Y 912008

分类号 O1　　　　　　密级 无

UDC 511

河海大学

硕士学位论文

车辆路径问题遗传算法的设计与分析

商丽媛

指导教师： 丁根宏　副教授　　河海大学理学院

南京市西康路1号

申请学位级别 理学硕士　专业名称 应用数学

论文提交日期 2006年5月　论文答辩日期 2006年5月

学位授予单位和日期 河海大学　　2006年6月

答辩委员会主席： 戴万阳　教授　论文评阅人： 戴万阳

徐小明

2006年5月　　中国　　南京

摘 要

物流是一个新兴学科，配送是现代物流的一个重要内容，合理安排车辆配送路线可以降低运输成本，提高经济效益。车辆路径问题是一类在物流配送调度中具有广泛应用的组合优化问题，属于强 NP 难题。有时间窗装卸问题比具有简单约束的车辆路径问题更加难以求解。

本文首先介绍了遗传算法在解决简单约束车辆路径问题上的应用，改进了交叉算子，为研究有时间窗装卸问题的遗传算法作了充分准备。

本文详细分析了有时间窗装卸问题的数学模型，深入研究解决此问题的分组编码遗传算法，将禁忌思想用于产生可行解的启发式插入搜索算法之中，并构造出适用于多目标的适应度函数，设计新的数据结构，对分组编码遗传算法进行有效实现。

在分组编码遗传算法中提出路径调整思想，设计出一种多策略分组编码遗传算法。采用多组通用算例测算，将多策略分组编码遗传算法与其它算法进行比较，其求解结果和计算时间都有明显改进，验证了多策略分组编码遗传算法能够有效稳定地收敛到所求问题的解。

关键词：车辆路径问题，有时间窗装卸问题，遗传算法，分组编码遗传算法，多策略分组编码遗传算法

Abstract

Logistics is an emerging discipline and distribution is an important element of modern logistics. Well arrangement for vehicle routes distribution can cut down transport cost and improve efficiency. Vehicle Routing Problem (VRP) is a combination optimization problem in transportation logistics, which has been applied in many fields. It is a strong NP problem. Pickup and Delivery Problem with Time Windows (PDPTW) is more difficult to solve than VRP with simple conditions.

In this paper, Genetic Algorithms (GA) for VRP with simple conditions was introduced and cross operator in GA was improved. These researches made full preparation for us to study PDPTW.

The mathematical model of Pickup and Delivery Problem with Time Windows was analyzed and a deep study of Grouping Genetic Algorithm (GGA) was made. A taboo thought was used in insertion heuristics to produce feasible solution. A fitness function was defined for multi-objective programming. A new data structure was designed in this paper to implement GGA.

Path adjustment strategies were proposed in GGA, and Multi-Strategy Grouping Genetic Algorithm (MSGGA) was designed. A benchmark data set for PDPTW was used to test MSGGA. Compared with other algorithms the experimental results have shown that MSGGA could find better solutions and take less time for computing. This proves that MSGGA is steadier to reach the solution of the problem.

Keywords: Vehicle Routing Problem, Pickup and Delivery Problem with Time Windows, Genetic Algorithm, Grouping Genetic Algorithm, Multi -Strategy Grouping Genetic Algorithm

学位论文独创性声明：

本人所呈交的学位论文是我个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。与我一同工作的同事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。如不实，本人负全部责任。

论文作者（签名）：高丽媛　　2006 年 5 月 30 日

学位论文使用授权说明

河海大学、中国科学技术信息研究所、国家图书馆、中国学术期刊（光盘版）电子杂志社有权保留本人所送交学位论文的复印件或电子文档，可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外，允许论文被查阅和借阅。论文全部或部分内容的公布(包括刊登)授权河海大学研究生院办理。

论文作者（签名）：高丽媛　　2006 年 5 月 30 日

第一章 绪 论

1.1 车辆路径问题

1.1.1 研究背景

物流被称为 Physical Distribution（PD）[1]，是一门新兴交叉性综合学科，研究物流的目标是有效地管理和控制物流的全过程，在保证服务质量的前提下，实现消耗总费用最小[2]。

物流配送是指根据一定的客户需求，在物流点内进行分拣、配货等工作，并将货物及时交给收货人的一个过程，其核心部分为配送车辆的集货、货物配装及送货过程。进行配送系统优化，主要就是进行配送车辆优化调度。车辆路径问题（Vehicle Routing Problem，VRP）是一类在物流配送调度中具有广泛应用的优化组合问题，在现代物流中居于中心地位。VRP 最早由 Dantzig 和 Ramser[3]于 1959 年提出，引起运筹学、应用数学、组合数学、图论与网络分析、物流科学、计算机应用等学科研究人员的极大重视，成为运筹学与组合优化领域的热点问题。各国研究人员对该问题进行了大量的理论研究及实验分析，取得了重大进展，其研究成果在运输系统、公交车辆路线设计、快递收发系统、物资调配系统中都已得到了广泛应用。研究车辆路径问题的特点及算法具有重要的实际意义[4]。

1.1.2 车辆路径问题研究综述

1.1.2.1 车辆路径问题介绍

车辆路径问题（VRP）一般定义为：对一系列装货点和卸货点，组织适当的行车线路，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、发送量、交发货时间、车辆容量限制、行驶里程限制、时间限制等）下，达到一定问题的目标（如路程最短、费用最少、时间尽量少、使用车辆数尽量少等）[4]。

各国研究人员对 VRP 作了大量而深入的研究，早在 1983 年，Bodin[5]、Golden[6]等人在综述文章中就列举了 700 余篇文献。在 Christofides[7]（1985），Golden 和 Assad[8]（1988）编辑的论文集，以及 Altinlemer 和 Gavish[9]（1991），Laporte[10]（1992），Salhi[11]（1993）等的综述文章中都对 VRP 进行了详尽的阐述。目前，车辆路径问题的形式已有很大发展，在运输、水运、航空、通讯、电力、工业管理和计算机等领域得到广泛应用，解决该问题的算法也已应用于解决航空乘务员轮班安排、轮船公司运送货物经

过港口与货物安排的优化设计、交通车线路安排、生产系统中的计划与控制等多种组合优化问题。

在VRP中，最常见的约束条件有：容量约束、优先约束、车型约束、时间窗约束、相容性约束、随机需求、运输中心、回程运输、最终时间期限和车速随时间变化等。根据已查阅的文献，从以上几个方面归结VRP主要分为以下几类：

（1）有容量约束的车辆路径问题（Capacitated Vehicle routing problem，CVRP）

（2）有时间窗的车辆路径问题（Vehicle Routing Problem with Time Windows，VRPTW）

（3）装卸一体化问题（Pickup-Delivery Problem，PDP）

（4）有时间窗的装卸问题（Pickup and Delivery Problem with Time Windows，PDPTW）

（5）多车场车辆路径问题（Multiple Depot Vehicle Routing Problem，MDVRP ）

（6）随机车辆路径问题（Stochastic Vehicle Routing Problem，SVRP）

（7）多车种车辆路径问题（Mixed Fleet VRP，MFVRP）

（8）有回程运输的车辆路径问题（Vehicle Routing Problem with Backhauls，VRPB）

（9）有最后时间期限的车辆路径问题（Period Vehicle Routing Problem，PVRP）

（10）车速随时间变化的车辆路径问题（Time Dependent Vehicle Routing Problem，TDVRP）

1.1.2.2 车辆路径问题的复杂性

计算复杂性的研究是离散优化的重要内容之一，通过对问题复杂性的研究，可以确定求解算法的研究方向。VRP是一个典型的有约束组合优化问题，也是离散优化的重要内容之一，著名的旅行商问题（Traveling Salesman Problem，TSP）是它的一个特例（即当车辆路径问题只包括一条路径，且没有能力约束时，就成为旅行商问题）。VRP是一个困难的组合优化问题，到目前为止，仅有一些相对较小规模的VRP能保证被求解到最优。

VRP被提出后，各国研究人员对其计算复杂性进行了研究。Lenstra和Rinnooy Kan[12]对VRP的计算复杂性进行了综述和分析；Karp[13]证明了旅行商问题（Traveling Salesman Problem，TSP）和有向旅行商问题（Directed Traveling Salesman Problem，DTSP）为NP难题；Papadimitriou[14]证明了多重旅行商问题（Multiple Traveling Salesman Problem，MTSP）为NP难题；Fredrickson[15]等人证明了多重中国邮递员问题（Multiple

Chinese Postmen Problem，MCPP）和多重有向中国邮递员问题（Multiple Directed Rural Postman Problem，DRPP）为NP难题。Savelsbergh[16]提出有时间窗约束的VRP（VRPTW）不仅问题本身是NP难题，在车队大小固定时寻找一个可行解的过程也是NP难题。Lenstra和Rinnooy Kan[12]证明了几乎所有类型的VRP均为NP难题。

自从VRP被证明为NP难题后，研究人员进行了各种求解算法的研究。传统求解VRP的方法有精确解法和启发式算法。由于VRP是一个强NP难题，当问题规模不断增大，其计算量呈指数增长，传统算法的效率和解的质量出现急速下降，甚至无法求解。启发式算法简单、易于操作，被广泛应用于VRP的近似最优求解。

1.1.2.3 车辆路径问题的算法

根据客户需求是否已知，VRP可分为确定性VRP和非确定性VRP。确定性VRP研究客户的需求信息、数量和地理位置已知的问题；非确定性VRP包括随机VRP和模糊VRP，在这些问题中，客户的需求量或客户数量不定。VRP类问题都属于最优化及规划问题，涉及非光滑函数，用经典的函数解法难以求解。下面主要介绍确定性VRP的求解算法，分为精确算法和启发式算法。

精确算法指可求出其最优解的算法，主要有：(1)分枝定界算法（Branch and Bound）及相关的分枝分割法（Branch and Cut），分枝定界算法由Laporte[17]等人提出，T. K. Ralphs，L. Kopman[18]等用并行分枝分割算法得到了TSP和CVRP较为满意的解。(2) K阶中心树和相关算法，由Christofides[19]等人提出。M. L. Fisher[20]对它做了改进，可求解有134个客户的VRP问题。(3)动态规划法，由Eilon[21]等人提出，Christofides[22]提出了状态空间松弛，可求解有50个客户的VRP。(4)集分割和列生成。VRP的集分割是由Balinski[23]等人提出的，Rao[18]等人引入了列生成方法进行求解，其本质为最短路径算法结合分枝定界算法。Desrocher[24]用它求解有100个客户点的VRPTW问题。(5)三下标车辆流方程，由Fisher[25]等人针对带能力约束、时间窗口以及无停留时间的VRP问题提出，并计算有50~199个客户点的VRP。(6)二下标车辆流方程，由Laporte[25]等人提出，求解客户规模为60的对称CVRP和DVRP。精确算法基于严格的数学手段，在问题可求解的情况下，计算结果优于启发式算法，但由于无法避免计算量的指数爆炸问题，精确算法只能有效地解决小规模的VRP，各算法所适用的问题特点也不同。

启发式算法是在状态空间中的改进搜索算法，它对每一个搜索的位置进行评价，得到最好的位置，再从这个位置进行搜索直到目标。在启发式搜索中，对位置的估价

十分重要，采用不同的估价可以有不同的效果。由于 VRP 是强 NP 难题，高效的精确算法存在的可能性不大，为此要构造高质量的启发式算法。现有文献中提出的启发式算法有以下几类：（1）Clarke-Wright 节约算法由 Clark 和 Wright[26]提出，用来解决车辆数不固定的 VRP，Golden, Nelson，Pilessens[27]等使用适当的数据结构降低其复杂度。（2） Sweep 算法，由 Wren, Gillett[28]等人提出。（3）Chrisofides-Mingozzi-Toth [29]两阶段算法主要面向 CVRP 和 DVRP。（4）禁忌搜索（Tabu Search，TS）由 Glover[30]等人最先应用于 VRP。E. Taillard[31] 按角度和路径重心，对问题的空间进行分割，再用禁忌搜索结合模拟退火对子问题求解，实现求解的并行化。（5）模拟退火算法（Simulated Annealing)是基于 MonteCarlo 迭代求解思想的随机搜索算法，Osman[32]，Teodorovic[33] 等人利用模拟退火算法求解 VRP 中的无时间窗单向问题或满载问题。（6）遗传算法（Genetic Algorithm，GA）。J. Lawrence[34]最先将该方法用于有效求解有时间窗口的 VRP(VRPTW)。Barnier[35]将它与约束满足问题（CSP）技术结合。张涛[36]等人通过遗传算法来保证搜索的全局性，用 3-opt 算法来加强局部搜索能力，得到针对 VRP 的混合算法，目前已可以求解较大规模的问题。本文重点研究了分组编码遗传算法在有时间窗装卸问题(PDPTW)上的应用。（7）重复匹配方法，由 P. Wark[37]等人提出，可求解较大规模的车辆路径问题。（8）蚁群优化算法 ACO 可以有效求解 TSP，研究人员将其推广应用于 VRP。Bernd Bullnheimer[38] 和 Ismail[39]等利用简单 ACO，可以求解较大规模的带时间窗 VRP，所求解优于 GA，SA 等算法。

1.2 本文介绍的三类车辆路径问题

本文重点研究解决有时间窗装卸问题（PDPTW）的遗传算法，作为前期准备，本文作者对遗传算法解决具有简单约束条件的 VRP（包括有容量约束的车辆路径问题 CVRP 和有时间窗的车辆路径问题 VRPTW）进行了初步研究。

1.2.1 有容量约束的车辆路径问题（CVRP）

有容量约束的车辆路径问题（Capacitated Vehicle routing problem，CVRP）是由一个服务中心（或车场）的若干车辆向多个客户点进行配送服务，在已知待服务客户点和出发点的位置、客户需求及车辆最大负载的前提下，设计车辆配送路径，规划设计方案，使运输成本最小化，即总代价最小（使用车辆尽量少，行车总距离尽量短）。CVRP 实际是多目标组合优化问题，一般以派出车辆最少（运输路线条数最少）为首要目标，行车总距离最短，即总代价最小为次要目标。CVRP 要求满足以下条件及假设：（1）

所有的配送车辆以配送中心为起点并最终回到配送中心；（2）每条配送路径上各客户点的需求量之和不超过车辆的负载量；（3）每个客户点的需求仅由一辆车一次满足。

1.2.2 有时间窗的车辆路径问题（VRPTW）

有时间窗的车辆路径问题（Vehicle Routing Problem with Time Windows，VRPTW）给定了配送中心和若干客户点，要求配送中心派出车辆对每个客户点送货一次且仅一次，并尽可能（或必须）在规定时间范围内到达客户点（配送中心也可能有时间范围限制），否则将因停车等待或配送延迟而产生损失，目的是设计最小成本的配送路线方案。VRPTW 除了必须实现 CVRP 的要求，还要考虑时间窗的限制。根据时间约束的严格程度，VRPTW 分为两类：软时间窗 VRP 和硬时间窗 VRP，前者要求尽可能在时间窗内到达客户点，后者则要求必须在时间窗内到达。本文所指的时间窗限制均是根据软时间窗的意义规定。

1.2.3 有时间窗装卸问题（PDPTW）

有时间窗装卸问题（Pickup and Delivery Problem，PDPTW）[40]是一类具有现实意义的组合优化问题。每个运输需求任务包括一个装货点需求、一个卸货点需求、所运货物的重量和类型；车队的每辆车从车库出发，沿优化的路线为客户服务，并最终返回车库；每辆车有最大装货量和装载货物类型的限制；所有车库、装货点和卸货点的地理位置可以不同，但必须在规定的时间窗口范围内访问。 有时间窗装卸问题（PDPTW）与有时间窗的车辆路径问题（VRPTW）根本的不同就是附加了前序限制，即限定每个需求的装货点必须在对应卸货点前被同一辆车访问。假定车辆数不限，且所有车辆有相同的负载和相同的起点，即所有路径都是从这个点开始并结束。可以利用优化理论产生一个合理的目标函数，如所用的车辆总数、行车总距离、总计划时间或者他们的混合目标。

PDPTW 被定义为一个整数规划问题[41]，是 VRPTW 的一般性推广，VRP 又是 TSP 的推广，而 TSP 已被证明是一个 NP-hard 问题（Lenstra 和 Rinnoy Kan[12]），因此，PDPTW 也具有相同的复杂性。这类问题的规模较大，复杂度很高，有许多实际的约束条件，解决难度大。PDPTW 在实际生活中应用很广，从车辆路径规划，飞机航线规划，到电路设计，柔性制造系统等，合理地解决此类问题能节省资源和能源，缩短工作时间，提高工作效率，从而大大降低成本，但由于它的复杂性，目前这方面的研究相对较少，特别是国内很少有相关的文献。因此，研究 PDPTW 的特点以及算法具有重要

的实际意义。

1.3 本文创新点及主要内容

1.3.1 本文创新点

本文作者查阅大量国内外有关遗传算法解决车辆路径问题（VRP）方面的研究资料，并对其进行整理、思考和研究。在初步研究遗传算法解决简单约束 VRP（包括 CVRP 和 VRPTW）的基础上，完善 PDPTW 模型，研究用于解决此问题的分组编码遗传算法（Grouping Genetic Algorithm，GGA），融入启发式插入搜索算法，设计了新的数据结构，为了有效找到 PDPTW 最优解，提出三种路径调整策略，形成多策略分组编码遗传算法（Multi-strategy Grouping Genetic Algorithm， MSGGA）。通过用 Pascal 语言编程，采用一般性算例测算，将结果与文献[42]结果比较，对 MSGGA 的算法性能进行分析。

本文主要创新点如下：

（1）完善了 PDPTW 的数学模型，构造了较为实用的目标函数。改进了分组编码遗传算法中用来生成可行解的启发式插入搜索算法，加入禁忌思想，使其便于产生可行解。根据分组编码方案设计了新的数据结构。

（2）在分组编码遗传算法的基础上提出了单车局部反向搜索、单个需求重排和部分需求重排三种路径调整策略，设计出多策略分组编码遗传算法。

（3）多策略分组编码遗传算法计算 100 个客户点的 PDPTW 算例，与文献[42]算法进行结果比较，算法缩短了计算时间，其收敛性和稳定性都有明显提高，并得出 4 个算例的更好解，取得了先进的研究成果。

1.3.2 本文主要内容

本文的各章节的主要结构如下：

第一章　概括叙述车辆路径问题的背景和研究现状、本文介绍的三类车辆路径问题以及本文主要工作内容。

第二章　对遗传算法进行介绍，包括遗传算法的基本流程与关键参数，一种与编码方式和选择策略无关的收敛性判据以及遗传算法的特点及改进。介绍遗传算法解决简单约束车辆路径问题（包括 CVRP 和 VRPTW）的研究成果。

第三章　描述有时间窗装卸问题（PDPTW）的模型，总结现有解决该问题的算法。

论述解决有时间窗装卸问题的分组编码遗传算法（Grouping Genetic Algorithm，GGA），设计三种路径调整策略，提出多策略分组编码遗传算法（Multi-strategy Grouping Genetic Algorithm， MSGGA）。

第四章 介绍 PDPTW 通用算例，用多策略分组编码遗传算法测算，对结果进行分析，与其它算法结果比较，说明多策略分组编码遗传算法（MSGGA）的性能。

第五章 论文总结和研究前景展望。

第二章　简单约束VRP遗传算法研究

为更好地研究应用遗传算法解决PDPTW，本文作者初步研究了运用遗传算法解决简单约束VPR（包括CVRP与VRPTW），对遗传算法的交叉算子进行了改进，并测算了简单通用实例，本章简单介绍研究情况。

2.1 遗传算法

2.1.1 概述

遗传算法（Genetic Algorithm，GA），是一类以达尔文自然进化论与遗传变异理论为基础的求解复杂全局优化问题的仿生型算法，它借鉴生物界自然选择和自然遗传机制，以概率论为基础，在解的空间中进行随机化搜索，最终找到问题的最优解。近年随着人工智能应用领域的不断扩大，传统基于符号处理机制的人工智能方法在知识表示、信息处理和解决组合爆炸等方面所遇到的困难越来越明显，J. Holland[43]于1975年提出的遗传算法正是适合于解决大规模问题的一种高度并行、随机和自适应的优化算法。

遗传算法是一种通用的优化算法，不受限制性条件的约束，两个最显著特点是全局空间搜索和隐含并行性。遗传算法进行全空间并行搜索，将搜索重点集中于性能高的部分，能够提高效率，不易陷入局部极小。遗传算法具有并行性，通过对种群的遗传操作可处理大量的模式，容易并行实现。随着计算机技术的发展，遗传算法越来越得到人们的重视，在机器学习、模式识别、图象处理、神经网络、优化控制、组合优化等领域得到了成功应用[44]。

2.1.2 遗传算法的基本流程与关键参数

2.1.2.1 遗传算法的基本流程

遗传算法是一类随机优化算法，通过对个体（染色体）的评价和对个体基因的作用，利用已有信息来指导搜索，改善优化质量的状态。标准遗传算法的操作是一个迭代过程，从一个具有M个体的初始群体出发，不断循环地执行选择、交叉、变异操作，直到满足停止准则。标准遗传算法的主要步骤可描述如下：

（1）初始化。设置最大进化代数T，随机生成M个体作为初始群体$P(0)$。

（2）个体评价。计算群体$P(t)$中各个体的适应度。

（3）判断算法收敛准则是否满足。若满足输出搜索结果；否则执行以下步骤。（4）根据适应度值大小以一定方式执行复制操作。

（5）按交叉概率执行交叉操作。

（6）按变异概率执行变异操作。

（7）返回步骤 2。

上述标准遗传算法中，适应度值是对个体（染色体）进行评价的指标，与个体的目标值存在一种对应关系；复制操作通常采用比例复制，即复制概率正比于个体的适应度值；交叉操作通过交换两个父代个体的部分信息构成后代个体，使得后代继承父代的有效模式，有助于产生优良个体；变异操作通过随机改变个体中某些基因而产生新个体，有助于增加种群的多样性，避免早熟收敛，得到更好的个体。

2.1.2.2 遗传算法的关键参数

遗传算法的设计一般是按以下步骤进行的：（1）确定问题的编码方案；（2）确定适应度函数；（3）遗传算子的设计；（4）算法参数的选取，主要包括种群数目、交叉与变异概率、进化代数等；（5）确定算法的终止条件。

1）编码方法

编码是在遗传算法中如何描述问题的可行解（即把一个问题的可行解从其解空间转换到遗传算法所能处理的搜索空间）的转换方法，很大程度上决定了如何进行群体的遗传进化运算及其效率。对于具体问题设计合理的编码方案是遗传算法的应用难点之一，常用方法大致可以分为二进制编码、浮点数编码和符号编码方法三大类。本文在第二章解决简单约束 VRP 所用的遗传编码是适合解决组合优化问题的符号编码，第四章多策略分组编码遗传算法则采用了 Falkenauer[45]提出的分组编码方式。

2）适应度函数

遗传算法使用适应度值来度量群体中个体在优化计算中有可能达到、接近或有助于找到最优解的优良程度，适应度值高的个体遗传到下一代的概率较大。度量个体适应度值的函数称为适应度函数（Fitness Function）。评价个体的适应度的一般过程为：（1） 对个体的编码进行解码，得到个体的表现型；（2） 由个体的表现型可以得到对应个体的目标函数；（3） 根据最优化问题的类型，由目标函数按一定的转换规则求出个体的适应度值。遗传算法中，群体的进化过程就是以个体的适应度值为依据，通过反复的迭代，寻求适应度值较大的个体，最终得到最优解或近似最优解。本文在第四章多策略分组编码遗传算法中设计的适应度函数适用于多目标问题，可以通过改

变权值的大小来达到适合不同目标要求的解。

3）遗传算子

选择操作是模仿自然界的“优胜劣汰”，确定如何从父代群体中按某种方法选取个体遗传到下一代群体的运算。选择操作建立在对个体适应度值进行评价的基础上，目的是利用父代群体中不同个体的优良基因，提高全局收敛性和计算效率[46]。

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（将群体中的N个个体以随机的方式组成$N/2$对配对个体组），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。交叉算子的设计和实现与所要解决的问题密切相关，一般要求它既不能太多地破坏个体编码中表现优良性状的模式，又能够有效地产生出一些较好的新个体模式。交叉算子设计包括确定交叉点的位置和进行部分基因交换，需要和编码设计统一考虑[47]。

变异算子是指个体（染色体）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。变异算子设计包括确定变异点位置和进行基因值替换两个方面。

从遗传算法中产生新个体的能力来说，交叉算子是主要的方法，决定着全局的搜索能力；变异算子只是辅助方法，但也必不可少，因为它决定着遗传算法的局部搜索能力。只有交叉算子和变异算子相互配合，共同完成对搜索空间的全局和局部搜索，才能以良好的搜索性能完成最优化问题的寻优过程[46]。

4）运行参数

遗传算法中需要的运行参数主要有个体编码串长度、群体规模、交叉概率、变异概率、终止代数等。这些运行参数对遗传算法运行的性能有着较大的影响，需要认真选取。

2.1.3 遗传算法收敛性的统一判据

遗传算法早期的基础理论主要是 Holland 的 Schema 定理及隐含并行性原理，后来又有了建筑块假设。当前，关于遗传算法基础理论的研究在 Schema 理论的拓广与深入、遗传算法的马氏链分析和遗传算法的收敛理论三个方面进行[44]。迄今为止，遗传算法的理论研究仍主要针对标准遗传算法(SGA)模型，复杂遗传算法由于其本身的多样性，理论方面的研究相当分散，尚未取得引人注目的结果。复杂遗传算法理论研究的大部

分结论都是通过计算机仿真来说明的，数学上严格完整且令人信服的解释仍需努力探索[48]。本节将系统分析遗传算法的收敛性并介绍一种与编码方式和选择策略无关的收敛性统一判据，给出证明。

2.1.3.1 收敛性的定义[49]和全局收敛的条件

定义 2.1（遗传算法全局收敛性定义）假设 $X_t=\{x_1(t),x_2(t),x_3(t),...,x_N(t)\}$ 为遗传算法的 t 代种群，$x_i(t)$ 为 t 代种群中的个体 $(i=1,2,...,N)$，N 为种群规模(即种群中个体的数量)。设 $Z_t=\max\{f(x_i(t)\mid i=1,2,...,N)\}$ 为 t 代种群中个体的最优适应度值，$f^*=\max\{f(x)\mid x\in S\}$ 表示全局最优适应度值，S 为个体空间，x 为 S 中的任意一个个体。遗传算法的全局收敛性定义可以表达为

$$\lim_{t\to\infty}p\{Z_t=f^*\}=1 \tag{2.1}$$

其中 $p\{Z_t=f^*\}$ 表示第 t 代种群中的最优个体成为全局最优的概率。

下面介绍一般可测状态空间上遗传算法的收敛性：

考虑优化问题： $$\max\{f(x);x\in S\} \tag{2.2}$$

x 为优化状态，S 为可测的有界紧致空间，$f(x)$ 为有界适应度值函数。假定问题的最优解集 $S_{opt}=\{x;f(x)=f_{\max}\}$ 非空，$\Phi(*)$ 为可测空间的一个测度函数，为了便于分析，这里假设 $\Phi(S_{opt})>0$。

$S^N=S\times S\times...\times S$ 为种群空间，称 S^N 中的每一个 $s=\{s_1,...,s_N\}$ 为包含 N 个个体的一个种群，种群 s 的适应度值为 $f(s)=\max\{f(x_i);x_i\in s\}$，定义全局最优解集为 $S_{opt}^N=\{s;\exists s_i\in s,s_i\in S_{opt}\}$。

假设 $\xi(t)$ 为 t 时刻的种群状态，$\xi_M(t)$、$\xi_C(t)$ 和 $\xi_S(t)$ 分别为变异后、交叉后和选择后的种群状态，则 $\xi(t)$、$\xi_M(t)$、$\xi_C(t)$ 和 $\xi_S(t)$ 均为 S^N 上的随机变量，并且 $\xi(t+1)=\xi_S(t)$。由于各遗传操作不随时间变化，因此相应的转移概率函数也不随时间变化。

令变异、交叉和选择操作引起的转移概率函数分别为：

$$P_M(s,A)=P_M(\xi_M(t)\in A\mid\xi(t)=s)$$

$$P_C(s,A)=P_C(\xi_C(t)\in A\mid\xi_M(t)=s)$$

$$P_S(s,A)=P_S(\xi_S(t)\in A\mid\xi_C(t)=s)$$

其中，$A\subset S^N$，遗传算法可用状态空间S^N上的平稳马尔可夫链$\{\xi(t);t\in Z^+\}|$来描述，其转移概率由下式给出：

$$P\{s,A\}=\int_y\int_z P_M(s,dy)P_C(y,dz)P_S(z,A) \tag{2.3}$$

算法要实现全局收敛，首先要求任意初始种群经有限步到达全局最优解，其次算法必须由保优操作来防止遗失最优解。假设各遗传操作满足如下性质：

性质 2.1 对任意时刻t，若$\xi(t)\in S_{opt}^N$，则$\xi_M(t+1)\in S_{opt}^N$；若$\xi_M(t)\in S_{opt}^N$，则$\xi_C(t+1)\in S_{opt}^N$；若$\xi_C(t)\in S_{opt}^N$，则$\xi_S(t+1)\in S_{opt}^N$。

性质 2.1 说明各遗传操作具有保优性。

性质 2.2 存在某一时刻t_0，对于任意初始种群$\xi(1)=s$，任意可测子集$A\in S_{opt}^N$和某一$\delta(s)>0$，转移概率函数满足$P(1,t_0;s,A)\geq\delta(s)\Phi(A)$。

性质 2.2 说明对于任意初始种群，最优集在有限步数内可达。

定理 2.1（一般可测空间上遗传算法的收敛定理） 设X^N为有界紧致可测空间，若以公式（2.3）表述的马尔可夫链满足性质 2.1 和性质 2.2，则对于任意初始种群$\xi(1)=s$，马尔可夫链全局收敛到最优集，即存在不变概率测度π，使得$\pi(S_{opt}^N)=1$且$\lim\limits_{t\to\infty}\|\mu_\varepsilon-\pi\|=0$。同时，存在$t_0>0$和正实数$\delta>0$，使得$\|\mu_t-\pi\|\leq(1-\delta)^{[t/t_0]-1}$，其中$\mu_t$为$\xi(t)$的概率分布，$[t/t_0]$表示取不超过$t/t_0$的最大整数。（$t_0$可视为由任意初始种群到$S_{opt}^N$的首达时间，$\delta$可认为是相应的概率。）

定理 2.1 的证明见文献[44]。

由定理 2.1 可知,收敛速度由δ或t_0来控制，增加δ或降低t_0是提高算法收敛速度的途径。

2.1.3.2 遗传算法的收敛性分析

与遗传算法收敛性有关的参数主要包括种群规模，交叉概率和变异概率；相关操作，即选择、交叉和变异等遗传算子。选择操作是为了避免有效基因的损失，使高性能的个体能够以更大的概率生存，提高算法的全局收敛性和计算效率。交叉操作用于组合生成新的个体，在解的空间中进行搜索，同时降低对有效模式的破坏。当交叉操作产生的后代适应度值不再进化并且没有达到最优时，就意味着遗传算法的早熟收敛。变异操作在一定程度上克服了这种情况，有利于增加种群的多样性。

遗传算法本质上是对个体（染色体）模式所进行的运算，通过选择算子将当前种群中的优良模式遗传到下一代种群中，利用交叉算子进行模式重组，利用变异算子进行模式突变。通过这些操作，种群逐渐淘汰较差模式，向好的模式进化，最终就可得到问题的解。模式过多将影响遗传算法的效率，在资源有限的条件下，必须进行优胜劣汰，使遗传算法向适应度值高的模式方向发展。导致早熟的直接原因是模式丢失，一旦出现模式丢失，遗传算法的搜索效率将受到影响，以致出现早熟收敛与停滞现象。文献[50]提出用补偿算子的方法进行基因补偿，可以解决基因丢失，并且克服一定程度的早熟收敛，实例证明这是有效的。

改进的遗传算法可以提高收敛性，但分析角度不同。下面介绍一种与编码方式和选择策略无关的收敛性统一判据及其证明[51]。

2.1.3.3 一定条件下的全局收敛性判据

假设一个全局优化问题的可行域是I（I的秩为N），表示任意的一个有限解集；种群规模为N固定不变，交叉概率$P_c \in (0,1)$，变异概率$P_m \in (0,1)$，遗传算法不改变运行过程中的选择机制。下面给出定理：

定理2.2 令$A(t)$表示t代种群，$A(t)=\{a_i(t)\mid i\in[1,n],t\in N,a_i(t)\in I\}$这里$a_i(t)$表示一个个体，代表可行域中的一个解；$f:I\to R$表示一个适应度函数，它分配给每个个体一个实数值；$I^*$是可行域$I$的一个子集，

$$I^*=\{a^*\mid a^*=\arg\max\{f(a)\mid a\in I\}\} \tag{2.4}$$

每个子集都有全局最大适应度；第t代种群的最优个体是$a^*(t)=a_j(t)\in A(t)$，这里$a_j(t)$满足$f(a_j(t))\ge f(a_i(t)\in A(t))$。

变异、交叉和选择操作中，$a^*(t)$和它的适应度将随机变化。但是每一代种群都至

少存在一个$a^*(t)$，因此，当t趋于∞时，$a^*(t)$是否在子集I^*上收敛，暗示着遗传算法是否全局收敛，即如2.4式所示：

$$p(\lim_{t\to\infty}a^*(t)\in I^*)=1\Rightarrow p(a^*\in\lim_{t\to\infty}A(t))=1 \tag{2.5}$$

$a^*(t)$的演化过程可以通过齐次有限马尔可夫链来描述。为表明收敛性，这样的马尔可夫链[52]可以简单地说成是一个$a^*(t)$链。$a^*(t)$链的状态空间S中的每个状态唯一地对应于可行域I中的一个解，且$|S|=|I|=N$。将S中的所有状态按照适应度$f(a^*(t))$降序排列，从1到N，可以导出遗传算法的全局收敛性的定理。

定理 2.3 令$P=(p_{ij})$为一个$a^*(t)$链的转移矩阵，状态空间为S，S^*为状态子空间，子空间中的每个元素分别对应子集I^*中的一个解。当$t\to\infty$时，如果$a^*(t)$从任意一个状态I以概率1转变成S^*，即如果

$$\lim_{t\to\infty}\sum_{j\in S}P_{ij}^t=1 \qquad (i\in S) \tag{2.6}$$

则遗传算法以概率1收敛于全局最优解。

推论 2.1 如果$a^*(t)$链的状态空间为S，子空间为S^*，其状态转移矩阵$P=(p_{ij})$是一个正定的随机矩阵，则遗传算法收敛到全局最优解。

证明：正定矩阵同时也是一个简单矩阵，它遵循定理2.1，转移矩阵P的极限分布在无非零输入的情况下是唯一的稳定分布。因此，对于任意$i\in S$，

$$s.t.\quad \lim_{t\to\infty}\sum_{j\in S^*}P_{ij}^t=1-\lim_{t\to\infty}\sum_{j\in S-S^*}P_{ij}^t=1-\sum_{j=|S^*|+1}^{|S|}P_j\pi$$

因此，方程（2.6）不满足，证明完毕。

推论 2.2 如果$a^*(t)$链的状态空间为S，状态子空间为S^*，其状态转移矩阵$P=(p_{ij})$是可约的随机矩阵，同时$P=\begin{pmatrix}C & 0\\ R & T\end{pmatrix}$，$R\neq0$且$T\neq0$，则遗传算法能够以概率1收敛到全局最优值。

证明：由于标准遗传算法不能以概率1收敛，则有

$$P^\infty=\lim_{t\to\infty}(p_{ij}^t)=(\pi,\ldots,\pi)_{1\times N}^T,\quad \pi=(p_1,\ldots,p_{|S^*|},0,\ldots,0)$$

当$1 \le j \le |s^*|$时，$p_j \ne 0$，且$\sum_{j=1}^{|s^*|} p_j = 1$。可以得出，对于任意$i \in S$，

$s.t.\quad \lim_{t \to \infty} \sum_{j \in S^*} p_{ij}^t = \sum_{j=1}^{|S^*|} p_j = 1$，满足方程（2.6），证明结束。

若$a^*(t)$链的转移矩阵的结构可以确定，则定理2.2和定理2.3可以直接用于遗传算法收敛性的判断[53]。

2.1.4 遗传算法的特点及改进

遗传算法利用生物进化思想实现优化过程，具有以下特点：（1）遗传算法将问题参数编码成个体（染色体）后进行操作，这使遗传算法不受函数约束条件的限制。（2）遗传算法的搜索过程是从问题解的一个集合开始的，具有隐含并行搜索的特性，减小了陷入局部最优的可能。（3）遗传算法使用的遗传操作是随机操作，同时根据个体的适应度值进行搜索，无需其他信息。（4）遗传算法具有全局搜索能力，善于解决复杂问题。

实际应用遗传算法时，往往出现早熟收敛等缺点，因此出现了许多用来改进遗传算法的策略。概括起来有以下几方面：（1）改变遗传算法的组成成分或操作技术。（2）采用混合遗传算法；（3）运用动态自适应技术，在进化过程中调整算法控制参数和编码；（4）采用非标准的遗传操作算子；（5）并行遗传算法。

2.2 CVRP 遗传算法

2.2.1 CVRP 问题描述及数学模型

CVRP的描述：一个中心车场，拥有装载能力为Q的车辆k辆，对n个客户点进行货物配送，每辆车从中心车场出发给若干个客户送货，最终回到中心车场，客户点i的货物需求为q_i，且$q_i < Q$，记车场编号为0，客户编号为$1,2,3,\ldots,n$，客户点i,j之间的距离为c_{ij}，求满足车辆数最小，车辆行驶总路程最短的运送方案。

定义变量

$$x_{ijk} = \begin{cases} 1 & \text{车辆}k\text{由}i\text{到}j \\ 0 & \text{否则} \end{cases} \qquad y_{ki} = \begin{cases} 1 & \text{车辆}k\text{访问}i \\ 0 & \text{否则} \end{cases}$$

则此问题的数学模型如下：

$$\min \quad m \tag{2.7}$$

$$\min z = \sum_{i}\sum_{j}\sum_{k} c_{ij} x_{ijk} \tag{2.8}$$

约束条件：

$$\sum_{k} y_{ki} = 1 \qquad (i = 0,1,\ldots,n) \tag{2.9}$$

$$\sum_{i} x_{ijk} = y_{kj} \qquad (j = 0,1,\ldots,n, \quad k = 1,\ldots,m) \tag{2.10}$$

$$\sum_{i} x_{jik} = y_{kj} \qquad (j = 0,1,\ldots,n, \quad k = 1,\ldots,m) \tag{2.11}$$

$$\sum_{i} q_i y_{ki} \le Q \qquad (k = 1,\ldots,m) \tag{2.12}$$

在上述模型中，（2.7）（2.8）代表目标函数，分别是最少车辆数和最短路径长度；（2.9）表示每个客户点只有一辆车服务且只服务一次；（2.10）（2.11）两式保证访问一个客户点的一辆车也离开这一点；（2.12）定义了车辆是带装载能力约束的。一般情况下，添置车辆的固定费用远高于车辆的行驶费用。对于上述多目标问题，在满足客户需求和车辆负载的限制条件下先确定最少车辆数，再最小化运行路程。车辆数为 $m \ge \sum_{i} q_i / Q$ 的最小整数。

2.2.2 CVRP 的遗传算法

由上述模型可知，求解 CVRP 的关键是合理确定车辆数以及每个车辆访问客户的路径顺序，在满足车辆载重量和客户需求约束条件的情况下使得总路程最小。这是一个整数规划问题，属于 NP 完全问题，不容易求得精确解，为求较好的近似解，构造遗传算法如下：

1）个体（染色体）编码

CVRP 数学模型的解向量可以表示成一条长度为 $k+n+1$ 的个体编码串 $(0, i_1, i_2, \ldots, i_p, 0, \ldots, 0, i_q, \ldots, i_r, 0)$，如(012304506780)的个体编码可以解释为：第一辆车从中心车场出发，经过客户 1,2,3 后，回到中心车场 0 形成子路径 1，第二辆车也从中心车场出发，途经客户 4,5 后回到中心车场，形成子路径 2，第三辆车从中心车场出发，途经 6,7,8 后，回到中心车场。

2）适应度计算

利用车辆负载约束，设计惩罚函数：

$$F = M\sum_{k}\max(\sum_{i}q_i y_{ki} - Q,0) \tag{2.13}$$

M 为惩罚系数，取 M 为一个较大的正数。

采用下式作为适应度函数：

$$fitness = 1/(\sum_{i}\sum_{j}\sum_{k}c_{ij}x_{ijk} + F) \tag{2.14}$$

3）选择

将每代种群中的个体，按照适应度值排序，保留最佳个体，直接进入下一代种群，剩下的个体用轮盘赌法选择产生。

4）交叉

由于个体编码中有重复0，根据一般的部分匹配交叉，结果中会出现邻近两位全为0，以及个体中车辆路径条数大于或小于 k 的非可行解。针对上述情况，将部分匹配交叉算子(PMX)进行改进，既使得子代个体良好地继承了父代个体的特征，也可以产生不同的个体。对于两个完全相同父代个体，首先改变其中一个父代个体的某段路径长度，以及另一父代个体的先后顺序，然后再通过PMX算子得到与父代不同的个体，这样可以使得种群得到更大的改变。这种交叉操作方式通过对父代个体中较好的部分路径的继承和组合来加快搜索速度。

具体方法：如果选择的两个父代个体不相同，分别在两个父代个体中选择相同的第 m 车辆路径，比较这两段路径的长度，选择较短的路径为交换部分，进行交换，交换过后的路径结果可能有重复的客户点，将原父代个体中与交换部分重复的客户点，按照由小到大的顺序将其替换为不在此个体的客户点，并保持父代个体的零元素位置，这样可以得到两个新的子代个体。如果选择的两个父代个体完全相同，将其中一个父代个体中用于分隔路径的0与任一非0且其相邻元素也非0的元素交换，将另一个父代个体基因串逆排，得到不相同的两个新子代个体，将这两个个体进行上述交叉操作。例如两个父代个体同时为（02078653140），经过改进的交叉操作后生成子代个体(02135687040)和(02870653140)。

5）变异

随机选取个体中不同基因（客户点）进行两点互换，产生一个新个体。这样对原个体有较大改动，可以达到较好的变异效果。

算法步骤：

step1　选用 $0,1\ldots,n$ 编码，0为中心车场，$0,1,\ldots,n$ 代表客户点，设置终止条件和种

群规模；

step2 迭代次数$t=0$；随机生成初始种群$P(0)$；

step3 对种群中的每一个个体，计算适应度值；

step4 最优选择，保留当前适应度值最好个体；

step5 剩余个体进行轮盘赌选择，进行交叉和变异操作，生成下一代种群P；

step6 若满足算法终止条件，则停止；否则，令迭代次数$t=t+1$，转 step3。

2.2.3 计算实例

应用*Matlab*语言编制遗传算法程序用于求解 CVRP 问题，对一个有 8 个商店和 1 个配送中心，车容量为 8 的车辆路径问题进行求解，客户 1-8 的需求量分别为：1.5，1，3.5，2，1，3，2，2。用 0 表示车场中心，各客户之间及与中心的距离如下表 2.1：

表 2.1 客户点距离表

距离	0	1	2	3	4	5	6	7	8
0	0	40	60	75	90	200	100	160	80
1	40	0	65	40	100	50	75	110	100
2	60	65	0	75	100	100	75	75	75
3	75	40	75	0	100	50	90	90	150
4	90	100	100	100	0	100	75	75	100
5	200	50	100	50	100	0	70	90	75
6	100	75	75	90	75	70	0	70	100
7	160	110	75	90	75	90	70	0	100
8	80	100	75	150	100	75	100	100	0

设置种群为 100，算法迭代 100 次，经过试验得，最优行驶路径：0-1-3-5-8-0、0-2-6-7-4-0，路径总长度 655，与文献[54]结果相同。

2.3 VRPTW 遗传算法

2.3.1 VRPTW 问题描述及数学模型

VRPTW 描述为：一个中心车场，拥有装载能力为Q的车辆k辆，对n个客户点进行货物配送工作，客户点i的货物需求为q_i，且$q_i<Q$,车辆必须在一定时间范围$[e_i,l_i]$内到达，求满足需求的路程最短的车辆行驶路径安排方案，并使用尽量少的车辆。设车场编号为 0，客户编号为$1,2,3,\ldots,n$，定义变量

$$x_{ijk}=\begin{cases}1 & 车辆k由i到j\\0 & 否则\end{cases}\qquad y_{ki}=\begin{cases}1 & 车辆k访问i\\0 & 否则\end{cases}$$

问题的数学模型如下：

$$\min \quad m \tag{2.15}$$

$$\min Z=\sum_i\sum_j\sum_k c_{ij}x_{ijk} \tag{2.16}$$

约束条件：

$$\sum_k y_{ki}=1 \qquad (i=1,\ldots,n) \tag{2.17}$$

$$\sum_i x_{ijk}=y_{kj} \qquad (j=1,\ldots,n,\quad k=1,\ldots,m) \tag{2.18}$$

$$\sum_j x_{ijk}=y_{ki} \qquad (i=1,\ldots,n,\quad k=1,\ldots,m) \tag{2.19}$$

$$\sum_i q_i y_{ki}\le Q \qquad (k=1,\ldots,m) \tag{2.20}$$

$$e_j\le t_j=\sum_{i\ne j}\sum_k x_{ijk}(t_i+t_{ij}+s_i)\le l_j \qquad (j=1,\ldots,n) \tag{2.21}$$

在模型中，c_{ij}代表i,j之间的距离，t_i为车辆在客户点i开始服务的时间，t_{ij}表示车辆从客户点i到j的行驶时间；s_i表示车辆在客户点i的服务时间；$t_0=0$，$s_0=0$；（2.15）（2.16）代表目标函数，分别是最少车辆数和最短路径长度；（2.17）表示每个客户点只有一辆车服务且只服务一次；（2.18）（2.19）两式保证访问一个客户点的一辆车k也离开这个点；（2.20）定义了车辆是带装载能力约束的；（2.21）代表时间窗约束。由于出动一辆车的固定成本远远大于车辆的行驶成本，因此寻求能完成任务的最少车辆数是减少成本的有效方法，但VRPTW无法确定使用的车辆数，需要优化算法进行计算。

2.3.2 VRPTW的遗传算法

由上述模型可知，求解VRPTW的关键是合理确定车辆数以及每个车辆访问客户的路径顺序，在满足车辆负载约束和时间窗约束条件的情况下使得总路程最小。这是一个整数规划问题，属于NP完全问题，不易求得精确解，为求较好的近似最优解，构造遗传算法如下：

1）优先关系的确定[55]

优先关系指的是客户被服务的先后次序，它可以根据起点到各客户点的距离确定，

也可以根据每个客户点的时间窗来确定，还可以通过加权因子由二者共同来确定。在满足车容量和时间窗的约束前提下，访问与起点 0 距离成本较小的客户。构造出下面的评价函数：

$$P(j)=\omega_1\frac{|t_{0j}-e_j|}{|l_j-e_j|}+\omega_2\frac{|t_{0j}-l_j|}{|l_j-e_j|}+\omega_3\frac{c_{0j}}{\max\limits_{1\le k\le n}c_{0k}} \tag{2.22}$$

上式中ω_1，ω_2，ω_3为权重系数，满足$0\le\omega_1,\omega_2,\omega_3\le 1$，且$\omega_1+\omega_2+\omega_3=1$；式中前两项分别为：从起点 0 访问客户$j$的时间窗口左右边界的绝对差与整个时间窗口宽度的比值，显然，当$t_{0j}\in[e_j,l_j]$时，$\frac{|t_{0j}-e_j|}{|l_j-e_j|}\le 1$，$\frac{|t_{0j}-l_j|}{|l_j-e_j|}\le 1$同时成立，而当$t_{0j}\notin[e_j,l_j]$时，$\frac{|t_{0j}-e_j|}{|l_j-e_j|}$，$\frac{|t_{0j}-l_j|}{|l_j-e_j|}$二者中至少有一项大于 1；第三项为距离因素，显然，$\frac{c_{0j}}{\max\limits_{1\le k\le n}c_{0k}}\le 1$总是成立。把客户按照其评价值从小到大的顺序进行排列，就得到各客户点被服务的优先关系。在式中，如果$\omega_1=\omega_2=0,\omega_3=1$，则评价函数只考虑距离因素，不考虑时间窗因素。反之，如果$\omega_1,\omega_2\ne 0,\omega_3=0$，则评价函数只考虑时间窗因素，不考虑距离因素。

2）个体（染色体）编码

对于种群个体采用的长度为n的整数序列（n表示顾客的数量）进行编码。个体中整数的排列顺序表示所有车辆访问客点的顺序，例如 123456789。

3）解码过程

在一个个体中，按照从左到右的顺序，满足优先关系的基因确定一个单车路径。例如 8 个客户点的优先关系为 1 2 3 4 5 6 7 8，个体 2 1 4 7 3 8 5 6，个体根据优先关系可以划分为（2），（1 4 7），（3 8），（5 6），此个体的车辆总数为 4。这样，每个个体代表不同的路径方案，同时可以自动搜寻个体车辆总数。

4）个体的评价

对经过解码得到的某个体的所有单车路径，检查是否满足车辆负载约束和时间窗约束，若二者都满足，则该个体对应问题的一个可行解，求出其目标函数值$f(i)$，否则，该个体对应非可行解，此时赋予$f(i)$一个较大整数M，$f(i)=f(i)+M$，定义适

应度函数为 $F(i)=1/f(i)$。

5）选择

将每代种群中的个体，按照适应度值排序，保留最佳染色体，直接进入下一代种群，剩下的个体用轮盘赌法选择产生。

6）交叉

采用部分匹配交叉算子（PMX），这种算子首先要确定两个交叉点，然后将两个交叉点之间的子路径进行互换。假设有 2 个父代个体 P_1、P_2，P_1 为 1 2|4 5 7 6|3 8 9，P_2 为 2 1|5 4 3 7|8 6 9,交叉位置用“|”表示。先把 P_1 中的 4 5 7 6 顺序赋给子代个体 P_1' 的前 4 个位置，然后用 P_2 中的元素逐个与 4 5 7 6 相比，若相同，则置之不用；若不相同，就将其顺序放在 P_1' 的后续位置，从而得到子代个体 P_1' 为 4 5 7 6 2 1 3 8 9，同样方法可以得到子代个体 P_2' 为 5 4 3 7 1 2 6 8 9。部分匹配交叉算子的特点在于能够最大程度地确保子代个体能够继承父代个体的基因属性，通过对父代个体中较好部分路径的继承和组合来加快搜索速度。

7）变异

对种群进行变异操作，随机选取其中某个体路径的基因（客户点）进行两点互换，产生一个新的个体。

算法步骤：

step1 选用 $0,1,\ldots,n$ 编码，0 为中心车场，$0,1,\ldots,n$ 代表各客户点，设置终止条件和种群规模；

step2 迭代次数 $t=0$；随机生成初始种群 $P(0)$；

step3 对种群中的每一个个体，计算适应度值；

step4 最优选择，保留当前适应度值最好个体；

step5 剩余个体进行轮盘赌选择，进行交叉和变异操作，生成下一代种群 P；

step6 若满足算法终止条件，则停止；否则，令迭代次数 $t=t+1$，转 step3。

2.3.3 计算实例

应用 *Matlab* 语言编制遗传算法程序用于求解 VRPTW 问题。对一个有 8 个商店和 1 个配送中心的 VRPTW 进行求解，用 0 表示车场中心，各客户之间及与中心的距离如

表 2.1。

表 2.2 客户点的需求及特征

客户	1	2	3	4	5	6	7	8
需求	1.5	1	3.5	2	1	3	2	2
服务	1	2	1	3	2	2.5	3	0.8
时间	[1,4]	[4,6]	[1,2]	[4,7]	[3,5]	[2,5]	[5,8]	[1.5,4]

中心仓库拥有5辆车的车队，车辆允许容量为8，行驶速度为50。

设置种群为 100，算法迭代 100 次，经过多次试验比较得，所需车辆数为 3，最优行驶路径：0-8-5-7-0、0-6-4-0、0-3-1-2-0，路径总长度 910，与文献[56]的节约算法结果相同。

2.4 小结

通过分析 CVRP 和 VRPTW 问题，可以对确定性车辆问题有较为深入的认识，有助于研究 PDPTW 复杂问题的数学模型。通过对遗传算法解决这两个问题的初步研究，了解遗传算法在解决车辆路径问题这一组合优化问题方面的发展，并在此基础上对遗传算法改进了交叉算子，得到了较好解。这些工作为下面提出多策略分组编码遗传算法解决 PDPTW 作了一定准备。

第三章 解决PDPTW的多策略分组编码遗传算法（MSGGA）

3.1 PDPTW问题描述及数学模型

PDPTW问题描述：假定车辆的数目不受限制，且每辆车的最大负载量相同。令N为运输需求集合，对每个需求任务$i \in N$，一个大小为q_i的负载量要从初始点N_i^+运送到终点N_i^-（$+q_i$为装载，$-q_i$为卸货），分别将$N^+ \equiv \bigcup_{i \in N} N_i^+$和$N^- \equiv \bigcup_{i \in N} N_i^-$定义为初始点集合和终点集合，假定$N^+$与$N^-$不相交，令$V \equiv N^+ \bigcup N^-$，$n = |V|$，令$M$和$m$分别为车辆集合和车辆数。每辆车有相同的最大负载量$Q$，车辆从车场$O$点出发并最终返回到$O$点，车场$O$点没有货物负载需求。$\forall i, j \in V \bigcup O$，令$d_{ij}$表示两点间运行距离，$t_{ij}$代表运行时间，令$[e_i, l_i]$表示时间窗，即客户点$i$必须在这个时间段内被服务，$s_i$表示车辆在客户点$i$的停留时间。以下给出两个定义：

定义3.1 车辆k的装卸路径R_k是一条贯通子集$V_k \subset V$的有向路，满足如下条件：

1. R_k开始并结束于O。
2. $\forall i \in N$，N_i^+和N_i^-全部或者其中之一属于V_k。
3. 如果N_i^+和N_i^-全部属于V_k，N_i^+要在N_i^-之前被服务。
4. V_k中的每个客户点被车辆k服务一次且只有一次。
5. 车辆在任何时间的负载量不超过Q。
6. 对任何客户点i，车辆k到达时间A_{ik}满足$A_{ik} \le l_i$，当$A_{ik} < e_i$时，车辆必须在i点等待，等待时间$W_{ik} = \max\{0, e_i - A_{ik}\}$，离开时间$D_{ik} = \max\{A_{ik}, e_i\} + s_i$。

定义3.2 一个装卸需求是一个路径集合$R \equiv \{R_k \mid k \in M\}$，满足如下条件：

1. $\forall k \in M$，R_k是车辆k的一条装卸路径。
2. $\{V_k \mid k \in M\}$是V的一部分。

$f(R)$为需求R根据特定目标函数得到的值。PDPTW 可以定义为一个优化问题，即寻找一个装卸需求R，使得函数值$f(R)$最小。

PDPTW 有多种不同的优化目标，本文着重研究的优化目标如下：

1. 最小化使用车辆总数，这在总运行花费中占有最大比重。
2. 最小化行车总距离，指在计划中所有路线的总长度最小。
3. 最小化等待总时间，使在所有客户点的总等待时间最短。

定义两个二进制变量：

$$x_{ijk}=\begin{cases}1 & \text{车辆}k\text{由}i\text{到}j\\ 0 & \text{否则}\end{cases} \qquad y_{ki}=\begin{cases}1 & \text{车辆}k\text{访问}i\\ 0 & \text{否则}\end{cases}$$

z_{ik}定义为当车辆k访问客户点i时车辆的负载量，定义多目标函数要满足使用车辆总数，行车总距离和等待总时间三个目标的最小化，原则是由前到后重要性依次减弱，PDPTW 的数学模型可定义为：

$$\min \quad m \tag{3.1}$$

$$\min \quad \sum_{k\in M}\sum_{i\in V}\sum_{j\in V} d_{ij}x_{ijk} \tag{3.2}$$

$$\min \quad \sum_{k\in M}\sum_{i\in V} W_{ik} \tag{3.3}$$

约束条件：

$$\sum_{k\in M} y_{ik}=1 \qquad (\forall i\in N) \tag{3.4}$$

$$\sum_{k\in M}\sum_{j\in V} x_{ijk}=1 \qquad (\forall i\in V) \tag{3.5}$$

$$\sum_{i\in V} x_{iok}=1 \qquad (\forall k\in M) \tag{3.6}$$

$$\sum_{j\in V} x_{ojk}=1 \qquad (\forall k\in M) \tag{3.7}$$

$$\sum_{i\in V} x_{ihk}-\sum_{j\in V} x_{hjk}=0 \qquad (\forall h\in V \quad \forall k\in M) \tag{3.8}$$

$$(z_{ik}+q_j)x_{ijk}\le z_{jk} \qquad (\forall i\in V\cup 0 \quad \forall j\in V \quad \forall k\in M) \tag{3.9}$$

$$0\le z_{ik}\le Q \qquad (\forall i\in V \quad \forall k\in M) \tag{3.10}$$

$$e_j\le t_j=\sum_{i\ne j}\sum_{k} x_{ijk}(t_i+t_{ij}+s_i)\le l_j \qquad (\forall j\in V) \tag{3.11}$$

$$D_{pk}\le A_{qk} \qquad (\forall k\in M \quad \forall i\in V \quad p=N_i^+, q=N_i^-) \tag{3.12}$$

模型中，t_i为车辆在客户点i开始服务的时间，$t_o=0$。约束条件说明：（3.4）保证每个需求被安排一辆车；（3.5）表示每个客户点只能被服务一次；（3.6）和（3.7）保证每辆车从车场出发并回到车场；（3.8）说明如果一辆车到达一个点，它也必须离开这个点；（3.9）—（3.10）保证了负载容量限制；时间窗和前序约束限制是由（3.11）—（3.12）表述的[57]。

3.2 PDPTW 解决方法

目前，国内外关于 PDPTW 的研究已经取得了一些进展，求解的方法可以分为最优化方法、近似算法和亚启发式算法。

3.2.1 最优化方法

最先应用于解决 PDPTW 的优化算法用来解决比一般 m-PDPTW 要简单的 1-PDPTW，其中最重要的是随机规划算法[58][59]。进入 90 年代，针对 m-PDPTW 的优化算法，Dumas[41]提出一种分枝界定法。Savelsbergh 和 Sol [40]运用了嵌入启发式和特殊分类方案来加速搜索，得出需求为 30 个的测试算例的最优解。由于在处理算例时，计算时间根据问题不同有变化，Savelsbergh 和 Sol 认为这种优化算法稳定性差，不适合一般性问题。

3.2.2 近似算法

由于 PDPTW 具有一定复杂性，出现了几种启发式近似算法，多数根据先聚类后分配路径的方式组织解，例如 Bodin 和 Sexton [60]提出的一种迭代过程，在每次迭代中先将需求指派给车辆，然后对每辆车路径解决 1-PDPTW。Desrosiers[61]提出一种小聚类的概念，即服务一个或多个需求，每个聚类都是一个单满载路径，用启发式方法产生许多小聚类，就可以解决一系列局部问题。Ioachim[62]提出一种精确算法来产生小聚类的最优集。启发式插入算法已经成功解决了 VRPTW(Solomon)[63]，试验表明它也适用于 PDPTW。Jaw[64]提出用启发式插入算法解决与 PDPTW 问题相似的 DARPTW，Madsen[65]在此基础上改进了算法。Van der Bruggen[66]等人使用了两阶段局部搜索算法解决 1-PDPTW，采用 Lin 和 Kernighan[67]为解决 TSP 提出的弧交换法进行搜索。Paraftis[68]提出相似的弧交换过程。Toth 和 Vigo[69]采用了针对需求的改进方法。

3.2.3 亚启发式算法

近似算法在搜索中常常陷入局部最优，亚启发式算法由于避免了局部最优被人们

广泛应用。目前已经出现许多亚启发式算法解决 VRPTW，解决 PDPTW 应用广泛的是禁忌搜索算法。Gendreau[70]等人提出的禁忌算法，采用适应记忆的方法，加入一个变动边界的结构，并在不同操作过程中进行检测。Nanry Barnes[71] 在三种距离移动概念的基础上提出了解决 PDPTW 的禁忌搜索算法。Lau 和 Lim[72]提出的禁忌搜索算法，采用局部启发式插入算法来构造初始解，算法的搜索邻域定义方式与 Nanry[71]等人一致，用 Nanry 提出的 9 个有 100 个客户点的算例进行测试，给出了基于 27 个 Solomon VRPTW 算例产生的 PDPTW 算例的详细描述，计算了其近似最优结果。Li 和 Lim[73] 则提出禁忌模拟退火的混合算法，用三种不同的需求交换来定义局部搜索的邻域结构，在所有 56 个 Solomon[63] VRPTW 算例的基础上构造了 PDPTW 算例，并给出了近似最优结果。目前针对 PDPTW 的遗传算法仍然很少，多数文献用遗传算法解决 1-PDPTW，如文献[74]。文献[42]是最早成功应用分组编码遗传算法解决 m-PDPTW 的，本文在其基础上作了较大改进，提出了多策略分组编码遗传算法。国内关于 PDPTW 的研究刚刚起步，还处于探索阶段，关于问题解决方法的研究比较单一，集中在禁忌搜索算法的研究上。如贾永基[75]等人运用加入快速局部搜索的混合禁忌搜索算法解决此问题，蓝伯雄[76]等人用概率式禁忌搜索算法试验了有 200 个需求点的问题。本文研究用遗传算法思想解决 PDPTW 问题有一定现实意义。

文献[42]提出的遗传算法应用了 Falkenauer(1998)[45]所提出的分组遗传编码，被称为分组编码遗传算法(Grouping Genetic Algorithm，GGA)。本章研究文献[42]介绍的分组编码遗传算法（GGA），改进了用来生成可行解的启发式插入搜索算法，使其具有一定禁忌机制，设计了高效率的数据结构，对算法进行重现，并编程在计算机上实现。经过算例测算，虽然其计算时间有较大减小，但是在对最优解的搜索上并没有明显改进，尤其是对客户点随机分布和客户点时间约束相近的问题。这些问题一般有多个可行解，搜索空间较大，分组编码遗传算法的随机搜索不能保证可以稳定搜索到最优解，并且容易出现“早熟”。为了使得分组编码遗传算法可以在较短时间有效收敛到最优解，本文在其基础上加入三种路径调整策略，提出了多策略分组编码遗传算法（Multi-strategy Grouping Genetic Algorithm，MSGGA）。此算法有效解决了原分组编码遗传算法的不足，经过对算例的计算结果分析，多策略分组编码遗传算法表现出了更加优良的收敛性和稳定性。

3.3 分组遗传编码

用遗传算法解决问题时，首先要确定个体（染色体）并对它进行编码处理。对于 PDPTW 来说，标准遗传算法由二进制表述的个体编码已经不适用于这一类包含复杂信息的问题。与 VRPTW 相比，PDPTW 有明确的前序限制，即装载点必须在相应同需求的卸货点前服务，隐含需要满足的条件是同一对需求的装载点和卸货点需要安排同一辆车。由于编码时要考虑到问题各种情况所包含的信息，很难找到一种适合的方案，因此遗传算法在 PDPTW 上没有得到广泛应用。用遗传算法解决 1-PDPTW 的编码方式与解决 VRPTW 相似，即选择客户点顺序编码作为所有位置的排列（详见本文第二章对 VRPTW 遗传算法的描述），但是这种编码方式无法应用于处理如 m-PDPTW 等进一步的复杂问题。

PDPTW 具有负载，前序和时间窗口的限制，所有客户点的可能排列中的绝大部分会导致非可行解，因此要运用特殊的机制以确保产生的解是可行的。Potter 和 Bossomaier[77]，Jih 和 Hsu[78]以及 Schonberger[79]都运用了基于交换的编码。Jung 和 Haghani[80]运用了一种称为随机键的概念来将 PDPTW 的组合部分和路径同时融入编码中，但实际结果并不理想。Kopfer[81]，Blanton 和 Wainwright[82]将所有任务需求的交换来代替点的交换，组成一个解的间接编码方式，将其应用于 VRP 和 VRPTW，这种基于顺序的编码还没有被应用到 PDPTW 上，但是这种编码在解决大型组合问题上显现出了严重不足。

文献[42]应用由 Falkenauer [45]提出的基于分组的编码方案，此编码方式既可以保持个体所代表解的可行性，又便于对其进行后续的遗传操作。分组编码个体的每个基因代表了一组目标而不是单个目标，起初用来解决纯组合问题如背包问题。下面具体叙述分组编码如何应用于 PDPTW。

个体（染色体）中的每个基因代表安排给某一辆车的所有需求组合，即这辆车的行驶路径。个体长度，即个体基因数量是不确定的，并且依赖于解路径所需的车辆数。图 3.1 描述了一个由 3 辆车运送 7 个需求的简单例子的解的编码：

如图 3.1 第一个图形（由左至右）可知，7 个需求分别由标号为 1、3、8 的三辆车服务，每辆车都有相对独立的行驶路径，满足问题所要求的前序条件，在满足负载和时间约束的情况下，这三辆车路径的组合，就是满足这个简单 PDPTW 的一个可行解。第二个图形所表示的同样是需求在路径中不同的分布情况。如车辆 1 先满足需求 2 然后满足需求 4，车辆 3 先满足需求 1 和 6 的装载点要求，然后满足其卸货点要求，车辆 8 的情况更加复杂，不同需求之间的装载点和卸货点是互相交叉的，以上的种种情况在

实际计算时都要考虑到。由于单独考虑装载点和卸货点比较复杂，所以分组编码将共同属于一个需求的装载点和卸货点作为一个整体进行考虑，如图 3.1 下方的车辆与需求的关系图。这里将车辆编号是为了便于进行下一步遗传操作中的交叉和变异，具体描述见 3.5 节。

在生成个体时，本文采用了一种启发式插入搜索算法（见 3.6 节），以保证个体的单个基因是可行的，即每辆单车的行驶路线满足问题给定的负载限制、时间限制和前序限制。应用插入法在生成种群时保证了个体解的可行性，并且在进行遗传操作时能够比较容易地产生可行解。

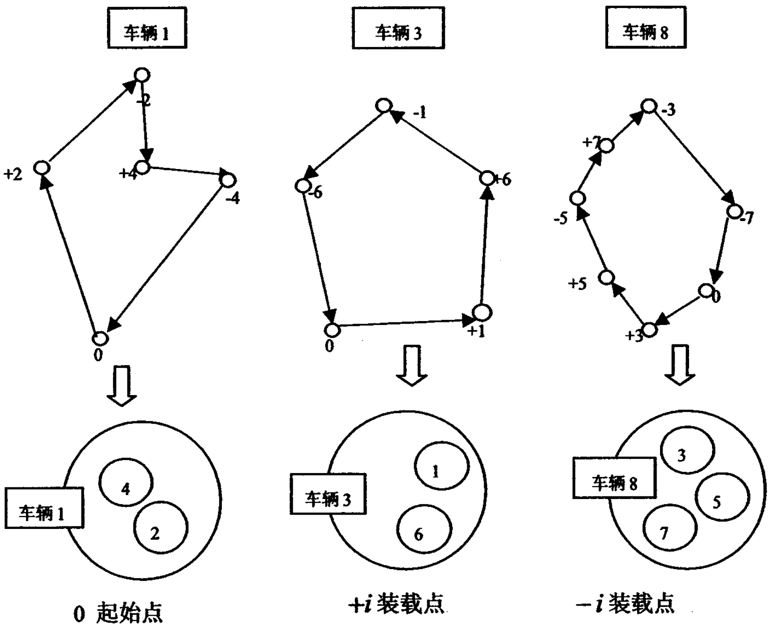

图 3.1 车辆与需求关系图

3.4 适应度函数

由于 PDPTW 比较复杂，存在多个目标，所以不能简单地通过计算某一个目标来评价个体的优劣，要将各个目标进行综合考虑。一般认为有三个需要优化的方面，即最小化使用车辆总数，最小化行车总距离和最小化车辆等待总时间。出于实际情况考虑，通常都是在最小化使用车辆总数的情况下最小化行车总距离，同时使得车辆的等待总时间尽可能小。在本文所运用的遗传算法中，为实现上述目标构造了一个有适应调节能力的惩罚函数，采用了将车辆数，行车距离，等待时间加权相加作为适应度值进行评价，分别对每个目标值乘一个权值（惩罚系数），根据目标的优先程度调整权值的大小。*kv* 代表使用车辆总花费的惩罚系数，*kd* 代表行车总距离的惩罚系数，*kw* 代

表等待总时间的惩罚系数。在解决实际问题时，根据问题的不同需求，相应设置 kv 、kd 和 kw 的值，使得所求得解侧重需要达到的目标。根据一般认知，在一个计划中车辆个数比行车总距离更重要，设置 kv 值远远超过 kd 和 kw 。本文设计的适应度函数可以使算法根据问题的不同要求进行调节，以搜索产生适合各种情况的可行解。

本文采用的适应度值函数如下：

$$\min \quad kv\times m+kd\times\sum_{k\in M}\sum_{i\in V}\sum_{j\in V}d_{ij}x_{ijk}+kw\times\sum_{k\in M}\sum_{i\in V}W_{ki} \qquad (3.13)$$

3.5 遗传算子

3.5.1 最优排序操作

在用遗传算法解决 PDPTW 时，希望在遗传操作中更多地获得不同的可行个体，本文对种群进行最优选择处理，首先用适应度值进行排序，然后将经过保留一定个数最优解操作后的种群进行下一步遗传处理。在本文遗传算法中，经过交叉算子、变异算子的种群都要经过最优选择操作，以使适应度值较优的个体始终处于种群的有利位置。

3.5.2 交叉

交叉算子关系到遗传算法的计算速度和最优解的产生，与遗传编码方式有密切关系，文献[42]根据上述遗传分组编码应用了一种组合交叉算子。具体思想如下：

1. 首先选取两个个体作为父代，分别在两个父代个体随机确定需要交叉的基因片段，基因片段由一个或多个单车路径（基因）组成，见图 3.2。

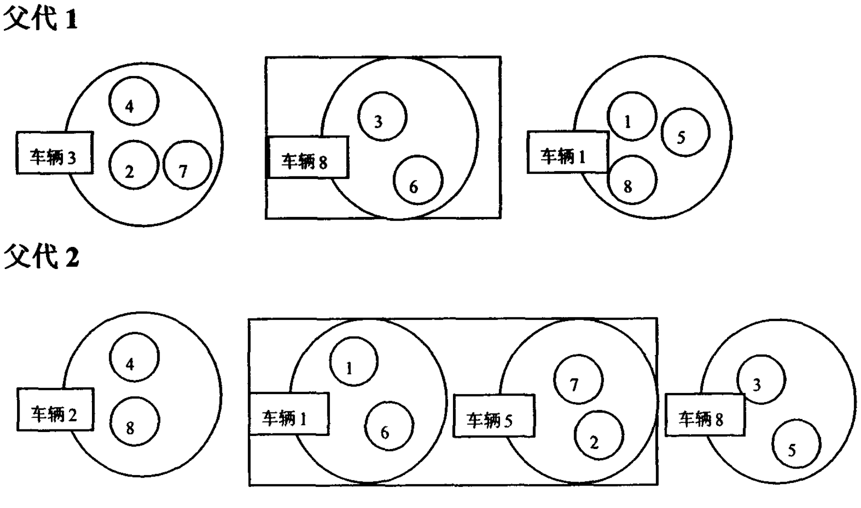

图 3.2 父代个体示意图

2. 将父代 2 的基因片段直接插入父代 1 基因片段（方框内）开始的位置。父代 2 的

基因片段（路径）直接采用，不进行重组。这种操作会产生一部分需求出现两次的结果（如 1、2、6、7），并且会有两个路径被安排给同一辆车（如车辆 1）。如图 3.3：

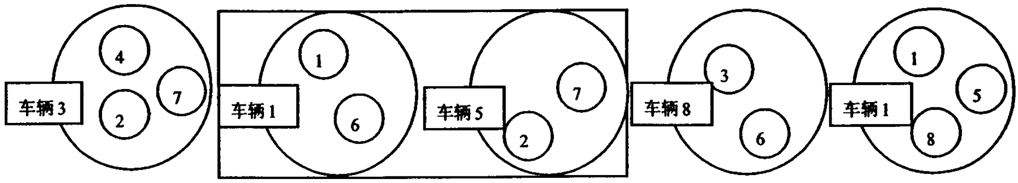

图 3.3 交叉操作示意图

3.为消除需求或车辆重复的情况，保持插入父代 2 片段路径需求不变，将原父代 1 中与插入片段相同的需求去除（如 1、2、6、7），将原父代 1 中与插入片段车辆相同的车辆去除（如车辆 1），去除车辆所剩的需求就成为没有被安排的需求。所得结果如图 3.4：

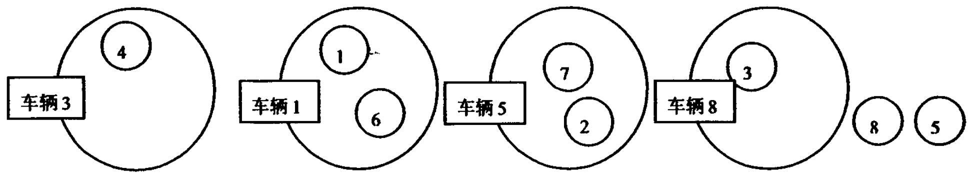

图 3.4 消除重复操作示意图

4. 将未安排需求重新用启发式插入算法随机插入个体。这可能会需要增加额外车辆以确保可行性，结束后，就可以获得一个完整的子代 1，见图 3.5。

子代 1

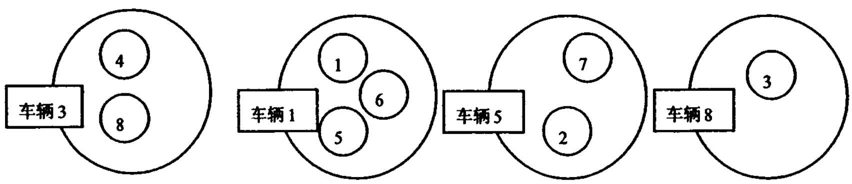

图 3.5 生成子代示意图

5. 交换父代角色，重复 2～4 步，可以产生第二个子代。

步骤描述：

begin

$for\quad i=1\quad to\quad [popsize/2]$

选取种群 P 中不同的两个个体作为父代个体 p_1, p_2；

分别确定 p_1, p_2 的交叉片段，运用上述交叉算子进行交叉，产生两个子代个体 p_1', p_2'；

保留 p_1', p_2' 为种群的第 $popsize + i*2-1$ 和 $popsize + i*2$ 个体，形成大小为 $2*popsize$ 的新种群 P'

end

上述交叉过程中，由于交换了一辆车完整的单车路径，使得交换部分在从一个父代到另一个父代的交换过程中保持了此单车路径的可行性，这样使算法在交叉算子中产生可行解的过程简化，有利于产生新的子代，保持种群进化后的多样性。在解决 CVRP 或 VRPTW 的遗传算法的交叉算子中，通常对以符号编码的父代个体中一部分客户点进行交叉（详见第二章对遗传算法的描述），本小节介绍的交叉算子则是基于相似原理，将一部分车辆路径进行交叉操作。

3.5.3 变异

变异算子是选取种群中具有较差适应度值的个体，通过对这些个体进行一定规则的操作，优化个体，从而使得整个种群得到改进。文献[42]采用的变异算子的主要思想有以下几步：1）在个体中随机选择一个基因片段。2）从个体中去除这个基因片段以及相应的路径（指这些路径中包含的所有需求），并将这些需求重新设定为未安排状态。3） 用启发式插入法将所有未安排的需求重新插入个体，并且可以适当安排新车辆以保证可行性。

步骤描述：

begin

for $i=1$ *to* *popsize*

选取个体 p_i；

随机选取 p_i 中的几个单车路径作为变异片段；

将这些单车路径包含的需求重新设置为未安排状态；

用启发式插入算法将需求插入个体，形成新个体 p_i'；

保留 p_i' 为种群的第 $popsize + i$ 个体；

end

3.6 启发式插入搜索算法

在对文献[42]介绍的启发式插入算法进行研究的基础上，本文进行了改进，在算法中融入禁忌思想。在本文所描述的 MSGGA 中，产生初始种群，交叉算子和变异算子都要应用到这种启发式插入搜索算法，它保证了在产生解的过程中路径的可行性，其所包含的禁忌思想也可以在产生个体时进行初步优化。这种启发式插入法贯穿于整个遗传算法中，插入的对象是未安排的需求（即没有安排给任一辆车的需求任务，包括装载点及其对应卸货点）。下面进行具体介绍：

将一个未安排需求插入当前路径，对路径中的所有车辆的单车路径可行插入都要分别检验，寻找根据评判准则（如适应度值）所规定的最优插入单车路径。对路径中的某个单车路径，试验路径中所有装载点和卸货点可以插入的位置，并检验是否满足前序，负载以及时间窗限制，以保证所产生个体路径的可行性。

由于装载点和卸货点是成对出现并且有先后顺序，一般先考虑装载点的插入，然后对属于同需求的相应卸货点进行相同操作。如果装载点和对应卸货点都可以顺利插入当前的单车路径，则完成了对这个需求任务的插入；如果装载点和对应卸货点中的任何一个客户点无法插入当前的单车路径，则将此需求对下一个单车路径进行插入检验；如果此需求在多个单车路径的插入都是可行的，则比较两个单车路径由于加入此需求后所产生的花费（包括距离和等待时间），取花费较小的单车路径作为此需求的插入路径。这种禁忌思想的加入，使得在将需求插入当前路径时，就已经有目标地选择优化性能较好的单车路径，因此，启发式插入算法具有自动寻优的特性。如果对于当前路径，此需求任务不能插入任何一个已存在的单车路径，则为这个需求分配一个新的车辆（即这辆车的新单车路径暂时由当前需求初始化）。

启发式插入算法的步骤：

Step 1 选取未安排需求i。

Step 2 对个体路径p中某个单车路径k进行需求i的插入检测，如果需求i（包括装载点和卸货点）插入单车路径k是可行的，计算此单车路径由于插入需求所增加的花费，包括行驶距离和等待时间。

如果需求i的装载点和卸货点中有一个不能插入单车路径k中，则检测对路径中另一个单车路径插入需求i的情况。

Step 3 经过对个体路径p中所有单车路径进行插入检测,如果需求i不能可行地插入任何现有单车路径中，则在个体路径p中为需求i安排一个新的单车路径，并计算由

此产生的花费。

如果需求i可以插入一个或多个现有单车路径，则比较这些可行单车路径由于插入需求i而增加的花费，选取花费最小的单车路径作为需求的最终插入单车路径。

Step 4 重复 step 1 到 step 3，直到所有未安排需求被插入个体路径p中。

用上述启发式插入搜索过程可以产生初始种群，即对所有的运输需求应用插入法，当所有需求被插入后，可以产生一个初始可行解，重复这一过程，直到初始解的个数达到需求数量的种群。在遗传算法的交叉和变异算子中，也要应用到启发式插入算法以使得经过这些操作得到的种群个体保持可行性。

本文改进的启发式插入搜索算法根据分组遗传编码的特性，在考虑问题要求的同时始终保持了所产生解的可行性，使得 MSGGA 具有良好的计算性。

3.7 数据结构

算法底层的设计就是数据结构的设计和过程与函数的设计，用高级语言表达，就是构造数据类型的定义和过程与函数的说明。数据结构用来反映一个数据的内部构成，是数据存在的形式。构造数据类型（数组类型、记录类型、（有限制的）集合类型、文件类型）对应于复杂的数据结构，复杂的数据结构允许成分数据本身具有复杂的数据结构，因而，构造数据类型允许复合嵌套；记录结构的特点是：（1）成分数据的个数固定，它们之间没有自然序，处于平等地位。每一个成分数据被称为一个域并赋予域名。（2）不同的域有不同的结构，不同的数据类型。（3）可以随机访问，途径是域名。

文献[42]并未给出其实现 GGA 的数据结构，本文根据分组编码的特点，设计了新的数据结构。

（1）将客户点已知信息设计为客户信息记录类型，包括：客户点编号、坐标、客户点需求、时间窗、要求服务时间和同需求对应装载（卸货）点编号。

（2）将车辆个体路径中的单车路径的各项信息设计为车辆信息记录类型，包括：车辆离开车场中心 0 后到达的第一个客户点编号、车辆行车总距离和车辆等待总时间。

（3）将客户点被车辆服务的信息设计为客户服务信息记录类型，包括：车辆离开该客户点的时刻、车辆离开该客户点时的负载、车辆在到达该点前访问的客户点和车辆在离开该点后访问的客户点。

（4）根据分组编码方式，将个体（染色体）包含各项信息设计为个体信息记录类型，包括：个体路径所包括单车路径的个数及车辆编号顺序、各单车路径车辆信息、

各客户点服务信息。

上面描述的数据结构可以很好地实现分组编码思想，它既包含了 PDPTW 问题的各项信息，同时也考虑到在算法实现过程中便于产生可行解和计算适应度值。从编程实现的运行效果来看，上述数据结构有效。

下面图 3.6 说明了个体记录结构的关系：

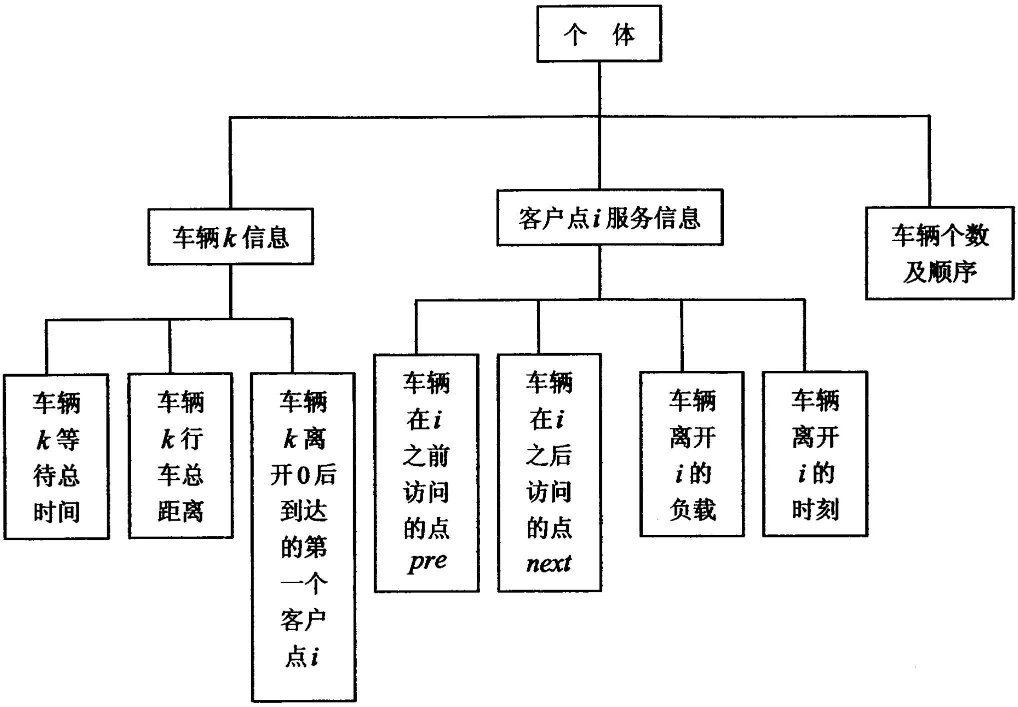

图 3.6 个体记录关系示意图

3.8 GGA 搜索步骤及流程图

本文在改进了启发式插入搜索算法（见 3.6 节）并设计新数据结构（见 3.7 节）的基础上，运用分组编码方式（见 3.3 节）和 3.5 节介绍的遗传算子，以 3.4 节描述的适应度函数为评价函数，实现了文献[42]介绍的分组编码遗传算法（GGA），下面介绍 GGA 的搜索步骤及流程。

GGA 搜索包括以下部分：（1）算法利用启发式插入算法来得到初始解和可行子代；（2）通过个体（序列）表现 PDPTW，进行遗传编码；（3）进行交叉，变异等遗传操作。

GGA 算法步骤描述：

input ：参数 *popsize*（种群大小），*m*（最大迭代次数）

begin

初始化种群 P（产生 $popsize$ 个初始个体）；

计算 P 中个体适应值，排序保优；

$iterationtime = 0$；

while $(iterationtime \neq m)$

对种群 P 进行交叉操作产生新种群 P'；

计算 P' 中个体适应值，排序保优；

对种群 P' 进行变异操作产生新种群 P''；

计算 P' 中个体适应值，排序保优；

$iterationtime++$；

end

return 种群中的最优个体作为最终近似最优解

end

GGA 流程图：

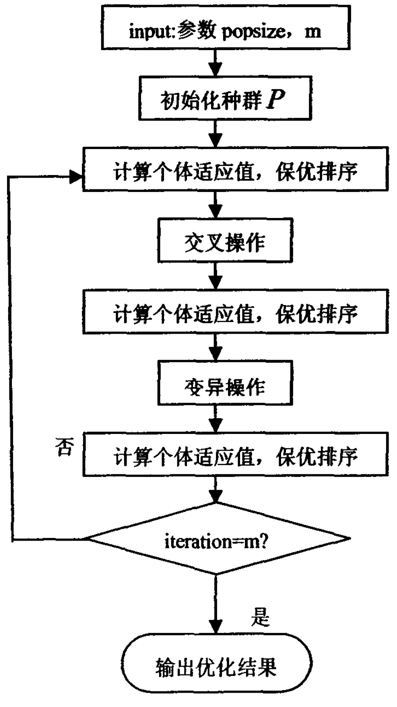

图 3.7 GGA 流程图

选用 Li 和 Lim[73]提出的具有 100 个客户的 PDPTW 算例对 GGA 进行测算，发现 GGA 在解决客户点平面位置分布局部聚集情况的算例可以很快收敛到稳定的最优解。但是解决客户点平面位置出现随机分布的情况时，最优结果不稳定，易出现早熟。这是由于客户分布越广泛，可行解越多，算法的搜索空间变大，在计算过程中，算法很快淘汰适应度值较差的模式，陷入局部最优。为使 GGA 可以比较稳定地收敛到全局最优解，需要在算法中加入新的思想，本文重点论述的多策略分组编码遗传算法（MSGGA）就是基于这点提出的。

3.9 路径调整策略

用 Li 和 Lim[73]提出的具有 100 个客户的 PDPTW 算例对 GGA 进行测算，在进行了大量试验后，选取了运用 GGA 不易求得最优解的几个算例，将试验求得的最好解与近似解比较，发现车辆路径安排相差不大，一般为单车路径个别客户位置需要调整，或者两条单车路径中的需求调整。为对 GGA 进行改进，本文设计了三种路径调整策略，分别解决在 GGA 过程中出现的这几种情况。

3.9.1 单车局部反向搜索策略

单车局部反向搜索策略是针对 PDPTW 解路径中的每辆单车路径进行优化。在结果比较中，总结了多个算例的结果特点，发现最优解与次优解的单车路径十分相似，只是局部有某些客户点的位置互换或有些客户点片段位置反转。这是由于 GGA 进化中的启发式插入算法与需求插入的顺序有密切的关系，并没有考虑如何进行单车路径的最优求解。对单车路径求最优解本身也是一个 NP 问题，一般定义为有负载、时间窗和前序约束的旅行商问题 TSP-CSTWPC（Travelling Salesman Problem with Capacity , Soft Time Windows and Precedence Constraints)[83]。本节介绍的单车局部反向搜索策略是对单车路径中两个客户点在路径中的位置进行交换，或者将路径某个片段的位置进行反转，目标是使单车路径的花费减少，从而使得总 PDPTW 解路径得到优化。单车路径调整策略的思想有以下几个步骤：

1. 在已有单车路径的客户点中按照一定规则选取两个交换点 $cross1$ 和 $cross2$；
2. 检测 $cross1$ 与 $cross2$ 之间客户点交换位置或者客户点片段位置的反转可行性（包括满足前序条件、位置交换后满足车辆路径的可行性以及使得单车路径总花费减少)；
3. 如果操作可行并使得花费减少，则交换客户点的位置或进行反转；否则，回

到 1 开始。

3.9.2 单个需求重排策略

在结果比较中，另一类常见的情况是两个不同单车路径中交换了一个需求，即这个需求插入不同单车路径会导致不同的结果。GGA 启发式插入算法是随机的，在算法初期过程中随机将需求插入某一路径；接下来的交叉操作是针对整个单车路径进行的，没有需求位置的变化；变异算子只是个体中某些单车路径的集体重排，并不是针对某个需求，降低了由单个需求变化产生优化的概率。因此，本节介绍的单个需求重排策略通过对个体中每个需求进行重新插入操作，将需求调整至最适合的单车路径中，使得整个车辆路径得到优化。具体步骤如下：

begin

for $i = 1$ *to* nr （nr 为已知需求个数）

在个体 p 中将需求 i 去除；

将需求 i 重新插入个体 p，形成新个体 p'；

end

end

3.9.3 部分需求重排策略

经过研究，第三类比较少见的情况与 3.9.2 节介绍的问题相似之处是两个不同单车路径有差别，但是差别较大，至少相差两个需求。这种情况比较少见，多数由于算例的客户点分布广泛，随机性大，有多个可行解，GGA 不容易收敛到最优解造成的。经过 GGA 生成的解个体一般都能得到较好的优化，本节介绍的部分需求重排策略在实际操作中作为另一种变异算子出现，即在迭代进行到某个设定时间，强制进行部分需求重排，使其即避免由于 3.5.3 介绍的变异算子进行重排时破坏了个体路径原有的结构，又可以进行较大程度的调整，跳出局部空间。下面是部分需求重排策略的具体步骤：

begin

随机选取 2-6 个需要插入个体 p 的需求；

在个体 p 中去除这些需求；

重新插入 p 形成新个体 p'；

end

3.10 多策略分组编码遗传算法(MSGGA)

经过上述三种调整策略的多策略分组编码遗传算法（MSGGA），优化了由 GGA 产生的解个体，使车辆路径得到改进。根据算例测算结果比较，三种情况出现的可能性并不是一致的，一般是单车路径需要调整的情况出现最多，其次是需要进行单个需求重排，部分需求重排的情况较少，通常出现于客户点分布广泛，可行解空间较大的算例。在对 GGA 进行路径局部调整时，对三种策略设置三个参数，指定迭代次数达到某程度时对种群或者种群最优解进行指定的某种局部调整，参数可以根据算例的特点设置。通常的规则是进行多次单车路径局部反向搜索调整，中等规模单个需求重排以及少数部分需求重排操作。本文将加入了路径调整策略的分组编码遗传算法称为多策略分组编码遗传算法（Multi-strategy Grouping Genetic Algorithm，MSGGA），以下是 MSGGA 的步骤及流程。

MSGGA 的步骤：

input ：参数 *popsize* (种群大小)，*m*（最大迭代次数）

begin

初始化种群 P（产生 *popsize* 个初始个体）；

计算 P 中个体适应值，排序保优；

$iteration = 0$；

while $(iteration \neq m)$

对 P 进行交叉操作产生新种群 P'；

计算 P' 中个体适应值，排序保优；

if $iteration == m1$（设定迭代次数 1）

对种群 P' 个体进行少量需求重排操作，形成新种群 P''；

else

对种群 P' 个体进行 GGA 变异操作形成新种群 P''；

计算 P' 中个体适应值，排序保优；

$iteration++$；

if $iteration == m2$（设定迭代次数 2）

对进行单车局部调整操作；

if *iteration* == *m*3（设定迭代次数 3）

进行单个需求重排操作；

end

return 种群中的最优个体作为最终近似最优解

end

MSGGA 的流程图：

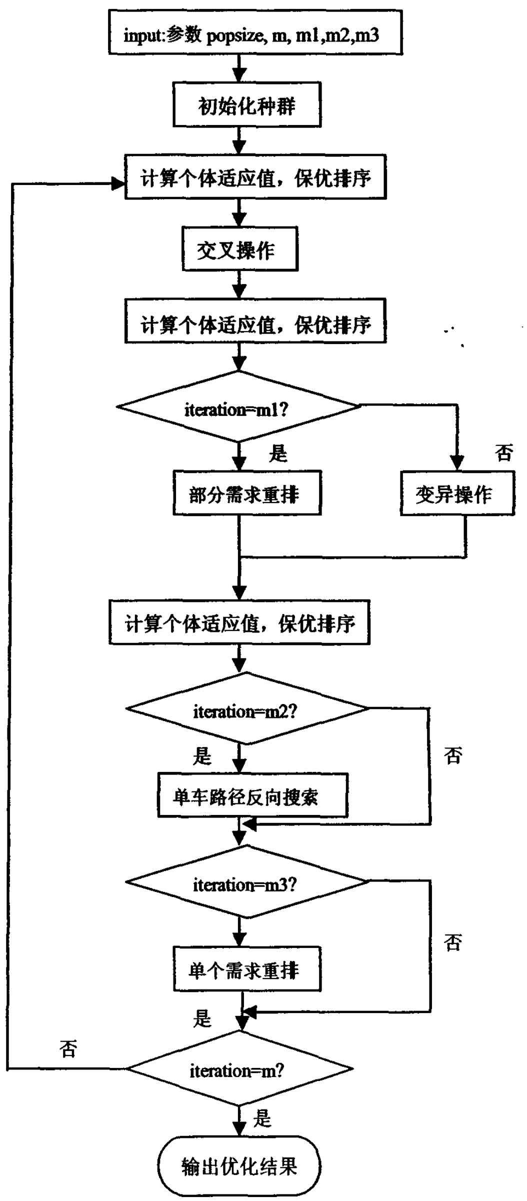

图 3.8 MSGGA 流程图

第四章　算例与算法分析

4.1 算例说明

自70年代车辆路径问题被提出以后，经过长时间的发展，很多确定性问题如TSP、VRP、VRPTW都已经有了公认的用来测算算法的算例集。通过比较不同算法解决同一算例的计算结果（包括解的精确性和计算时间），可以获知不同算法解决此问题的优劣。同时，一种优良的算法也可以找到作为NP-问题算例的更好解。虽然PDPTW在现实世界中有广泛应用，但是由于约束条件多，计算复杂，一直到90年代末，都没有用来测试解决PDPTW算法的一般性算例。

由于PDPTW在约束条件上只比VRPTW增加了前序条件（即装载点与对应卸货点在同一条单车路径中，并且装载点在单车路径中的位置在卸货点之前），所以研究人员都希望从已知 VRPTW 的算例中产生合理的 PDPTW 算例。目前用来测试算法的VRPTW算例是Solomon[63]提出的，此算例集包含情况比较广泛，可以用来测算各种算法。Li 和 Lim[73]在2001年提出了一系列PDPTW算例，用来测算他们在文章中提出的禁忌模拟退火算法的性能。其中100个客户点的算例是由Solomon提出的 VRPTW算例集合得出。算例产生方法如下：将每个算例给定参考解的客户点随机配对，如果参考解路径的客户点是奇数，则加入辅助虚构点。为保证参考解的可行性，虚构参考点应与其对应点配置一致，每个算例至少有100个点，相当于至少50个需求。

为从 Solomon 的 VRPTW 算例中产生 PDPTW 测试问题，Li 和 Lim 在最优解不知道的情况下假设了最优解。根据 Solomon 的分类，算例分6类，即LC1、LC2、LR1、LR2、 LRC1 和 LRC2。通过对这 6 类算例中客户点的平面位置分布进行分析，发现在LC1和LC2类算例中，客户点的平面位置是局部聚集的；在LR1和LR2类算例中，客户点的平面位置是随机分布的；在 LRC1 和 LRC2 类算例中，客户点的平面位置部分聚集，部分随机分布。LC1、LR1和LRC1类算例设定车辆容许负载量较小，客户点有局部排列趋势；而 LC2、LR2 和 LRC2 类算例设定车辆容许负载量较大，客户点分布的范围较大。目前，大多数解决 PDPTW 问题的算法都用 Li 和 Lim 的 PDPTW 算例来测算。本文也采用这个算例对MSGGA进行测试，并将测试结果与文献[42]中的结果进行比较。

4.2 算法性能分析

为了研究本文设计的多策略分组编码遗传算法（MSGGA）的性能，用 Pascal 语言编程在计算机上仿真实现，对 100 个客户点的 PDPTW 算例进行运算，所有测试都是在 PC（Pentium4，2.8GHz）机上进行的。由于 MSGGA 本质上是随机的，为能够更好地测算算法的稳定性和收敛性，对每个算例进行 MSGGA 的 30 次运算，收集实验数据，并与文献[42]给出的 Li 和 Lim 的禁忌搜索算法以及 GGA 进行结果比较，对 MSGGA 的性能进行综合分析。

Li 和 Lim[73]在 2001 年用禁忌模拟退火算法给出了 PDPTW 算例的最优解，但是没有说明其方法的计算时间是算法的平均运行时间还是得出最优解时特定的运行时间。Giselher[42]于 2003 年在 PC 机（Pentium4，2GHz）上用分组编码遗传算法（GGA）测算相同的 PDPTW 算例，给出了最优解，平均最优解和算法平均运行时间。本文将每个算例进行指定次数试验，给出了经过 MSGGA 计算的 100 个客户点 PDPTW 算例的最优解，平均最优解和算法收敛到最优解的平均运行时间。在每次试验中，固定 MSGGA 的种群为 200，迭代次数为 50。

表 4.1 列出了 Li 和 Lim 禁忌算法、GGA 和 MSGGA 三种算法的结果数据。三种算法采用的目标准则是一致的，都是首先考虑优化使用车辆总数，然后考虑行车总距离，可以直接进行结果比较。符号意义如下：case no： 算例标号；td (best)：最优行驶距离；nv(best)：最少车辆总数；ct：最少计算时间；td (avg)：平均最优行驶距离；nv(avg)：平均车辆总数；ct(avg)：平均计算时间，单位时间以秒计算。表 4.1 中，同一算例最优解的不同结果用黑体标出。

对表 4.1 所列三种算法的最优解进行分析。在解决客户点平面位置局部聚集的 Lc1、Lc2 类算例时，三种方法都达到了比较稳定的最优解。Lc104 算例 GGA 只得出 10 辆车的最优解，MSGGA 则求得了 9 辆车的最优解，并且结果优于 Li 算法，这证明本文适应度函数的设置对目标的调整十分有效；Lc203 算例 MSGGA 与 GGA 都没有得到与 Li 算法相同的解，经过试验发现 MSGGA 在这个算例较早收敛；Lc204 算例 MSGGA 与 GGA 得到相同的比 Li 算法更好的解。

客户点平面位置随机分布的 Lr1、Lr2 类算例比 Lc1、Lc2 类算例复杂，这是由问题的离散性决定的，出现了较多的不同结果。Lr101 算例 MSGGA 和 GGA 都没有得到 Li 算法的最优解；Lr109、Lr201 算例 MSGGA 和 GGA 得到相同的较 Li 算法更好的解，Lr110、Lr203、Lr208 算例 MSGGA 与 Li 算法得到相同的最优解，并且优于 GGA；MSGGA 得到了 Lr209 算例优于另外两种算法的更好解；Lr211 算例比较特殊，MSGGA

和 GGA 都没有找到 Li 算法得到的车辆总数更少的最优解，MSGGA 搜索到相同车辆数下比 GGA 更好的解。

客户点平面位置局部聚集局部随机分布的 Lrc1、Lrc2 算例出现了最多的不同结果。Lrc101、Lrc202、Lrc203 算例，MSGGA 所得最优解与 Li 算法相同，并且无论车辆总数还是行车总距离都要优于 GGA；Lrc102、Lrc108、Lrc204、Lrc206、Lrc207 算例 MSGGA 与 GGA 求得相同的最优解，这些最优解车辆总数与 Li 算法结果是相同的，但行车总距离有了较大改进；Lrc107 算例 MSGGA 与 Li 算法最优解相同，仅比 GGA 差 0.01；MSGGA 得到了 Lrc201 算例优于 Li 算法与 GGA 结果的更好解。

经过详细对比，发现 MSGGA 和 GGA 在解决客户点局部聚集的 Lc1、Lc2 以及客户点部分局部聚集的 Lrc1、Lrc2 算例表现了良好的寻优性，在车辆总数相同的情况下，MSGGA 与 GGA 可以找到比 Li 算法更好的路径解。MSGGA 在继承了 GGA 优良性质的同时，又在一些具有特殊性的算例如 Lc104 上表现出禁忌算法的优点，MSGGA 在解决这些算例时，可以搜索到使用车辆数最少的最优解，这是 GGA 所欠缺的。这是由于本文所设计的具有自动调节性能的适应度函数以及加入的三种路径调整策略发挥了重要作用。几乎所有的算例，MSGGA 都能够在有限迭代次数中达到比较稳定的最优解，无论从行驶距离还是车辆总数都达到文献[42]所列 Li 算法与 GGA 求得的最好解，其中 Lc104、Lr209、Lr211 和 Lrc201 算例 MSGGA 得出了更好的结果。

对 MSGGA 和 GGA 的平均最优解进行分析。56 个算例中，用 GGA 测算，有 17 个算例不能以 100%收敛到最优解，其中有 3 个属于 Lc（客户点局部聚集）算例，Lr（客户点随机分布）算例中不能收敛到最优解的有 14 个。用 MSGGA 测算，只有 7 个算例不能收敛，2 个属于 Lrc（客户点部分聚集部分随机分布）算例，5 个属于 Lr 算例。可以看出 MSGGA 在收敛到最优的稳定性上明显优于 GGA，尤其是在客户点随机分布的 Lr 算例的解决上，MSGGA 的性能比 GGA 更加优良。这说明 MSGGA 中启发式插入法的改进以及加入的三种路径调整策略十分有效。虽然计算机平台不同，无法确切比较算法的计算时间，但是从表 4.1 结果看，MSGGA 收敛到最优解运行的时间远远小于另外两种算法。

考察算法的性能除了从理论上进行严格证明外，算法解决问题的有效性也是重要参考标准。本文提出的多策略分组编码遗传算法（MSGGA）在解决 PDPTW 方面与文献[42]介绍的分组编码遗传算法（GGA）比较，在求解最优解的稳定性和收敛性上都有明显提高，具有很好的求解精确性。通过计算通用算例与现有结果进行比较，MSGGA

求解结果和计算时间都优于文中提到的其它算法，说明其在求解 PDPTW 上十分有效。

4.3 算例结果举例

下面图 4.1—图 4.6 给出 100 个客户点的 PDPTW 算例 Lc101、Lc201、Lr101、Lr201、Lrc101 和 Lrc201 的最优路径结果图形作为参考：

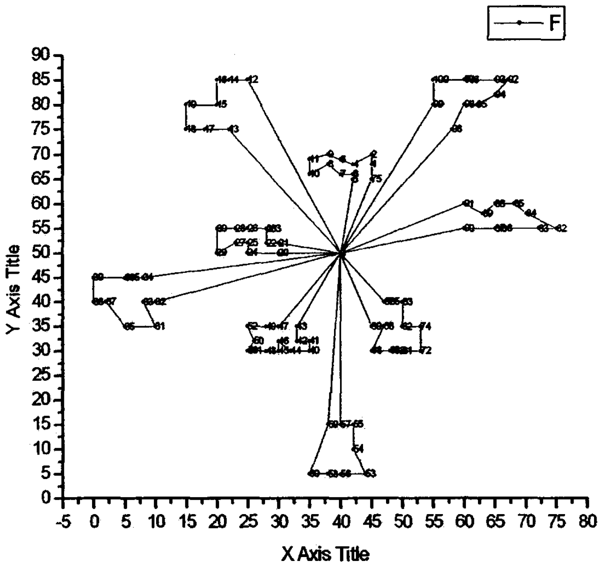

图 4.1 Lc101 算例路径图形

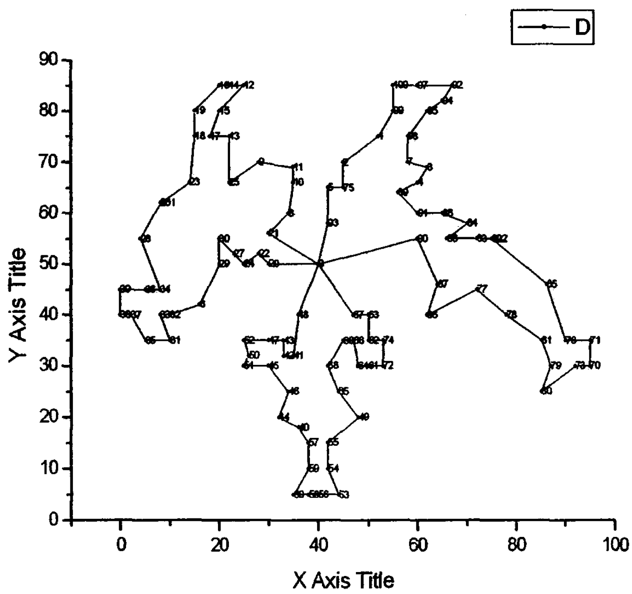

图 4.2 Lc201 算例路径图形

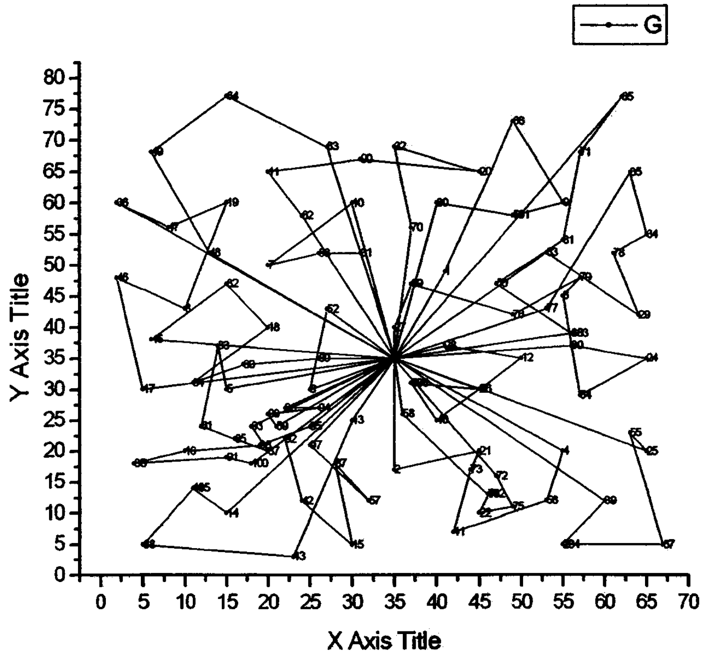

图 4.3 Lr101 算例路径图形

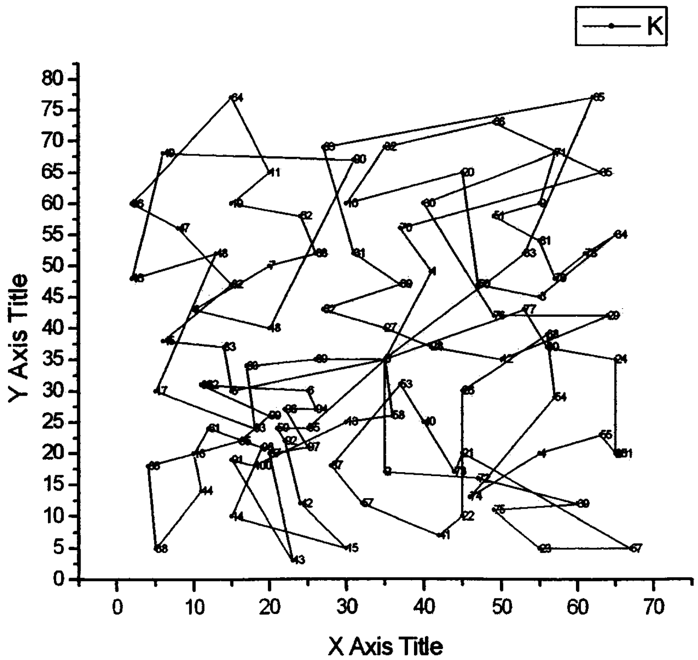

图 4.4 Lr201 算例路径图形

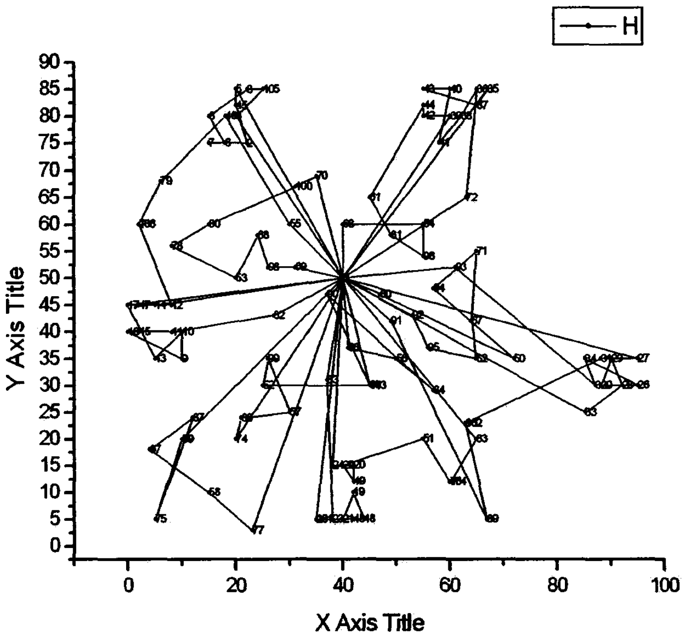

图 4.5 Lrc101 算例路径图形

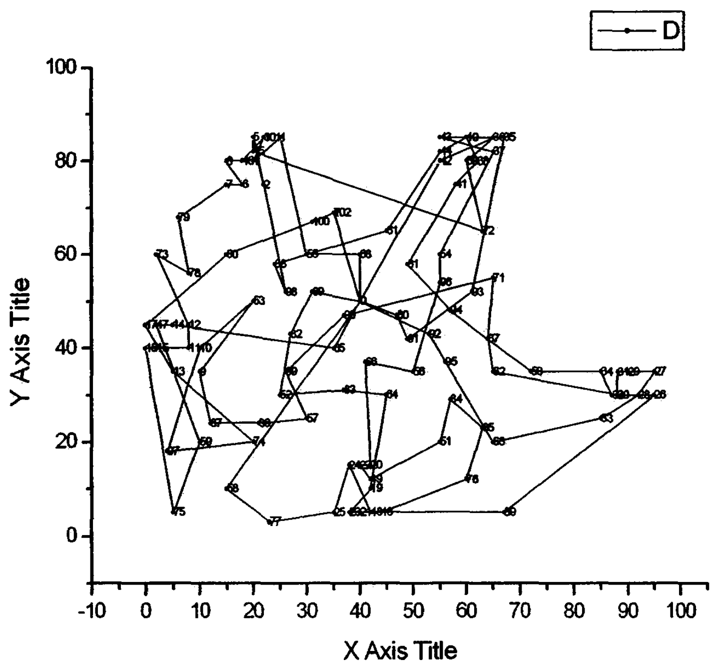

图 4.6 Lrc201 算例路径图形

表 **4.1**　三种算法结果比较

	Li and Lim (2001)			GGA (2003)					MSGGA				
case no	td (best)	nv (best)	ct	td (best)	nv (best)	td (avg)	nv (avg)	ct (avg)	td (best)	nv (best)	td (avg)	nv (avg)	ct (avg)
Lc101	828.94	10	33	828.94	10	828.94	10.00	55.03	828.94	10	828.94	10.00	0.80
Lc102	828.94	10	71	828.94	10	828.94	10.00	68.67	828.94	10	828.94	10.00	3.49
Lc103	827.86	10	191	827.86	10	827.86	10.00	85.33	827.86	10	827.86	10.00	14.66
Lc104	**861.95**	**9**	1254	**818.60**	**10**	818.67	10.00	139.90	**860.01**	**9**	861.56	9.00	54.03
Lc105	828.94	10	47	828.94	10	828.94	10.00	58.33	828.94	10	828.94	10.00	1.35
Lc106	828.94	10	43	828.94	10	828.94	10.00	63.60	828.94	10	828.94	10.00	2.32
Lc107	828.94	10	54	828.94	10	828.94	10.00	65.97	828.94	10	828.94	10.00	3.29
Lc108	826.44	10	82	826.44	10	826.44	10.00	81.00	826.44	10	826.44	10.00	6.84
Lc109	827.82	10	255	827.82	10	827.82	10.00	118.10	827.82	10	827.82	10.00	20.54
Lc201	591.56	3	27	591.56	3	591.56	3.00	59.27	591.56	3	591.56	3.00	0.40
Lc202	591.56	3	94	591.56	3	591.56	3.00	115.50	591.56	3	591.56	3.00	6.45
Lc203	**585.56**	3	145	**591.17**	3	591.17	3.00	191.67	**591.17**	3	591.17	3.00	35.19
Lc204	**591.17**	3	746	**590.60**	3	620.38	3.00	346.13	**590.60**	3	590.60	3.00	149.99
Lc205	588.88	3	190	588.88	3	588.88	3.00	94.07	588.88	3	588.88	3.00	3.53
Lc206	588.49	3	88	588.49	3	588.49	3.00	124.20	588.49	3	588.49	3.00	8.53
Lc207	588.29	3	102	588.29	3	588.35	3.00	137.03	588.29	3	588.29	3.00	12.39
Lc208	588.32	3	178	588.32	3	588.75	3.00	141.93	588.32	3	588.32	3.00	16.24

case no	Li and Lim (2001) td (best)	nv (best)	ct	GGA (2003) td (best)	nv (best)	td (avg)	nv (avg)	ct (avg)	MSGGA td (best)	nv (best)	td (avg)	nv (avg)	ct (avg)
Lr101	**1650.78**	19	87	**1650.80**	19	1650.80	19.00	67.00	**1650.80**	19	1650.80	19.00	0.57
Lr102	1487.57	17	1168	1487.57	17	1488.74	17.00	89.53	1487.57	17	1487.57	17.00	1.65
Lr103	1292.68	13	169	1292.68	13	1292.68	13.00	82.87	1292.68	13	1252.68	13.00	6.61
Lr104	1013.39	9	459	1013.39	9	1013.16	9.43	136.83	1013.39	9	1013.39	9.00	28.18
Lr105	1377.11	14	69	1377.11	14	1377.11	14.00	70.97	1377.11	14	1377.11	14.00	1.10
Lr106	1252.62	12	87	1252.62	12	1252.63	12.00	81.33	1252.62	12	1252.62	12.00	3.74
Lr107	1111.31	10	287	1111.31	10	1112.26	10.03	95.67	1111.31	10	1111.31	10.00	10.54
Lr108	968.97	9	415	968.97	9	969.02	9.00	102.50	968.97	9	968.97	9.00	15.64
Lr109	**1239.96**	11	348	**1208.96**	11	1233.99	12.00	103.80	**1208.96**	11	1226.21	11.60	7.23
Lr110	**1159.35**	**10**	547	**1165.83**	**11**	1176.20	11.13	142.13	**1159.35**	**10**	1168.24	10.60	18.06
Lr111	1108.90	10	179	1108.90	10	1113.60	10.20	105.53	1108.90	10	1108.90	10.00	14.22
Lr112	1003.77	9	638	1003.77	9	1040.34	10.27	168.70	1003.77	9	1048.10	9.80	43.62
Lr201	**1263.84**	4	193	**1253.23**	4	1260.41	4.20	103.43	**1253.23**	4	1253.23	4.00	5.62
Lr202	1197.67	3	885	1197.67	3	1255.78	4.03	234.50	1197.67	3	1218.64	3.00	41.64
Lr203	**949.40**	3	1950	**952.29**	3	962.82	3.00	381.50	**949.40**	3	949.40	3.00	92.39
Lr204	849.05	2	2655	849.05	2	879.01	2.27	486.50	849.05	2	849.05	2.00	210.86
Lr205	1054.02	3	585	1054.02	3	1078.84	3.03	166.37	1054.02	3	1054.02	3.00	34.48
Lr206	931.63	3	747	931.63	3	941..67	3.00	259.03	931.63	3	931.63	3.00	59.53
Lr207	903.05	2	1594	903.05	2	927.58	2.43	418.77	903.05	2	903.05	2.00	171.34
Lr208	**734.85**	2	3572	**736.00**	2	760.95	2.03	531.07	**734.85**	2	734.85	2.00	332.24

	Li and Lim (2001)			GGA (2003)					MSGGA				
case no	td (best)	nv (best)	ct	td (best)	nv (best)	td (avg)	nv (avg)	ct (avg)	td (best)	nv (best)	td (avg)	nv (avg)	ct (avg)
Lr209	**937.05**	3	2773	**932.43**	3	955.96	3.07	236.90	**930.59**	3	930.59	3.00	90.40
Lr210	964.22	3	1482	964.22	3	972.34	3.00	286.13	964.22	3	964.22	3.00	87.87
Lr211	**927.80**	**2**	4204	**888.15**	**3**	897.03	3.07	478.53	**884.29**	**3**	890.18	3.00	320.61
Lrc101	**1708.80**	**14**	119	**1703.21**	**15**	1703.21	15.00	76.07	**1708.80**	**14**	1708.80	14.00	1.41
Lrc102	**1563.55**	**13**	152	**1558.07**	**12**	1558.07	12.00	95.53	**1558.07**	**12**	1558.07	12.00	5.03
Lrc103	1258.74	11	175	1258.74	11	1258.74	11.00	92.80	1258.74	11	1258.74	11.00	10.79
Lrc104	1128.40	10	202	1128.40	10	1128.40	10.00	108.23	1128.40	10	1128.40	10.00	26.90
Lrc105	1637.62	13	179	1637.62	13	1637.62	13.00	88.17	1637.62	13	1637.62	13.00	3.23
Lrc106	**1425.53**	11	459	**1424.73**	11	1424.73	11.00	95.63	**1424.73**	11	1424.73	11.00	4.82
Lrc107	**1230.15**	11	154	**1230.14**	11	1230.14	11.00	97.73	**1230.15**	11	1230.15	11.00	13.04
Lrc108	**1147.97**	10	650	**1147.43**	10	1147.43	10.00	111.67	**1147.43**	10	1147.43	10.40	25.28
Lrc201	**1468.96**	4	266	**1407.21**	4	1407.21	4.00	101.47	**1406.94**	4	1406.94	4.00	8.79
Lrc202	**1374.27**	3	987	**1385.25**	4	1385.25	4.00	161.70	**1374.27**	3	1381.21	3.40	27.28
Lrc203	**1089.07**	3	1605	**1093.89**	4	1093.89	4.00	268.17	**1089.07**	3	1089.07	3.00	94.91
Lrc204	**827.78**	3	3634	**818.66**	3	818.66	3.00	457.30	**818.66**	3	818.66	3.00	196.94
Lrc205	1302.20	4	639	1302.20	4	1302.20	4.00	147.20	1302.20	4	1302.20	4.00	16.61
Lrc206	**1162.91**	3	445	**1159.03**	3	1159.03	3.00	139.59	**1159.03**	3	1159.03	3.00	27.99
Lrc207	**1424.60**	3	607	**1062.05**	3	1062.05	3.00	218.97	**1062.05**	3	1062.05	3.00	68.18
Lrc208	852.76	3	4106	852.76	3	852.76	3.00	322.13	852.76	3	855.49	3.00	248.35

第五章 结论与展望

5.1 结论

本文主要研究运用多策略分组编码遗传算法解决有时间窗装卸问题(PDPTW)。在初步研究运用遗传算法解决简单约束车辆路径问题（VRP）的基础上，完善了 PDPTW 的数学模型，改进分组编码遗传算法[42]用来生成可行解的启发式插入搜索算法，设计新的数据结构，在此基础上加入三种路径调整策略，提出一种多策略分组编码遗传算法。通过计算机仿真实现，对有 100 个客户点的通用 PDPTW 算例进行测算，将结果与文献[42]进行比较， 验证多策略分组编码遗传算法在解决 PDPTW 上具有良好的收敛性和稳定性。本文的主要成果内容如下：

（1）完善了 PDPTW 的数学模型，构造了较为实用的目标函数。利用惩罚函数的思想，设计具有调节能力的适应度函数，使得在多策略分组编码遗传算法的操作中，可以根据问题目标的不同要求，调节求解的侧重点，得到更符合需要的解。

（2）改进了分组编码遗传算法中用来生成可行解的启发式插入搜索算法，使其尽可能避免出现非可行解。改进启发式插入算法具有一定禁忌机制，可以使算法求解在生成可行解的同时达到一定的初步优化。

（3）根据分组编码方案设计了新的数据结构，包含了 PDPTW 问题的各项信息，在算法实现过程中便于产生可行解和计算适应度值。从编程实现的运行的效果来看，此数据结构很有效。

（4）在分组编码遗传算法的基础上提出了单车局部反向搜索、单个需求重排和部分需求重排三种路径调整策略，设计出多策略分组编码遗传算法。

（5）用多策略分组编码遗传算法计算 100 个客户点的 PDPTW 算例，与文献算法进行结果比较，算法缩短了计算时间，其收敛性和稳定性都有明显提高，并得出 4 个算例的更好解，取得了先进的研究成果。

用本文提出的多策略分组编码遗传算法计算文献[73]介绍的 100 个客户点的 PDPTW 通用测试集，并将结果与文献[42]已知最优解进行比较。多策略分组编码遗传算法在计算结果精度和计算时间上都达到了很好的效果，说明算法在解决 PDPTW 上是有效的，具有广阔的发展前景。

5.2 研究展望

本文提出的多策略分组编码遗传算法，在求解 PDPTW 中取得了很好的效果，但是在许多方面仍有待进一步的提高。可以从以下几方面进行继续研究：

（1）研究现实情况更多元化、数学模型更复杂的车辆路径问题。随着物流技术的发展，对车辆路径问题的求解要求越来越高，在现实生活中也会出现具有各种约束条件的装卸问题（Pickup and Delivery Problem，PDP），例如本文未考虑的多个车场多种车型的 PDP 问题和动态或随机 PDP 问题。多数都是现实存在的复杂约束或者不确定性车辆路径问题，目前仍未得到有效解决。应用本文提出的多策略分组编码遗传算法（MSGGA）解决大规模此类问题具有重要实际意义。

（2）检验多策略分组编码遗传算法（MSGGA）在此类组合优化问题上的可推广性。将本文提出的多策略分组编码遗传算法（MSGGA）应用于同类型车辆路径问题以及相关的组合优化问题，将其发展为解决此类问题的一种有效通用算法。

（3）对分组编码遗传算法进行理论分析。从理论基础的角度研究分组编码遗传算法及本文提出的多策略分组编码遗传算法的收敛性等方面的问题，进一步发展遗传算法的理论研究。

综上所述，将多策略分组编码遗传算法应用于解决 PDPTW，在现实生活中有重要意义。随着实际问题的不断出现和算法的继续发展，各种解决方案也会越来越多，有利于车辆路径问题研究的进一步进行，这些问题的解决必然会对物流业产生深刻积极的影响。

参考文献

[1]刘志强．物流配送系统设计[M]．北京：清华大学出版社，2004 ：10－23．

[2]丁立言，张铎．物流企业管理[M]．北京：清华大学出版社，1999 ：12-15．

[3]Dantzig G，Ramser．J．The truck dispatching problem[J]．Management Science，1959，(6):：80-91．

[4]李军，郭耀煌．物流配送车辆优化调度理论与方法[M]．北京：中国物资出版社，2001：11-17．

[5]Bodin L.D，Golden.B.L，Assad.A.A，Ball.M．Routing and Scheduling of Vehicles and Crews：The State Art[J]．Computers & Operations Research，1983，(10)：7-108．

[6]Golden B.L. Introduction to Recent Advances in Vehicle Routing methods[J]. Transportation Planning Models，1984，(12)：36-45．

[7]Christofides N，Mingozzi A，Toth P.Eds．The Vehicle Routing Problem[J]．Combinationrial Optimization，1979，(11)：315-338．

[8]Golden B.L，Assad．A. Vehicle Routing：Methods and Studies[M]．B.V：Elsevier Science Publishers，1988：36-45．

[9]Altinkemer.K，Gavish.B，Parallel Savings Based Heuristic for the Delivery Problem[J]．Operation Research，1991，(16)：456-469．

[10]Laporte.G．The Vehicle Routing problem：An Overview of Exact and Approximate Algorithms[J]．European Journal of Operational Research，1992，(59)：345-358．

[11]Salhi.S，Rand G.K．Incorptating Vehicle Routing into the Vehicle Fleet Composition Problem[J]．European Journal of Operational Research，1993，(59)：323－330．

[12]Lenstra J.K.，Rinnooy K．Complexity of Vehicle Routing and Scheduling Problem[J]．Networks，1981，(11)：221-227．

[13]Karp.RM，Upfal.E，Wigderson.A．Constructing a perfect matching is in random NC[J]．Combinatorica，1986，6(1)：35-48．

[14]Papadimitriou，C.H.等著，组合最优化、算法和复杂性[M]. 北京：清华大学出版社，1988：36-58.

[15]Frederickson G N，Hecht M S，Kim C E．Approximation algorithm for some routing problems[J]．SIAM J Comput，1978，7：178-193．

[16]Savelsgergh,M.W.P．Local search in routing problems with time windows[J}．Operations Research，1985，33(4)：285-305．

[17]Desrosiers.J，Laporte.G，Sauve.M，Soumis.F. Vehicle Routing With Full Load[J]. Computers & Operations Research，1988，15(3)：219-226.

[18]TK.Ralphs，L.Kopman，WR.Pulleyblank，LE.Trotter. A branch and cut algorithm for the vehicle routing problem (preliminary draft).
Available at ftp:/Ibranchandcut.org/pub/reference/vrp.ps.

[19]Christofides，Mingozzia，Tothp. Exact algorithms for the vehicle routing problem，based on spanning the shortest path relaxation[J]. Mathematical Programming，1981，20：255-282.

[20]Fisher.M.L. Optimal solution of vehicle routing problems using minimum k-trees [J]. Operations Research ，1994，42(4)：626-642.

[21]Christofieds.N，S.Eilon. An Algorithm for Vehicle Dispatching Problem[J]，Operational Research Quarterly，1969，(20)：309-318.

[22]Balinski.M，Quandt.R. On An integer program for A delivery problem[J]. Operations Research，1962，(12)：300-304.

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

[24]Laporte.G, Nobert.Y ， Desrocher.M . Optimal routing under capacity and distance restrictions[J]. Operations Research，1985，(33)：1050-1073.

[25]Fisher，M.L. JaiKumar. A generalize assignment heuristic for vehicle routing[J]. Networks 1981：(11) ：109-124.

[26]Clark.G，Wright.J. Scheduling of vehicles from a central depot to a number of delivery points[J]. Operations Research，1964，(12)：568-581.

[27]Paessens.H. The saving algorithm for the vehicle routing problem[J]. European Journal of Operational Research，1988，(34)：336-349.

[28]Gillett.B，Miller.L. A heuristic for the vehicle routing problem[J]. Operations Research，1996，44(2)：286-304.

[29]Christofides.N，Mingozzia，Toth.P.Cominatorial. Optimization[M]. Wiley，Chichester，1979：78-91.

[30]Glover.F. Future paths for integer programming and links to artificial intelligence[J]. Computers and Operations research，1986，(13)：533-549.

[31]Tailllard E. Parallel interative search method for vehicle routing problem[J]. Operations Research

Socitey，1994，45(10)：1156-1167.

[32]Osman，I.H.，Laporte G. Metaheuristics： A Bibliography[J]. Annals of Operations Research ，1996，(63)：513-623.

[33]Teodorovic D，Pavkovic G. A simulated annealing technique approach to the vehicle routing problem in the case of stochastic deman[J]. Transportation Planning and Technology，1992，16(1) ：261-273.

[34]Lawrence S，Mohammad A. Parametric experimentation with a genetic algorithmic configuration for solving the vehicle routing problem[J]. Proceedings—Annual Meeting of the Decision Sciences Institute，1996：488-490.

[35]Nicolas，Barnier. Optimization by hybridization of a genetic algorithm with CSP techniques [J]. Evolutionary Computation Proceedings，1998，4(9)：645-649.

[36]张涛，张玥杰，王梦光．不确定车辆数的车辆路径问题模型和混合算法．系统工程理论方法应用，2002，11(2)：121-124.

[37]Wark P，Holt J. A repeated matching heuristic for the vehicle routing problem[J]. Operations Research Socitey，1994，45(10)：1156-1167.

[38]Bernd Bullnheimer，Richard F. Hartl，Christine Strauss An improved ant system algorithm for the vehicle routing problem[J]. Annals of operations research，1999，(89)：319-328.

[39]Ismail Ellabib，Otman A.Basir，Paul Calamai. An experimental study of a simple and colony system for the vehicle routing problem with time windows[J]. Computer Science，2002：53-64.

[40]Savelsbergh M.W.P，Sol M. The General Pickup and Delivery Problem[J]. Transportation Science，1991，(29)：17-29.

[41]Dumas Y，Desrosiers J，Soumis F. The Pickup and Delivery Problem with Time Windows[J]. European Journal of Operational Research，1991，(54)：7-22.

[42]Giselher，Pankratz. A Grouping Genetic Algorithm for the Pickup and Delivery Problem with Time Windows[J]. OR Spectrum，2005，(27)：21-41.

[43]J. Holland. Adaptation in natural and artificial systems[M]. The University of Michigan Press，1975，MIT Press，1992：45-78.

[44]王凌．智能优化算法及其应用．北京：清华大学出版，2001：36-37.

[45]Falkenaue E. Genetic Algorithms and Grouping Problems[M]. Wiley，Chichester，1998：53-67.

[46]周明，孙树栋编著．遗传算法原理及应用[M]．北京：国防工业出版社，1999：36-39.

[47]王正志，薄涛．进化计算[M]．长沙：国防科技大学出版社，2000：28-39.

[47]王正志，薄涛．进化计算[M]．长沙：国防科技大学出版社，2000：28-39.

[48]张文修，梁怡．遗传算法的数学基础[M]．西安：西安交通大学，2000：14-35.

[49]韩炜，廖振鹏．关于遗传算法收敛性的注记[J]．地震工程与工程震动，1999，(4)：13-16.

[50]熊伟清，赵杰煜.遗传算法的早熟收敛[J]．宁波大学，2001，(6)：23-27.

[51]王莉．遗传算法的收敛性统一判据[J]．自动化技术与应用，2004，23(6)：16-19.

[52]Davigc，Stephenm．Conver gencecriterion for genetic algorithms[J]．Society for Industrial and Applied Mathematics，2000，30(1)：269-282.

[53]陈湘州，黎志明，刘祖润．一种改进的整数编码遗传算法在车辆路径优化问题中的应用[J]．南方冶金学院学报，2004，(01)：36-60.

[54]邹彤，李宁，孙德宝．不确定车辆数有时间窗的车辆路径问题的遗传算法[J]．系统工程理论与实践，2004，(06)：0134-05.

[55]张丽萍，柴跃廷，曹瑞．有时间窗的车辆路径问题的改进遗传算法[J]．计算机集成制造系统—CIMS，2002，(06)：0451-04.

[56]李军．有时间窗车辆路线安排问题的启发式算法[J]．系统工程，1996，14(5)：45-50.

[57]Chuin Lau，Zhe Liang．Pickup and Delivery with Time Windows：Algorithms and Test Case[J]．International Journal on Artifical Intelligence Tools，2002：445-472.

[58]Psaraftis HN．A Dynamic Programming Solution to the Single Vehicle Many-to-Many Immediate Request Dial-a-Ride Problem[J]．Transportation Science，1980，(14)：130-154.

[59]Desrosiers J，Dumas Y，Soumis F．A Dynamic Programming Solution of the Large-Scale Single-Vehicle Dial-a-Ride Problem with Time Windows[J]．The American Journal of Mathematical and Management Sciences，1986，(6)：301-325.

[60]Bodin L，Sexton T．The Multi-Vehicle Subscriber Dial-a-Ride Problem[J]．TIMS Studies in the Management Sciences，1986，(26)：73-86.

[61]Desrosiers J，Dumas Y，Soumis F．The Multiple Vehicle Dial-a-Ride Problem[J]．Public Transport Springer Berlin，1988：15-27.

[62]Ioachim I，Desrosiers J，Dumas Y，Solomon M，Villeneuve D．A Request Clustering Algorithm for Door-to-Door Handicapped Transportation[J]．Transportation Science research Part B 1995，29(20)：243-257.

[63]Solomon M. Algorithms for the Vehicle Routing and Scheduling Problem with Time Windows Constrains[J]．Operations Research，1987，35(2)：254-265.

[64]Jaw JJ, Odoni AR, Psaraftis HN, Wilson NHM. A Heuristic Algorithm for the Multi Vehicle Advance Request Dial-A-Ride Problem with Time Windows[J]. Transportation Research Par B 1986, (20): 243-257.

[65]Madsen OBG, Ravn HF, Rygaard JM. A Heuristic Algorithm for a Dial-a-Ride Problem with Time Windows: Multiple Capacities and Multiple Objectives[J]. Annals of Operations Research, 1995(60): 193-208.

[66]Van der Bruggen, Lenstra JK, Schuur. Variable Depth Search for the Single Vehicle Pickup and Delivery Problem with Time Windows[J]. Transportation Science, 1993(27): 298-311.

[67]Lin S, Kernighan B. An Effective Heuristic Algorithm for the Traveling Salesman Problem[J]. Operations Research, 1973, (21): 498-516.

[68]Psaraftis HN. k-Interchange Procedures for Local Search in a Precedence-Constrained Routing Problem[J]. European Journal of Operational Research, 1983, (13): 391-402.

[69]Toth P, Vigo D. Fast Local Search Algorithms for the Handicapped Persons Transportation Problem[J]. Kluwer, Norwell, 1996: 677-690.

[70]Gendreau M, Guertin F, Potvin JY, Seguin R. Neighborhood Search Heuristics for a Dynamic Vehicle Dispatching Problem with Pick-ups and Deliveries[J]. Technical Report CRT-98-10, 1998: 37-58.

[71]Nanry WP, Barnes JW. Solving the Pickup and Delivery Problem with Time Windows using Reactive Tabu Search[J]. Transportation Research Part B, 2000, (34): 107-121.

[72]Lau HC, Liang Z. Pickup and Delivery with Time Windows, Algorithms and Test Case Generation[J]. Computer Society, 2001, (13): 333-340.

[73]Li H, Lim A. A Metaheuristic for Solving the Pickup and Delivery Problem with Time Windows[J]. Computer Society, 2001(13): 160-167.

[74]Wan-Rong Jih, JaneYung-Jen Hsu. A Family Competition Genetic Algorithm for the Pickup and Delivery Problems with Time Windows[J]. Bulletin of the College of Engineering, 2004, (90): 89–98.

[75]贾永基，谷寒雨，席裕庚．一类货运车辆调度问题的混合禁忌搜索算法[J]．信息与控制，2004，33(6)：724-728.

[76]蓝伯雄，张跃．求解带时间窗的装一卸载问题的概率式禁忌搜索算法[J]．中国管理科学，2004，12(2)：66-72.

[77]Potter T, Bossomaier T. Solving Vehicle Routing Problem with Genetic Algorithms. Computer

Society，Los Alamitos，California，1995：788-793.

[78]Jih，W.-R，Hsu，Y.-J. Dynamic Vehicle Routing Using Hybrid Genetic Algorithms[J]. Computer Society，Los Alamitos/Califonia，1999：453-458.

[79]Schonberger.J，Kopfer.H，Mattfeld.DC. A Combined Approach to Solve the Pickup and Delivery Selection Problem[J]. Operations Research Proceedings，2002：150-155.

[80]Jung S，Haghani A. A Genetic Algorithm for Pick-up and Delivery Problem with Time Windows[J]. Transportation Research Board，2000：1-7.

[81]Kopfer.H，Pankratz.G，Erkens.E. Die Entwicklung eines hybriden Genetischen Algorithmus fur das Tourenplanungsproblem[J]. Operational Research，Spektrum，1994，(16)：21-32.

[82]Blanton.JL，Wainwright RL. Multiple Vehicle Routing with Time and Capacity Constraints using Genetic Algorithms[J]. Forrest S(ed) Proceedings of the Fifth International Conference on Genetic Algorithms，1993：452-459.

[83]Kjetil，Fagerholt. Ship scheduling with soft time windows：An optimization based approach[J]. European Joural of Operational Research，2001，(131)：559-571.

致 谢

首先深深感谢导师丁根宏副教授在学习、工作和生活上给予我的悉心指导、热情关心和无私帮助。丁老师渊博的知识、精益求精的治学态度、敏锐的洞察力、豁达的胸怀、对学术的严格要求和无私奉献、乐于助人的精神等，都对我有着潜移默化的影响。在这里谨向我的导师致以衷心的感谢和崇高的敬意！

感谢河海大学理学院的各位老师在学习上和生活上的关怀。各位老师的学识和勤勉的工作作风将是我努力的目标。

感谢 2003 级理学院硕士班的所有同学，近三年的研究生生活，有了他们的陪伴和鼓励，让我的研究生生活更加丰富多彩。非常感谢他们陪我渡过了人生一段美丽的时光。

感谢参加论文评审和答辩的各位专家、教授。对他们付出的辛勤劳动表示诚挚的谢意！

最后，把本文献给我的家人和亲戚朋友，感谢他们多年来在精神上和物质上的无私支持与鼓励。父母无私的关爱是我勇往直前的动力，在以后的工作和学习中我将继续努力奋斗。

商丽媛

2006 年 4 月