{ijk} \quad \dots(2-1)$$

$$\text{s.t. } \sum_i q_i y_{ki} \leq Q, (k = 1, 2, \dots, K) \quad \dots(2-2)$$

$$\sum_k y_{ki} = 1, (i = 1, 2, \dots, n) \quad \dots(2-3)$$

$$\sum_k y_{k0} = K \quad \dots(2-4)$$

$$\sum_i x_{ijk} = y_{ki} \quad (j = 1, 2, \dots, n) \quad \dots(2-5)$$

$$\sum_j x_{ijk} = y_{ki} \quad (i = 1, 2, \dots, n) \quad \dots(2-6)$$

$$x_{ijk} \in \{0,1\} \quad (i,j=0,1,\dots,n; \quad k=1,2,\dots,K)$$

...(2-7)

$$y_{ki} \in \{0,1\} \quad (i=0,1,\dots,n; \quad k=1,2,\dots,K)$$

...(2-8)

模型中，配送中心编号为 0，客户点编号为 0,1,2...n，配送中心和客户点均用点*i*(0,1,2...n)来表示。

目标函数（2-1）是使车队总的服务费用最低；约束（2-2）为车辆装载能力约束，即每条配送线路*k*的送货量不能超过车的最大载重量；约束（2-3）保证每个客户都被服务；约束（2-4）保证车辆都从配送中心出发并返回到配送中心；约束（2-5）、（2-6）保证如果客户点*i, j*在车辆*k*的行驶线路上，那么客户点*i, j*将由车辆*k*服务。

2.2.4 车辆路径问题的分类

车辆路径问题被提出后，Linus, Bodin 和 Golden、Bodin、Ehlers、Lenstra 和 Savelsbergh 等许多学者对 VRP 从不同角度按不同目标进行了多种分类<sup>[37]</sup>。如按任务特征可以划分为纯装问题或纯卸问题和装卸混合问题；按车辆载货状况可分为满载问题和非满载问题；按车场（或货场、配送中心等）数目分类可分为单车场问题和多车场问题；按车辆类型数可以划分为包括单车型问题和多车型问题；按车辆对车场的所属关系可以划分为车辆开放问题和车辆封闭问题；按运送货物类型可分为单一货物和多种货物，多种货物又包括可相容货物和不相容货物。按优化目标数可划分为单目标问题和多目标问题等等。

根据上述分析可以将 VRP 问题的主要类型归结为如下表 2-1。

表 2-1 车辆路径问题分类一览表

Tab 2-1 Table of the Classification of VRP

分类依据	类型	问题描述
任务特征	纯装问题或纯卸问题	车辆在任务点装货或卸货，即集货或送货问题
	装卸混合问题	每项任务有不同的装货点或卸货点，即集送货一体化问题
车辆载货状况	满载问题	货运量不小于车辆容量
	非满载问题	货运量小于车辆容量
车场数目	单车场问题	车场是唯一的
	多车场问题	车场是不唯一的

表 2-1 车辆路径问题分类一览表（续）

Tab 2-1 Table of the Classification of VRP

车辆类型	单车型问题	所有车辆类型和容量相同
	多车型问题	执行任务的各车辆的类型和容量不完全相同
车辆对车场所属关系	车辆开放问题	车辆可以不返回其出发车场
	车辆封闭问题	车辆必须返回其出发车场
运送货物类型	单一货物	货物为同种类型的货物
	多种货物	运输货物的种类多于一种
优化目标数	单目标问题	某一项指标较优或最优
	多目标问题	同时要求多项指标较优或最优

根据上述约束条件和任务要求的不同组合，车辆路径问题在学术研究和实际应用上产生了许多不同的延伸和变化型态，包括带能力约束的车辆路径问题（Capacitated Vehicle Routing Problems, CVRP）、带时间窗的车辆路径问题（Vehicle Routing Problems with TimeWindows, VRPTW）、追求最佳服务时间的车辆路径问题（Vehicle Routing Problems with Defined Time, VRPDT）、多车种车辆路径问题（Fleet Size and mix Vehicle Routing Problems, FSVRP）、车辆多次使用的车辆路径问题（Vehicle Routing Problems with Multiple use of Vehicle, VRPPM）、随机需求车辆路径问题（Vehicle Routing Problems with Stochastic Demand, VRPSD）、动态车辆路径问题（Dynamc Vehicle Routing Problems, DVRP）、考虑收集的车辆路径问题（Vehicle Routing Problems with Backhauls, VRPB）、满载或非满载 VRP、双向 VRP 等<sup>[38]</sup>。以上分类中带有时间窗约束的车辆路径问题（VRPTW）是本文的主要研究对象，将在下一节进行详细阐述。

2.3 带时间窗的车辆路径问题（VRPTW）

随着现在人民生活水平的提高，消费观念也正在发生转变，消费者要求商家提供便捷、周到的服务，对商品配送提出了更高的要求。在配送过程中，如果客户要求在一定的时间范围内被访问，则配送路径问题就成为带时间窗的车辆路径问题（Vehicle Routing Problem with Time Windows, 简称 VRPTW）。下面的内容为此类问题的详细介绍。

### 2.3.1 时间窗的概念及描述

时间窗是一个时间段 $[e_i, l_i]$ ，是由客户 $i$ 要求的最早服务时间 $e_i$ 和最晚服务时间 $l_i$ 确定的一个服务时间区间。带时间窗的车辆路径问题（VRPTW）是一般车辆路径问题（VRP）的扩展问题，即在VRP问题的基础上给每个客户加上服务所允许的时间窗（Time Windows）约束，VRP就扩展成为了VRPTW。

VRPTW问题可简单描述为：在不违背车辆容量限制、时间限制等约束条件的前提下，合理制定运输时的车辆配送路径方案，以尽可能小的成本满足处在不同地理位置的客户对货物和服务时间的要求。

若配送中心违反了客户的时间窗约束，则势必会产生一定的损失，这些损失会产生因违反客户时间窗约束而导致的时间效应成本，配送中心在追求成本最低化的情况下，有必要将时间效应成本纳入到考虑范围之内。产生时间效应成本的因素包含以下几个方面<sup>[39]</sup>：

（1）客户为减少库存的压力而要求配送中心及时（Just In Time）配送，若配送车辆不能在约定的时间窗内到达，必须赔偿客户的损失或者甚至被拒收；

（2）配送中心的服务质量和商业信誉也会因违反时间窗而受到一定的损失，或许会损失一些潜在的客户，这种损失也需要适当的加以量化表示；

（3）车辆若在客户的最早收货时间之前到达，将产生等候时间，因为等候时车辆及人员闲置，并无实际产出，所以会产生人员闲置成本和机会成本，而且有可能因车辆为了等候而不当的停放造成城市交通拥挤，此时所牵涉的成本已包括额外的社会成本，这也是过去所忽略的问题。

### 2.3.2 时间窗的分类

VRPTW相对于VRP，必须额外考虑到运送时间与时间窗口，故对VRP问题除了空间方面的路径（Routing）考虑之外，还必须要加上时间安排上的（Scheduling）考虑。时间窗的划分方式有以下两种<sup>[40]</sup>：

#### （1）按形式来分类

按形式来分，时间窗有单边和双边时间窗之分，双边时间窗是指客户要求在某段时间中获得服务，常见的单边时间窗对应的车辆路径问题为带最后期限（Deadline）的VRP（VRPTD），即从双边时间窗中去掉了对最早服务时间的限制，客户不再要求车辆最早服务时间，而只要求在某个时间点之前为其提供服务。目前，研究较多的是带双边时间窗的车辆路径问题，即只有在客户指定的最早时间之后和最晚时间之前，客户才能接受服务。若车辆在客户指定的最早时间之前到

达客户所在地，则必须等待到了客户要求的最早时间，才能对客户提供服务；车辆还必须是最晚服务时间之前为客户提供货物或服务，否则就视为延误，可以采取一定的惩罚措施。

(2) 按客户满意度分类

由于 VRPTW 问题的复杂性，如果严格按照客户设定的服务时间为其服务，可能造成企业的配送成本增加。如果允许在某些客户点适当地延误，可能使运输成本大为减少，即对该延误现象给予一定的惩罚。所以根据决策者在对客户满意度和成本二者权衡时偏好的不同，按照客户满意度来分，时间窗还可以分为硬时间窗 (Hard Time Windows, 简称 HTW) 和软时间窗 (Soft Time Windows, 简称 STW) 以及混合时间窗 (Mixed Time Windows, 简称 MTW) 三种情况。

1) 硬时间窗 (Hard Time Windows): 指配送车辆必须在规定时间段内将配送货物送达到客户手中，客户拒绝接受在此时间段之外提供的服务。如图 2-3 为一惩罚函数 (Penalty Function)，当配送货物送达时间超过了规定的时间段 ( $e, l$ )，其惩罚值  $P(t)$  等于一个非常大的正值，以表示硬时间窗的限制。

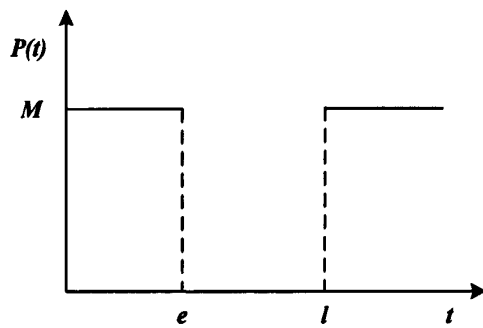


图 2-3 硬时间窗

Fig 2-3 Hard Time Windows

硬时间窗的车辆路径问题 (VRPHTW) 中，车辆可以在最早服务时间之前到达客户所在地，但必须等待直至到了最早服务时间才能对客户提供服务；另外车辆不能在最晚服务时间之后到达客户所在地。如在 Just-In-Time 生产系统中，送货车辆到达时间若迟于指定的最晚到达时间，会导致整个生产线的延误和闲置，造成时间成本的增加和效率的降低，因此这种情况下往往不允许违背时间窗的约束。对于 VRPHTW 问题来说，服务所有客户所需的车辆数目也是决策变量，一般来说，所用车辆越少，对应的成本也越小，如何用最少的车辆在不违背车辆容量及客户时间窗的约束的前提下，为所有客户提供服务，也是决策者关心的一个问题。

2) 软时间窗 (Soft Time Windows): 指配送车辆如果无法将货物在特定的时段 (如图 2-4 中的 ( $e, l$ )) 内送到客户手中，则必须按照违反时间的长短施以一定的罚金或其它惩罚法则。图 2-4 就是一种可能的惩罚函数。

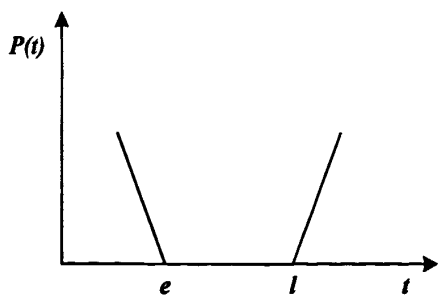


图 2-4 软时间窗

Fig 2-4 Soft Time Windows

带软时间窗的车辆路径问题（VRPSTW）相对 VRPHTW 来说放松了对时间窗的约束，这是因为在实际情况中，由于道路交通流量，车辆的行驶速度以及客户的需求时间等不确定因素导致货物无法在规定的时间内送到，如果采用硬时间窗的约束进行优化，将导致成本非常之高，甚至可能无解。而 VRPSTW 只要求车辆应尽可能在客户指定的时间窗内以最小的总成本为所有客户提供服务，允许一定程度的延误现象。显然，VRPSTW 较 VRPHTW 具有更好的通用性。在实际运输规划中，决策者往往需要在客户满意和成本二者之间权衡，偏好不同决定了可以允许延误的程度不同。在 VRPSTW 中，车辆如果在最早服务时间之前到达客户所在地，需等待直至到了最早服务时间后才能提供服务；如果在最晚服务时间之后到达客户所在地，则可能导致客户满意度降低，隐性成本的增加。因此应尽量避免延误现象的发生，如果发生，应加以惩罚，惩罚的具体程度可由决策者设定。

3) 混合型时间窗（Mixed Time Windows）：在系统中有些客户属于硬时间窗，有些属于软时间窗：对同一客户，又可能软、硬时间窗混合使用。在实际的配送工作中，车辆如果能在最佳时间段（如图 2-5 中  $(e, l)$ ）内将货物送达到客户手中，则不处罚；若在图 2-5 中的  $[a, e)$  或  $(l, b]$  时段内才送达，则客户的满意度降低，将处以一定的惩罚（转化为惩罚函数  $P(t)$ ），并且客户不接受在上述两个时间段以外的时间  $[-\infty, a)$  或  $(b, \infty]$  收货。

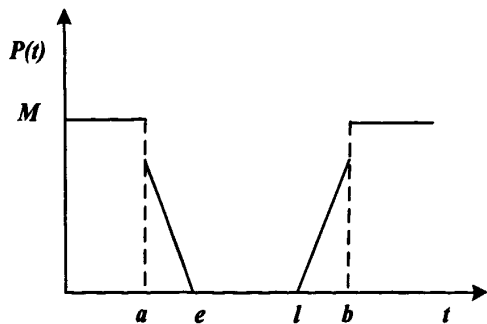


图 2-5 混合时间窗

Fig 2-5 Mixed Time Windows



VRPSTW 与 VRPHTW 的区别仅在于处理延误的方式, VRPSTW 允许发生延误, 但对延误发生的程度和频度有一定限制, 这是通过对延误的惩罚程度来体现的。在对 VRPSTW 建模时, 可通过提高对延误现象的惩罚, 使惩罚成本高至不允许延误的发生来解决 VRPHTW。同时, 可以把硬时间窗看作是惩罚项为零的一种情况, 对 VRPSTW 建模时, 表现为目标函数增加了一项惩罚项, 具体体现在对应服务成本的增加。本文研究的时间窗主要倾向于软时间窗, 即 VRPSTW, 这是因为考虑到在实际情况中的不确定因素, VRPSTW 较 VRPHTW 放松了对时间窗的约束, 具有更好的通用性。

## 2.4 本章小结

本章主要对车辆路径问题进行概述, 为本文以后的章节打下理论基础。本章首先引入了经典的旅行商问题 (TSP), 然后阐述了一般车辆路径问题 (VRP) 的描述、组成要素、基本模型及其分类, 进而引出带时间窗约束的车辆路径问题 (VRPTW), 并对三类时间窗, 即硬时间窗、软时间窗和混合型时间窗进行了详细分析, 为本文后续章节进行带时间窗车辆路径问题的数学模型构建与求解算法设计打下了坚实的理论基础。

3 带时间窗车辆路径问题求解算法研究

车辆路径问题（Vehicle Routing Problem，简称 VRP）是极具魅力的组合优化问题，半个世纪以来，许多的专家学者从车辆路径优化问题的基本问题出发，根据不同的约束和目标，构建了不同的模型，并有针对性地开发出了不同的算法。综合来看，用于求解 VRPTW 的方法大致可以分为两大类：精确算法和启发式算法，而启发式算法又可以分为传统启发式算法和现代启发式算法。精确算法是针对具体的问题和模型利用数学规划技术可求出最优解的算法；启发式算法主要是基于直观或者凭借经验，开发出能够朝着最优解的方向搜索或靠近的一类算法。精确算法的计算量一般随问题规模的增大呈指数增长，只有在需点数和路段数较少时才有可能寻求其精确解，因此，如何针对车辆路径问题的特点，构造运算简单、寻优性能优异的启发式算法，这不仅对于配送系统而且对于许多可转化为车辆路径问题求解的组合优化问题都具有十分重要的意义。

下面对 VRPTW 问题的主要求解方法予以综述，为了清晰展现各种算法之间的关系，作出分类图 3-1 来示意。

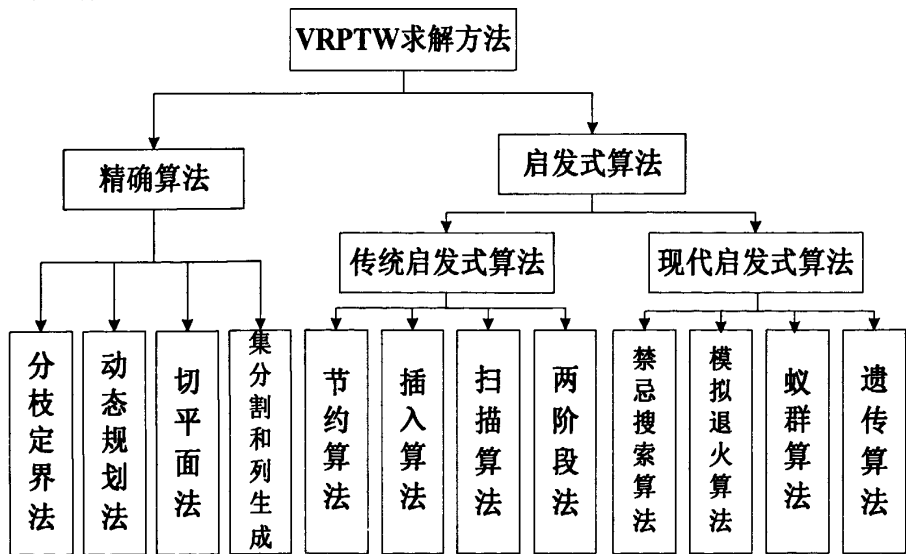


图 3-1 VRPTW 求解方法分类图

Fig 3-1 Category Map of Solving VRPTW

3.1 精确算法

精确算法是基于运筹学的优化算法，通常运用线性规划（包括经过了专门处理的分枝定界法、割平面方法和标号法）和非线性规划等数学规划技术，以求得

问题的最优解,是理论上的准确求解方法。其中常用于求解 VRPTW 的精确算法主要有如下几种方法:

#### (1) 分枝定界法

该算法是 Fisher<sup>[41]</sup>最早提出的,它的核心思想是将解空间划分为若干个小的子空间,再依次搜索各个子空间,通过与当前最好解的比较,可以剪去那些不可能产生最优解的分枝,只搜索那些能产生最优解的分枝。Araque<sup>[42]</sup>曾利用分支定界法求解了车辆路径问题。Kolen 曾利用此方法求解含时间窗约束的车辆路径问题,其实验的节点数范围为 6-15。当节点数为 6 时,计算机演算所花费的时间大约 1 分钟;当节点数增长至 12 时,计算机有内存不足的现象产生,所以分枝定界法比较适用于求解小型问题。

#### (2) 动态规划法

Psarafis<sup>[43]</sup>提出了一种动态规划算法,用来规划车辆配送路线,该方法只能解决访问点很少的调度问题。此方法的核心观念为最佳化原理,即运用数学逻辑来处理一连串相互关连的决策问题,并采取系统化步骤以求得使整体最有利的策略。求解主要步骤如下:

- 1) 将整个问题分解为许多相互关联的局部问题,以便分别分析处理;
- 2) 从最后一个阶段的各种状态中,找出最有利的决策方案;
- 3) 利用“反策推算法”,自最后一个阶段逐步向前项阶段推进,直到各阶段的最有利策略均找出为止。

#### (3) 切平面法

Padberg 和 Rinaldi<sup>[44]</sup>首次利用切平面法求解大型旅行商问题。此方法与分枝定界法类似,也是在求解与整数规划相对应的线性规划上,不断增加新的约束,也就是另外加入线性约束条件,以切掉对应于非整数规划的所有可行解的集合,使问题可达到整数线性规划求解的形式,从而获得最优解。

#### (4) 集分割和列生成

VRP 的集分割是由 Balinski<sup>[45]</sup>等人提出的。它直接考虑可行解集合,在此基础上进行优化,因此建立的 VRP 模型最为简单。其缺陷在于如果问题所受约束不严格,则所需要计算的状态空间非常大。另外,要确定每个可行解的最小成本也是件困难的事情。对于其中规模相对较小、约束严格的问题,可通过线性松弛,引入割平面进行求解。于是 Rao<sup>[46]</sup>等人引入了列生成方法进行求解。在该方法中,原问题被转化为简化问题,考虑的范围是所有可能的可行解的子集。在此基础上反复求解。通过引入优化对偶变量 $\lambda$ ,对该简化问题松弛,通过计算列的最小边际成本,确定最优解。其算法本质上是最短路径算法,同时结合了分枝定界算法。

## 3.2 启发式算法

启发式算法 (Heuristic Algorithm) 是指一种基于直观或经验构造的算法<sup>[47]</sup>, 该算法不需要或需要很少的关于问题的先验信息, 在与研究问题有关的模型及算法中寻求其间的联系, 从中得到启发, 去发现适于解决该问题的思路和途经。启发式方法作为一种逐次逼近的算法, 能适应不同领域的优化问题求解, 并在大多数情况下都能得到比较满意的解。因此, 启发式算法是目前关于复杂优化问题求解的一类有效方法, 已成为解决配送路径优化问题的重要方法。对于启发式算法大致上可以将其分为两类: 传统启发式算法 (Classical heuristics) 和现代启发式算法 (Meta-heuristics), 大部分传统启发式算法是在 1960-1990 年间提出的, 而现代启发式算法是在 1990 年以后出现和发展的。下面分别予以简要介绍。

### 3.2.1 传统启发式算法

传统启发式算法 (Classical heuristics) 是搜索解空间中相对有限的一部分子空间, 力求在比较合理的时间内取得比较好的解的方法。常用于求解 VRPTW 问题的传统启发式算法主要有节约算法、插入算法、扫描算法、两阶段算法。

#### (1) 节约算法

节约算法是求解 VRP 和 VRPTW 的一类最为经典的启发式算法之一, 该算法最早由 Clark 和 Wright<sup>[7]</sup>于 1954 年提出, 通常也简称为 C-W 算法。该算法的思想是: 根据顾客点之间连接可以节省的距离 (节约值) 最大的原则, 将不在线路上的顾客点依次插入到路线中, 直到所有的点都被安排进路线为止。对于 VRPTW, 这种算法的复杂度为  $O(n^4)$ 。Paesens<sup>[8]</sup>等人通过使用适当的数据结构和技巧, 使其复杂度降低为  $O(n^3)$ 。节约算法两节点的节约值的计算公式与意义如下所示:

$$s(i, j) = d(i, 0) + d(0, j) - d(i, j)$$

其中  $d(i, 0)$  表示从顾客  $i$  到场站的距离,  $d(i, j)$  表示从顾客  $i$  到顾客  $j$  的距离。算法开始时, 以所有客户节点均采用直接往返的送货方式作为初始可行安排。然后计算任意两个节点的节约值, 并将节约值按由大到小的顺序进行排列。最后根据排序结果以及可行性条件, 对路径进行合并, 直到无法找到更好的解。图示如 3-2 所示。

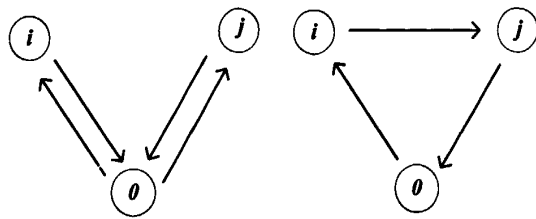


图 3-2 节约算法示意图

Fig 3-2 Schematic Diagram of Saving Algorithm

(1) 插入算法

插入算法是指通过第  $k$  步迭代，将第  $k$  个节点插入到路线中。算法的关键在于确定在第  $k+1$  步可以被插入到路线中的点以及该点的最佳插入位置。因此，该算法由两个关键的部分组成：第一部分是节点选择阶段，即确定下一步被插入到路线中的顾客节点；第二部分是路径插入阶段，即确定所选择的顾客节点在路线中的最佳插入位置。最初是 Solomon<sup>[48]</sup> 将该方法引进到带有时间窗口的车辆路径问题中。这种方法是逐个将未访问过的客户加入到路径当中。假设一条路径为  $R_p = \{c_1, \dots, c_m\}$ ，下一个加入到当前路径的客户是如下选取的：计算所有未访问客户的最佳插入点；把一个客户插入到  $R_p$  中的任何一条边，计算新路径的总旅行距离，选出使总旅行距离达到最小的位置。然后在未访问客户中选择一个使总旅行距离最小的客户，即是下一个加入到  $R_p$  中的客户，将其插入到最佳位置。重复此过程，直到插入任何一个未访问过的客户或者导致当前路径超载，或者不满足该客户的时间窗口时，则重新开辟一条路径。再按照一定的规则选出一条路径的第一个客户。再重复上面的过程，直到所有的客户都插入到合适的路径为止。

(3) 扫描算法

扫描算法最早由 Gillett 和 Miller<sup>[49]</sup> 于 1974 年提出的一种直接针对实际问题的基于平面几何的求解方法。算法的思想是：先以极坐标的方式表示各个客户点的区域位置，再任取一个客户点作为起点以顺时针（逆时针）方向，以车辆容量为限制条件（或再加入其它限制条件）对区域进行分割，使不超过车辆容量（或满足其它限制条件）的需求点组成一个区域，将一个区域指派一辆车。在形成区域后，再对每一组区域中的各点安排路线。不断地分组构造路径，直到所有的点均已分配为止。如图 3-3 所示。

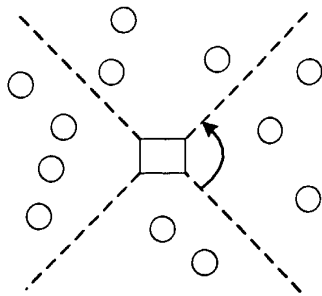


图 3-3 扫描算法示意图

Fig 3-3 Schematic Diagram of Sweeping Algorithm

Solomon<sup>[50]</sup>于 1983 年将此方法应用于求解 VSPTW 问题，与原扫描法不同点在于最后的求解各顾客群排程，以插入法进行各顾客群的排程，并检查时间可行性。若有顾客点无法满足时间窗的约束，则先排除此顾客点；若所有的顾客群都已排入行程，则所有的顾客点都已被服务，则完成路线的建构；若有顾客点尚未被服务，则沿原扫描方向，将剩余的尚未被服务的顾客点重复进行扫描与插入的步骤，直到所有的顾客点都被服务。

(4) 两阶段法

两阶段法的求解过程分为两个阶段：第一阶段按最小路径的原则形成初始解，然后用 k-OPT 算法对所得的各子路径分别进行优化。第二阶段是在各子路径间进行点的交换,以减小总行程,然后用 k-OPT 算法对点交换后的子路径进行优化。该算法的优点是：在计算过程中,考虑了所需要访问的点数量增加的情况。在两阶段法求解过程中，常常采用交互式优化技术，把人的主观能动作用结合到问题的求解过程中，其主要思想是：有经验的决策者具有对结果和参数的某种判断能力，并且根据知识直感，把主观的估计加到优化模型中去。这样做通常会增加模型最终实现并被采用的可能性。

3.2.1 现代启发式算法

现代启发式算法 (Meta-heuristics)，也叫智能优化算法，相对于传统启发式算法，它不要求在每次迭代中均沿目标值下降方向，而允许在算法中适当接受目标值有所上升甚至不可行的解，其目的是能够跳出局部搜索邻域。现代启发式算法强调的是对解空间进行全局搜索，特别是对富有希望的区域进行纵深试探以达到较好的解。下面对图 3-1 所示的四种主要求解 VRPTW 的现代启发式算法做如下介绍。

(1) 禁忌搜索算法

禁忌搜索算法 (Tabu Search Algorithm, 简称 TS)，最早由 Glover 提出<sup>[51]</sup>，TS

的基本思想是给定一个当前解（初始解）和候选解产生函数（邻域结构），然后在当前解的邻域中确定若干候选解。若最佳候选解对应的目标值优于目前为止搜索到的“最好解”（best-so-far），则忽视其禁忌特性，用其替代当前解和“最好解”。若不存在上述候选解，则在候选解集中选择非禁忌的最佳候选解为新的当前解，而无视它与当前解的优劣。两种情况下都将相应的对象加入禁忌表，并修改禁忌表中各对象的任期。如此重复上述迭代搜索过程，直到满足终止准则。禁忌搜索算法流程图如图 3-4 所示。

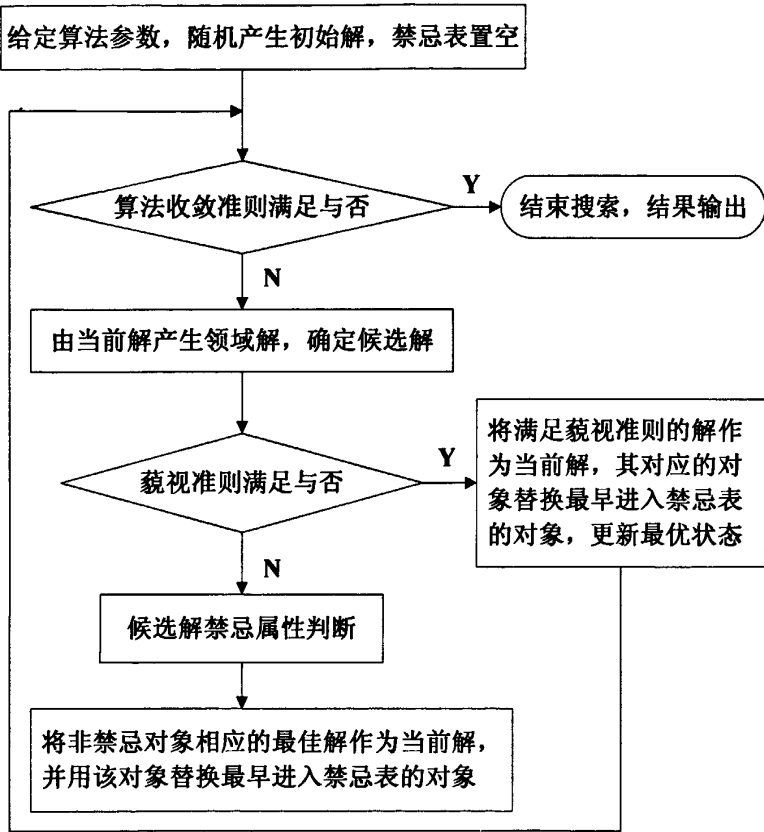


图 3-4 禁忌搜索算法流程图

Fig 3-4 Flow Chart of the Tabu Search Algorithm

与传统的优化算法相比，禁忌搜索算法的主要特点是：在搜索过程中可以接受劣解，因此具有较强的“爬山”能力；新解不是在当前解的邻域中随机产生，而或是优于“best so far”的解，或是非禁忌的最佳解，因此选取优良解的概率远远大于其他解。由于 TS 算法具有灵活的记忆功能和藐视准则，并且在搜索过程中可以接受劣解，所以具有较强的“爬山”能力，搜索时能够跳出局部最优解，转向解空间的其他区域，从而增强获得更好的全局最优解的概率，所以 TS 算法是一种局部搜索能力很强的全局迭代寻优算法。

## (2) 模拟退火算法

模拟退火算法 (Simulated Annealing, 简称 SA), 1996 年, Chiang 和 Russell<sup>[52]</sup> 提出 VRPTW 问题的模拟退火算法, 它是一种基于热力学退火原理的随机搜索算法, 源于 20 世纪 50 年代, 到 80 年代开始应用于组合优化领域。SA 将优化的目标函数视作能量函数, 模仿物理学中固体物质的退火处理, 在热平衡条件下, 物体内部处于不同状态的概率服从 Boltzman 分布, 若退火步骤恰当, 则最终会形成最低能量的基态。该算法在求解优化问题时, 不但接受对目标函数 (能量函数) 有改进的状态, 还以某种概率接受使目标函数恶化的状态, 从而可使之避免过早收敛到某个局部极值点。模拟退火算法基本步骤如下:

1) 任意选取一个初始状态  $i$  (初始解  $f_0$ ), 令迭代次数  $k=1$ ; 并设定一个较高的起始温度  $T$ ;

2) 求目标函数  $f(x_i)$  的函数值;

3) 如果该函数值在该温度下满足内循环停止条件转 4), 否则在当前状态  $i$  的一个领域内随即选一个新状态  $x_j$ , 若  $\Delta f_{ij} = f(x_j) - f(x_i) \leq 0$ , 将  $x_j$  取代  $x_i$  成为新状态, 若  $\Delta f_{ij} \geq 0$ , 以  $\exp(\Delta f_{ij}) > \text{random}(0,1)$  的概率接受新状态  $x_j$ 。若不接受新状态, 重复步骤 3)

4) 按照退火时间表降低温度  $t$ , 迭代次数  $k=k+1$ , 若满足终止条件, 输出结果, 否则回到步骤 2);

与传统的优化算法相比, 模拟退火算法的特点是: 对目标函数无特殊要求, 从理论上可以找到全局最优解, 且解的质量较好; 方法灵活, 可以考虑多种约束, 易于并行处理; 在应用过程中发现迭代初值好坏对其搜索质量影响不大, 能较好跳出局部最优, 因此易于与其他算法构成高效的组合算法。但冷却温度  $T$  难以控制, 因为只有  $T$  的下降与  $\lg(1+t_r)$  成比例 ( $t_r$  为模拟时间) 才能得到最优解。在实际应用中, 一般只能求得近似解, 其计算量比较大、耗时多, 影响了其实用性。

1983 年, Kirkpatrick 等意识到组合优化与模拟退火的这种相似性, 将该算法应用于求解组合优化问题中, 之后有不少学者将该算法应用于求解 VRP 问题。Tavakkoli-Moghaddam 等<sup>[53]</sup>利用一种混合的模拟退火算法求解了带容量约束的 VRP 问题, 以得到能够使车辆费用最小和车辆容量利用率最大的解。在应用该算法时, 他们提出了基于最近邻算法的 NNBH 算法, 用于生成模拟退火算法的初始解; 而在更新解时, 他们利用了 1-opt 和 2-opt 交换算法。事实上, 模拟退火算法自身能够克服初值依赖性, 但是好的初始解能够改善解的收敛速度。Attahiri<sup>[54]</sup>也利用模拟退火算法来求解 VRP 问题, 在求解过程中, 他们利用了 3-opt 交换算法。

## (3) 蚁群算法

蚁群算法 (Ants Colony Algorithm, 简称 AC) 是一种生物仿生算法, 该算法



是上世纪 90 年代由 M.Dorigo 等<sup>[55]</sup>学者从自然界蚁群觅食行为中受到启发而提出的一种优化算法。算法通过模拟蚁群搜索食物的过程,达到求解比较困难的组合优化问题之目的。由于其所具有的并行性和分布性,蚁群算法特别适合大规模启发式搜索,应用该算法求解 VRP 问题已成为最近研究的一个新方向<sup>[56]</sup>。用蚁群算法求解 VRP 问题的基本步骤为:

- 1) 算法的初始化,包括读取数据,计算各客户点之间的距离,对连接同一客户点的各条边按距离排序,计算从客户点出发的各条边的选择概率等;
- 2) 蚁群进行搜索;
- 3) 判断是否调整比例参数,若条件为真则进行调整;
- 4) 判断是否产生新的阶段最优路线,如是,转步骤 5),否则转步骤 6);
- 5) 判断是否产生新的全局最优路线;
- 6) 判断是否满足终止条件,若满足,退出程序,否则返回步骤 2)。

蚁群算法被提出后,不少学者利用该算法来求解各种 VRP 问题, Gambardella 等人<sup>[15]</sup>应用蚁群算法对 VRPTW 进行路径改进。首先构造两组相互协作的人工蚁群,第一个用来最小化车辆数,第二个用于最小化总路径长,并以共用解的方式建立协作关系。蚂蚁算法具有较强的寻优能力、并行性和可扩充性,其缺点是初期信息素匮乏,求解速度慢<sup>[57]</sup>。

#### (4) 遗传算法

遗传算法 (Genetic Algorithm, 简称 GA) 是基于 Darwin 进化论和 Mendel 的遗传学说而发展起来的全局优化的随机化搜索方法,其最早由密执根大学的 John.H.Holland 教授于 1975 年提出<sup>[58]</sup>。遗传算法作为一种基于自然选择和遗传变异等生物进化机制而发展起来的高度并行、随机、自适应的全局性概率搜索算法,可用于解决诸如控制及函数的优化、机器学习等许多方面的问题。因遗传算法在求解组合优化问题上所表现出的良好特性,其也被广泛地应用于 VRP 领域,本文也是基于该算法研究车辆路径优化问题,下一节将对本算法进行详细的分析。

### 3.3 遗传算法

#### 3.3.1 遗传算法基本思想

遗传算法是模拟生物在自然环境中优胜劣汰、适者生存的遗传和进化过程而形成的一种具有自适应能力的、全局性的概率搜索算法。

遗传算法是从代表待优化问题潜在解集的一个种群开始,而种群则由经过基因编码的一定数目的个体组成。基因编码成染色体,每个个体由染色体构成,每

个个体实际上是带有染色体特征的实体。染色体作为遗传物质的主要载体,是多个基因的集合,其内部表现(即基因型)是多个基因的某种组合,它决定了个体的形状的外部表现。因此,在一开始需要实现从表现型到基因型的映射即编码工作。初代种群产生之后,按照适者生存和优胜劣汰的原理,逐代演化产生出适应度越来越好的个体。在每一代中,根据问题域中个体的适应度的优劣,选择一些适应度高的个体,基于这些选出的适应度高的个体,并借助于自然遗传学的交叉、变异算子,产生出代表新解集的下一代种群。这个过程将导致种群像自然进化一样,使后代种群比前代种群具有更高的适应度,更加适应于环境。在优化过程结束后,末代种群中的最优个体经过解码,即可以作为问题的近似最优解<sup>[59]</sup>。

### 3.3.2 遗传算法相关概念

遗传算法是基于进化论和遗传学说发展起来的全局优化的随机化搜索算法,因而在这个算法中要用到各种进化和遗传学的概念。现对其中的主要概念做如下的说明。

#### (1) 个体 (Individual) 或染色体 (Chromosome)

个体或染色体,是生物学中的概念,在遗传算法中通常用一个所谓的串来表示:  $X=x_1 x_2 \cdots x_i$ , 其中  $x_i$  是串  $X$  的基本单元,称为基因 (Gene)。在遗传算法中,根据问题的不同,个体一般可分为二进制串,整数串,实数串,复数串等。

#### (2) 编码 (Coding) 和解码 (Decoding)

由于遗传算法是设计用来解决遗传空间中的问题,不能直接应用于实际的解空间,所以需要将解空间转换到遗传空间。编码操作就是指从表现型映射到基因型,从而将实际的解空间映射为相应的可以被遗传算法处理的遗传空间;解码操作是指从基因型映射到表现性,它是为了将遗传算法处理后的遗传空间映射到实际的解空间。二者是相反的操作过程。

#### (3) 种群 (Population) 和种群规模 (Population Size)

种群即个体的集合。使用遗传算法解决问题时,从随机选择的多个初始解开始进行迭代搜索,这多个初始解的集合以及每次迭代生成的一组新解就构成一个种群。种群规模表示群体中所含个体的数量。种群规模的取值非常关键,种群规模过小可提高算法的运行速度,但却降低了种群的多样性,影响遗传算法的优化性能,容易陷入局部最优解;种群规模过大,会降低算法的运行速度。

#### (4) 适应度 (Fitness)

在研究自然界生物的遗传和进化现象时,生物学家使用适应度这个术语来度量某个物种对于其环境的适应程度。与此相类似,遗传算法中也使用适应度这个

概念来度量种群中各个个体在优化计算中有可能达到或接近于或有助于找到最优解的优良程度。度量个体适应度的函数称为适应度函数。适应度用来评价个体优劣的程度，是遗传操作的依据，并指导遗传算法的搜索方向。

#### (5) 遗传算子 (Genetic Operators)

遗传算法的进化是靠遗传算子作用于当代种群的个体上，然后根据下一代种群中个体的存亡，来实现“优胜劣汰，适者生存”的。遗传算法通过遗传算子的作用，逐代进化种群，从而使相应的解也逐代优化，并逐渐逼近最优解。常见的遗传算子包括选择算子 (Selection)、交叉算子 (Crossover) 和变异算子 (Mutation)。

选择算子根据一定的选择策略作用在一个基因串上，其作用是为了从当前群体中选出优良的个体，使它们有机会作为父代产生后代个体，适应度越高的个体被遗传到下一代的概率较高，等待交叉和变异对其进一步演化。常见的选择策略包括轮盘赌选择法、最优保存策略、确定式采样选择、无回放随机选择、无回放余数随机选择、排序选择和随机联赛选择等<sup>[60]</sup>，其中轮盘赌选择法应用较多。

交叉算子在遗传算法中用来产生新的个体，对群体中的个体进行随机配对，将两个配对的个体按某种方式相互交换其部分基因，形成两个新个体。交叉操作是遗传算法区别于其他进化运算的重要特征，它在遗传算法中起着关键作用，是产生新个体的主要方法，它决定了遗传算法的全局搜索能力。基本遗传算法的交叉算子一般分为点式交叉和均匀交叉。

变异算子是指个体编码中的某些基因用其它等位基因来替换，产生新个体的操作。在遗传算法中使用变异算子主要为了改善遗传算法的局部搜索能力，维持群体的多样性，防止出现早熟现象。常用的变异算子包括基本变异算子和逆转算子两种。

### 3.3.3 遗传算法工作流程

遗传算法是从达尔文进化论中得到灵感和启迪，借鉴自然选择和自然进化的原理，通过对自然界进化过程中自然选择、交叉、变异机理的模仿来完成对最优解的搜索所形成的一种优化求解方法。Goldberg 总结了一种统一的最基本的遗传算法——基本遗传算法 (Simple Genetic Algorithm, 简称 SGA)<sup>[60]</sup>，基本遗传算法只使用基本遗传算子，其结构简单，是其它遗传算法的雏形和基础。基本遗传算法的执行过程是典型的迭代过程，如图 3-4 所示，其必要的工作内容如下：

(1) 构造满足条件的染色体。由于遗传算法不能直接处理解空间中的解，所以必须通过编码将解表示成适当的染色体。实际问题的染色体有多种编码方式，染色体编码方式的选取应尽可能的符合问题约束，否则将影响计算效率。

- (2) 随机产生初始群体。初始群体是搜索开始的一组染色体，其数量应适当选择。
- (3) 计算每个染色体的适应度。适应度是反应染色体优劣的唯一指标，遗传算法就是要寻求适应度最大的染色体。
- (4) 使用选择、交叉和变异算子产生子群体。这三个算子是遗传算法的基本算子，其中选择体现了优胜劣汰的自然规律，交叉体现了有性繁殖的思想，变异体现了进化过程中基因突变。
- (5) 重复步骤 (3)、(4) 直至满足终止条件时停止进化。

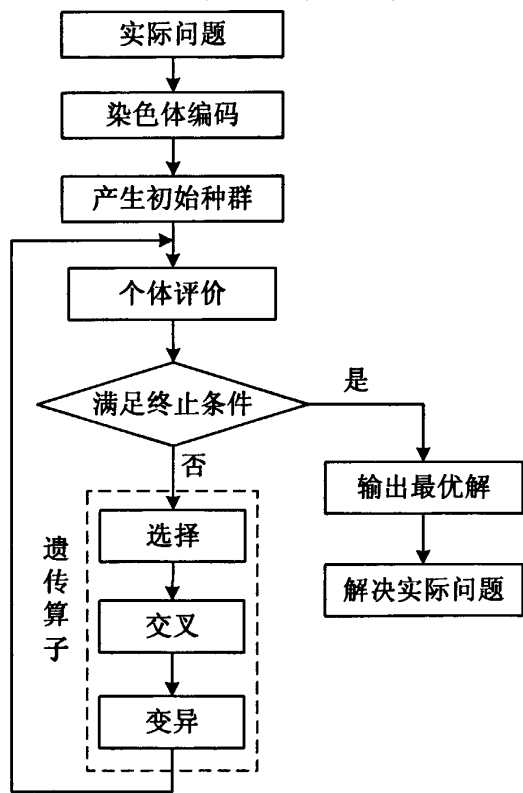


图 3-5 基本遗传算法工作流程图

Fig 3-5 The Workflow Chart of Simple Genetic Algorithm

3.3.4 遗传算法特点

由于遗传算法是一种基于自然选择原理和自然遗传机制的概率搜索算法，它模拟自然界中的生物进化机制，以实现特定目标的优化，与其他方法相比，具有如下特点<sup>[61]</sup>：

- (1) 自组织、自适应和自学习性。应用遗传算法求解问题时，在编码方案、适应度函数以及遗传算子确定以后，算法将利用进化过程中获得的信息自行组织

搜索;

(2) 遗传算法搜索不是直接作用在实际问题的变量上,而是作用在对参数集已经进行编码的遗传空间的个体上,所以遗传算法可直接对各种结构对象进行操作,应用领域广泛,具有一定的通用性;

(3) 遗传算法同时搜索解空间的许多点,即一个种群数目的点,而不是从一个单点上进行寻优,是从一组解迭代到另一组解,因而可以有效地防止搜索过程收敛于局部最优解,能有效求得全局最优解或近似最优解;

(4) 遗传算法的应用范围广泛,不依赖于实际问题空间的知识,它仅根据适应度函数的值来评价种群中个体的优劣程度,并在此基础上进行遗传操作,问题本身所具有的目标函数的限制极少,既可连续又可离散,不要求可微,也不要求其他先决条件和辅助信息;

(5) 遗传算法使用概率的规则来指导搜索方向,不采用确定性搜索规则,是一种启发式搜索,对搜索空间没有任何特殊要求,且不需要导数、梯度等信息,适用范围更广,因而能搜索离散的复杂空间,适用于大规模,甚至是超大规模的复杂优化问题;

(6) 遗传算法的本质并行性。遗传算法的内在并行性非常适合大规模地并行处理,比如可以同时使用数台计算机各自独立进行独立种群的进化,运行过程中甚至不需要进行任何通信,等到运算结束时才通信比较,选取最佳个体。GA 适合在目前所有的并行机或分布式系统上进行并行处理。此外遗传算法内含并行性,它采用种群的方式同时对多个个体进行搜索,这种伪并行的方式使得 GA 有较高的计算效率;

(7) 遗传算法的流程具有较强的通用性和灵活性,通过调整适应度函数和各种控制参数,可将 GA 应用于其他优化问题的求解。

### 3.4 常用算法概括与比较

车辆路径问题是 NP-hard 问题,随着客户数量增加,以及约束条件的复杂化,车辆路径问题求解越来越复杂,如何针对车辆路径问题的特点,构造运算简单、寻优性能优异的算法,这不仅对于物流配送而且对于许多可转化为车辆路径问题求解的组合优化问题具有十分重要的意义。

综上所述,各种优化算法在一定时期、一定情况下都有各自的优点,都有解决某一类问题的优越性,下面对上述各种优化算法进行比较分析,通过表格 3-1 来展现各自的特点。

表 3-1 各类优化算法比较分析表

Tab 3-1 Comparison of the Different Optimization Algorithms

类别	基本方法	优点	缺点	适用性
精确算法	分支定界法	搜索效率高,可求最优解	计算周期长	适用于解小规模组合优化问题
	动态规划法	可求最优解	计算时间随变量的增加成指数倍增长	适用于约束条件严格的规模较小问题
	切平面法	可求最优解	计算时间长	适用于求解小规模问题
	集分割和列生成	直接考虑可行解、可求最优解	解的最小成本难以确定	适用于约束条件严格的小规模问题
传统启发式算法	节约算法	扩充性好、车辆利用率高	规模较大问题所得解质量较差	求解规模小的问题
	插入算法	可得到较满意解	计算速度慢,所得解并非最优	适用于小规模问题
	扫描算法	可与插入法结合求解	扫描每一个点,速度慢	适用于小规模CVRP问题
	两阶段法	易于改善技术结合起来求解	所得解的改进较复杂	适用于小规模CVRP与DVRP问题
现代启发式算法	禁忌搜索算法	可通过规则提高搜索效率	涉及复杂领域结构影响求解速度	适用于较大规模问题
	模拟退火算法	采用随机松弛技巧	搜索结果不能保证是最优	适用于对已有路径进行改造
	蚁群算法	正反馈性、协同性、易与其他算法结合	变量需不断调整,搜索时间长,易陷入局部最优	适用于多目标优化问题
	遗传算法	鲁棒性、并行性、搜索能力强	不能保证最大概率收敛到全局最优解	适用于复杂优化问题

通过表格 3-1 对各类算法的分析比较,可以看出不同优化算法有其不同的优缺点与适用范围。总的来说,精确算法基于严格的数学手段,可以求得最优解,其解通常要优于其他算法,但由于引入严格的数学方法,其计算量一般随问题规

模的增大而呈指数增长,使该类算法只能有效求解中小规模的路径优化问题,对于带有各种约束的大规模问题就不太适用;而传统启发式算法与精确算法相比,在精度上不占优势,可以在有限的时间内找到满意的次优解或可行解,但仍不太适用于现在实际遇到的问题;现代启发式算法作为近年来较为流行的优化算法,在解决实际中的各种复杂问题上具有不可比拟的优势。相比其它启发式算法,遗传算法的性能相对稳定,具有全局搜索能力强,鲁棒性强以及计算效率高等特点和优势,适用于解决大规模优化问题,遗传算法已日益成为各领域研究者考虑采用的工具。因此,本文选择采用遗传算法作为本文带时间窗车辆路径问题的求解算法。

### 3.5 本章小结

本章对可求解带时间窗车辆路径问题的各种优化算法进行了研究与对比分析,为本文求解 VRPTW 的方法选择提供了参考和依据。本章首先介绍了求解 VRPTW 的精确算法以及启发式算法,包括其中的各种主要算法原理、概念与特点等;然后注重对技术性较强的遗传算法原理、工作流程、特点等进行了论述;最后对各种优化算法进行了对比,分析了各种算法在求解车辆路径问题时的优缺点和适用范围。通过对各种算法进行对比分析可以看出,遗传算法作为解决大规模优化问题的现代启发式算法,具有搜索能力强、鲁棒性强以及计算效率高等特点和优势,因此最终确定采用遗传算法作为本文 VRPTW 的求解算法,文章将在第四章详细介绍带时间窗车辆路径问题的数学模型建立及其求解算法设计。

## 4 VRPTW 模型构建与算法设计

### 4.1 VRPTW 模型构建

#### 4.1.1 问题描述

随着经济社会发展, 客户对物流配送服务质量及时效性要求越来越高, 如何在保持高度准时、快速配送的要求下, 降低配送成本是物流配送所要面临的新挑战。配送路线中的时间窗约束是基于客户允许车辆最早和最晚服务时间这一事实提出的, 对于多品种, 少批量, 多批次的配送问题, 尤其应该考虑时间窗约束。在实际的配送中, 通常认为多用一辆车所带来的成本的增加总是超过其因总行驶距离缩短带来的成本的节省, 因此寻求尽可能少的车辆数完成任务是节省总成本的有效方法, 所以, 研究带时间窗车辆路径问题具有重要的现实意义。

在需求点较多、分布不均匀且道路网复杂的情况下, 合理有效地对配送路线进行优化必须有明确的目标。本文制定配送路线方案的优化目标主要考虑以下几个方面:

(1) 合理安排车辆配送以实现效益最高或配送成本最低。效益是企业追求的主要目标, 成本对企业效益有直接的影响, 选择成本最低化是物流配送公司提高经济效益的重要途径。

(2) 按照客户的要求的时间窗送货以提高配送服务质量。能够满足客户的要求, 也可提高物流公司的信誉, 保留公司的客户, 是物流公司取得效益的根本保证。

(3) 合理利用运力。在运力非常紧张、运力与成本或效益有一定相关的情况下, 为了节约运力、充分运用现有运力, 而不需外租或新购车辆, 通常认为使用一辆车的成本远大于其总行驶距离缩短所带来的可变成本的节省。另外, 运力利用合理也可缓解城市日益严重的交通拥挤问题。

综上所述, 有时间窗口约束的配送路径问题优化目标除了考虑因距离造成的运输变动成本外, 还应考虑车辆启用数量变动成本以及因违反时间窗约束所造成的时间效应成本。因此, 本文所要解决的带时间窗车辆路径问题 (VRPTW) 可具体表述如下: 配送中心拥有相同容量的  $K$  辆车, 车辆的最大负荷为  $Q$ , 现有  $n$  个客户节点  $(1, 2, \dots, n)$  的运输任务需要完成, 第  $i$  个客户节点的需求量为  $d_i$  ( $i=1, 2, \dots, n$ ) 且有  $\max d_i \leq Q$ , 完成客户  $i$  的任务需要的时间为  $T_i$ , 且客户  $i$  的服务最好在时间窗口  $[e_i, l_i]$  内开始, 若车辆到达客户  $i$  的时间早于  $e_i$ , 则车辆需在  $i$  处等待, 如果到



达时间晚于  $l_i$ , 则处以一定的惩罚, 每个客户能且只能由一辆车一次完成运送任务, 每辆车从配送中心出发, 完成运送任务后最终都回到配送中心。最终目标是满足车辆容量限制、时间限制等一系列约束条件下找出使配送总成本最省的优化路径方案。

#### 4.1.2 基本假设

为了将现实中的带时间窗约束的车辆路径问题 (VRPTW) 抽象为数学模型, 本文建立如下的基本假设:

- (1) 物品流向为单向, 即纯送货;
- (2) 只有一个配送中心, 且每条线路的开始和结束位置都在配送中心, 所有车辆必须在规定的时间内返回配送中心;
- (3) 配送中心和客户点的位置坐标已知;
- (5) 车辆为同种车型, 且最大容量已知;
- (4) 车辆在配送过程中不得超过其额定载重量;
- (6) 每个客户节点的需求量已知;
- (7) 每个客户必须且只能被访问一次;
- (8) 每个客户都有一个指定的服务时间窗口  $[e_i, l_i]$ , 到达客户时刻最好在此时间范围内;
- (9) 每辆车只服务一条路线;
- (10) 各道路均顺畅, 不考虑交通堵塞拥挤等特殊情况。

#### 4.1.3 惩罚函数

以往对含时间窗约束的车辆配送问题的研究常仅对行驶成本进行探讨, 但事实上, 若配送中心违反了客户的时间窗约束, 势必会产生一定的经济损失, 这些损失可认为是因违反客户时间窗约束而导致的时间效应成本, 因此配送中心在追求总成本最小化的情况下, 有必要将时间效应成本纳入到考虑范围之中。

违反时间窗所造成的成本应以适当方式表达出来, 避免在设计路线时为了节省车辆行驶成本而忽略了其他无形成本, 因此提出以函数型态仿真顾客的时间窗, 这一函数也可看作是配送中心违反客户时间窗时所面对的惩罚成本函数。对惩罚成本采取较为合理的规定, 可使配送中心不至于为了节省路线成本而降低服务品质, 一般惩罚成本的额度是根据决策者对成本和客户满意之间的权衡而定, 但其共同特征是车辆到达时间偏离时间窗约束越多, 其惩罚成本越高。因此, 根据实

际情况有所不同, 惩罚成本可能成抛物线增长或指数增长等, 本文假设惩罚成本为线性增加以简化问题。因此, 在定义惩罚成本函数之前, 对惩罚成本做如下前提假设:

- (1) 惩罚成本以函数方式表示;
- (2) 当车辆在客户时间窗内服务, 配送中心不须支付任何惩罚成本;
- (3) 时间窗宽度越窄者, 其惩罚成本的边际效应越高;
- (4) 车辆不论早到或晚到, 随着违反程度的增加, 惩罚成本呈直线增加。

定义的惩罚成本函数表达式如下:

$$P_i(s_i) = \begin{cases} p(e_i - s_i) & s_i < e_i \\ 0 & e_i \leq s_i \leq l_i \\ q(s_i - l_i) & s_i > l_i \end{cases} \quad \dots(4-1)$$

上式亦可统一表示为:

$$P_i(s_i) = p \max(e_i - s_i, 0) + q \max(s_i - l_i, 0) \quad \dots(4-2)$$

式中的用到的参数符号作以下的定义:

$e_i$ ——客户  $i$  允许服务时间窗的始点

$l_i$ ——客户  $i$  允许服务时间窗的终点

$s_i$ ——车辆到达客户  $i$  的时间

$p$ ——车辆提前到达客户所处发生的等待单位时间的机会成本

$q$ ——车辆晚于时间窗到达客户所处以的单位时间惩罚值

若车辆  $k$  在时间窗  $e_i$  之前到达客户节点  $i$ , 则车辆在此等待, 发生机会成本损失  $p(e_i - s_i)$ ; 若车辆在时间窗  $l_i$  之后到达点  $i$ , 则服务被延误, 产生延误成本  $q(s_i - l_i)$ ; 车辆在时间窗  $[e_i, l_i]$  内到达, 则时间效应成本为 0。其中  $p$  和  $q$  取值的大小可根据实际情况来定, 对于某些重要的客户或对时间要求较苛刻的客户,  $p$  和  $q$  可取较大的值。对于硬时间窗问题可看作是软时间窗的一种情况, 即当  $p$  和  $q$  的值取无穷大的正数  $M$  时, 软时间窗惩罚函数便可修正为硬时间窗情形, 其惩罚函数表达式如下:

$$P_i(s_i) = M \max(e_i - s_i, 0) + M \max(s_i - l_i, 0) \quad \dots(4-3)$$

#### 4.1.4 参变量定义

根据以上对 VRPTW 的描述和基本假设, 现进行如下的变量和参数符号定义:

$N$ : 配送中心需服务的客户总数目;

$i, j$ : 单个客户,  $i, j = (0, 1, \dots, N)$ ,  $i, j = 0$  代表配送中心;

$k$ : 各车辆编号,  $k = (1, \dots, K)$ ;

$c_{ij}$ : 从客户  $i$  到客户  $j$  的运输成本, 这里考虑距离, 其中  $i \neq j$ ;

$\alpha$ : 车辆单位运输距离的成本;

$\beta$ : 单位车辆的启用固定成本;

$Q$ : 车辆的最大载重量;

$d_i$ : 客户  $i$  的需求量, 且  $\max d_i \leq Q$ ;

$e_i$ : 客户  $i$  允许的 earliest 服务时间;

$l_i$ : 客户  $i$  允许的 latest 服务时间;

$s_i$ : 车辆  $k$  到达客户  $i$  时的时刻, 其中  $s_0 = 0$ ;

$t_{ij}$ : 车辆从客户  $i$  行驶到客户  $j$  的时间, 其中  $i \neq j$ ;

$f_i$ : 车辆完成客户  $i$  任务所需的服务时间 (停留时间);

$w_i$ : 若车辆提前到达客户点  $i$  则必须的等待时间, 其中  $w_0 = 0$ ;;

$T_0^k$ : 车辆  $k$  的出发时间;

$T_R^k$ : 车辆  $k$  的要求返回时间;

$M$ : 非常大的正数。

$$x_{ijk} : \begin{cases} x_{ijk} = 1 & \text{当车辆 } k \text{ 从点 } i \text{ 行驶到点 } j \text{ 时 } (i \neq j) \\ x_{ijk} = 0 & \text{其他} \end{cases}$$

#### 4.1.5 数学模型

通过上述分析, 以配送总成本最小化为目标函数, 建立带时间窗车辆路径优化问题的数学模型如下:

$$\min F(i, j, k) = \alpha \sum_{k=1}^K \sum_{j=0}^N \sum_{i=0}^N c_{ij} x_{ijk} + \beta \sum_{k=1}^K \sum_{j=1}^N x_{0jk} + \sum_{k=1}^K \sum_{i=1}^N P_i(s_i) \quad \dots(4-4)$$

$$s.t \quad \sum_{k=1}^K \sum_{j=1}^N x_{ijk} \leq K \quad i = 0 \quad \dots(4-5)$$

$$\sum_{j=1}^N x_{ijk} = \sum_{j=1}^N x_{jik} \leq 1 \quad i = 0, k \in \{1, 2, \dots, K\} \quad \dots(4-6)$$

$$\sum_{k=1}^K \sum_{j=0}^N x_{ijk} = 1 \quad i \in \{1, 2, \dots, N\} \quad \dots(4-7)$$

$$\sum_{k=1}^K \sum_{i=0}^N x_{ijk} = 1 \quad j \in \{1, 2, \dots, N\} \quad \dots(4-8)$$

$$\sum_{i=0}^N \sum_{j=0}^N d_i x_{ijk} \leq Q \quad k \in \{1, 2, \dots, K\} \quad \dots(4-9)$$

$$\sum_{i=0}^N x_{ip} - \sum_{j=0}^N x_{pj} = 0 \quad p \in \{1, 2, \dots, N\}, k \in \{1, 2, \dots, K\} \quad \dots(4-10)$$

$$T_0^k + \sum_{i=0}^N \sum_{j=0}^N x_{ijk} (t_{ij} + f_i + w_i) \leq T_R^k \quad k \in \{1, 2, \dots, K\} \quad \dots(4-11)$$

$$\sum_{k=1}^K \sum_{i=0}^N x_{ijk} (s_i + f_i + w_i + t_{ij}) = s_j \quad j \in \{1, 2, \dots, N\} \quad \dots(4-12)$$

$$x_{ijk} \in \{0, 1\} \quad i, j \in \{1, 2, \dots, N\}, k \in \{1, 2, \dots, K\} \quad \dots(4-13)$$

$$P_i(s_i) = p \max(e_i - s_i, 0) + q \max(s_i - l_i, 0) \quad \dots(4-14)$$

以上数学模型中各式简要说明如下：

目标函数（4-4）表示最小化配送总成本。它由三项组成：车辆行驶距离成本、车辆启用数目成本以及时间延误或提早的惩罚成本。

约束（4-5）表示从配送中心出发的车辆数不超过  $K$ ；

约束（4-6）表示每辆车均从配送中心出发并最终返回配送中心；

约束（4-7），（4-8）表示每个客户点恰好被一车辆访问一次；

约束（4-9）表示每辆车所承担的任务总和不超过车量的载重量限制；

约束（4-10）表示流守恒条件，即车辆到达某点后，必须离开该点；

约束（4-11）表示车辆出发时间和返回时间均在规定的时间窗内；

约束（4-12）表示车辆从客户  $i$  到达客户  $j$  的时间约束；

约束（4-13）表示整数化约束，限制了  $x_{ijk}$  只能取 0 或 1；

约束（4-14）是根据式（4-2）表示的惩罚成本函数表达式。

上述 VRPTW 是包含了 TSP 以及 VRP 问题的更为一般性的问题。如果没有以上的（4-11）、（4-12）式关于时间窗的约束，那么该模型就变成一个简单的 VRP 模型。除此之外，还可通过令  $e_i = 0$ ,  $l_i = M$  来实现；如果只有一辆车可以使用（ $K=1$ ），则问题转化成 TSP 问题；如果去掉约束（4-9）式，也即去掉车辆容量约束，则得到一个 m-TSPTW 问题。总之，上述所构建的基于直观描述的 VRPTW 数学模型具有以下特点：考虑的目标函数和约束条件较为全面，接近实际情况；目标函数、变量和约束条件的表示较为自然、直观，易于理解；模型便于设计求解算法；模型具有较好的可扩充性和通用性。

## 4.2 遗传算法设计

针对上节建立 VRPTW 数学模型，本节将根据第三章介绍的遗传算法基本理论设计适用于该模型求解的遗传算法，其中主要包括染色体编码与解码、初始群体、适应度函数、遗传算子等要素的设计。以下就求解 VRPTW 的遗传算法的具

体设计方案进行详细介绍。

#### 4.2.1 编码与解码

编码是应用遗传算法时需解决的首要问题,选择适当的解的表达方式是遗传算法解决实际问题的基础。由于遗传算法应用的广泛性,迄今为止已经人们根据不同类型的问题提出了不同的编码方式,大致可以分为二进制编码、实数编码、整数或字母排列编码和针对具体问题的一般数据结构编码等几种结构。其中,二进制编码是最常用的编码方式,它的编码符号集由二进制符号 0 和 1 组成,具有编码解码操作简单,易于进行遗传操作等优点<sup>[62]</sup>,但不适用于基于次序的优化问题,它会导致频繁生成无效解,降低遗传算子的搜索效率,给遗传操作和解码造成很大的复杂性。编码方案的选取取决于问题的实际情况,对于带时间窗的车辆路径问题,其可行解为对应于所有被访问客户和访问路径的组合,而遗传算法不能直接处理该解空间的解数据,因此,需将车辆路径问题的解的形式转换成遗传空间的基因型串结构数据,以便于遗传计算。

带时间窗车辆路径问题是一种基于次序的组合优化问题,为减少无效解的生成,本文对染色体采用自然数编码,即序数编码方式。假设配送中心有车辆数  $k$  台,客户点  $l$  个,采用增加  $k+1$  个虚拟配送中心可形成一条长为  $k+l+1$  的染色体编码串  $(0, i_{11}, i_{12}, \dots, i_{1a}, 0, i_{21}, i_{22}, \dots, i_{2b}, 0, \dots, 0, i_{k1}, \dots, i_{kc}, 0)$ , 其中染色体相邻两个 0 之间表示一条子路径。文献<sup>[63]</sup>采用了这种客户点与虚拟配送中心共同排列的编码方法。例如,对于 9 个客户由 3 辆车来服务的个体编码 (0532071690840) 可以解释为:第一辆车从配送中心车场出发,经过客户 5, 3, 2 后回到配送中心车场,形成子路径 1,第二辆车也从中心车场出发,途径客户 7, 1, 6, 9 后回到中心车场,形成子路径 2,第三辆车从中心车场出发,途径 8, 4 后回到中心车场,形成子路径 3。其对应的三条配送路径方案如下 (0 表示配送中心):

子路径 1:  $0 \rightarrow 5 \rightarrow 3 \rightarrow 2 \rightarrow 0$

子路径 2:  $0 \rightarrow 7 \rightarrow 1 \rightarrow 6 \rightarrow 9 \rightarrow 0$

子路径 3:  $0 \rightarrow 8 \rightarrow 4 \rightarrow 0$

上述子路径内是有序的,若子路径 2 中点 7、1 相互交换位置,会使函数值改变;而子路径之间是无序的,若子路径 1 和子路线 2 相互交换位置,是不会改变目标函数值的。上述编码方式车辆数是固定的,即所有个体中的车辆数是相同的,虽然配送中心 0 作为不同路径之间的分隔符易于将不同路径区分开来,但在以后的交叉操作阶段,会产生大量无效解,且采用这种编码方式还有可能大幅度缩小遗传算法的搜索空间。

为了避免上述不利因素,本文研究的车辆数事先不确定的 VRPTW 问题编码没有将作为路径分隔符的配送中心 0 加入到染色体结构中,而是直接将所有路径中被访问的客户依次编码至一条染色体中,对于以上三条子路径编码为 (5 3 2 7 1 6 9 8 4),采用这种比较直观的基于客户点直接排列的染色体编码方式,可保证每个客户节点均被访问且只被访问一次,从而可大大地简化了对模型约束条件的处理。客户点直接排列的编码方式虽然没有表示各条路径分界点的基因位,但这有利于将来的交叉操作,在交叉算子和变异算子的操作中都无需考虑配送中心的位置。

解码操作是编码操作的逆过程,即将染色体的编码向量映射为满足全部约束条件的可行解的过程,本文解码操作采用与路径构造类似的过程进行解码。首先初始化一条新路径,然后将染色体中的基因值顺序插入到当前路径中,若一个基因值的插入导致该路径的负荷超过了车辆的最大容量或返回配送中心的时间晚于最晚返回时间,则开始构建新的路径,重复上面操作,直至所有客户均被插入到路径中。

#### 4.2.2 初始群体

遗传算法是一种群体型搜索方法,必须为遗传操作准备好若干个体组成的初始群体。遗传算法的搜索是从一代种群中多个个体同时进行的,因此第一代初始染色体种群被作为遗传操作的对象,成为进化的起点。群体规模影响遗传优化的最终结果,以及遗传算法的执行效率,当群体规模太小时,遗传算法的优化性能一般不会太好,而采用较大的群体规模可以减少遗传算法陷入局部最优解的机会,但较大规模的种群则是以计算的时间复杂度和空间复杂度为代价的。对于一般染色体长度不大的解来说,种群大小取值在 20~200 间比较合适,但是对于大规模算例可以根据染色体的长度适当增大种群的规模以方便取得更优的解空间<sup>[60]</sup>。研究表明,遗传算法搜索最优解不依赖于初始种群,但如果初始种群能均匀地分布在解空间,则有助于遗传算法跳出局部最优。因此为了使初始种群尽可能地均匀分布在解空间,本文采用随机方式生成初始种群。由于染色体采用整数编码方式,即每个个体(记为  $i$ ) 为 1 到  $n$  的自然数的一个全排列,因此本文初始化种群时随机产生  $popsiz$  个如此个体作为初始种群,其中  $n$  为客户节点数目,  $popsiz$  为种群规模。

#### 4.2.3 选择算子

选择算子是从一个旧种群选择生命力顽强的个体位串进行复制，从而产生新种群的过程。选择算子模仿了生物进化过程中“适者生存，优胜劣汰”的自然法则，是遗传算法搜索过程中至关重要的一步。它的目的是为了从当前种群中选出优良的个体，使它们有机会作为父代为下一代繁殖子孙，即适应度高的个体能以较大的概率被遗传到下一代。轮盘赌选择算子具有简单易操作等特点，是目前遗传算法中最基本也是最常用的选择算子。该算子是一种基于比例的选择，一种回放式的随机采样方法，它利用各个个体适应度所占比例的大小决定其子孙保留的可能性，这种选择方式与赌博中的赌盘操作原理颇为相似，故命名为轮盘赌选择法<sup>[61]</sup>。

如图 4-1 所示为一赌盘示意图，整个赌盘被分为大小不同的一些扇面，指针指向各个扇面的概率与各个扇面的圆心角大小成正比：圆心角越大，停在该扇面的可能性也越大；圆心角越小，停在该扇面的可能性也越小。与此类似，在遗传算法中，整个群体被各个个体所分割，各个个体的适应度在全部个体的适应度值中所占的比例（即相对适应度）大小不一，这个比例值瓜分了整个赌盘盘面，它们也决定了各个个体被遗传到下一代群体的概率。

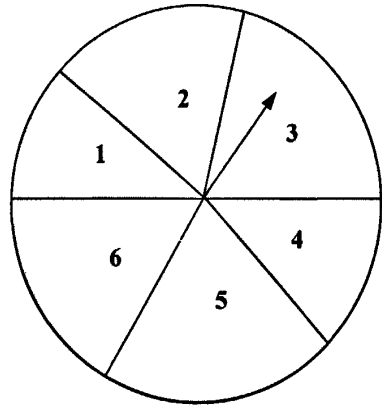


图 4-1 轮盘赌选择示意图

Fig 4-1 Schematic Diagram of Roulette wheel Selection

因此，如上所述：轮盘赌选择是指从群体中选择一些个体，这些个体被选中的概率与它们的相对适应度值成正比，个体的相对适应度值越高，被选中的概率就越大。但这并不保证相对适应度值大的个体就一定能选入下一代，仅仅说明它有大的概率被选中。假设某个染色体 $i$ ，其适应度为 $f_i$ ，种群大小为 $M$ ，其具体执行过程为：

- 1) 对各染色体 $i$ 计算适应度 $f_i$ ；
- 2) 计算种群中 $M$ 个染色体的适应度总和 $\sum_{i=1}^M f_i$ ；

3) 计算各染色体  $i$  被选中的概率  $p_i = f_i / \sum_{i=1}^M f_i$ ;

4) 对各染色体  $i$  计算累积概率  $q_i = \sum_{j=1}^i p_j$ ;

5) 使用模拟赌盘操作, 产生  $[0, 1]$  之间的随机数, 确定各染色体被选中的次数。

上述轮盘赌选择具有简单易操作的特点, 但由于随机操作的原因, 这种选择方法的选择误差比较大, 有时会出现“退化”现象, 即适应度较高的个体失去选择机会, 最终使得遗传算法的结果产生剧烈的震荡, 难以收敛到最优解<sup>[62]</sup>。本文算法提出改进的轮盘赌选择算子, 它在传统轮盘赌思想的基础上加入了排序和加大随机数产生次数的思想, 同时结合了最佳个体保存选择策略, 从而可提高选择算子的选优性能。

设种群大小为  $M$ , 父代种群  $Z = \{a_1, a_2, \dots, a_i, \dots, a_M\}$ , 其中每个个体的适应度大小为  $f(a_i)$ , 子代群体初始状态为  $X = \{\}$ , 则该选择算子的具体执行过程说明如下:

1) 将所有个体按其适应度值由大到小进行排序, 排序后的种群为

$Z' = \{b_1, b_2, \dots, b_i, \dots, b_M\}$ , 其中  $f(b_{i-1}) > f(b_i) > f(b_{i+1})$ ; 同时, 计算出父代种群中适应度最大个体  $k$ , 即  $f(a_k) = \max(f(a_1), f(a_2), \dots, f(a_M))$ ;

2) 计算出群体  $Z'$  中所有个体的适应度总和  $\sum_{i=1}^M f(b_i)$ ;

3) 计算出每个个体被选取的概率  $P_{b_i} = f(b_i) / \sum_{i=1}^M f(b_i)$ ;

4) 计算每个个体的累积概率  $Q_i = \sum_{j=1}^i P_{b_j}$ ;

5) 转动  $M$  轮轮盘:

① 产生  $M$  个  $[0, 1]$  之间的均匀随机数  $r$ 。如果  $r \leq Q_1$ , 则选择染色体  $b_1$ ; 否则, 选择第  $i$  个染色体  $b_i$  ( $2 \leq i \leq M$ ), 使得  $Q_{i-1} \leq r \leq Q_i$

② 统计各区间的  $\xi$  值  $\xi_1, \xi_2, \dots, \xi_M$ , 其中  $\xi_i$  是落在  $i$  号区域的随机数个数

③ 取最大的  $\xi$  值  $\xi_j = \max\{\xi_1, \xi_2, \dots, \xi_M\}$  所在区间  $j$  对应的个体  $b_j$  为本轮转动后所选中的个体  $M_i$ , 即

$$\begin{cases} M_i = b_j & \text{其他} \\ M_i = b_{\min(i_1, i_2, \dots, i_l)} & \text{若存在多个相同 } \xi \text{ 值区间 } \xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_l} (l \leq M) \end{cases}$$

④ 将  $M_i$  并入  $X$ , 即  $\begin{cases} X(0) = \emptyset \\ X(t) = X(t-1) \cup M_i \end{cases}$

⑤ 若选出的个体数到达种群大小, 则转 7), 否则转 ①;

6) 找出子代种群  $X$  中适应度最低个体  $m$ :  $f(c_m) = \min(f(c_1), f(c_2), \dots, f(c_M))$ ;



7) 用个体  $k$  代替个体  $m$ ;

8) 存储所有新选出的个体, 并且返回。

由上述执行过程可知, 结合排序选择和最佳个体保存策略的改进轮盘赌选择算子相比于传统轮盘赌选择算子有以下三方面的优势:

(1) 将传统轮盘赌算子中的每产生一个随机数就确定一个个体改进为每产生  $M$  个随机数才能确定一个个体, 即产生随机数总数量由原来的  $M$  增至  $M^2$ , 这样能更精确的体现随机数的作用, 减小选择误差。

(2) 将所有个体按其适应度有大到小进行排列的操作, 从而使高适应度的个体在前, 低适应度的个体在后, 以便当它们的  $\xi$  值相同时, 高适应度的个体会被优先选中, 从而可以较大的概率抛弃适应度低的个体。

(3) 针对传统的轮盘赌算子可能会使种群中适应度最好的个体遭到破坏的缺陷, 本文提出的算子结合了最佳个体保存策略, 即在每代操作结束后用父代适应度最高的个体替换子代适应度最差的个体, 保证了最优个体顺利进入下一代遗传操作, 从而可提高算法的运行效率、加快种群的收敛速度。

#### 4.2.4 交叉算子

在生物的自然进化过程中, 两个同源染色体通过交配而重组, 形成新的染色体, 从而产生出新的个体或物种。交配重组是生物遗传和进化过程中的一个主要环节。模仿这个环节, 在遗传算法中使用交叉算子来产生新的个体, 交叉算子是用双亲基因生成新染色体, 体现两性繁殖生物进化原理。交叉算子为选择操作提供了新的个体来源, 在遗传算法中起着重要作用。一方面, 它使得在原来的群体中的优良的个体的特性能够在一定的程度上保持; 另一方面, 它使得算法能够搜索新的基因空间, 从而使得新的群体中的个体具有多样性。对采用整数直接排列的个体编码, 若采用简单的一点交叉或多点交叉算子, 必然以极大概率导致未能完全遍历所有客户的非法配送路径方案, 解决这一问题的方法是对交叉算子做适当修正, 目前采用较多的方法是部分匹配交叉 (Partially Matched Exchange, 简称 PMX)<sup>[61]</sup>。在 PMX 操作中, 先随机产生两个位串交叉点, 定义这两个点之间的区域为匹配区域, 并使用位置交换操作交换两个父串的匹配区域。本文所研究的 VRPTW 采用了客户排列的自然数编码方式, 因此交叉算子采用了适用于自然数编码问题的类 PMX<sup>[64]</sup>。类 PMX 方法不同于传统的交叉算子, 不是直接交换染色体的交叉段, 而是先将交叉段加入到对方染色体的首个基因前, 之后再逐个去掉原个体部分中与交叉段基因相同的基因, 从而得到交叉后的个体。现举例说明类 PMX 操作过程:



(2) 将变异区域基因按反序插入到原位置中得到新个体  $\text{temp}_1 = "9 | 3 7 8 5 2 | 10 4 6 1"$ 。操作过程示意如图 4-3 所示。

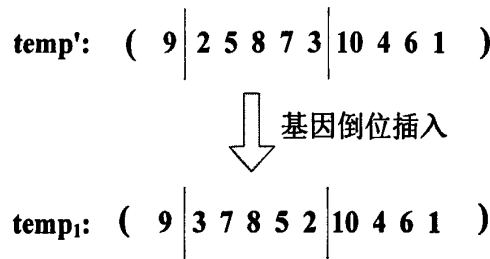


图 4-3 倒位变异操作示意图

Fig 4-3 The Illustration of the Inverse Operation

## 4.2.6 适应度函数

在生物进化过程中，“适者生存，优胜劣汰”的自然法则是通过生物物种的环境适应能力作为衡量其质量优劣的标准。遗传算法中，衡量一个个体质量优劣的标准是适应度。适应度是评价个体优劣和进行遗传操作的重要依据，个体的适应度越大，被选择到下一代的概率越大<sup>[65]</sup>。适应度函数具有非负性，任何情况下都大于等于 0，其值越大越好。而目标函数值可能有正负值，且有时取最大值，有时取最小值，因此需要在目标函数与适应函数之间进行适当交换<sup>[66]</sup>。

在遗传算法中，个体的适应度越大，表示个体的性能越好，而 VRPTW 是最小化组合优化问题，其目标是使配送总成本最小，即目标函数值最小，因此需要将目标函数转化为适应度。本文算法采用了如下所示的转换方式将目标函数转化为适应度函数：

$$f_i = \frac{1}{z_i} \quad i \in \{1, 2, \dots, \text{popsize}\} \quad \dots(4-15)$$

上式中  $z_i$  为群体中第  $i$  条染色体对应的目标函数值，反映了第  $i$  条染色体所对应的配送总费用； $f_i$  为第  $i$  条染色体对应的适应度，其值决定了该染色体产生后代的概率。

## 4.2.7 终止进化规则

遗传算法是随机搜索算法，为了结束遗传算法的演化循环，必须预先设定终止规则，使得当遗传算法在满足所设定的终止规则时，则停止演化循环。常见的终止规则有以下几种：

- (1) 达到了预先设定的目标。
- (2) 种群中的最优个体在连续若干代中都没有再获得改进。
- (3) 达到了预先设定的进化代数。

预先设定进化代数能有效控制算法的运行时间和算法的求解精度，因此本文算法采用事先确定进化代数作为终止规则，即判断进化的代数是否为要求代数  $N$ ，若是，则停止进化，选性能最好的染色体所对应的配送路径集合作为所求 VRPTW 问题的最优解输出；反之，继续执行进化运算。

### 4.3 本章小结

本章以第二章和第三章车辆路径问题及求解算法研究为基础，首先对 VRPTW 进行了问题描述，进而讨论了惩罚函数的设立，并在相关前提假设条件的设定下，考虑配送距离、配送车辆数、配送及时性对配送成本的影响，建立了以配送总成本最小化为目标函数的 VRPTW 数学模型，最后设计了适合于该模型求解的遗传算法，主要包括染色体编码与解码、初始种群、适应度函数和遗传算子等，为第五章的算例分析提供理论与技术支持。

## 5 算例分析

### 5.1 算例说明

Benchmark Problems<sup>[67]</sup>是 Solomon 在 1983 年设计的 VRPTW 标准测试题库, 该题库共包含 56 个不同实例, 这些实例皆假设求解单一配送中心和单一车种的 VRPTW。这 56 个实例中的每个问题可包括 25 个、50 个和 100 个客户, 每个客户有相应的时间窗约束及一定的需求量, 每辆车有最大允许载重量, 每两个客户点间的几何距离即为它们之间的运行时间。题库中所有问题的假设如下:

- (1) 每个问题中分别包含 25、50、100 个客户点, 每个客户皆有客户编号、位置 X、Y 坐标、客户需求量、客户的开门和关门时间以及固定的服务时间;
- (2) 单一配送中心, 并限制车辆最晚返回的时间;
- (3) 单一服务车种, 有最大可使用车数与车容量之限制;
- (4) 所有客户分布在  $(0,25)^2$ 、 $(0,50)^2$ 、 $(0,100)^2$  的平面坐标上;
- (5) 配送时的距离以欧基里德距离公式计算, 即时间与距离为同单位换算。

Solomon 题库可分为 C 型、R 型和 RC 型三个独立的数据集, 三个数据集之间惟一的区别在于客户点坐标的设置方式以及时间窗的参数设置不同, C 型例题的客户以结构性的方式设置坐标, R 型例题的客户以均匀分布的方式设置坐标, RC 型例题的客户则以 R 型与 C 型混合的方式设置坐标; C1, R1, RC1 等 1 型问题的车辆排程时间较短, 车容量也较小, 因此每条配送线路中可排的客户较少; C2, R2, RC2 等 2 型例题则反之。

### 5.2 试验数据

根据上述对 Benchmark Problems 的介绍, 本文选用 R101 算例和 C204 算例作为求解 VRPTW 的试验用例。为了保证算例的可行性和科学性, 本文定义 R101 和 C204 算例中客户要求的最早服务时间都大于配送中心的最早工作时间, 最晚服务时间都小于配送中心的最晚工作时间, 同时需满足所有客户对货物的需求均不超过车辆的容量限制, 对两算例分别描述如下。

#### (1) R101 算例

R101 问题可具体描述为: 设有一配送中心为 25 个处于不同地理位置、且对送货时间有不同要求的客户提供服务。其中配送中心的车辆是同质的, 都有相同的

载重量，均为 200 单位；车辆在所有客户处的服务时间均为 10 单位（时）；配送中心的时间窗定义为[0, 230]。R101 算例的具体试验数据在表 5-1 中列出。

表 5-1 R101 试验数据

Tab 5-1 Test Date of R101

序号	X 坐标	Y 坐标	需求量	最早接受 服务时间	最晚接受 服务时间	服务时间
0	35	35	0	0	230	0
1	41	49	10	161	171	10
2	35	17	7	50	60	10
3	55	45	13	116	126	10
4	55	20	19	149	159	10
5	15	30	26	34	44	10
6	25	30	3	99	109	10
7	20	50	5	81	91	10
8	10	43	9	95	105	10
9	55	60	16	97	107	10
10	30	60	16	124	134	10
11	20	65	12	67	77	10
12	50	35	19	63	73	10
13	30	25	23	159	169	10
14	15	10	20	32	42	10
15	30	5	8	61	71	10
16	10	20	19	75	85	10
17	5	30	2	157	167	10
18	20	40	12	87	97	10
19	15	60	17	76	86	10
20	45	65	9	126	136	10
21	45	20	11	62	72	10
22	45	10	18	97	107	10
23	55	5	29	68	78	10
24	65	35	3	153	163	10
25	65	20	6	172	182	10

表 5-1 中的第一列数据代表配送中心和客户节点的编号，0 代表配送中心；第二、三列数据分别表示配送中心或客户的  $x$  坐标和  $y$  坐标，本文假定客户之间的距

离是欧氏距离，即车辆从客户  $i$  行驶到客户  $j$  处的行驶路程等于行驶时间，且  $d_{ij} = d_{ji}, i \neq j$ ，距离满足三角不等式；第四列表示客户的货物需求量；第五、六列分别表示客户要求的最早可接受服务的时间和最晚可接受服务的时间；第七列是客户的服务时间。下图 5-1 表示的是 R101 算例中客户节点位置分布，其中用粉红色三角形代表配送中心，蓝色方点代表客户节点。

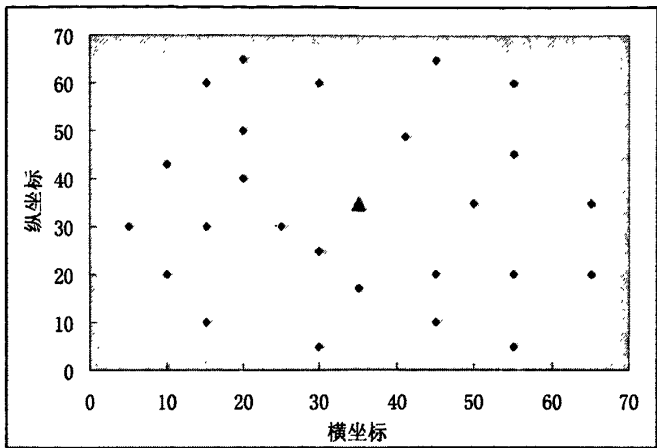


图 5-1 R101 客户位置分布图

Fig 5-1 The Scattergram of Location for R101 Customers

(2) C204 算例

C204 问题可描述为：假设配送中心为 100 个处于不同地理位置、且对送货时间有不同要求的客户提供服务，配送中心的所有辆车容量均为 300 单位，车辆在所有客户处的服务时间均为 90 单位（时），车辆的时间窗定义为  $[0, 1236]$ 。配送中心及客户节点的位置分布如图 5-2 所示，C204 算例的具体试验数据详见附录。

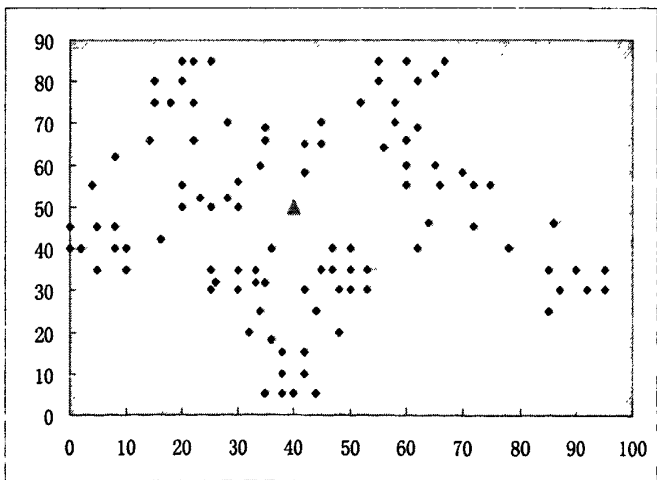


图 5-2 C204 算例客户位置分布图

Fig 5-2 The Scattergram of Location for C204 Customers

5.3 试验参数设定

本文采用 R101 和 C204 算例作为试验数据，求解如何安排车辆和行驶路径使最终的配送总成本最小化的路径优化问题。在求解过程中，需设定的相关试验参数如下。

(1) 模型参数设定

根据上述对 R101 和 C204 算例的描述，对模型的试验参数设定如表 5-2 所示：

表 5-2 模型参数设置表

Tab 5-2 Table of the Parameter for Test Model			
参数	意义	R101 取值	C204 取值
$\alpha$	运行单位距离成本	8	8
$\beta$	启用单位车辆成本	60	100
$p$	等待单位时间成本	0.5	1
$q$	延误单位时间成本	1.5	1.5

(2) 遗传算法运行参数设定

遗传算法的运行参数对算法的性能有较大影响，本文算法需要设定的运行参数主要有染色体长度、种群规模、交叉概率、变异概率、终止进化代数等，具体运行参数取值如表 5-3 所示：

表 5-3 遗传算法参数设置表

Tab 5-3 Table of the Parameter for Genetic Algorithm			
参数	意义	R101 取值	C204 取值
$L$	染色体长度	25	100
$N$	种群规模	80	400
$P_c$	交叉概率	0.85	0.85
$P_m$	变异概率	0.01	0.02
$Gen$	终止进化代数	100	300

5.4 算法实现主要步骤

本节根据第四章提出的遗传算法设计方案，应用 MATLAB 7.1 软件编制用于实现 VRPTW 模型求解的计算程序，实现该算法对带时间窗车辆路径问题的求解过程。其程序计算的具体步骤为：

Step1: 使用客户直接排列的自然数编码方式，构造可行配送路线的染色体；



Step2: 设置控制参数, 如  $\alpha$ 、 $\beta$ 、 $p$ 、 $q$ 、 $P_c$ 、 $P_m$ 、 $Gen$  等;

Step3: 标记进化代数  $gen=0$ , 随机产生包含  $N$  个染色体的初始群体  $P(0)$ ;

Step4: 若满足算法终止条件, 则输出结果, 停止, 否则继续;

Step5: 标记循环变量  $i=0$ ;

Step6: 计算适应度;

Step7:  $i=i+1$ ;

Step8: 若  $i \leq N$ , 回到 Step5, 否则转 Step9;

Step9: 根据适应度大小, 按基于排序和最佳个体保存策略的改进轮盘赌法复制下一代染色体;

Step10: 进行类 PMX 交叉、倒位变异操作, 重组  $P(gen+1)$ ;

Step11: 标记进化代数  $gen=gen+1$ ;

Step12: 若满足算法终止条件, 则终止, 否则转 Step4.

通过上述步骤的迭代搜索, 得到种群中最优染色相应的配送路径, 将其作为所求问题的最终优化结果输出, 完成带时间窗车辆路径问题的自动寻优过程。上述主要步骤的程序设计和实现过程如下所述:

#### (1) 种群初始化

种群初始化通过利用 `initializega` 代码随机产生 `popsiz` 个个体, 构成初始种群, 完成了问题解集的初始化过程, 且将此时的进化代数记为零。初始化种群的 `initializega` 主要代码如下:

```
function [t] = initializega(popsiz,custnum)
```

```
for i = 1: popsiz
```

```
    t(i,:) = randperm(custnum);
```

#### (2) 适应度计算

适应度函数是评价遗传空间中染色体性能的惟一量度, 本文利用 `Fitness` 代码计算群体中每条染色体的适应度, 以此指导算法的搜索过程。适应度 `Fitness` 的主要代码如下:

```
function [fitness,carNum,totalDis,totaldd,totallyw] = Fitness(vecAns)
```

```
    vecLODtmp = VECLD(vecAns);
```

```
    vecAnstmp = vecAns;
```

```
    num = 0;
```

```
    carNum = 0;
```

```
    totalDis = 0;
```

```
    totalTimdd = 0;
```

```
    totalDimyw = 0;
```

```

num_total = length(vecAns);
while(1)
    vecLODtmp = vecLODtmp(num+1:end);
    vecAnstmp = vecAnstmp(num+1:end);
    num1 = rule1(vecLODtmp);
    num2 = rule2(vecAnstmp);
    num = min(num1, num2);
    carNum = carNum + 1;
    vecAnscore = vecAnstmp(1:num);
    totalDis = totalDis + calcDist(vecAnscore);
    totaldd = totaldd + calcTimDD(vecAnscore);
    totalyw = totalyw + calcTimYW(vecAnscore);
    num_total = num_total - num;
    if num_total <= 0
        break;
    end
end
fitness = 1/(BPdis*totalDis + APcar*carNum + CPdd*totaldd + CPyw*totalyw);

```

### (3) 选择操作

根据 4.2.4 节提出的结合排序和最佳个体保存策略的改进轮盘赌选择算子, 对每代种群的所有染色体按其适应度从大到小进行排序, 以复制进入下一代的等待交叉和变异操作染色体。完成选择操作的 select 关键代码如下:

```

function newmatrix = select(Ansmatrix, vecfitness)
    vecfitness = vecfitness/sum( vecfitness);
    maxFitness = max(vecfitness);
    index = find(vecfitness == maxFitness);
    MaxAns = Ansmatrix(index, :);
    [vecfitness, mindex] = sort(vecfitness, 'descend');
    Ansmatrix = Ansmatrix(mindex, :);
    for m=1: popsize
        vecNum = zeros(1, popsize);
        for j=1: popsize
            randp = rand(1);
            for i=1: popsize

```

```

        if randp < vecfitness(i)
            break;
        else
            randp = randp - vecfitness(i);
        end
    end
    vecNum(i) = vecNum(i) + 1;
    [maxFitness, index] = max(vecNum);
    newmatrix(m,:) = Ansmatrix(index(1), :);
    vecfitness = MatrixcaleFitness(newmatrix);
    [minfitness, index] = min(vecfitness);
    newmatrix(index,:) = MaxAns(1, :);

```

#### (4) 交叉操作

采用 4.2.5 节类 PMX 交叉算子, 依照一定的交叉概率对父代染色体进行交叉产生新的染色体。完成交叉操作的 crossover 关键代码如下:

```

function [chromc, chromd] = crossover(chroma, chromb)
    chromc = chroma;
    chromd = chromb;
    n = length(chroma);
    vestart = srandom(n);
    veoff = srandom(n);
    if(vestart > veoff)
        vebetween = vestart;
        vestart = veoff;
        veoff = vebetween;
    end
    chromc = chromb(vestart:veoff);
    chromd = chroma(vestart:veoff);
    j = veoff - vestart + 1 + 1;
    for i=1:length(chroma)
        if(find(chromc == chroma(i)))
            continue;
        end
        chromc(j) = chroma(i);

```

```

        j = j + 1;
    end
    j = veoff - vestart + 1 + 1;
    for i = 1:length(chromb)
        if(find(chromd == chromb(i)))
            continue;
        end
        chromd(j) = chromb(i);
        j = j + 1;
    end

```

### (5) 变异操作

根据 4.2.6 节关于倒位变异算子的设计方案, 按一定的变异概率对父代染色体进行变异操作。完成倒位变异操作的 `mutation` 关键代码如下:

```

function newvector = mutation(vector)

    newvector = vector
    n = length(vector);
    vestart = srandom(n);
    veoff = srandom( n );
    if(vestart > veoff)
        vebetween = vestart;
        vestart = veoff;
        veoff = vebetween;
    end
    veczj = 0;
    j = veoff;
    for i=vestart:veoff
        veczj = newvector (j);
        newvector(j) = newvector(i);
        newvector(i) = veczj;
        i = i+1;
        j = j-1;
        if (i>= j)
            break;
        end
        i = i-1;
    end

```

### (6) 可行化操作

依据 4.2.1 节关于染色体编码与解码所述,构造的染色体为一串基于客户节点排列的自然数序列,本文用代表配送中心的“0”元素作为路径分隔符分割染色体中的不同路径,以直观表示所需车辆数目及其访问客户节点的次序。可行化操作的 decode 主要代码如下:

```
function codevecAns = decode (vecAns)

    codevecAns = [];
    vecLODtmp = VECLD( vecAns);
    vecAnstmp = vecAns;
    num = 0;
    carNum = 0;
    num_total = length( vecAns);
    codevecAns (1) = 0;
    len = 2;
    old_len = 1;
    while(1)
        vecLODtmp = vecLODtmp(num+1:end );
        vecAnstmp = vecAnstmp(num+1:end );
        num1 = rule1(vecLODtmp);
        num2 = rule2(vecAnstmp);
        num = min(num1, num2);
        carNum = carNum +1;
        num_total = num_total-num;
        codevecAns(len:len+num -1) = vecAns(old_len:old_len+num-1);
        len = len+num ;
        old_len = old_len+num ;
        codevecAns(len) = 0;
        len = len+1;
        if num_total<= 0
            break;
```

## 5.5 试验结果分析

由上述分析,在 CPU 为 Inter Core T2400、内存为 2GB、操作系统为 Windows

XP 的计算机上运行 MATLAB 7.1 求解本文所要解决的 R101 路径优化问题，为了使得到的解更科学有效，本文对 R101 算例进行了 10 次随机测试，10 次测试中取得的最优试验结果如表 5-4 所示。

表 5-4 R101 算例最优试验结果

Tab 5-4 The Best Solution of R101 Instance			
目标 结果	最优解	车辆数	达到最优解代数
计算结果	5937.2	7	67
路径编号	停靠客户点顺序		
1	0-11-19-18-10-0		
2	0-2-21-3-24-0		
3	0-14-15-6-17-0		
4	0-5-16-13-0		
5	0-7-8-0		
6	0-12-9-20-1-0		
7	0-25-23-22-4-0		

上述 R101 算例试验结果的最优解为 5937.2，共生成由 7 条路径组成的最优配送路径组合，经检验，所有路径均满足车辆容量限制和最晚返回配送中心时间约束。其最优配送路径方案如图 5-3 所示。

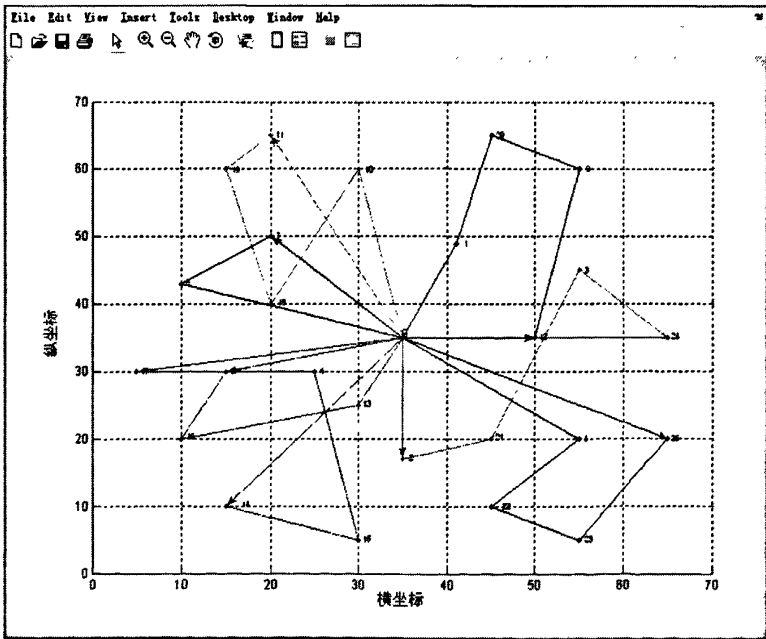


图 5-3 R101 最优配送路径示意图

Fig 5-3 The Optimal Delivery Route of R101 Instance

该配送方案中车辆访问各客户点顺序描述如下：

路径 1（车辆 1）：配送中心→客户节点 11→客户节点 19→客户节点 18→客户节点 10→配送中心；

路径 2（车辆 2）：配送中心→客户节点 2→客户节点 21→客户节点 3→客户节点 24→配送中心；

路径 3（车辆 3）：配送中心→客户节点 14→客户节点 15→客户节点 6→客户节点 17→配送中心；

路径 4（车辆 4）：配送中心→客户节点 5→客户节点 16→客户节点 13→配送中心；

路径 5（车辆 5）：配送中心→客户节点 7→客户节点 8→配送中心；

路径 6（车辆 6）：配送中心→客户节点 12→客户节点 9→客户节点 20→客户节点 1→配送中心；

路径 7（车辆 7）：配送中心→客户节点 25→客户节点 23→客户节点 22→客户节点 4→配送中心。

在获得上述 R101 最优试验结果的优化过程中，其优化结果与进化代数的关系如下图 5-4 所示

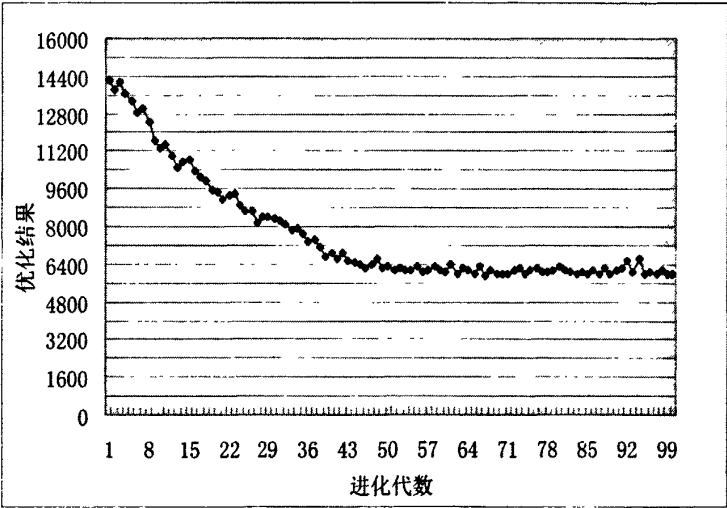


图 5-4 R101 最优解进化过程示意图

Fig 5-4 Schematic Diagram of R101 Evolving Processes of the Optimal Solution

从上图 5-4 可以看出，优化过程的开始阶段，优化曲线下降比较陡，随着进化过程的进行，优化曲线变化趋于平缓，并且在第 67 代收敛到该问题的最优解 5937.2。由于随机生成初始群体，在优化过程开始阶段的群体适应度较差，因此所得到的解相对较差。但随着进化过程的进行，遗传算法的自动寻优机制指导其搜索过程朝着目标更优化的方向收敛，解逐渐逼近最优，从 45 代左右后，解的波动

性开始趋于相对稳定，并在第 67 代搜索到上述最优解。该问题最优解的收敛情况表明了本文设计的遗传算法在解决带时间窗路径优化问题中的可行性，同时也表明了本文所构建的 VRPTW 模型的合理性。

为了说明本文设计的改进遗传算法在求解带时间窗路径问题上具有普遍适用性，本文还对具有 100 客户点的 C204 算例进行了试验，其中的参数设置如 5.3 节所述。在同一台计算机上对该算例进行 10 次随机计算后，取得的最优试验结果如表 5-5 所示。

表 5-5 C204 改进遗传算法最优试验结果

Tab 5-5 The Best Solution of C204 Instance Based on IGA

目标 结果	配送总成本	车辆数	达到最优解代数
计算结果	16558.6	7	226
路径编号	停靠客户点顺序		
1	0-16-97-3-83-82-85-0		
2	0-80-81-78-96-63-62-69-55-53-58-56-60-59-0		
3	0-6-32-33-38-47-42-41-43-48-26-23-18-19-17-25-0		
4	0-21-22-20-24-27-30-29-34-36-39-28-14-15-12-13-9-11-8-0		
5	0-54-57-49-65-68-64-61-72-66-50-40-46-45-51-52-31-35-37-44-0		
6	0-10-5-75-2-1-99-100-92-94-95-98-7-4-89-91-88-86-84-0		
7	0-93-90-67-74-76-79-73-77-71-70-87-0		

本文设计的改进遗传算法在 226 代获得了上述 C204 算例试验结果，取得的优化解为 16558.6，生成由 7 条路径组成的的最优配送路径组合，且所有路径都满足了车辆容量约束和最晚返回配送中心时间约束。其最优配送路径方案如图 5-5 所示。



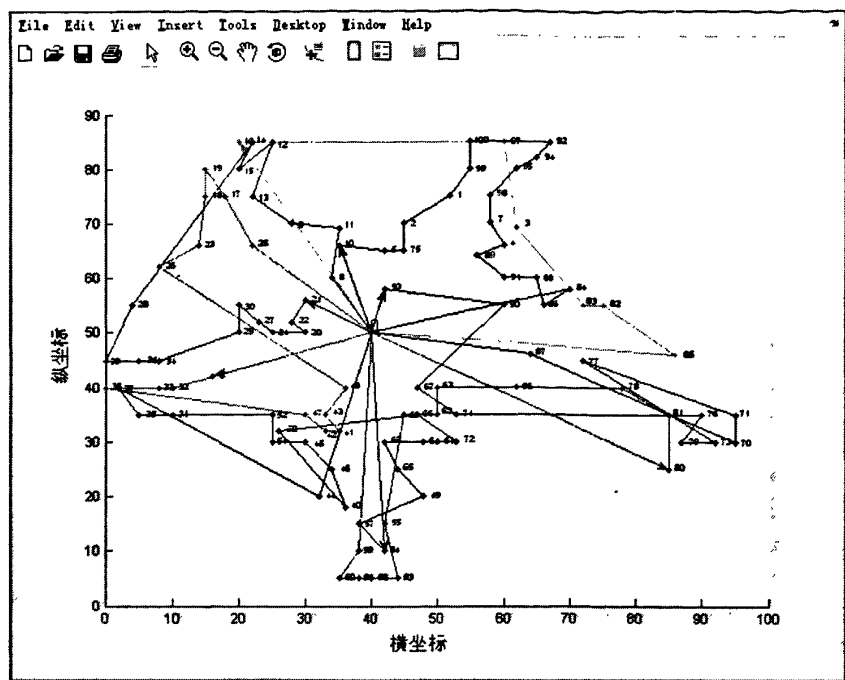


图 5-5 C204 改进遗传算法最优配送路径示意图

Fig 5-5 The Optimal Delivery Route of C204 Instance Based on IGA

为了验证本文所设计的改进遗传算法的优劣性，特在同一台计算机上采用基本遗传算法对上述 C204 算例进行了求解。为了使算法具有可比较性，基本遗传算法求解过程使用的试验参数与改进遗传算法的试验参数设定相同，如 5.3 节所述。在同样的条件下，采用基本遗传算法进行 10 次随机计算后得到的最优试验结果及配送路径方案分别如表 5-6 和图 5-6 所示。

表 5-6 C204 基本遗传算法最优试验结果

Tab 5-6 The Best Solution of C204 Based on SGA

目标 结果	配送总成本	车辆数	达到最优解代数
计算结果	21572.3	7	261
路径编号	停靠客户点顺序		
1	0-67-62-60-69-55-53-58-56-54-59-49-65-68-64-61-72-66-44-0		
2	0-74-57-40-46-45-51-50-52-31-35-32-6-47-42-41-43-0		
3	0-48-38-23-26-28-39-37-36-34-33-29-30-27-24-22-21-20-0		
4	0-15-13-25-9-11-8-12-14-0		
5	0-16-19-18-17-10-5-99-100-92-4-3-94-97-95-98-7-1-0		
6	0-2-75-89-91-93-84-76-79-77-87-71-0		
7	0-70-73-81-85-78-82-83-86-88-90-96-63-80-0		

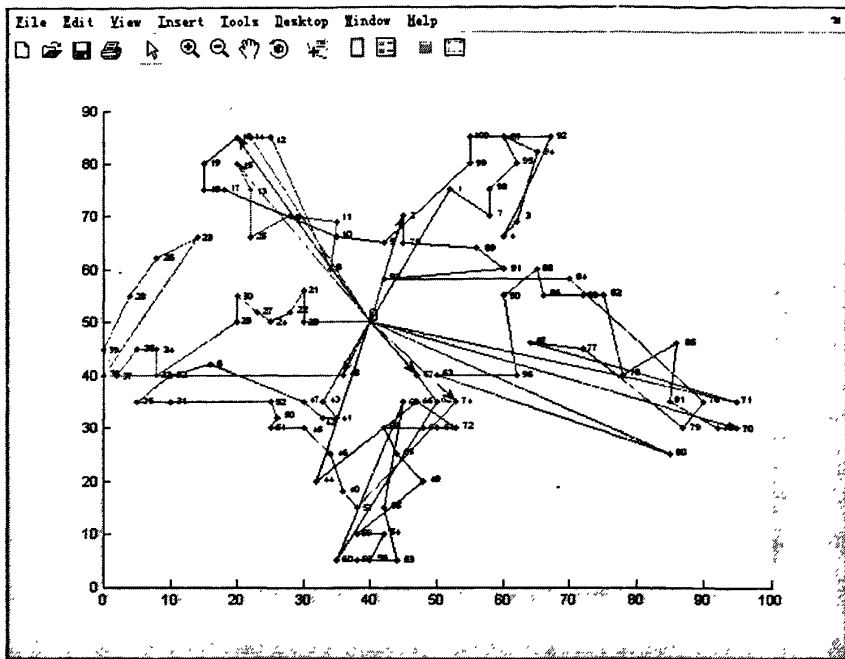


图 5-6 C204 基本遗传算法最优配送路径图

Fig 5-6 The Optimal Delivery Route of C204 Instance Based on SG

下图 5-7 表示的是本文设计的改进遗传算法与基本遗传算法在上述运算条件下，所得到的结果分析图，图中用 SGA 代表基本遗传算法，IGA 代表改进遗传算法。

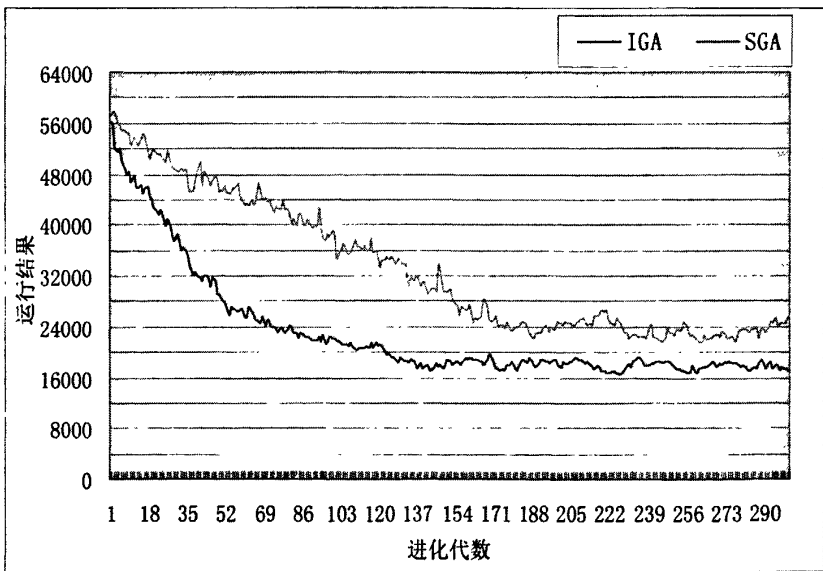


图 5-7 进化过程比较分析图

Fig 5-7 The Comparison of Evolutionary Processes of the Two Algorithms

从图 5-7 的比较可以看出，采用基本遗传算法所得到的优化曲线波动性相对较大，收敛速度较慢，在 261 代才收敛到最优值 21572.3；而本文设计的改进遗传算

法所得到的优化曲线总体波动性相对较小,呈现出明显的收敛趋势,在 120 代左右就开始趋于相对收敛状态,并在 226 代搜索到最优值 16558.6,且在相同进化代数时,几乎每次运行结果都优于基本遗传算法的运行结果。因此,相对于基本遗传算法,本文设计的改进遗传算法具有更好的性能。

由上述分析可知,本文的改进遗传算法在求解质量、求解速度、解的稳定性等方面都优于基本遗传算法,是解决带时间窗车辆车辆路径问题的一种可行且有效方法。另外,在实现上述算法的过程中,可以发现遗传算法求解问题时获得的解存在着某种程度上的不确定性。产生不确定性的原因主要在于:遗传算法是一种随机化技术指导搜索方法,遗传算法的选择算子、交叉算子、变异算子在进行遗传操作时都需要随机产生相应的随机数,随机数产生的好坏也在一定程度上影响着遗传算法的搜索性能,所以即使对同一问题使用相同的参数设置也可能得到不同的试验结果。此外由于不同参数所获解的性能往往不同,因此一般需要反复调整或凭经验设定参数,使搜索到的解产生了很大的不确定性。

## 5.6 本章小结

本章在前面几章的基础上,选用 R101 算例和 C204 算例作为试验用例,利用 MATLAB 软件有效地实现了本文改进遗传算法对两算例的自动寻优过程,验证了本文所建模型及算法设计的合理性和可行性;同时,运用基本遗传算法实现了 C204 算例的求解,通过其与改进算法的试验结果的对比分析,说明了本文算法在解决 VRPTW 问题方面的优越性,从而证实了本文提出的改进算法是一种可行且有效的算法。

## 6 总结及展望

### 6.1 全文总结

车辆路径问题（VRP）是物流配送过程中的关键环节，配送路径是否合理直接影响到配送速度、成本和效益。采用科学、合理的方法优化配送路径，对降低配送成本，提高服务水平，增强竞争实力，具有重要意义。带有时间窗约束的车辆路径问题是典型的组合优化问题，具有很高的计算复杂性。本文对带时间窗车辆路径优化问题（VRPTW）进行了探索和研究，主要研究内容总结如下：

（1）在国内外关于车辆路径问题的研究基础上，分析了车辆路径问题的描述、组成要素及分类，进而引出带时间窗的车辆路径问题（VRPTW），并着重介绍了时间窗的概念及分类。

（2）分析了求解带时间窗车辆路径问题的各种优化算法，主要对精确算法、传统启发式算法及现代启发式算法的原理、概念与特点等进行了分析，并归纳出各种优化算法的优缺点及其适应范围，确定遗传算法作为本文解决带时间窗车辆路径问题的优化算法。

（3）在对带时间窗车辆路径问题系统描述的基础上，考虑配送距离、配送车辆数及配送及时性对配送成本的影响，建立了以最小化总配送成本为目标函数的 VRPTW 数学模型。其中设定的关于违背客户时间窗的惩罚成本函数，兼顾了车辆提前到达所产生的等待机会成本与延迟到达所产生的延误惩罚成本。

（4）在基本遗传算法的基础上，设计了求解带时间窗的车辆路径问题的改进遗传算法，主要设计方案包括客户直接排列的自然数编码方式、结合排序和最佳个体保存策略的轮盘赌选择算子、类部分匹配交叉算子和倒位变异算子等，以提高算法寻优性能。

（5）通过算例验证了本文所建模型和算法设计的可行性和有效性。应用 MATLAB 软件实现了本文提出算法与基本遗传算法的求解过程，通过试验结果的对比分析，显示出本文提出算法在解决 VRPTW 问题方面的优越性，达到了保持群体多样性、克服早熟收敛、加快搜索速度，提高算法总体搜索能力的效果。

### 6.2 研究展望

由于本文对带时间窗车辆路径优化问题的研究是以探索的目的进行的，加上

时间和理论水平的局限, 本文的研究还存在一定的不足和不完善之处。以下几点可作为未来的研究方向:

(1) 虽然本文研究 VRPTW 时考虑了车辆载重及时间窗限制, 比一般车辆路径问题更接近现实, 但总体来说, 与实际问题还有一定差距。实际配送活动中, 经常会遇到随机型的物流配送问题, 如客户的不确定, 客户需求的不确定以及车辆行驶时间的不确定性、车辆的载重量不同等。因此, 今后应针对随机型的车辆路径问题开展深入的研究。

(2) 实际的车辆路径问题中, 常有某些客户必须优先服务, 这就是具优先次序的车辆路径问题, 应以适当的模型加以表达。

(3) 本文研究的是遗传算法求解带时间窗车辆路径问题, 对于其它智能优化算法如禁忌搜索算法、蚁群算法、模拟退化算法也需要进一步研究比较, 以找到求解该类问题的最优算法。

## 参考文献

- [1]周大帅等. 一种解决物流配送问题的算法研究[J].佳木斯大学学报. 2006, 24(3):417-419.
- [2]G.B.Dantzig and J.H.Ramser. The Truck-dispatching Problem[J]. Management Science. 1959,6: 80-91.
- [3]张炯等. 有时间窗配送车辆调度问题的禁忌搜索算法[J]. 北方交通大学学报. 2004,28(2):103-110.
- [4]Savelsbergh M. Local Search for Routing Problem with Time Windows[R]. Annals of Operations Research, 1985,16(4):285-305.
- [5] Baker J.E. Adaptive Selection Methods for Genetic Algorithms[D]. In Proc.of the 1<sup>st</sup> Int'l. Conton Genetic Algorithms. 1985:110-111.
- [6]M.M.Solomon. On the Worst-Case Performance of Some Heuristics for the Vehicle Routing and Scheduling Problem with Time Window Constraints[R]. Networks. 1986,16:61-174.
- [7]Clarke G, Wright J.W. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points[J]. Operations Research. 1964,12:568-581.
- [8]Paessens H. The Saving Algorithm for the Vehicle Routing Problem[J]. European Journal of Operations Research.1988,34:336-344.
- [9]Olatz Arbelaiz, Clemente Rodriguez. Comparison of Systems Based on Evolutionary Search and Simulated Annealing to Solve the VRPTW Problem.International Journal of Computational Intelligence and Applications[J]. 2004,4(1):27-39.
- [10]Jorg Homberger, Hermann Gehring. Two Evolution Metaheuristics for the Vehicle Routing Problem with Time Windows. INFOR[J]. 1999,37(3).
- [11]Hoong Chuin Lau AND Zhe Liang. Pick And Delivery with Time Windows: Algorithms And Test Case Generation. International Journal on Artificial Intelligence Tools [J]. 2002,3:455-472.
- [12]Wee-Kit.A Hbrid Search Algorithm for the Vehicle Routing Problem with Time Windows. International Journal On Artificial Intelligence Tools [J]. 2001,10(3):431-449.
- [13]Fisher M L,Jaikumar R. A Generalized Assignment Heuristic for Vehicle Routing[J]. Networks, 1981,11:109-124.
- [14]Wark P, Holt L. A Repeated Matching Heuristic for the Vehicle Routing Problem[J]. Operations Research Society. 1994,45(10):1156-1167.
- [15]L.M.Gambardella,E.Taillard and G.Agazzi. MACS-VRPTW:A Multiple Ant Colony System for Vehicle Routing Problem with Time Windows. New Ideas in Optimization,.1999,8:63-76.
- [16]B.Ombuki, M.Nakamura.M.Osamu. A Hybrid Search Based on Genetic Algorithm and Tabu Search for Vehicle Routing, presented at the 6th International Conference on Artificial Intelligence and Soft Computing,Banff,Canada,. 2002,7:176-181.
- [17]Brasy O. A reactive variable neighborhood search for the vehicle routing problem with time windows[J]. INFORMS Journal on Computing. 2003,15(4):347-368.
- [18]R.A.Russell,W.C.Chiang. Scatter search for the VRP with time window[J]. European Journal of Operational Research. 2006,169:606-622.
- [19]H.I.Calvete,C.Gale,M.J.Oliveros. A goal programming approach to vehicle routing problems with soft time windows[J]. European Journal of Operational Research. 2007,177:1720-1733.
- [20]N.Azi,M.Gendreau,J.Y.Potvin.An exact algorithm for a single-vehicle routing problem with time

- windows and multiple routes[J]. European Journal of Operational Research. 2007,178:755-766.
- [21]李军, 郭耀煌. 物流配送车辆优化调度理论与方法[M]. 中国物资出版社. 2001.
- [22]覃俊, 蓝雯飞, 兰华荣. 用遗传算法求解旅行商问题. 中国民族学院学报. 2000,1:25-28.
- [23]李大卫, 王莉, 王梦光. 一个求解带有时间窗口约束的车辆路径问题的启发式算法[J]. 系统工程. 1998,16(4):20-29.
- [24]谢秉磊, 李军, 郭耀煌. 有时间窗的非满载车辆调度问题的遗传算法[J]. 系统工程学报. 2000,15(3):290-294.
- [25]周贤伟, 李光远. 利用 GPS 技术的货物运输车辆路径问题的遗传算法研究[J]. 全球定位系统. 2001,26(4):50-52.
- [26]张丽萍, 柴跃廷, 曹节. 有时间窗车辆路径问题的改进遗传算法[J]. 计算机集成制造系统. 2002,8(6):451-454.
- [27]宋厚冰等. 有时间窗约束的车辆路径问题的改进遗传算法[J]. 交通与计算机. 2003,21 (4).
- [28]宾松等. 求解带软时间窗的车辆路径问题的改进遗传算法[J]. 系统工程. 2003,21(6):12-15.
- [29]刘小兰等. 有时间窗的车辆路径问题的近似算法研究[J]. 计算机集成制造系统. 2004,10(7):825-831.
- [30]刘诚, 陈治亚, 封全喜. 带软时间窗物流配送车辆路径问题的并行遗传算法[J]. 系统工程. 2005,23(10):7-11.
- [31]杨宇栋, 朗茂祥, 胡思继. 有时间窗车辆路径问题的模型及其改进模拟退火算法研究[J]. 管理工程学报. 2006,20(3):204-107.
- [32]张丽艳, 庞小红, 夏蔚军等. 带时间窗车辆路径问题的混合粒子群算法[J]. 上海交通大学学报. 2006,40(11):1890-1900.
- [33]张海刚, 顾幸生, 王军伟. 基于改进免疫遗传算法的带硬时间窗车辆调度问题的实现[J]. 微电子学与计算机. 2007(6):218-221.
- [34]Christofides N.Mingozzi A.Toth P.The Vehicle Routing Problem[C]. Combinational Optimizatoin. NewYork' Johnly Wiley. 1979.
- [35]钱颂迪. 运筹学[M]. 清华大学出版. 1990.
- [36]郎茂祥. 配送车辆优化调度模型与算法研究[M]. 电子工业出版社大学. 2009, (3):3-6.
- [37]符卓. 开放式车辆路径问题及其应用研究[D]. 中南大学. 2003,11: 3-42.
- [38]张强, 荆刚, 陈建岭. 车辆路线问题研究现状及发展方向[J]. 交通科技. 2004(1).
- [39]赵鲁华. 城市配送中心车辆调度优化研究[D]. 吉林大学. 2005,5:41-42.
- [40]林郁承. 基于聚类分析和遗传算法的带时间窗车辆路径问题研究[D]. 福建农林大学. 2009,4:13-16.
- [41]M.L.Fisher. Optimal Solution of Vehicle Routing Problems Using Minimum K-trees. Operations Research. 1994,42:626-642.
- [42]Araque,J.R, Kudva,G. A branch-and-cut algorithm for vehicle routing problems.Annals of Operations Research. 1994,9:37-59.
- [43]Psarafis,Potvin,Jean-Yves. Vehicle routing and scheduling with dynamic traveltimes. Computers and Operations Research, April, 2006, Part Special Issue: Optimization Days 2003:1129-1137.
- [44]Rinaldi, Padberg. A new test rig for measuring the inertia properties of vehicles and their subsystems. American Society of Mechanical Engineers, Design Engineering Division (Publication) DE, Proceedings of the ASME Design Engineering Division-2004, 2004:849-859.
- [45]Balinski M, Quandt R. On an integer program for a delivery problem[J]. Operations Research.

1962,12:300-304.

[46]Rao M R ,Ziont S. Allocation of transportation units to alternative trips-A column generation scheme with out-of-kilter sub-problems [J]. Operations Research. 1968,12:52-63.

[47]Jacques Renaud,Fayez F Boctor. A sweep-based algorithm for the fleet size and mix vehicle routing problem[J]. European Journal of Operational Research. 2002,140(3):618-628.

[48]Solomon M.M. Algorithms for Vehicle Routing and Scheduling Problem with Time Window Constraints.Operations Research. 1987,35:254-264.

[49]Gillett.B.E,Miller.L.L.A heuristic algorithm for a vehicle-dispatch problem.Operation Research. 1974,22(2).

[50]Solomon M.M. Vehicle routing and scheduling with time windows constraints: models and algorithms. Ph.D.dissertation, Dept of decision sciences,University of Pennsylvania.1983.

[51]F.Glover. Future Paths for Integer Programming and Links to Artificial Intelligence, Computers and Operations Research. 1986,13:533-549.

[52]W.Chiang and R.Russell. Simulated Annealing Metaheuristics For The VehicleRouting Problem With Time Windows. Annals of Operations Research. 1996, 63:3-27.

[53]Tavakkoli-Moghaddam R, Safaei N, Gholipour Y. A hybrid simulated annealing for capacitated vehicle routing problems with the independent route length[J]. Applied mathematics and computation. 2006:445-454.

[54]Attahiru SA. A 3-opt based simulated annealing algorithm for vehicle routing problems[J]. Computers and Industrial Engineering. 1991: 635-639.

[55]Marco Dorigo M, Gambardella L.M, luca Maria. Ant Colonies for the Traveling Salesman Problem[J]. Biosystems. 1997,43(2):73-81.

[56]Kwang M.S,Weng H.S. Multiple Ant-colony Optimization for Network Routing[C]. Proceedings of the First International Symposium on Cyber Worlds. 2002:1-5.

[57]丁建立, 陈增强等. 遗传算法与蚂蚁算法的融合[J]. 计算机研究与发展. 2003,40(9):31-36

[58]Holand J.H. Adaptation in Natural and Artificial Systems[M]. Massachusetts:MIT Press. 1975

[59]席裕根, 柴天佑等. 遗传算法综述. 控制理论与应用[M]. 1996,(6):26-28.

[60]雷英杰, 张善文等. MATLAB 遗传算法工具箱及应用[M]. 西安电子科技大学出版社. 2005.

[61]王小平, 曹立明. 遗传算法-理论、应用与软件实现[M]. 西安交通大学出版社. 2002.

[62]邱平. 车辆路径问题研究[D].大连海事大学. 2007.3.

[63]武亚丽, 段富. 免疫算法在物流车辆优化调度中的应用[J].电子商务与物流 2007,23(9).

[64]李仁安, 袁际军. 基于改进遗传算法的物流配送路线优化研究[J]. 武汉理工大学学报. 2004.12:99-101.

[65]周森. 基于遗传算法的物流运输中的车辆路径问题研究[D]. 对外经济贸易大学. 2005.9.

[66]谢开贵等. 免疫算法与其他随机优化算法的比较分析[J].重庆大学学报. 2003,26(11):43-47.

[67]Solomon benchmark problems[z]. <http://ecos.dei.uc.pt/index.php/EvoVRP>.



## 附录 A

## 试验数据 C204

CUST NO	XCOORD	YCOORD	DEMAND	READY TIME	DUE TIME	SERVICE TIME
0	40	50	0	0	3390	0
1	52	75	10	0	3272	90
2	45	70	30	0	3279	90
3	62	69	10	0	3270	90
4	60	66	10	1261	1421	90
5	42	65	10	0	3284	90
6	16	42	20	0	3274	90
7	58	70	20	0	3273	90
8	34	60	20	2887	3047	90
9	28	70	10	2601	2761	90
10	35	66	10	0	3283	90
11	35	69	10	2698	2858	90
12	25	85	20	0	3261	90
13	22	75	30	2405	2565	90
14	22	85	10	0	3260	90
15	20	80	40	2216	2376	90
16	20	85	40	0	3259	90
17	18	75	20	0	3266	90
18	15	75	20	0	3264	90
19	15	80	10	0	3260	90
20	30	50	10	0	3290	90
21	30	56	20	0	3288	90
22	28	52	20	0	3287	90
23	14	66	10	1643	1803	90
24	25	50	10	0	3285	90
25	22	66	40	2504	2664	90
26	8	62	10	0	3265	90
27	23	52	10	0	3282	90
28	4	55	20	0	3263	90
29	20	50	10	0	3280	90
30	20	55	10	0	3279	90
31	10	35	20	0	3266	90
32	10	40	30	0	3268	90
33	8	40	40	0	3266	90
34	8	45	20	0	3267	90
35	5	35	10	0	3261	90
36	5	45	10	0	3264	90
37	2	40	20	0	3260	90

38	0	40	30	1063	1223	90
39	0	45	20	0	3259	90
40	36	18	10	0	3267	90
41	35	32	10	0	3281	90
42	33	32	20	0	3280	90
43	33	35	10	0	3283	90
44	32	20	10	1913	2073	90
45	30	30	10	0	3277	90
46	34	25	30	0	3274	90
47	30	35	10	0	3281	90
48	36	40	10	0	3289	90
49	48	20	10	0	3268	90
50	26	32	10	0	3277	90
51	25	30	10	0	3275	90
52	25	35	10	0	3278	90
53	44	5	20	1256	1416	90
54	42	10	40	0	3259	90
55	42	15	10	1065	1225	90
56	40	5	30	0	3255	90
57	38	15	40	0	3264	90
58	38	5	30	1442	1602	90
59	38	10	10	0	3259	90
60	35	5	20	0	3254	90
61	50	30	10	0	3277	90
62	50	35	20	120	280	90
63	50	40	50	0	3285	90
64	48	30	10	0	3278	90
65	44	25	10	0	3274	90
66	47	35	10	0	3283	90
67	47	40	10	12	172	90
68	42	30	10	0	3279	90
69	45	35	10	680	840	90
70	95	30	30	0	3241	90
71	95	35	20	0	3242	90
72	53	30	10	0	3276	90
73	92	30	10	0	3244	90
74	53	35	50	213	373	90
75	45	65	20	0	3284	90
76	90	35	10	2131	2291	90
77	72	45	10	2900	3060	90
78	78	40	20	0	3260	90
79	87	30	10	2608	2768	90
80	85	25	10	0	3248	90
81	85	35	30	0	3252	90
82	75	55	20	0	3264	90
83	72	55	10	0	3267	90
84	70	58	20	1641	1801	90
85	86	46	30	0	3253	90

86	66	55	10	0	3273	90
87	64	46	20	3097	3257	90
88	65	60	30	0	3273	90
89	56	64	10	0	3278	90
90	60	55	10	0	3279	90
91	60	60	10	0	3277	90
92	67	85	20	694	854	90
93	42	58	40	0	3291	90
94	65	82	10	0	3259	90
95	62	80	30	0	3262	90
96	62	40	10	0	3275	90
97	60	85	30	0	3259	90
98	58	75	20	0	3269	90
99	55	80	10	407	567	90
100	55	85	20	502	662	90

## 作者简历

2004年9月至2008年7月在青岛理工大学管理学院攻读土地资源管理专业学士学位,2008年9月至今在北京交通大学交通运输学院攻读系统工程专业硕士学位。

硕士研究生期间主要参与科研项目:

- 1) 基于 GIS 的铁路电务综合管理系统研究与开发 2008-2009
- 2) 专用铁路职工培训与考核系统设计与开发 2009
- 3) 专用铁路运输安全预警及决策支持系统研究与开发 2009

## 独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京交通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。

学位论文作者签名：蒋波 签字日期：2010年6月12日

学位论文数据集

表 1.1： 数据集页

关键词*	密级*	中图分类号*	UDC	论文资助
物流配送；时间窗；车辆路径问题；遗传算法	公开	U116.2	65.012.34	
学位授予单位名称*		学位授予单位代码*	学位类别*	学位级别*
北京交通大学		10004	硕士学位	
论文题名*		并列题名		论文语种*
基于遗传算法的带时间窗车辆路径优化问题研究				中文
作者姓名*	蒋波		学号*	08121266
培养单位名称*		培养单位代码*	培养单位地址	邮编
北京交通大学		10004	北京市海淀区西直门外上园村 3 号	100044
学科专业*		研究方向*	学制*	学位授予年*
系统工程		信息系统工程	2 年	2010 年
论文提交日期*	2010 年 6 月			
导师姓名*	王喜富		职称*	教授
评阅人	答辩委员会主席*		答辩委员会成员	
	袁振洲 教授		刘红璐、吴惠群、张晓东、陈佳娟、李晓峰	
电子版论文提交格式 文本（ ） 图像（ ） 视频（ ） 音频（ ） 多媒体（ ） 其他（ ） 推荐格式：application/msword； application/pdf				
电子版论文出版（发布）者		电子版论文出版（发布）地		权限声明
论文总页数*	70			
共 33 项，其中带*为必填数据，为 22 项。				

作者：[蒋波](#)  
学位授予单位：[北京交通大学](#)  
被引用次数：3次

本文读者也读过(4条)

1. [王容](#) [基于IA与TS的带时间窗车辆路径优化算法研究](#)[学位论文]2011
2. [陈平平](#) [基于电力GIS的最短路径优化算法的应用与分析](#)[学位论文]2010
3. [赵彤](#), [范厚明](#), [王桂琳](#), [张婧瑶](#), [董国松](#), [李佳书](#), [ZHAO Tong](#), [FAN Hou-ming](#), [WANG Gui-lin](#), [ZHANG Jing-yao](#), [DONG Guo-song](#), [LI Jia-shu](#) [带时间窗的应急救援物资配送车辆路径优化模型研究](#)[期刊论文]-[物流技术](#)2010, 29(10)
4. [刘希洋](#) [应用改进型蚁群算法求解车辆路径优化问题的研究](#)[学位论文]2010

引证文献(3条)

1. [邬开俊](#), [王铁君](#) [基于改进差分进化的车辆路径优化算法](#)[期刊论文]-[计算机工程与应用](#) 2013(13)
2. [侯秀喆](#), [黄家城](#), [李文勇](#) [动态车辆路径问题的遗传算法研究](#)[期刊论文]-[西部交通科技](#) 2012(11)
3. [李娅](#), [王东](#) [多策略优化的蚁群算法求解带时间窗车辆路径问题](#)[期刊论文]-[计算机与数字工程](#) 2013(4)

引用本文格式：[蒋波](#) [基于遗传算法的带时间窗车辆路径优化问题研究](#)[学位论文]硕士 2010