

## 摘 要

近年来国内物流产业正在快速发展，物流市场在壮大的同时，竞争日趋激烈。随着物流业向全球化、信息化及一体化发展，配送在整个物流系统中的作用变得越来越重要。顾客对物流服务质量要求的提高，以及行业竞争使物流企业格外注重在物流配送作业上的科学与高效管理。

配送是物流系统中一个直接与消费者相连的重要环节，是货物从物流节点送达收货人的过程。运输系统是配送系统中最重要的一个子系统，运输费用占整体物流费用中的 50%，所以降低物流成本首先要从降低物流配送的运输成本开始。其中运输线路是否合理直接影响到配送速度、成本和效益。选取合适的运输路线，可以加快对客户需求的响应速度，提高服务质量，增强客户对物流系统的满意度，降低服务上的运营成本。本文针对配送中的核心问题——车辆路径问题（Vehicle Routing Problem, VRP），采用遗传算法进行了深入的研究。

遗传算法是模拟自然界生物进化过程与机制求解问题的一类自组织与自适应的人工智能技术。它使用群体搜索技术，通过对当前群体施加选择、交叉、变异等一系列遗传操作，从而产生新一代的群体，并逐步使群体进化到包含或接近最优解的状态。由于其具有思想简单、易于实现、应用效果明显等优点而被众多应用领域所接受，并在组合优化、模式识别、机器学习、人工生命、管理决策等领域得到了广泛的应用。遗传算法给我们呈现出的是一种通用的算法框架，该框架不依赖于问题的种类。

本文在认真分析国内外对 VRP 研究的基础上，将遗传算法进行了一系列的改进，如采用整数编码、引入罚函数对约束进行处理、对交叉算子和变异算子进行改进并对由此操作产生的不合法个体进行相关操作，直到得到合法的子代个体才能进行下一次迭代。通过 MATLAB 实现该算法，并通过实例证明了该算法是求解 VRP 的一个较好方案。这对物流配送企业实现配送路径优化、降低成本和提高物流经营管理水平、更快的响应顾客，最终增加企业的竞争力具有重要的参考价值。

关键字：配送；车辆路径问题（VRP）；遗传算法

## Abstract

Logistics industry has developed quickly in recent years. With the logistics market becoming stronger; the competition has become more and more drastic. Distribution's effect in the whole logistics system is more important with the development of logistics industry to globalization, digitalization and integration. The high quality of logistics service required by customers and competition in the logistics service make logistics companies pay more and more attention to scientific and efficient logistics distribution management.

Distribution is an important link in logistics, which is joined to consumers directly. Transportation system is the most important subsystem in distribution system, and the cost of transportation occupies 50% of that in whole logistics. So, reducing the cost of logistics begins with reducing that of transportation of logistics distribution. In the problem, whether the transportation route is rational will affect the speed, cost and benefit of distribution. Selecting rational vehicles' route can help to achieve quicker response speed to customers, improve the quality of service, increase customers' satisfaction to logistics system and reduce operational cost of service merchants. This paper has studied the core problem of the distribution —VRP by using genetic algorithm(GA).

Genetic algorithm is an artificial intelligence technology of self-organization and adaptation, which imitates natural organisms' evolutionary process and mechanism to solve problems. It produces new colony and makes it achieve or approach optimization result by using colony search technique, choosing, crossover and mutation. Because of the algorithm's simple elements, achieving easily and obvious application effect, it is widely applied in combination optimization, mode identification, machinery learning, artificial being, management decision-making and so on. Genetic algorithm applies a frame of a kind of currency algorithm, and this frame is foreign to the varieties of problems.

This paper puts forward an improved genetic algorithm on the base of analysing

the research on VRP in China and abroad. The improved genetic algorithm uses integer coding, penalty function to solve constraints, improves crossover operators and mutation operators and uses some operations to deal with illegal individuals until they are legal. At last we realizes the improved genetic algorithm by MATLAB. The algorithm given in this paper can obtain an optimized solution effectively and has been proved to be a good scheme to solve VRP. So, it has a high value to logistic enterprises to decrease the cost, to improve the level of logistics management, to response customers more quickly and to improve the rivalship power of the enterprise.

**Key Words:** distribution, vehicle routing problem(VRP), genetic algorithm(GA)

## 第1章 绪论

### 1.1 选题背景和意义

物流活动的历史很长，但对物流研究的历史却很短。对于物流尚没有统一的定义。但普遍认为物流有广义和狭义之分。狭义的物流，比较常用的是美国物流管理协会的定义：“物流是为满足消费者而进行从起点到终点间的原材料、中间过程库存、最后产品及相关信息有效流动和储存计划、实施和控制管理过程。”也就是说，物流是指与采购、供应、生产、销售等企业经营活动相对应的将信息传递、运输、配送、库存控制、仓储、装卸搬运以及包装等物流活动综合起来的一种系统化集成式管理。它的目的是提高企业的效益（销售额的提高和利益的扩大），它的任务是以尽可能低的成本为顾客做出最好的服务，即通过时间（快速送达）、物流指令（优良的运送、无差错运送）、备货（所需要的商品及其数量）、库存（低库存）、信息（在库、断货信息、运送中的信息、送达信息）等重要资源的控制，提高物流服务的品质，并降低从原材料供应——商品市场——产品向最终顾客过渡的整个过程中的物流成本，来实现企业的高收益。

广义的物流是指从企业的经营战略角度出发，将物流过程与客户要求、企业内部生产过程以及供应商的制造资源整合在一起，形成一种影响力量，从而达到快速反应市场、降低风险成本、实现高效益、赢得竞争优势的一种管理过程，即现在人们通常所说的供应链管理。图 1-1 描述了供应链物流过程。

物流（Logistics）是供应链中最重要的组成部分，是商品从生产者经过诸流通环节最终到达消费者手中的过程。物流业则是专门从事物流活动的行业，从企业销售成本和商品价格组成角度考察，物流业蕴藏着巨大的商机。现代物流已被公认是企业在降低物资消耗、提高劳动生产率以外的“第三利润源泉”，也是企业降低经营成本，提高产品竞争力的重要途径，因而受到国内外各行业的极大重视，并得到较快发展。

配送是指企业或物流代理企业在确认顾客的订购信息之后将商品从各配送中心（或零售点）送到消费者（顾客）手中的过程。从概念范畴上讲，物流包



含配送。配送是物流系统中的一个重要环节，由于它直接与消费者相连，是在集货、配货的基础上，完全按照用户的要求，包括种类搭配、数量、时间等方面的要求所进行的运送，是“配”和“送”的有机结合形式，因而地位十分突出。配送的一般流程如图 1-2 所示。

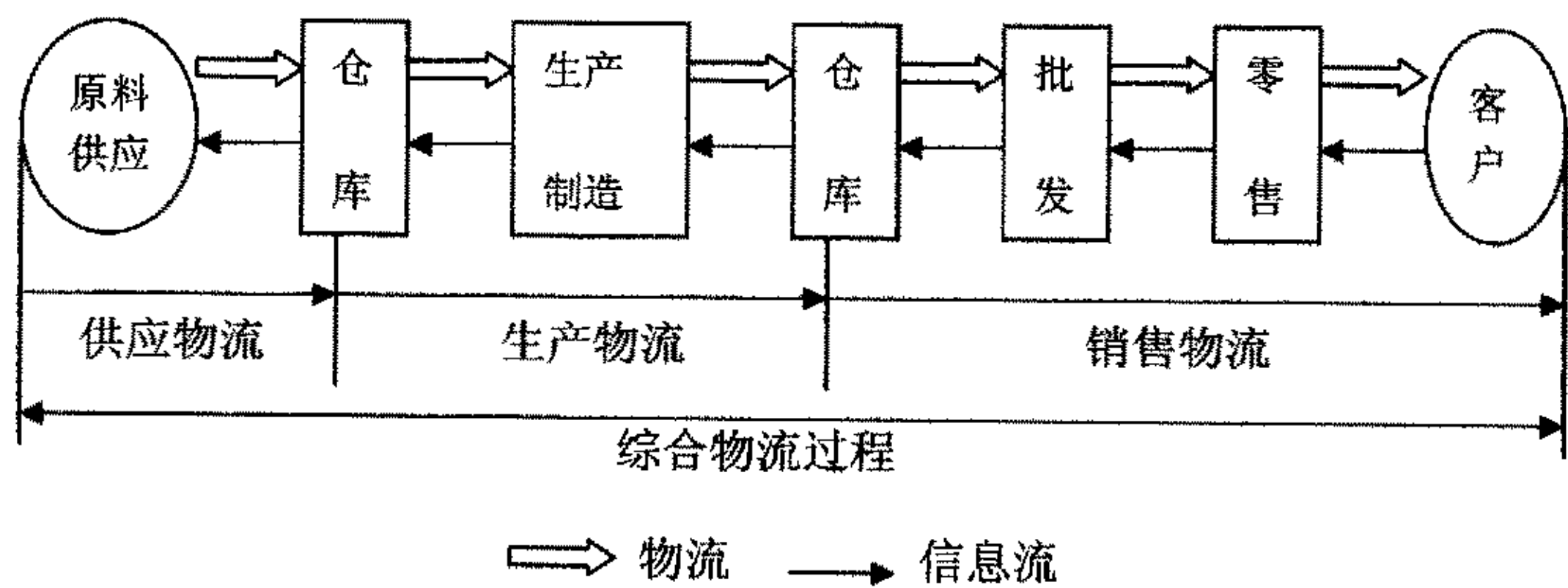


图 1-1 供应链物流过程图

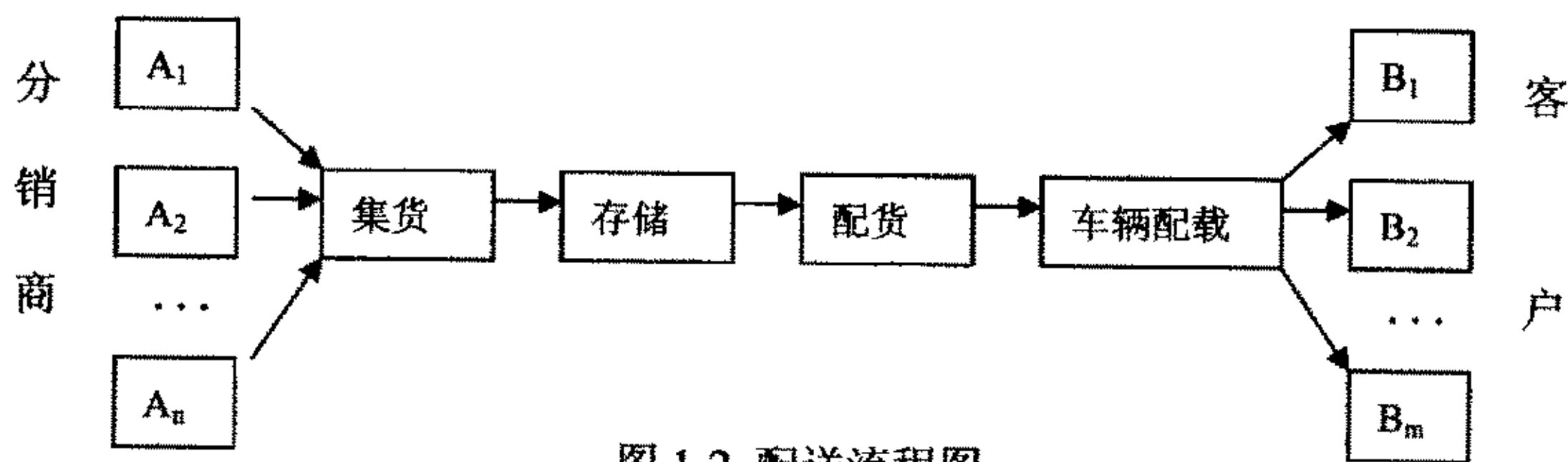


图 1-2 配送流程图

随着电子商务的发展以及新的物流配送模式的出现，存储已不是必然的环节。因此，配送工作主要包括以下几个部分：

- （1）集货作业：从生产工厂进货、并集结的过程。
- （2）配货作业：即货物的分拣作业，根据各用户的不同要求，在配送中心将所需要的货物挑选出来的过程。
- （3）配装作业：由于配装作业本身的特点，配装工作所需车辆一般为汽车，由于配送货物的质量和体积的差异，在配送货物时要考虑车辆的载重和容积，为使车辆的载重和容积得到充分利用，还要考虑一趟多送几户的问题。
- （4）送货作业：即按照预先确定的配送线路和时刻表将货物送到用户的手中。其中，配送线路合理与否对配送速度、成本、客户满意度影响很大，特别

是多用户配送线路的确定更为复杂。采用科学合理的方法来确定配送路线，是配送活动中非常重要的一项工作。

运输问题是物流决策中的关键问题。一般来说，除去产品中的采购成本外，运输成本比任何其他物流活动的成本所占的比例都要高。尽管运输的形式有很多种，但其中最重要的不外乎运输方式的选择，车辆调度与规划等内容。本文将针对其中的车辆路径问题作较深入地探讨。

车辆路径问题（Vehicle Routing Problem, VRP）也有学者称之为车辆调度问题（Vehicle Scheduling Problem, VSP）。一般认为，不考虑时间要求，仅根据空间位置安排线路时称为车辆路径问题，即 VRP；考虑时间要求，安排线路时称车辆调度问题，即 VSP。对 VRP 和 VSP，也有学者不加区分，只是加上具体约束定语，例如，将有时间要求的车辆调度问题称为 Vehicle Routing Problem with Time Window (VRPTW)。由于国内外文献习惯采用 VRP 表述车辆路径与调度问题，本文也一律称之为 VRP。VRP 涉及面广，需要考虑的因素较多。配送中心需按不同客户不同的订货要求进行组织配送，其主要内容是指根据确定配送货物量分配车辆和选择优化路线。由于从事物流配送的车辆货运工作尤其是从事城市配送的车辆货运工作条件复杂，不仅货运点多、货物种类繁多、道路网复杂，而且运输服务地区内运输网点分布也不均匀，同时，很多客户还对配送需求提出一些约束条件的要求，因此应用现代数学方法及计算机快速求解优化调度方案是国内外专家普遍探讨的重要课题。

本文本着为从事物流配送的企业提供智能化决策支持的目标出发，对物流配送及其车辆路径优化与调度问题进行分析，提出实用性较强的配送车辆路径问题的改进遗传算法。本文的研究一方面丰富了遗传算法在组合优化中的应用，另一方面为继续深入研究 VRP 及其算法实现打下基础，具有较大的理论意义和实践意义。

## 1.2 VRP 国内外研究现状

VRP 最早是由 Dantzig 和 Ramser<sup>[1]</sup>于 1959 年提出的，提出后很快引起运筹学、应用数学、组合数学、图论和网络分析、物流科技、计算机应用学科专家与运输计划制定者和管理者的极大重视，成为运筹学与组合优化领域的前沿和研究热点问题。各学科的专家对该问题进行了大量的理论研究及试验分析，取

得了很大进展。

VRP 一般定义为：对一系列装货点和（或）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，达到一定的目标（如路程最短、费用最少、使用车辆数量尽可能少等）<sup>[2]</sup>。

国外对 VRP 作了大量而深入的研究，例如早在 1962 年，Balinski 等人首先提出了 VRP 的集分割<sup>[3]</sup>，直接考虑可行解集合，在此基础上进行优化，建立了最简单的 VRP 模型。1971 年，Eilon 等人<sup>[4]</sup>提出将动态规划法用于固定车辆数的 VRP，通过递归方法求解。其后，Christofides 提出了状态空间松弛，极大的减少了状态空间。1974 年，Wren, Gillett 等人<sup>[5]</sup>提出了 Sweep 算法（扫描法）。1981 年，Christofides 等人提出了 k 度中心树和相关算法<sup>[6]</sup>，对固定车辆数 m 的 m-TSP 进行了 k 度中心树松弛。后来，M.L.Fisher 对这种方法做了进一步改进<sup>[7]</sup>，可求解有 134 个客户的 VRP。1991 年，Gendreau 等人<sup>[8]</sup>将禁忌搜索方法应用于 VRP。该算法求解过程中的邻域，是通过 GEM 过程得到的。它针对 VRP 比较好的启发式算法，可以成功地应用于许多经典的 VRP。其后 E.Taillard 等人<sup>[9]</sup>通过按角度和路径重心对原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，实现了对问题求解的并行化。1996 年，J.Lawrence<sup>[10]</sup>将遗传算法用于 VRP 的研究，并可有效求解带时间窗口的 VRP。鉴于传统的遗传算法是个大范围、粗粒度的寻优算法，Barnier<sup>[11]</sup>将其与约束满足问题（CSP）的技术相结合，通过遗传算法来处理 CSP 参数的子域（基因的适应度通过 CSP 解的计算得到），从而减小搜索空间，降低 CSP 问题目标函数和遗传算法的复杂度。

在我国，遗传算法、启发式算法以及一些混合算法被学者们广泛的利用。张涛等人<sup>[12]</sup>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。这类算法目前已可求解较大规模的问题（199 个客户）。肖鹏等人<sup>[13]</sup>通过构造 VRP 的染色体表达，采用基因换位算子进行染色体重组，实现了新颖的单亲遗传算法。李嘉等人<sup>[14]</sup>通过引入“车队模式”，提出了混合车队 VRP 求解框架，设计了基于遗传算法和禁忌搜索启发式的混合算法。方霞等人<sup>[15]</sup>利用免疫算法的全局搜索能力和收敛性，将新型的启发式算法用于解决 VRP。纪寿文等人<sup>[16]</sup>根据深圳市科技园的实际路网图，采用神经网络的方法对运输车辆优化调度进行了试验研究。王正彬等人<sup>[17]</sup>在分析 VRP 现有启发式算法的基础上，建立了考虑线路安排的物流配送方案模型，并提到出了求

解该问题的一种搜索算法。陈湘洲等人<sup>[18]</sup>引入一种进化逆转算子，改进了遗传算法求解 VRP 时的局部搜索能力。此外，崔雪丽、马良和范炳全等人<sup>[19]</sup>基于近些年出现的新型智能优化思想：人工蚂蚁系统，给出了一种可快速求解 VRP 的蚂蚁搜索算法。通过定义基本的人工蚂蚁状态转移概率，并结合局部搜索策略，用迭代次数控制算法的运行时间，从而使该方法具有使用意义和可操作性。经一系列数据测试和验证，并与若干已有的经典算法相比较，获得了较好的结果。章兢等人<sup>[20]</sup>构造了一种免疫克隆算法来求解 VRP，并在算法中引入了克隆选择、克隆删除、受体编辑、体细胞高频变异、抗体循环补充等思想。仿真计算结果表明，免疫克隆算法能快速收敛于全局最优解，克服了遗传算法中易陷入局部最优解和收敛速度慢的缺点，可有效地解决 VRP。杨善林等人<sup>[21]</sup>提出了一种基于蚁群优化（ACO）的混合算法来解决 VRP。首先提出一种 ACO 算子，然后加入局部搜索机制并使用基于问题的特定启发信息——节约量来改进算法。魏俊华等<sup>[22]</sup>提出了一种分段编码方法，编码中的各段表示相应车辆路径的需求城市集合，以非完全连通配送网络为研究对象，基于分段遗传编码，构造了车辆路径问题的遗传算法，通过对编码各段超载、包容等的判断，得出一个关于需求城市的最优化分。最后利用改进的迪杰斯特拉算法，根据最优化分中的需求城市，采取最优的配送路径集。计算实例证明了该方法的有效性。尹小峰等<sup>[23]</sup>针对了蚁群算法存在的过早收敛问题，引入节省量以及车辆载重利用率两种启发式信息对蚁群算法加以改进，并加入 2-opt 方法对问题求解进行局部优化，计算机仿真结果表明，这种混合蚁群算法对求解车辆路径问题有较好的改进效果。

总之，在我国国内，有关车辆路径问题的研究是在 20 世纪 90 年代以后才逐渐兴起的，比国外相对落后。随着顾客需求的变化运输车辆的调度显得日益重要，近年来，我国理论界逐渐开始关注车辆路径问题的研究，并已取得初步成果，总体来说，目前我国对车辆路径问题的理论研究仍相对薄弱，有待于进一步研究。

### 1.3 论文的主要工作

在前人关于物流配送中的车辆路径问题、遗传算法、组合优化等问题的求解方面所做的大量工作的基础上，论文在遗传算法改进方面并将其应用于 VRP 中进行了探索和尝试。本文主要工作如下：



(1) 根据物流配送的特点和要求,提出了对物流配送系统的规划与设计方  
案。

(2) 在考虑 VRP 的目标及其各种约束条件的基础上建立了数学模型;

(3) 针对基本遗传算法在应用中遇到的局部搜索能力差、计算量大、早熟  
收敛等问题,根据 VRP 实际问题的具体情况,通过对基本遗传算法进行改进,  
提出了相应的改进遗传算法。

(4) 将改进遗传算法应用于 VRP 求解,并采用 MATLAB 完成了对其算法  
的实现。通过具体数据计算分析,验证了改进的遗传算法在 VRP 应用中的可行  
性。

## 第2章 物流配送概述

VRP 研究主要是针对物流配送过程而言的,因此我们有必要了解物流配送的发展模式及目前发展中存在的一些问题。如果我们能迅速的寻求 VRP 的最优解,提高物流配送公司对客户的响应速度,某些存在的问题也就能迎刃而解。

### 2.1 物流配送模式

物流配送的模式有以下几种:

#### (1) 集中配送模式

在长期的流通实践中,很多从事送货活动的企业意识到,在货物运距较远、顾客较多且需求日趋复杂的情况下,不经过物流基地中转,直接从工厂或仓库进货,并将装好的货物送达客户的直达配送模式主要适用于大批量的配送或某些生产资料的配送。对大多数物流产品来说,直达配送有时候会浪费运力,增加物流成本。为了更有效的组织物流活动,许多企业纷纷在流通枢纽处设置配送中心,并以配送中心为基地集中进行送货、集中仓库、集中配送。集中配送模式的优势主要表现在,通过配送中心这种高效的物流环节来减少迂回、相向、重复、倒流等不合理运输现象,降低用户平均库存水平,控制总体物流费用,全面提高服务质量等多个方面。

#### (2) 即时配送模式

即时配送是根据用户提出的时间要求、供货数量和品种要求及时地进行配送的形式,即时配送可以满足用户(特别是生产企业)的急需,它是一种灵活的配送方式。对于配送企业来说,实施即时配送必须有较强的组织能力和应变能力;必须熟悉服务对象的情况。由于即时配送完全按照用户的要求运行,客观上能促使需求者压缩自己的库存,使其货物的“经常储备”趋近于零。

#### (3) 计划配送模式

发达国家在物流配送发展之初,过度强调完全按照顾客要求办事,而忽略了顾客要求对企业经营效率的影响,导致配送量急剧增加。与此相关,也带来运输效率下降、成本增加,配送服务水平日趋降低(如速度慢、准确性差)的

不良后果。物流企业为了兼顾用户满意水平和配送服务效率，在市场竞争中处于优势地位，在实践中推行“路线发送”和“时间表发送”等计划配送制度。即按照地区和配送货物的数量将配送时间间隔和配送具体线路。计划配送，一方面能够使每日的配送量相对稳定，从而有利于减少配送活动的波动；另一方面又可以提高配送设备（如运输工具）的利用率，避免浪费。这种配送方式一般对配送企业的要求比较严格，作业难度也较大。由于这种形式的配送计划性较强，准确度高，因此，一般只适用于在生产稳定、产品批量较大的用户中推行。

#### （4）共同配送模式

共同配送最早产生于日本。20 世纪 60 年代中期随着日本经济振兴和产品产量及消费量的日益扩大，交通运输量也在迅猛增加。当时由于道路拥挤，交通混乱，严重地困扰了配送货物的顺利开展。特别是在中小企业独立配送的形式之下，配送效率很难提高。面对这种现实，企业迫切希望联合行动，共同组织配送活动。

共同配送，即由若干个配送企业联合起来，对某一地区的用户进行配送服务的物流形式。它是在核心组织（配送中心）的统一计划、统一调度下展开的。由于共同配送是一种协作性的配送活动，因而有利于充分发挥各配送企业的整体优势，便于合理调配、调度运输工具和综合利用物流设施。对于参加协作的配送企业来说，不仅可以提高配送的服务水平，而且还可以借此扩大销售渠道和开展联合经营；对于用户来说，可以满足生产经营活动的整体需要，尤其有利于保证重点建设项目的需要。

共同配送是物流共同化发展的产物，促进了物流活动分工与协作，推动了物流结构的调整和物流资源的优化配置。正因为如此，许多国家的政府都积极向企业推广这种配送组织模式。目前有两种新的共同配送运作方式比较引人注目，一是多家用户联合设置接货点和货物处置场地，集中人力、物力开展配送；二是数个配送企业交叉利用他方的配送中心和机械设备进行配送。据 1996 年日本连锁经营协会的物流调查：大多数零售业均采用共同配送方式，共同配送率为 55.4%，其中的 41.4%由配送中心或物流中心进行配送。

#### （5）一体化配送模式

一体化配送模式产生于 20 世纪 90 年代的日本。目前在供应链管理中发挥着重要的作用。一体化配送也称“一揽子物流”，它是将物流与信息流紧密结合

在一起的高水平配送管理模式。这里“一揽子”有两层含义：一是“货架一揽子”，该模式以某一货架群作为整体对象，不问进货地点和形态，将全部商品备齐后集中上货；二是“业务一揽子”，该模式不仅包揽进货业务，而且还要求发货的商品完全不出现差错地以陈列的方式进入店铺。一体化配送以店铺作为服务网点，利用条形码（BC）、销售时点（POS）、电子数据交换（EDI）等信息技术，将一次性分品种进货和定时、定量配送等作业组合在一起，是全部进货业务的一揽子接收系统和进货体制，是零售业最为适宜的配送模式。

## 2.2 我国物流配送的现状与发展趋势

### 2.2.1 我国物流配送的现状

我国整个储运配送系统还不完善，存在着一系列问题，这些问题也影响着整个物流行业发展。

#### （1）物流业配送系统基础设施薄弱

物流配送系统整体功能落后，表现在①交通路径和运行设施满足不了要求。虽然近年来我国各种运输方式都得到发展，以公路为例，到2003年底，我国公路总里程已达到185万公里，在铁路，水运，航空及管道运输等方面也有较大的改善，但这些仍不能满足运输的要求。②交通装备水平落后，高效专用运输车辆少，国内区域运输，主要还是以汽车为主，而汽车中以中型油车运输为多。这种车消耗大，效率低，这就给高效运输带来困难。③仓库存储设施落后，第三方物流企业拥有的仓库70%是普通平房仓库，现代化立体自动化仓储设施比例极低，具有冷藏、保鲜、气调功能的仓库更少。④物流运输系统中使用的辅助装备标准化差，运输中使用的集装箱、托盘、货架等标准化程度低，有的利用欧美标准，有的采取日本标准，这就使物流储运配送各环节对接不便，影响了物流作业效率。

#### （2）我国物流配送规模和服务对象的规模都较小

我国第三方物流市场发展落后，主要表现了第三方物流市场数量、经营规模小，市场份额及服务功能少。大多数物流企业只是按用户要求，从事单一功能运输与仓储，很少能提供物流策略，深入到整个物流环节链的全过程管理。配送中心客户规模太小，配送中心难以从供应商手中接收多种大批量商品，不



利于转运、分类、保管、流通和加工，无法做到迅速、准确、便宜地配送货物，不利于实现规模经济效益。

### （3）物流企业信息化水平较低

物流企业的信息技术水平较低，是物流和配送发展的巨大障碍。由于我国物流企业总体上信息化起步较晚，信息化程度较低，对企业信息化存在着许多分歧，不少物流企业还处在电话联系、手工操作、人工装卸的低级阶段。许多现代信息技术，如条码技术、射频技术、EDI 技术、全球卫星定位系统技术等，都处于学习和起步阶段。

### （4）物流配送人才缺乏和物流配送观念的落后

物流和配送方面的人才短缺，是物流和配送发展的巨大障碍。国外物流和配送的教育和培训非常发达，形成了比较合理的物流和配送人才的教育培训系统，在相当多的大学和学院中设置了物流管理专业，并广泛地为工商管理各专业的学生开设物流课程；部分商业院校设置了物流方向的研究生课程和学位教育，形成了一定规模的研究生教育系统；在各国物流行业协会的领导和倡导下，全面开展了物流和配送的职业教育。我国在物流和配送方面的教育还很落后，虽然有一些高等院校设置了物流专业和课程，但为数较少，高级物流管理人才更是缺乏。

## 2.2.2 我国物流配送的发展趋势

在信息化时代，随着网络技术，电子商务、交通运输、物流管理的现代化，现代物流企业配送系统也将不断的实现现代化管理模式，从而出现了系统化、自动化、信息化、智能化的趋势，其主要表现如下。

（1）物流业配送模式由生产者和销售商自己进行改变成由第三方物流企业进行

随着物流的发展和分工深化，专业物流企业，即第三方物流企业将不断出现，专业化物流配送系统取代了原有生产者和销售商自营的物流配送系统。这样既能使生产者专心于自己所熟悉的生产业务，又能起到优化资源配置，降低配送费用，提高配送效率。由第三方物流企业负责商品的配送环节可以提高运输的网络效率，并能更好照顾顾客的要求，最大限度缩短运输周期，提高物流运输的反应速度。

（2）物流业配送系统走向计划化、规模化

以单位企业为主体的配送系统存在着车辆利用率低, 运力和仓储投资不合理现象。现代物流企业中配送系统的发展将扩大企业的规模, 扩大企业经营范围, 使配送更加计划化, 运行合理化, 达到服务的高效率。从而使物流企业更具有竞争能力。

### (3) 物流业中配送管理技术与方法现代化

与电子商务发展相结合, 随着运输规模的扩大和计算机的应用, 使信息传递处理逐渐采用 EDI 系统。计算机与自动化装置相结合, 必将使单位产品输运成本降低, 效率提高, 同时也提高了服务质量, 这就使物流业中配送管理的技术与方法进入现代化。

## 2.3 配送系统的规划与设计

### 2.3.1 设计原则

所谓配送作业就是利用配送车辆把用户订购的物品从制造厂、生产基地、批发商、经销商或配送中心, 送到用户手中的工作。车辆配送服务是直接面对用户的服务, 服务优劣对企业效益和信誉影响较大。因此, 在规划和设计物流配送系统时应注意以下几项原则:

#### (1) 时效性

快速及时, 即确保在客户指定的时间内交货是客户最重视的因素, 也是配送运输服务性的充分体现。配送运输是从客户订货到交货的最后环节, 也是最容易引起时间延误的环节, 影响时效性的因素有很多, 除配送车辆故障外, 所选择的配送线路不当、中途客户卸货不及时等均会造成时间上的延误, 因此, 必须在认真分析各种因素的前提下, 用系统化的思想和原则, 有效协调, 综合管理, 选择配送线路、配送车辆、送货人员, 使每位客户在其所期望的时间能收到所期望的货物。

#### (2) 安全性

配送运输的宗旨是将货物完好无损地送到目的地。影响安全性的因素有货物的装卸作业、运送过程中的机械振动和冲击及其他意外事故、客户地点及作业环境、配送人员的素质等, 这些都会影响配送运输的安全性, 因此, 在配送运输管理中必须坚持安全性的原则。

### （3）沟通性

配送运输是配送的末端服务，它通过送货上门服务直接与客户接触，是与顾客沟通最直接的桥梁，代表着公司的形象和信誉，在沟通中起着非常重要的作用，所以，必须充分利用配送运输活动中与客户沟通的机会，巩固和发展公司的信誉，为客户提供优质的服务。

### （4）方便性

配送以服务为目标，以最大限度地满足客户要求为优先，因此，应尽可能地让顾客享受到使捷的服务，通过采用高弹性的送货系统，如紧急送货、顺道送货与退货、辅助资源回收等，为客户提供真正意义上的便利服务。

### （5）经济性

实现一定的经济利益是企业运作的基本目标，因此，对合作双方来说，以较低的费用，完成配送作业是企业建立双赢机制加强合作的基础。所以客户的要求不仅是高质量、及时方便的配送服务，还必须提高配送运输的效率，加强成本控制与管理，为客户提供优质、经济的配送服务。

## 2.3.2 配送运输基本作业流程

现实中的配送管理非常困难。配送管理受到许多因素的影响，如订单处理很费功夫，难以制订配送计划，难以选择配送路径，配送效率低，难以按时交货，配送绩效评价基准不明确，司机及工作时间的不确定，易疲劳以及货物在配送过程中的遗失与损坏，运输车辆及冷藏车的制冷机的完好情况等。因此我们必须加强物流配送系统的规划和设计。同时在实际配送的过程中，还受许多不可控因素的影响，如用户的分布区域、道路交通网络、车辆通行限制、送达时间要求、车流量变化、道路施工、用户变动、车辆变化等。为此，必须成立一个规划和设计小组，对物流配送系统进行总体规划和设计，如根据客户分布地点划分基本配送区域，完成车辆配载，暂定配送的先后顺序，考虑货物、车辆的特征及费用安排车辆，选择配送路线，确定配送顺序，最后完成车辆积载。如图 2-1 所示。

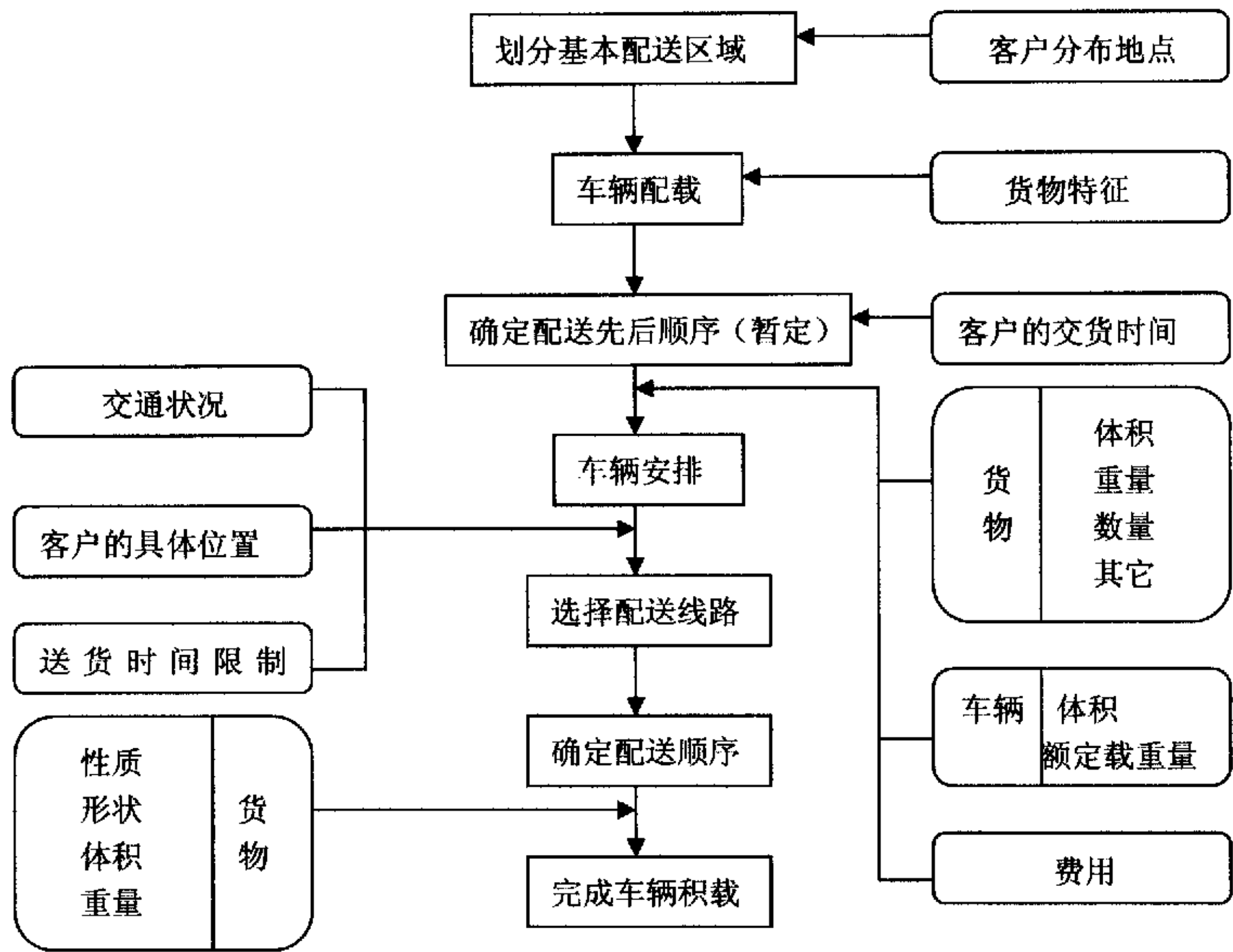


图 2-1 配送运输基本作业流程

2.3.3 配送系统的具体规划与设计

物流配送系统规划和设计是根据外部环境的变化，选择最佳的配送路线，结合内部环境的变化配置适宜的车辆并提高单位装载率，防止交错运输，尽可能降低配送差错率，不断改善流通加工、分拣、验货、包装等功能，使其处于最佳工作状态。

车辆配送是物流中心作业最终及最具体直接的服务表现，其服务要点有下列各项：

- （1）按时送货：按时送货是指按照客户要求的时间将商品送到客户手中，这一点在配送中是最重要的，为了满足客户的要求，必须提前做好规划。
- （2）按质送货：按质送货是指将商品完好无缺地送到客户手中。按质送货不仅要求将商品完整无缺地送到客户手中，而且要求配送人员向客户提供的优质服务。



(3) 成本最低：为了提高市场的竞争，在提供配送服务时，在保证按时送货和按质送货的前提下，价格也是比较重要的因素，因而若能让物流中心本身运作有效率，成本控制得当，自然对客户的收费也能较低廉，也就更能以经济性来抓住客户了。

在物流中心，或者货运公司，百货商场，在城市配送时，上面三个要求是必须的。

在配送实际中，由于受众多因素的影响，配送规划，往往是一个极其复杂的系统工程，如何来规划配送，才能达到上述三个要求呢？这是摆在物流企业面前的一个难题。

做这个配送系统的初步设计，是指在作配送规划时，运用计算机技术、图论、运筹、统计、GIS 等方面的技术，由计算机根据配送的要求，选出一个最佳的配送方案，包括配送路线，使用车辆，装载的商品等内容。设计的物流配送系统可以包括五个子系统，如图 2-2 所示。

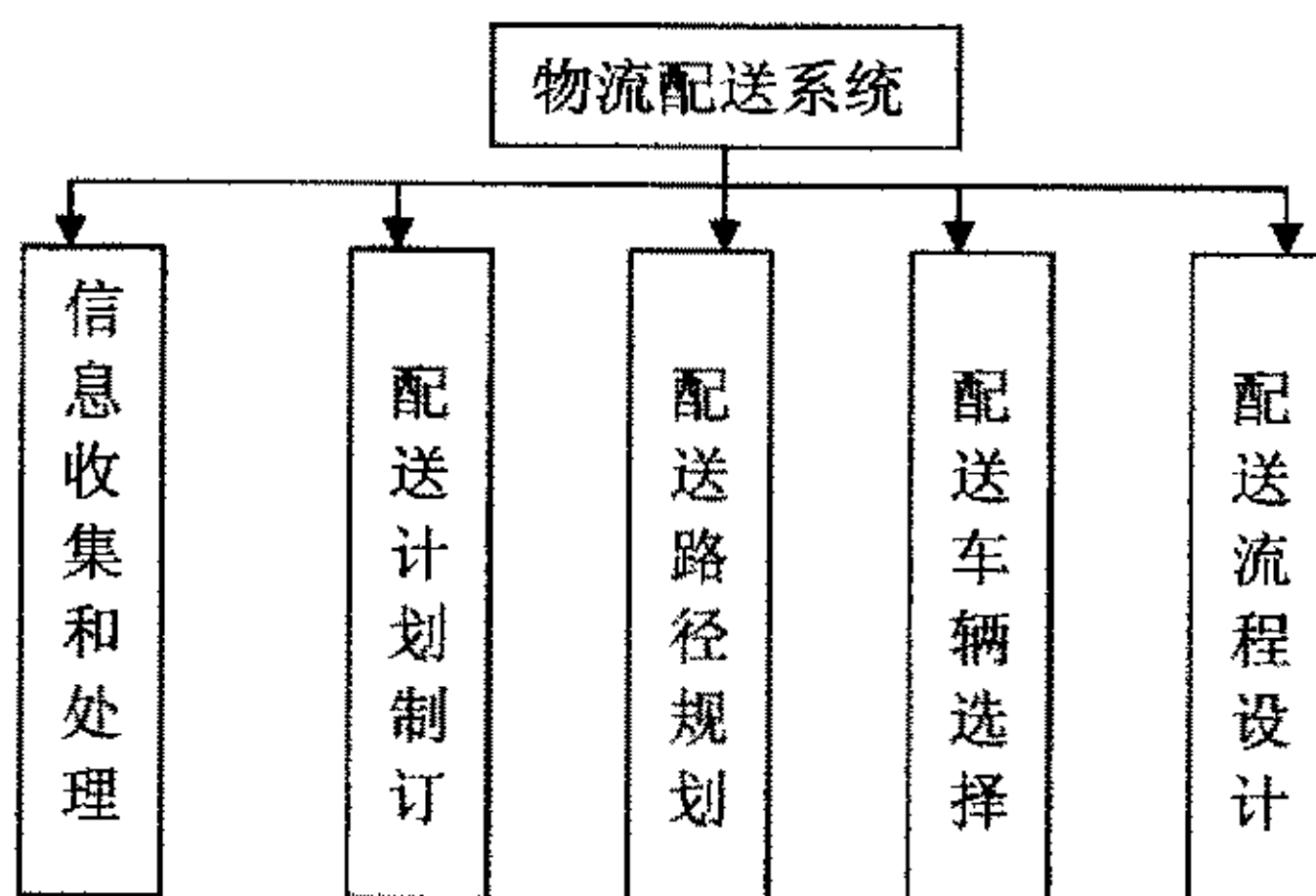


图 2-2 物流配送系统

根据配送系统这五个子系统，可以分模块设计并利用计算机实现整个配送系统的智能化，具体模块如下：

基本信息模块：提供系统数据库，包括地理数据库、客户基本资料、运输车辆资料（车辆基本信息、车辆维修、车辆保险及事故、车辆业绩、车辆经费、车辆装载量等）、司机及其它工作人员资料等。

订单管理模块：针对客户订单进行管理，是客户服务部门接受订单之后，将订单录入系统，从而开始单据在物流系统中的流转过程，同时对订单进行跟

踪管理，并与客户之间保持联系，向客户及时地提供订单的执行情况。

货物管理模块：包括收货管理、分货管理和拼箱管理，货物清点后将货物按种类进行分货并将小包装货物拼箱成大包装货物，这样易于货物的管理。

仓库管理模块：是物流管理系统中心环节，其包含本地库和外地库管理，本系统是专门针对第三方物流的仓储进行管理的，仓库管理系统实现了对仓储运作的全过程的管理，它包括入库、出库、盘点、不良品处理、库存时实查询等齐备的功能，同时系统实现了对货物的批次进行精确的跟踪，提供完善的单据报表，采用严格的权限控制，从而保证了仓库运作的严格、安全、快捷、高效、有序。为物流企业的仓库管理提供了一个良好的平台，并为企业电子商务的应用预留了可扩展的接口。系统通过互联网进行外地库的管理。

配送管理模块：是物流管理系统的中心模块，主要针对车辆现行情况进行配货，也是公司节约资金的重要环节，配送货物管理可以使企业更为有效地利用有限的资源。对配货中线路安排、车辆安排进行计算时本文后面要考虑的主要问题。

系统的执行过程如图 2-3。

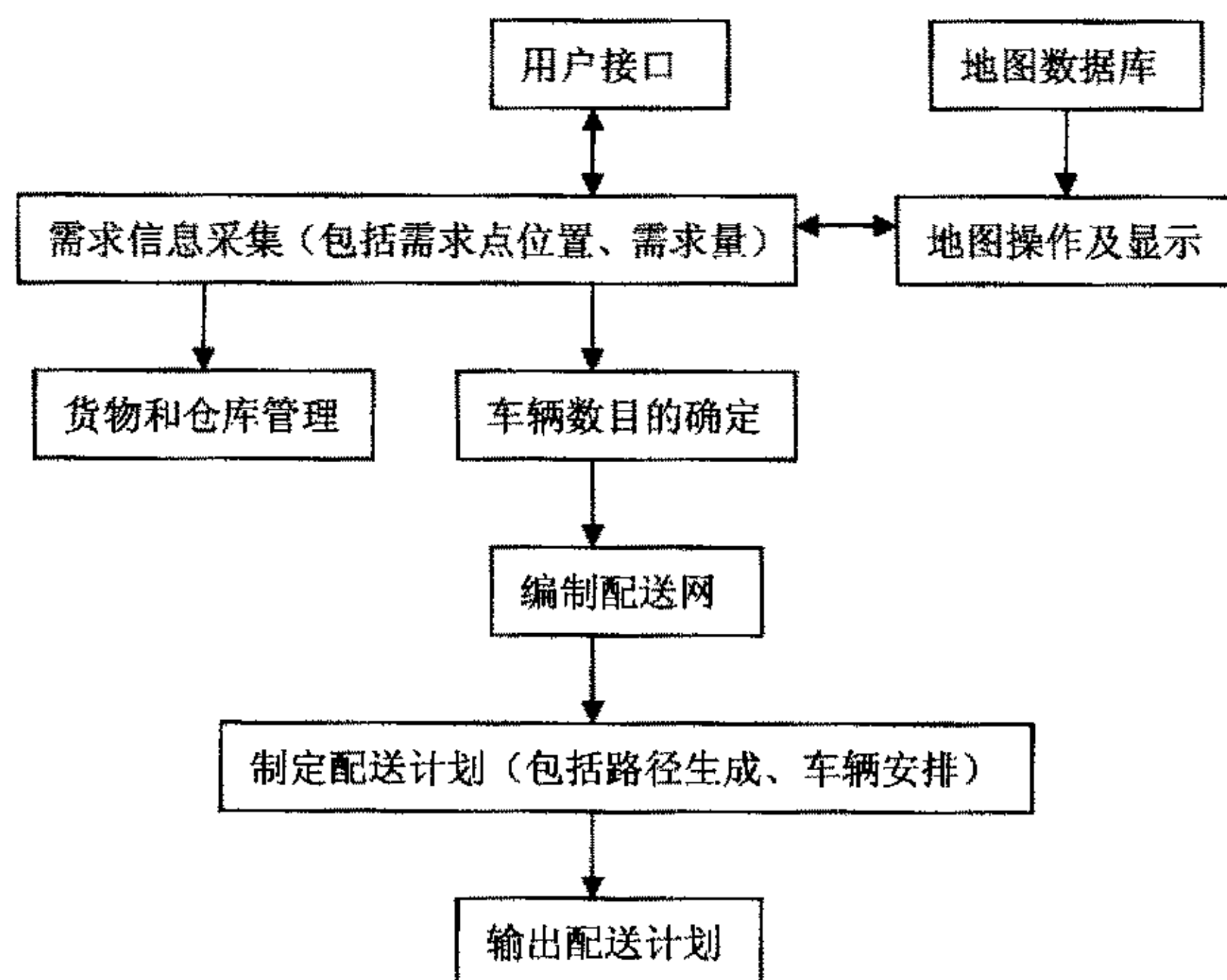


图 2-3 系统执行过程

此外,为了使系统更加完善,还可以使用 GIS 和 GPS 技术在车辆出门以后的管理,对车辆实行在途跟踪,物流中心可以根据实际情况,对在途的车辆进行各种指示,如:指示车辆顺路拉回退货的商品、回收商品的容器等。在途车辆也可以向物流中心发出各种紧急信息,请求处理意见等,如:遇上交通堵塞,请求物流中心重新指示行车路线,车辆损坏,请求维修支援等。

## 2.4 本章小结

本章首先介绍了物流配送的模式以及我国目前物流配送的发展及存在的一些问题,说明了解决物流配送的问题对提高物流服务水平具有重要的现实意义。然后针对配送系统的特点和要求,对配送系统作了初步的规划和设计。

## 第 3 章 VRP 的提出及各类 VRP 的算法

### 3.1 VRP 的由来

#### 3.1.1 旅行商问题的由来

旅行商问题 (Traveling salesman problem, TSP) 的形象化描述是指一个商人想要到  $n$  个城市推销商品, 希望选择一条道路, 使得经过所有城市一次且仅一次, 最后回到起点。问题的目标是帮助这个商人找到最短的路径。

TSP 以图论的形式描述为: 在图  $G=(V, E)$  中,  $V$  是点 (城市) 的集合,  $E$  是边的集合,  $G=\{(i, j) | i, j \in V\}$ 。点  $i$  与  $j$  的欧氏距离为  $d_{ij}$ , 设  $d_{ij}=d_{ji}$ 。目标是找到一个长度最小的闭合回路, 使得访问每个点一次且仅一次, 这条闭合回路也称作哈密顿回路。

此外, TSP 从组合优化的角度描述, 是一个有限集合下的最优化问题: 设有限集合  $D$  是有限个决策变量  $x$  组成的集合, 集合  $F=\{x \in D | g(x) \geq 0\}$  称为可行解区域,  $x \in F$  称为可行解, 用  $f: D \rightarrow R$  表示目标函数, 称满足

$$f(x^*) = \min\{f(x) | x \in F\} \quad (3-1)$$

的可行解  $x^*$  是组合优化问题  $(D, F, f)$  的最优解, 而模型  $(D, F, f)$  表示

$$\min f(x) \quad (3-2)$$

$$\text{s.t. } g(x) \geq 0 \quad (3-3)$$

$$x \in D \quad (3-4)$$

因为  $D$  是一个有限集合, 因此我们可以用穷举的方法证明最优解  $x^*$  一定存在, 但是随着问题规模的扩大, 用穷举法显然不现实。

TSP 已经被证明是 NP (Nonlinear Problem) 完全问题, 是典型的易于描述, 但是最优化求解很难的问题。主要原因是所谓的“组合爆炸”现象。例如, 对于置换排列描述的  $n$  城市 TSP 问题有  $(n-1)!/2$  种不同的排列, 状态数量随着问题规模呈超指数增长。 $n=20$  的时候, 使用穷举法求解, 即使计算机每秒能够处理 1 亿种排列, 也需要大约几百年的时间。



### 3.1.2 VRP 的由来

VRP 的原型就是 TSP。VRP 是对一系列特定位置和需求量的客户点，调用一定数量的车辆，从中心仓库出发，选择最优的行车路线，使车辆有序的访问各客户点，在满足特定的约束条件（如客户需求量，车辆载重量等）下，使得货物尽快到达客户点并且运输总费用最低。

从图论的角度来看，VRP 可以描述为基于无向图  $G=(V, A)$ ，其中  $V=\{v_0, v_1, \dots, v_N\}$ ， $V$  代表顶点集合， $E=\{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$ ， $E$  代表连接两两顶点的线段的集合，即边的集合。顶点  $v_0$  表示配送中心的位置，其他顶点则表示有待服务的顾客的集合。在  $E$  上可以定义一个非负的距离矩阵或成本矩阵  $D=\{d_{ij}\}$ ，此处  $d_{ij}$  代表从顶点（顾客） $i$  到顶点（顾客） $j$  的距离或成本。

若与 TSP 联系起来，VRP 就相当于在  $m$ -TSP 问题中给每个城市添加了一个固定的需求量，同时每一辆车（旅行商）都有一定的容量限制，即分配给每辆车服务顾客的需求量之和不能超过该车的载重量。所以一般的 VRP 也常常被称之为带有容量限制的车辆路径问题（Capacitated Vehicle Routing Problem, CVRP）。图 3-1 所示的是一个典型 VRP 问题的解，其中的方块代表配送中心，每辆车的载重量（即容量）为 30 个单位，括号内数字代表各个顾客点需求量。

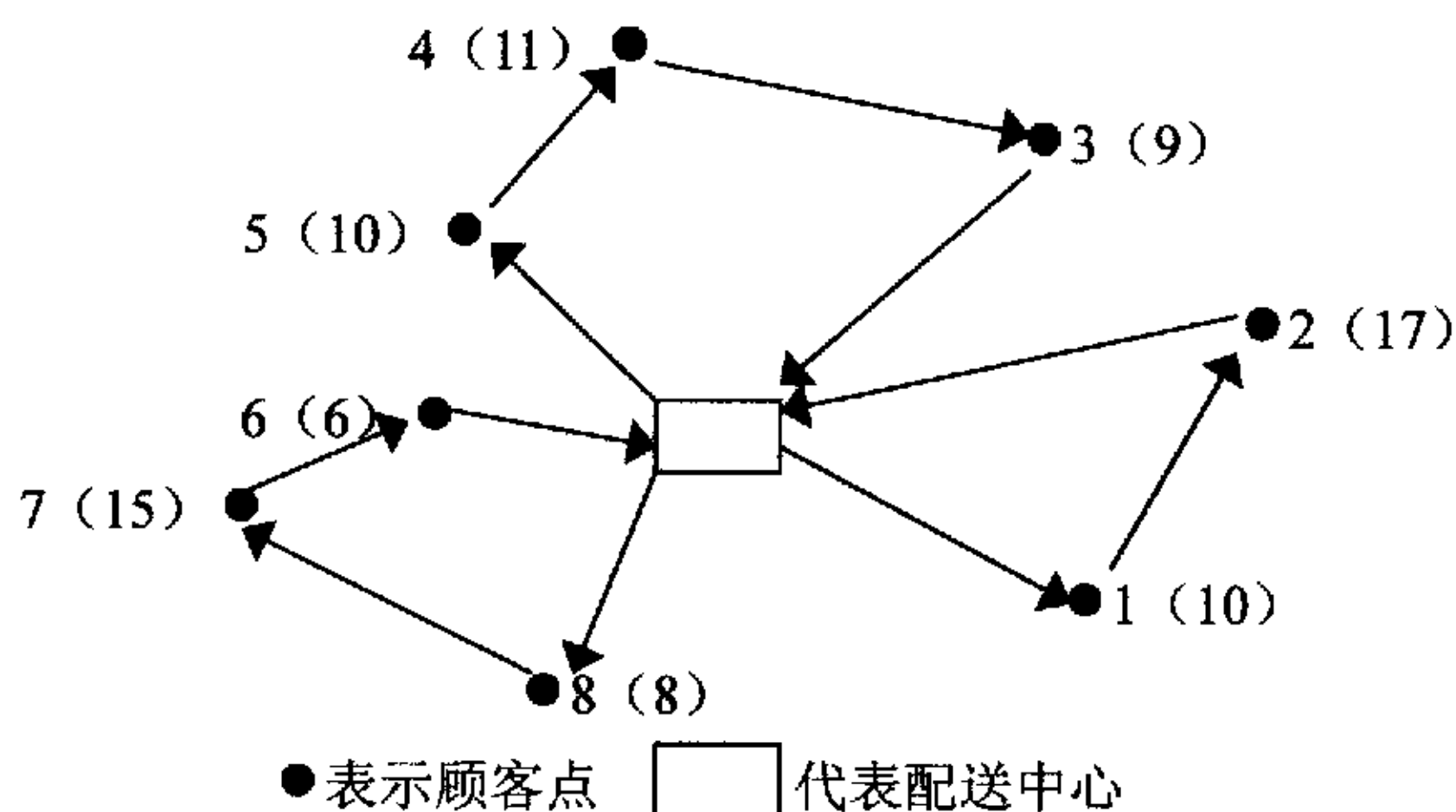


图 3-1 拥有三条路线的一般的 VRP 的一个可行解

研究 VRP 一般存在以下几个前提条件：

- (1) 被配送的是可混装的物资；

- (2) 各个用户的所在地和需求均已知;
- (3) 从配送中心到各用户的距离以及各用户之间的距离已知;
- (4) 配送中心有足够的资源以供配送, 并且拥有足够的运输能力。

## 3.2 各类 VRP 的算法

根据研究重点的不同, VRP 存在多种分类方式。按已知信息的特征可分为确定性 VRP 和不确定性 VRP, 其中不确定性 VRP 可进一步分为随机车辆路径问题(SVRP)和模糊车辆路径问题(FVRP); 按约束条件可以分为带有容量限制的車輛路径问题(CVRP)、带有时间距离约束的車輛路径问题(DVRP)以及带有时间窗的車輛路径问题(VRPTW); 按需求是否可以分割, 可以分为可分割的車輛路径问题和不可分割的車輛路径问题; 按每个顾客需求量是否超过车的容量来分, 可以分为满载車輛路径问题和非满载車輛路径问题; 按配送中心的多少来分可以分为单车场車輛路径问题(SVRP), 即一般車輛路径问题(VRP), 以及多车场車輛路径问题(MVRP), 其中 MVRP 又可以根据是否每辆车都有固定的终点车场分为终点车场固定的車輛路径问题和终点车场不固定的車輛路径问题。以下针对各种不同类型的 VRP 人们提出的一些算法进行了归纳:

### 3.2.1 确定性 VRP

确定性 VRP 是现实中最常见的类型。它针对的背景是: 已知客户/节点的确切需求信息, 以及客户/节点的位置。这种问题的实例如: 物流中心对各商场提交的商品需求信息进行处理, 决定第二天商品配送路径和过程。解决这类问题的算法有两种: 精确算法和人工智能算法。

#### 3.2.1.1 精确算法

精确算法可以分为三个大类: 直接树搜索算法、动态规划方法和整数线性规划。

##### (1) 分支定界算法

该方法是 Laporte 等人提出的。它利用了 VRP 和其放松形势  $m$ -TSP 间的关系。根据 Lenstra 等人所给出的  $m$  的上界  $m_v$ ,  $m$ -TSP 可转化为 1-TSP。关键步骤是, 引入  $m_v-1$  个伪出发点,  $n'=n+m_v-1$ ,  $v'=(1, \dots, n')$ ,  $A'=A \cup \{(i, j):$

$i, j \in v', i \neq j, i \text{ 或 } j \in v' \setminus v\}$ 。接下来用分支定界算法求解。目前可求解有 260 个客户的 VRP。

### (2) k 度中心树算法

该方法是 Christofides 等人提出的。对固定  $m$  的  $m$ -TSP 进行  $k$  度中心树松弛。所以该方法需要知道所需车辆数的下界。其模型是用边的角度建立的，出发点用一条边表示，其它点用两条边表示。通过拉格朗日松弛法，将其中的一个约束条件消去，并进一步将原来的最小化问题转化为 3 个易求解的子最小化问题，然后进行求解。目前可求解有 134 个客户的 VRP。

### (3) 动态规划法

动态规划法是由 Eilon 等人首先提出的。它针对的也是固定车辆数的 VRP，通过递归方法求解。为减小问题的计算规模，引入可行性规则或松弛过程减小状态的数量。其后，Christofides 提出了状态空间松弛，极大地减少状态数量。该方法要求：转换函数易于求解，映射出来的范围小，可求得很好的下界。目前可求有 50 个客户的 VRP。

### (4) 集分割和列生成

这种方法直接考虑可行解集合，在此基础上进行优化，因此建立的 VRP 模型最简单。但缺陷在于如果问题所受的约束不严格，则所需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很困难。对于其中规模相对较小的、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 等人引入了列生成方法进行求解，在此方法中，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。在此基础上重复求解。通过引入优化对偶变量向量，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，同时结合了分支定界算法。Desrocher 用它求解有 100 个客户的带时间窗问题的 VRP。

### (5) 三下标车辆流方程

Fisher 等人针对带能力约束、时间窗口以及无停留时间的 VRP 问题，提出了三下标车辆流方程。在该方程中，其中两个下标表示弧或边，另外一个下标表示特定车辆的序号。基于 Benders 的分解技术，他们提出了一种启发式算法，保证在有限的步骤内找到优化解。Fisher 等人用它计算了有 50~199 个客户的 VRP。Martello 和 Desrochers 分别提出了相应的改进算法。

### (6) 二下标车辆流方程

对于对称的 CVRP 和 DVRP, 可通过去掉表示车辆序号的下标, 引入所需车辆数的下界, 得到一个更为紧凑的方程。它所对应的算法结合了爬山法的思想, 其算法核心仍然是线性规划, 若得到的解是分数解, 则用分枝定界方法求其整数解。该方法是由 Laporte 等人提出的, 已用它来求解了规模为 60 的问题。

总的来说, 精确性算法基于严格的数学手段, 在可以求解的情况下, 其解通常要优于人工智能算法。但由于引入严格的数学方法, 因而无法避开指数爆炸问题, 从而使该类算法只能有效求解中小规模的确定性 VRP。具体到每个算法, 它们都有其适用范围和特点。给定下界和相关的分枝定界算法是从所要访问的点的角度出发建立的, 因此不仅适用于对称 VRP, 还适用于非对称的 VRP; 三下标车辆流方程在模型中有效引入了代表时间窗口的变量, 从而可适用于通用任务分配问题(GAP)和带时间窗口的 TSP(TSPTW); 二下标车辆流方程是由 TSP 的 SYM 方程扩展而来。由于去掉了代表车辆序号的下标, 形式上更为紧凑, 具有更少的约束条件, 所以仅适用于对称的 CVRP 和 DVRP, 而且特别适用于这两种 VRP 中约束条件比较宽松的问题。而 k 度中心树、动态规划和集分割等方法在对应的模型中, 以不同的方式给出能力约束、子回路约束等约束条件, 适用于约束条件严格的问题。

### 3.2.1.2 人工智能算法

在求解中小规模 VRP 时, 人工智能算法与精确算法相比, 在精度上不占优势。但在求解大规模 VRP 时, 人工智能方法总可以在有限时间里, 找到满意的次优解/可行解, 这是精确算法难以做到的。因此, 在实际应用中, 人工智能算法要更广泛。

由于 VRP 与 TSP 存在内在的本质联系, 因此有相当数量的 TSP 的启发式算法在考虑到 VRP 的特点和约束后, 可以用于 VRP, 诸如邻近点算法、插入算法、行程改进算法等。但是, 大多数的 VRP 的人工智能算法并不等同于 TSP 的启发式算法, 常见的有特色的人工智能算法有以下几种:

#### (1) Clarke-Wright 算法

该算法是由 Clark 和 Wright 提出的, 用来解决车辆数不固定的 VRP。该算法最初按所需访问的点数  $n-1$  (不含出发点) 生成同样数量的路径, 计算合并任意两条路径后可节省的成本量  $s_{ij}=c_{il}+c_{lj}-c_{ij}$ 。然后对可节省的成本量进行排序。最后根据排序结果以及可行性条件, 对路径进行归并 (这里的归并是指: 去掉分



别位于原来两条路径中的弧  $(i, 1)$  和  $(1, j)$ , 用  $(i, j)$  代替), 直到无法找到更好的解。这种算法的复杂度为  $n^2 \log n$ 。在 Golden, Nelson, Paessens 等人的研究结果中, 均通过使用适当的数据结构, 来降低它的复杂度。

#### (2) Sweep 算法

该算法是由 Wren, Gillett 等人提出的。即先计算出所要访问的点的极坐标, 并按角度大小排序。然后在满足可行性条件的前提下, 按角度大小归并到不同的子路径中。最后再根据 TSP 的优化算法对所得到的子路径进行优化。

#### (3) Chrisofides-Mingozi-Toth 两阶段算法

它主要面向 CVRP 和 DVRP。该算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, 然后再用 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。

#### (4) 禁忌搜索

Gendreau 等人最先将该方法应用于 VRP。先构造一系列的解, 然后对所得解不断地进行改进。该算法所得到的解不一定是可行解, 它们对可行性的偏离程度是通过目标函数里的罚函数来体现的。该算法求解过程中的邻域, 是通过 GENI 过程得到的。它针对 VRP 的比较好的启发式算法, 可以成功地应用于许多经典的 VRP。其后 E.Taillard 等人通过按角度和路径重心对原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, 实现了对问题求解的并行化。

#### (5) 遗传算法

J.Lawrence 最先将该方法用于 VRP 问题的研究, 并可有效求解带时间窗口的 VRP 问题。鉴于传统的遗传算法是个大范围、粗粒度的寻优算法, 因此, Barnier 将它与约束满足问题(CSP)的技术相结合, 通过遗传算法来处理 CSP 参数的子域(基因的适应度是通过对 CSP 解的计算得到的), 从而减小搜索空间, 降低 CSP 问题目标函数和遗传算法约束的复杂度。我国张涛等人则是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。这类算法目前已可以求解较大规模的问题(199 个客户)。

#### (6) 重复匹配

该方法是由 P.Wark 等人提出的。首先对每个客户生成一条子路径, 然后提供了总匹配成本和负载改变匹配成本, 作为归并路径的依据, 同时为满足自匹

配条件的集合提供分割手段, 以利于跳出局部最优。该算法也可求解较大规模的问题(199 个客户)。

Clarke-Wright 算法归并子路径的基本前提是要满足车辆的能力约束, 因此该算法适用于 CVRP; Sweep 通过角度大小对客户/节点加以聚类, 本质上是将距离近的客户归并到一个子路径中, 同时聚类要满足的可行性条件仍为车辆的能力约束, 因此, 该方法也适用于 CVRP; Chrisofides-Mingozzi-Toth 两阶段算法、遗传算法进行聚类是要满足可行性条件, 在这里指的是能力约束, 也可以是时间约束, 因此该方法适用于 CVRP 和 DVRP; 禁忌搜索的可行性条件也是可以选择的, 因此其适用范围与两阶段法相同; 重复匹配算法在其模型里同时考虑了时间约束和能力约束, 因此它适用于这类具有强约束的 VRP。

### 3.2.2 非确定性 VRP

#### 3.2.2.1 随机 VRP

现有的随机 VRP 是针对如下背景的配送问题: 每天要访问的客户或节点的数量和位置是固定的, 但每个客户每天的需求是不同的, 并各自满足一定的可能性分布或随机分布。如果限于时间或资源的关系, 调度员无法等到获得所有信息后才作决策, 则需要应用本节所讨论的方法。这类问题中典型的例子有: 银行运钞车傍晚时对市内各 ATM 的取款服务(为安全计, 每个运钞车所运现金数量不能超过一定限额, 这就构成了其能力约束); 中心邮局对各邮政支局的取邮件服务等。

对这种类型的问题, 目前出现的算法大部分可以归纳为以先验(即定)序列为基础的方法。该方法分为两个阶段: 在信息不完全(随机信息)的情况下确定先验序列; 在获得确定性信息的情况下进行决策。因此, 其随机模型的选择基于两点: 第一阶段的成本和第二阶段的期望成本。先验序列的确定方法又有两类, 一类是按二元可能性理论来确定, 另一类则是基于机会约束规划。

(1) 第一类方法的思想最早是在 Jaillet 博士的论文中提出的, 并在 Bertimas 的博士论文中基本建立完整的体系。在该方法中, 假定需求分布是二元(即在第  $i$  点有单位需求的概率为  $p_i$ , 则没有单位需求的概率为  $1-p_i$ )、离散的。根据该假定, 可推出先验序列的期望长度, 相应的上下界和渐进特性。Michel, Gendreau 等人以该理论为基础, 得到目标函数和罚函数, 以 Clark-Wright 算法生成初始解,

然后用禁忌搜索算法对初始解进一步优化,可以求解规模为46的问题。而Laporte基于L-Shaped方法,提出一个精确算法,可解决规模小于50的问题。

(2) 另一类方法是 Stewart, Dror, Laporte 等人分别使用机会约束规划,在一定的假定条件下,将SVRP转化为等价的、确定性的VRP,并找到了解的界,其方法的核心是让出错返回的概率不超过一定的界限。机会约束规划模型均以确定性VRP的三下标流和二下标流模型为基础,添加可能性约束条件和罚函数后建立起来的。所用的算法,也与确定性的算法有不少相通之处。如M.Dror就是在建立三下标流机会约束规划模型和罚函数模型后,利用Clark-Wright算法进行求解的。

第二阶段的策略为:按照先验序列,跳过需求为0的客户,直接访问下一个客户,如果车辆负荷超过了车辆最大载重量,则返回出发点卸载,并回到出现超载的地方,继续提供服务。

在随机VRP中,还存在LRP(位置路由问题)和需求可切分的SVRP等类型的问题。LRP需要在仓库(出发点)运作成本已给出的情况下,确定它们的位置,使仓库运作成本最小化,同时还要给出最优配送路径。G.Laporte等人给出的方法也是基于两阶段法,第一阶段确定仓库位置和设计路径,第二阶段采用出错返回方式,并分别建立了基于约束和基于边界罚函数的模型,最后用分枝定界法进行求解。B.Bouzaïene-ayari等人对需求可切分的SVRP问题,利用Dror等建立的模型和算法求解。总之,其他类型的SVRP的模型和求解算法到目前为止都是以基本型SVRP为基础。

### 3.2.2.2 模糊VRP

在实际的配送体系中,如航空快递的上门取件服务,某些待服务的客户/节点需求信息没有或无法给出准确的描述,如:“大约10千克”,“40到45单位”,“100单位左右”,“装了几件衣服的小箱子”,这样,就需要引入模型和算法来解决这类问题。D.Teodorovic等人是最早着手研究这类问题的人。他以模糊数表示客户点的需求信息,以倾向度为基础建立模糊判定规则,其核心是基于Sweep算法,并省略了Sweep算法的第二阶段,即初始化子路径后对它们的优化过程。

祝崇隽、刘民、吴澄等人以模糊可能性分布建立了VRP的基于置信度的三下标流模型,并提出了基于可能性分布的2-OPT算法。该算法引入了伪出发点,建立了以置信度为基础的判定规则,以遍历为终止条件,从而在全局层次进行

优化, 同时避免过多地扩大搜索空间。该算法已可解决较大规模的问题(>200 个客户)。

下面主要把针对确定性 VRP 的算法及其适用范围列成表 3-1:

表 3-1 各类 VRP 的算法及其适用范围

类型	算法			适用范围
确定性 VRP	精确性算法	精确性算法基于严格的数学手段, 在可以求解的情况下, 其解通常要优于人工智能算法。但由于引入严格的数学方法, 因而无法避开指数爆炸问题, 从而使该类算法只能有效求解中小规模的确定性 VRP。	给定下界和相关的分支定界算法	给定下界和相关的分枝定界算法是从所要访问的点的角度出发建立的, 因此不仅适用于对称 VRP, 还适用于非对称的 VRP。
			三下标车辆流方程	三下标车辆流方程在模型中有效引入了代表时间窗口的变量, 从而可适用于通用任务分配问题(GAP)和带时间窗口的 TSP(TSPTW)。
			二下标车辆流方程	二下标车辆流方程是由 TSP 的 SYM 方程扩展而来。由于去掉了代表车辆序号的下标, 形式上更为紧凑, 具有更少的约束条件, 所以仅适用于对称的 CVRP 和 DVRP, 而且特别适用于这两种 VRP 中约束条件比较宽松的问题。
			k 度中心树和相关算法	这三种方法在对应的模型中, 以不同的方式给出能力约束、子回路约束等约束条件, 适用于约束条件严格的问题。
			动态规划法	
			集分割和列生成	



续表 3-1 各类 VRP 的算法及其适用范围

类型	算法			适用范围
确定性 VRP	人工智能算法	在求解中小规模 VRP 时, 人工智能算法与精确算法相比, 在精度上不占优势。但在求解大规模 VRP 时, 人工智能方法总可以在有限时间里, 找到满意的次优解/可行解, 这是精确算法难以做到的。因此, 在实际应用中, 人工智能算法要更广泛。	Clarke-Wright 算法	此算法归并子路径的基本前提是要满足车辆的能力约束, 因此该方法也适用于 CVRP
			Sweep 算法	Sweep 通过角度的大小对客户/节点加以聚类, 本质上是将距离近的客户归并到一个子路径中, 同时聚类要满足的可行性条件是车辆的能力约束, 因此该方法也适用于 CVRP
			Chrisofides-Mingozi-Toth 两阶段算法	这三种方法进行聚类要满足的可行性条件可以是能力约束, 也可以是时间约束, 因此这两种方法适用于 CVRP 和 DVRP
			遗传算法	
			禁忌搜索	
非确定性 VRP			重复匹配	此方法在其模型里同时考虑了时间约束和能力约束, 因此适用于具有强约束的 VRP

### 3.3 本章小结

本章介绍了 VRP 的由来，并针对各种不同的 VRP 的算法作了详细的综述。本论文研究的对象是确定性 VRP，解决这类问题有两类算法，一种是精确算法，一种是人工智能算法。由于精确算法虽然精度高，但无法避开指数爆炸问题，因此该类算法只能有效求解中小规模的确定性 VRP，不具有一般代表性意义。本文选用的是人工智能算法中的遗传算法对 VRP 问题进行求解。

## 第4章 遗传算法基本理论

遗传算法 (Genetic Algorithm, 简称 GA) 是一种有效的解决最优化问题的搜索算法。它是由美国 J.Holland 教授于 20 世纪六、七十年代研究形成的一个较完整的理论和方法, 是模拟达尔文的遗传选择和自然淘汰的生物进化过程的计算模型<sup>[25]</sup>。遗传算法是一种自适应全局优化概率随机搜索方法, 它对优化对象既不要求连续, 也不要求可微, 目前已被广泛应用于组合优化、人工智能、人工生命领域, 并取得了良好效果。

### 4.1 遗传算法的基本理论

#### 4.1.1 基本概念

遗传算法是建立在自然选择和群体遗传学机理基础上的随机、迭代、进化, 具有广泛适用性的概率搜索方法。它模拟生物进化论中的“优胜劣汰、适者生存”的自然法则, 即若当前个体的性能越优, 则越容易传到下一代, 从而将个体的优良基因继承下去, 直至搜索到近似最优解或满意解。在此为了更好的理解遗传算法基本思想, 先介绍有关遗传算法的几个概念。

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

个体 (Individual) 或染色体 (Chromosome), 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 (有时也采用一个向量表示):  $X=x_1x_2\cdots x_i$ , 其中  $x_i$  是串  $X$  的基本单元, 称为基因 (Gene),  $i$  称为基因数。染色体上的一个有效信息段叫基因组 (Gene Group), 一般一个基因组对应于解中的一个优化参数。在遗传算法中, 根据问题的不同, 个体一般可分为 2 进制的 0、1 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; 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 $N$

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  $N$  表示群体中所含个体的数量。种群规模的取值非常关键，当  $N$  过小时，可提高算法的运行速度，但却降低了种群的多样性，有可能引起遗传算法的早熟现象，而当  $N$  过大时，又会降低算法的运行速度，所以一般建议的取值范围为 10-160。

### (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。基本遗传算法按与个体适应度成正比的概率来决定当前种群中每个个体遗传到下一代种群中的机会多少。

### (5) 遗传算子 (Genetic Operators)

遗传算子作用在基因串上，是模拟种群进化过程的主要手段。常用的遗传算子有如下三种：

#### ① 选择算子 (Selection Operators)

选择算子作用在一个基因串上，它根据一定的选择策略，以较大的概率从父代群体中选择适应值较大的个体复制到子代中，等待交叉和变异对其进一步演化。

目前，有多种不同的选择策略，如比例选择法（赌盘选择法）、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，各种策略在原理上是一致的，即在旧的群体中“随机”选择个体生成一个新群体，这种“随机”选择并非完全随机，它是基于一个个体相对于整个群体的适应值，根据个体的适应值确定选择的系数，按比例复制生成新个体加入新种群中。其中赌盘选择策略最为常用。



## ②交叉算子 (Crossover Operators)

交叉算子通常作用在两个基因串上，它把两个基因串中的某一部分相互交换产生两个新的个体。交叉算子又分为点式交叉和均匀交叉。

### A、点式交叉 (Pointal Crossover)

点式交叉首先随机地在两个父代串上选择一个 (单点式交叉) 或多个 (多点式交叉) 交叉点，然后交换 (单点式交叉) 或间断式交换 (多点式交叉) 父代串的对子串。简单的单点交叉如图 4-1 所示 (“|” 代表交叉点)。

父代 1	XXXX   XXXXX	子串 1	XXXXYYYYY
父代 2	YYYY   YYYYY	子串 2	YYYYXXXXX

图 4-1 点式 (单点式) 交叉操作示意图

### B、均匀交叉

均匀交叉是概率交换两个父串的每一位，其过程是：先随机产生一个与父串具有同样长度的二进制串，其中 0 表示不交换，1 表示交换。这个二进制串成为杂交模板 (Template)，然后根据该模板对两个父代进行交叉，所得的两个新串即为子串。

点交叉能以较大的概率保护优良的模式，均匀交叉则具有较强的搜索能力。这两种交叉都是针对二进制编码的交叉操作，对一些数值问题，常采用十进制 (实数) 编码，此时就需要根据点交叉和均匀交叉设计专门的十进制交叉算子。交叉算子是遗传算法中产生新个体的主要方法，所以交叉概率一般应取较大值，通常取值范围为 0.4—0.99。

## ③变异算子 (Mutation Operators)

变异算子作用在一个基因串上，它是遗传算法用以模拟生物在自然界的遗传环境中由于各种偶然因素引起基因突变而引入的一种算子，它按照一定的概率，随机选取符号串中的一个基因，改变其值，生成一个新的个体。

表 4-1 对常用的各种常用的遗传算子进行了总结。

以上对遗传算法中的一些基本概念做了阐述。在此，将本文中提到的遗传算法中的一些基本用语与自然遗传学中的基本用语的对应关系列表如下 (表 4-2)。

表 4-1 各种常用的遗传算子

遗传算子	操作	常用方法
选择算子	按照个体适应度的大小选择进入下一代的个体,或者选择直接进行交叉,变异的个体	赌盘选择法、随机联赛选择法、最佳个体保留选择法、排序选择法、精英策略等
交叉算子	将两个相互配对的染色体按照某种方式相互交换部分基因,从而形成两个新的个体	单点交叉、双点交叉、多点交叉、均匀交叉;最大保留交叉、顺序交叉(OX)、部分匹配交叉(PMX)、循环交叉(CX)以及各种基于一定知识的启发式交叉
变异算子	将个体中某个或某些基因按小概率扰动,维持种群的多样性,增强 GA 的全局搜索能力	基本位变异、均匀变异、高斯变异;倒置变异、对换变异;k 交换变异;基于一定知识的启发式变异等

表 4-2 自然遗传学和遗传算法中的基本用语对照表

自然遗传学	遗传算法
染色体 (Chromosome)	解的编码 (数据、数组、位串)
基因 (Gene)	解中每一个分量的特征
个体 (Individual)	解 (也叫染色体)
适应性 (Fitness)	适应度函数值
群体 (Population)	选定的一组解 (其中解的个数叫群体规模)
复制 (Reproduction)	根据适应函数值选区一组新解的过程
交叉 (Crossover)	通过交叉原则产生一组新解的过程
变异 (Mutation)	编码的某一分量 (基因) 发生变化的过程

对于 VRP 的优化问题,以前的研究已经发现采用整数编码可以取得较好的结果。为了避免生成不可行解,我们要设计专门的交叉算子和变异算子来解决

VRP。具体的设计在第 5 章说明。

#### 4.1.2 遗传算法执行过程

遗传算法基本的执行过程如下：

(1) 编码：遗传算法在进行搜索之前先将解空间的解数据表示成遗传空间的基因型串数据，这些串结构的不同组合便构成了不同的编码。

(2) 初始群体的生成：随机产生  $N$  个初始串结构，每个串结构成为一个个体（或叫染色体）， $N$  个个体构成了一个群体，群体内个体的数目  $N$  就是群体规模。群体内每个染色体必须以某种编码形式表示，编码的内容可以表示染色体的某些特征。随着求解问题的不同，它所表示的内容也不同。通常染色体表示被优化的参数，每个初始个体就表示问题的初始解。

(3) 适应值评估检测：适应值函数表明个体或解的优劣性。对于不同的问题，适应值函数定义的方式也不同。

(4) 选择：选择的目的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为下一代繁殖子孙。根据各个个体的适应度值，按照一定的规则或方法从上一代群体中选择出一些优良的个体遗传到下一代群体中。遗传算法通过选择运算体现了这一思想，进行选择的原则是适应性强的个体为下一代贡献一个或多个后代的概率大。这样就体现了达尔文的适者生存的原则。

(5) 交叉：交叉是两个染色体之间随机交换信息的一种机制。以实现给定的杂交概率  $P_c$ 。在选择出的个体中任意选择两个个体进行交叉运算或重组运算，产生两个新的个体，重复此过程直到所有要求交叉的个体交叉完毕。交叉操作是遗传算法中最主要的遗传操作。通过交叉操作可以得到新一代个体，新个体结合了其父辈个体的特性。交叉体现了信息互换的思想。

(6) 变异：变异首先在群体中随机选择一个个体，对于选中的个体以一定的概率随机的改变串结构数据中某个串的值。同生物界一样，遗传算法中发生变异的概率很低，通常取值在 0.001-0.01 之间。变异为新个体的产生提供了机会。根据需要可以实现给定的变异概率  $P_m$  在最好的个体中选择若干个体，并按一定的策略对选中的个体进行变异运算。变异运算增加了遗传算法找到最优解的能力。

(7) 检验停机条件：若满足收敛条件或固定迭代次数则停机；若不满足条件则转 (3) 重新进行进化过程。每一次进化过程就产生新一代的群体。群体内

个体所表示的解通过进化最终达到最优解。

其计算流程图可用图 4-2 表示。

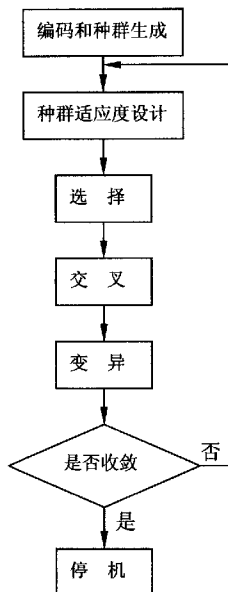


图 4-2 遗传算法的计算流程图

图 4-3 形象地说明了遗传算子的执行过程（图中 1、2、3、4、5、6、7、9 表示群体中的个体）。



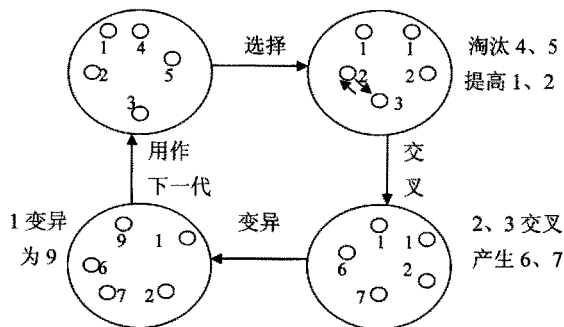


图 4-3 遗传算子的执行过程

#### 4.1.3 遗传算法的特征

遗传算法是基于自然选择和基因遗传学原理的搜索算法，与传统的优化算法如梯度法、动态规划法、分枝定界法等相比，遗传算法主要有以下几点不同之处<sup>[26]</sup>：

(1) 适用范围广：遗传算法不依赖于问题的具体领域，提供了一种求解复杂系统优化问题的通用框架，它并不是直接作用在参变量上，而是利用某种编码将优化问题的决定因素控制参数编码成长度有限、代码集有限的串，并且在编码串上进行操作，从中找出高适应值的串。用遗传算法解决问题不需要了解问题的内在性质，对所求问题没有太多的数学要求，不受函数约束条件（如连续性、导数存在、单极值等）的限制。因此用传统方法很难解决的问题，遗传算法都能解决，广泛应用于许多学科。

(2) 本质并行性：由于采用群体（一组编码）的方式组织搜索，可以同时搜索多个区域，因此遗传算法特别适合于大规模并行，有较好的全局搜索性能。

(3) 智能性：遗传算法的智能性包括自组织、自适应和自学习等。由于采用基于自然选择的策略，适应值大的个体具有较高的生存概率，进行交叉和变异等遗传操作时就可能产生与环境更适应的后代。

(4) 集中性：遗传算法首先在解空间中搜索期望值高的区域，它优先把适应性最好的作为母体样本串，适应度优于平均的串在下一代得到更好的发展。某一区域串数量的增加必然表明该区域适应度高。

(5) 遗传算法的结构式开放的，与问题无关，很容易与其它算法综合。

(6) 鲁棒性强：遗传算法用种群作为基本单位，采用 3 个不同作用的基本算子进行搜索，解随时间的增加趋于稳定，不受初始解的影响，而且不因实例的不同而蜕变。

(7) 遗传算法通过目标函数来计算适应值，而不需要其他的推导和附加信息，从而对问题的依赖性较小。

(8) 遗传算法利用了概率的转移规则，而不是确定性的规则。

## 4.2 应用遗传算法的关键

### (1) 串的编码方式

这本质是问题编码。一般把问题的各种参数用二进制进行编码，构成子串。但是研究证明，VRP 采用整数编码会取得良好的效果，所以在后面的编码方式设计时，采用整数编码方式。

### (2) 适应函数的确定

适应函数也称对象函数，这是问题求解品质的测量函数；往往也称为问题的“环境”，一般可以把问题的模型函数作为对象函数，但有时候需要另行构造。

### (3) 遗传算法自身参数设定

遗传算法自身参数有 4 个，即群体大小  $N$ 、遗传运算的终止进化代数  $T$ 、交叉概率  $P_c$  和比变异概率  $P_m$ 。群体大小  $N$  太小时难以求出最优解，太大则增加收敛时间，一般取  $N=10-160$ ；遗传运算的终止进化代数  $T$  一般取 100-500；交叉概率  $P_c$  太小时难以向前搜索，太大则容易破坏高适应值的结构，一般取  $P_c=0.4-0.99$ ；变异概率  $P_m$  太小时难以产生新的基因结构，太大时使遗传算法成了单纯的随机搜索，一般取  $P_m=0.01-0.2$ 。

## 4.3 遗传算法的改进策略

在基本遗传算法的研究中，主要目标之一就是使设计的算法是稳健的，即广泛适用于多种问题。尽管基本遗传算法是稳健的，但是就任何一个特殊领域而言，遗传算法一般不是最成功的最优化算法，它们往往比不上专门处理该领域问题的算法，那么怎样才能使遗传算法在实际中得到应用呢？近些年来，众

多学者一致致力于推动遗传算法的发展,对编码方式、控制参数的设定、选择方式和交叉、变异机理等进行了深入的探讨,引入了动态策略和自适应策略以改善遗传算法的性能,提出了各种变形的遗传算法,目的在于使其性能超过基本遗传算法。其基本途径概括起来有以下几个方面:

(1) 改进遗传算法的组成成分或使用技术,如选用优化控制参数、适合问题特征的编码技术等;

(2) 采用混合遗传算法;

(3) 采用非标准的遗传操作算子;

(4) 采用并行遗传算法。

同时,由于实际中很多问题带有一些约束条件,比如本文研究的 VRP 中就有车辆的装载量不大于车辆的最大承载量这一约束条件。为了解决如何满足约束的问题,近年来也有一些学者提出了一些遗传算法的改进策略,主要分为以下几类:

(1) 拒绝策略

拒绝策略抛弃进化过程中产生的不可行的染色体,这是遗传算法中普遍的做法。当可行的搜索空间是凸的且为整个搜索空间的适当的一部分时,这种做法是有效的,然而这样的条件也是比较苛刻的。例如对许多约束优化条件初始种群可能是由非可行染色体构成的,这就需要对它们进行修补。对于某些系统,允许跨过不可行染色体使用修复往往更能达到最优解。

(2) 修复策略

修复染色体使对不可行染色体采用修复程序使之变为可行的。对于许多组合优化问题,构造修复程序比较容易。已经证明,对于一个有多个不连通可行集的优化问题,修复策略在速度和计算性能上都远胜于其他策略。但是该方法的缺点是它对问题本身的依赖性,对于每个具体问题必须设计专门的修复程序,而对于某些问题,修复过程本身比原问题的求解更复杂。

(3) 改进遗传算子策略

解决可行性问题的一个合理办法是设计针对问题的表达式,以及专门的遗传算子来维持染色体的可行性。许多领域的实际工作者采用专门的问题表达式和遗传算子构成了非常成功的遗传算法,这是一个非常普遍的趋势。但是该方法遗传搜索受到了可行域的限制。

(4) 惩罚策略

上面的三种策略的共同优点是不会产生不可行解，缺点是无法考虑可行域外的点。对于约束严格问题，不可行解在种群中占的比例很大，这样将搜索限制在可行域内就很难找到可行解。惩罚策略就是在遗传搜索中考虑不可行解的技术。

构造带有惩罚项的适值函数一般有两种，一种是采用加法形式：

$$\text{val}(x) = f(x) + p(x)$$

其中， $x$  代表染色体， $f(x)$  是问题的目标函数， $p(x)$  是惩罚项。

对于极大化问题，则取：

$$\begin{cases} p(x) = 0, & \text{若 } x \text{ 可行} \\ p(x) > 0, & \text{其他} \end{cases}$$

对于极小化问题，则取：

$$\begin{cases} p(x) = 0, & \text{若 } x \text{ 可行} \\ p(x) < 0, & \text{其他} \end{cases}$$

另一种是乘法形式：

$$\text{val}(x) = f(x) * p(x)$$

此时，对于极大化问题，则取：

$$\begin{cases} p(x) = 1, & \text{若 } x \text{ 可行} \\ 0 \leq p(x) < 1, & \text{其他} \end{cases}$$

对于极小化问题，则取：

$$\begin{cases} p(x) = 1, & \text{若 } x \text{ 可行} \\ p(x) > 1, & \text{其他} \end{cases}$$

#### 4.4 遗传算法的应用领域和展望

目前，遗传算法在许多领域都有广泛的应用，例如：自适应控制（导弹控制）、工业优化控制（瓦斯管道控制）、规划设计（生产规划、并行机任务分配）、模式识别（模糊聚类）、组合优化（TSP 问题）、函数优化（经典应用领域）、图象处理（特征提取）、信号处理（滤波器设计）、机器学习（知识获取、知识优

化)、人工生命(生命的遗传进化)、遗传编程(遗传程序设计)、神经网络(学习算法的改进)等领域。

随着应用领域的扩展,遗传算法的研究出现了几个引人注目的新动向:

(1) 基于遗传算法的机器学习,这一新的研究课题把遗传算法从历来离散的搜索空间的优化搜索法扩展到具有独特的规则生成功能的崭新的机器学习算法。这一新的学习机制对于解决人工智能中知识获取和知识优化精练的瓶颈难题带来了希望。

(2) 遗传算法正日益和神经网络、模糊推理以及混沌理论等其它智能计算方法相互渗透和结合。这对开拓 21 世纪中新的智能计算技术将具有重要的意义。

(3) 并行处理的遗传算法的研究十分活跃。这一研究不仅对遗传算法本身的发展,而且对于新一代智能计算机体系结构的研究都是十分重要的。

(4) 遗传算法和另一个称为人工生命的崭新研究领域正不断渗透。所谓人工生命即是用计算机模拟自然界丰富多彩的生命现象,其中生物的自适应、进化和免疫等现象是人工生命的重要研究对象,而遗传算法在这方面将会发挥一定的作用。

(5) 遗传算法和进化规划(Evolution Programming, EP)以及进化策略(Evolution Strategy, ES)等进化计算理论日益结合。EP 和 ES 几乎是和遗传算法同时独立发展起来的,同遗传算法一样,它们是模拟自然界生物进化机制的智能计算方法,既同遗传算法具有相同之处,也有各自的特点。目前,这三者之间的比较研究和彼此结合的探讨正形成热点。

(6) 遗传算法在数据挖掘领域中的应用。数据挖掘是揭示存在于数据里的模式及数据间的关系学科,它强调对大量观测到的数据库的处理。它是涉及数据库管理,人工智能,机器学习,模式识别,统计学及数据可视化等学科的边缘学科。数据挖掘中常用的技术有人工神经网络、遗传算法、决策树、近邻算法和规则推导等。目前众多的学者研究如何有效的利用遗传算法方法来实现数据搜索。

## 4.5 本章小结

本章首先介绍了遗传算法的基本理论、算法流程。并指出了应用遗传算法应注意的关键问题和相应的改进措施,为下一章对遗传算法的改进设计做准备。



## 第 5 章 利用改进遗传算法求解 VRP

本章针对 VRP 及其约束条件建立数学模型,并对传统的遗传算法进行改进,用于 VRP 的研究中,最后用一组具体数据来说明此算法的可行性。

### 5.1 VRP 的数学模型

通常车辆路径问题的描述如下:有  $k$  个分仓库  $(1, 2, \dots, k)$ ,第  $i$  个分仓库的货运量为  $g_i(i=1, 2, \dots, k)$ ,由总仓库派出载重量分别为  $q_m$  的  $m$  辆车来承运,将所有货物运往总仓库。汽车从总仓库出发,最后回到总仓库。已知  $g_i \leq q_m$ ,求满足货运需求的最短行程路线。

也可描述为有一个配送中心,向  $k$  个顾客点送货。第  $i$  个顾客点的需求量为  $g_i(i=1, 2, \dots, k)$ ,由配送中心派出载重量分别为  $q_m$  的  $m$  辆车来承运。将货物运往各个顾客点,最后回到配送中心。已知  $g_i \leq q_m$ ,求满足各顾客点需求的最短行程路线。此两种表达方式是一样的。

VRP 有以下几个约束条件:

- (1) 车辆最大承载量约束:每辆车装载的货运总量不得超过车的最大承载量;
- (2) 每个分仓库仅由一辆车访问;
- (3) 对每辆车每天总的运行时间(或运行距离)有预定的上限;
- (4) 必须满足用户提出的到货时间要求;
- (5) 多个需求点之间的优先级关系,比如必须在访问  $i$  需求点之前访问需求点  $j$ 。

对某一具体问题,上述约束条件可能全部存在,也可能只存在一部分。

VRP 的最终目标:就是要使配送总成本最低,这里总成本包括固定成本和变动成本。减少固定成本,主要是要使得所使用的车辆数目最少。在下文我们将按照一定的方法来选择车辆数目。减少变动成本,主要是使得油费、路程中的机会成本最小,而此与行驶距离成正比,所以将变动成本最小简化为行驶距离最短。

为了安排路线，需要对使用的汽车数  $m$  有一个估计。在实际情况中，货物装（卸）车越复杂，约束条件越多，一辆车的实际载货量就会越小。我们可以使用文献[28]的公式来确定需要的汽车数：

$$m = [\sum g_i / \alpha q] + 1 \quad (5-1)$$

$[]$ 表示取整， $\alpha$  为参数， $0 < \alpha < 1$ ，约束条件越多，货物装（卸）车越复杂， $\alpha$  值越小。本文参考文献[24]， $\alpha$  取值为 0.85。

$c_{ij}$  表示点  $i$  到点  $j$  的运输成本，如时间、路程、花费等。总仓库的编号为 0，各分仓库的编号为  $i$  ( $i=1, 2, \dots, k$ )，各车的编号为  $s$  ( $s=1, 2, \dots, m$ )。定义变量如下：

$$x_{ijs} = \begin{cases} 1, & \text{车 } s \text{ 由 } i \text{ 驶向 } j \\ 0, & \text{否则} \end{cases} \quad (5-2)$$

$$y_{is} = \begin{cases} 1, & \text{点 } i \text{ 的货运任务由车 } s \text{ 完成} \\ 0, & \text{否则} \end{cases} \quad (5-3)$$

建立的模型的目标就是要使总的运输成本最小。而运输成本与车辆的行驶路径成正比，行驶路径越短，车辆的耗油量越少，司机的工作时间越少，总的运输费用也就越少。因此建立如下以行驶路径最短作为目标函数的数学模型：

$$\min Z = \sum_{i=0}^k \sum_{j=0}^k \sum_{s=1}^m c_{ij} x_{ijs} \quad (5-4)$$

$$\sum_{i=0}^k g_i y_{is} \leq q_s \quad \forall s \quad (5-5)$$

$$\sum_{s=1}^m y_{is} = \begin{cases} 1 & i=1, 2, \dots, k \\ m & i=0 \end{cases} \quad (5-6)$$

$$\sum_{i=0}^k x_{ijs} = y_{js} \quad j=1, \dots, k; \forall s \quad (5-7)$$

$$\sum_{j=0}^k x_{ijs} = y_{is} \quad i=1, \dots, k; \forall s \quad (5-8)$$

$$T_0^s + \sum_{i=0}^k \sum_{j=0}^k x_{ijs} (t_{ij} + S_i) \leq T_R^s \quad \forall s \quad (5-9)$$

$$t_i + S_i + t_{ij} - M(1 - x_{ijs}) \leq t_j \quad i=1, \dots, k; j=1, \dots, k; \forall s \quad (5-10)$$

上述模型中：

$q_s$  表示由车辆  $s$  的最大承载量；

$t_i$  为车辆到达仓库  $i$  的时间；

$t_{ij}$  为车辆从仓库  $i$  到仓库  $j$  的行车时间；

$S_i$  为车辆在仓库  $i$  处的停留时间；

$T_0^s$  为车辆  $s$  的出发时间；

$T_R^s$  为车辆  $k$  的要求返回时间。

式 (5-4) 为目标函数；

式 (5-5) 为汽车最大承载量约束：即每辆车装载的货运总量不得超过车的最大承载量；

式 (5-6) 保证了每个分仓库的运输任务仅由一辆车完成，而所有运输任务则由  $m$  辆车协同完成；

式 (5-7) 和 (5-8) 表示对任一由  $s$  服务的仓库  $j$ ，必定有另一（而且只有一个）由  $s$  服务的仓库（包括配送中心） $i$ ，车辆  $s$  从仓库  $i$  到达仓库  $j$ ，而对由  $s$  服务的仓库  $i$  同样存在由  $s$  服务的另一仓库，车辆  $s$  是从该仓库到达仓库  $i$  的，依此类推；

式 (5-9) 表示每辆车的行车路线的总耗时不超过一个事先定下的数值，比如有时候要求每辆车一天的行驶时间不超过某一极限；

式 (5-10) 对某个仓库，车辆达到时间限制在某一时间段内。比如有的需求点必须要求在上午 8:00-9:00 之间有车来访问，并且还规定了车辆在此处的停留时间。

本文考虑的一般 VRP，考虑上述约束条件式 (5-5)、(5-6)、(5-7) 和 (5-8)。

## 5.2 遗传算法的设计

在对算法进行改进之前，有必要先对照表 4-2 将下文将要提到的术语举例说明其含义。

例如 013402506780，称为一条染色体。染色体中每一个分量称为基因，上述染色体中的 0、1、2、3、4、5、6、7、8 都称为基因。染色体也叫个体，多

个个体组成的集合称为群体。

根据第三章介绍的遗传算法的基本理论，以及遗传算法应用的关键和改进的策略，对基本遗传算法做了如下改进：

### (1) 染色体编码

出于简单表示、计算机处理方便的目的，这里对染色体使用整数编码。0 代表总仓库，1, 2, ..., k 分别代表 k 个分仓库。染色体的长度为 k+m+1 (0,  $i_1$ ,  $i_2$ , ...,  $i_s$ , 0,  $i_j$ , ...,  $i_k$ , 0, ..., 0,  $i_p$ , ...,  $i_q$ , 0)，在整条染色体中，整数  $i_j$  表示第 j 个分仓库，0 的数目为 m+1 个，首尾都是 0，代表从总仓库派车出发，最后又回到总仓库。中间 m-1 个 0 把编码分为 m 段，形成 m 个子路径，表示由 m 辆车完成所有的运输任务。这样的染色体编码可以解释为：第一辆车从总仓库出发，经过  $i_1, i_2, \dots, i_s$  分仓库后，回到总仓库，形成子路径 1；第二辆车也从总仓库出发，经过  $i_j, \dots, i_k$  分仓库后，回到总仓库，形成子路径 2；m 辆车依次出发，完成所有的运输任务，构成 m 条子路径。

如染色体编码为 0123045067890，表示由 3 辆车完成 9 个分仓库的运输任务的路径安排：

子路径 1：总仓库→分仓库 1→分仓库 2→分仓库 3→总仓库

子路径 2：总仓库→分仓库 4→分仓库 5→总仓库

子路径 3：总仓库→分仓库 6→分仓库 7→分仓库 8→分仓库 9→总仓库

### (2) 初始化种群

为了能使算法收敛到全局最优，遗传群体应具有一定规模；但为了保证计算效率，群体规模也不能太大。一般群体规模取值在 10 到 100 之间。在初始化染色体时，先生成 k 个分仓库的一个全排列，再将 m+1 个 0 随机插入排列中。需要注意的是，必须要有 2 个 0 被安排在排列的头部和尾部，并且在排列中不能有连续的 2 个 0。这样就构成一条满足问题需要的染色体，重复这一过程，直至生成满足群体规模数的染色体。

### (3) 计算适应度函数

对一般的 VRP，我们应用 4.3 节中对约束条件的处理方法介绍中的惩罚策略，引入一个罚函数，将容量约束式 (5-5) 转为运输成本的一部分，运输成本变为：

$$Z = \sum_{i=0}^k \sum_{j=0}^k \sum_{s=1}^m c_{ij} x_{ijs} + M \sum_{s=1}^m \max(\sum_{i=0}^k g_i y_{is} - q, 0) \quad (5-11)$$

$$\text{式中, } M \sum_{s=1}^m \max(\sum_{i=0}^k g_i y_{is} - q, 0) \quad (5-12)$$

表示若违反容量约束处以的惩罚值。为了严格满足容量约束, 应有  $M \rightarrow \infty$ 。但考虑到计算机处理的不便,  $M$  可取一适当大的整数。

将运输成本转换为适应度函数:

$$f_i = z' / z \quad (5-13)$$

其中  $f_i$  为第  $i$  条染色体的适应度,  $z'$  当前最优染色体的运输成本,  $z$  为第  $i$  条染色体的运输成本。

最后计算染色体相对适应度:

$$p_i = f_i / \sum f \quad (5-14)$$

其中  $p_i$  表示第  $i$  条染色体的相对适应度,  $f_i$  表示第  $i$  条染色体的适应度,  $\sum f$  表示所有染色体的适应度总和。

#### (4) 判断停止进化条件

一般事先选取一个最大迭代次数, 当迭代次数大于设定的最大迭代次数时, 终止遗传算法。

#### (5) 遗传算子的设计

基于 VRP 问题的特征, 染色体中开始和结尾的基因都为 0 (因为要保证车辆从配送中心出发, 最后又要回到配送中心), 且在染色体中间不可能出现两个 0 代码连续出现的现象, 如果在操作过程中产生了这样的子代个体, 则视为是不合法的。为了避免在迭代过程中产生这种不合法的子代个体, 我们对遗传算子作了下面的设计:

##### ①选择算子

选择又称复制操作, 将每代种群的染色体中适应度最大的染色体直接复制, 进入下一代, 作为交叉、变异等遗传操作的对象。常用的选择方法有这样几种: 比例选择法、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。这些选择方法中最常用和最基本的选择方法就是比例选择法, 也叫赌盘选择法, 因为这种选择方式与赌博中的赌盘操作原理颇为相似。

如图 5-1 所示为赌盘示意图。整个赌盘被分为大小不同的一些扇面, 分别对



应着价值各不相同的一些赌博物品。当旋转着的赌盘自然停下来时，其指针所

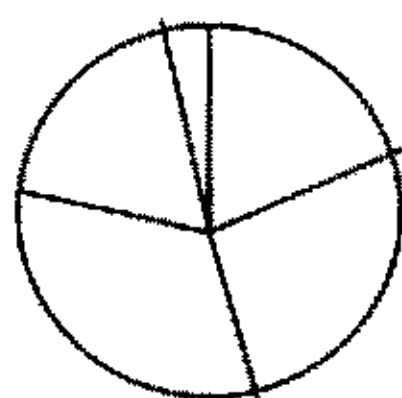


图 5-1 赌盘示意图

指扇面上的物品就归赌博者所有。虽然赌盘指针具体停止在哪一个扇面是无法预测的，但指针指向各个扇面的概率却是可以估计的，它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和中所占的比例（即相对适应度）也大小不一，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。

因此，如上所述：赌盘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适应度值成正比，个体的适应度值越高，被选中的概率就越大。但这并不保证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，其工作过程是这样的：

A、首先根据各个体的适应度计算出种群中所有个体的适应度总和（即  $\sum f$ ）；

B、其次计算出每个个体的相对适应度的大小，其值即为各个个体被选中的概率（即式（5-14）所算出的  $p_i$ ）；

C、最后再使用模拟赌盘操作（即[0, 1]之间的随机数）来确定各个个体被选中的次数。

## ②交叉算子

这里仿照类 PMX 方法。类 PMX 交叉过程如下<sup>[29]</sup>：假设有 2 个父代 A、B，A 为 12 | 4576 | 389，B 为 21 | 5437 | 869（“|”表示交叉位置）。先把 A 中的 4576 顺序赋给子体 A1 的前 4 个位置，然后用 B 中的元素逐个与 4576 相比，若相同则弃之不用，若不同，就将其顺序放在子体 A1 的后续位置，从而得到子体 A1 为 457621389，同样的方法可以得到子体 B1 为 543712689。该步骤进行交叉的 2 个个体来自前一步进行选择时产生的染色体，依类 PMX 方法，产生 2 个新

的子体作为下一代进行变异操作的父体。

然而在 VRP 这个特定问题的编码中, 每条染色体中都有  $m+1$  个 0 代码 (代表配送中心), 如果按照上述交叉产生了前文所述的不合法的子代, 那么将不能让它进入下一次迭代的群体中。

在此, 我们对类 PMX 交叉算子作了如下改进: 比如有 2 个父代 A、B, A 为 012 | 0340 | 560, B 为 023 | 4050 | 160 (“|”表示交叉位置)。按照上面的交叉方法, 得到的子体 A1 为 0340250160, B1 为 4050123060, 此时 B1 为不合法的子体, 我们不能让它进入下一次迭代的群体中。此时, 我们采取以下操作: 首先比较 B1 首尾是不是为 0 代码, 若有不为 0 的现象, 将中间任意一个 0 代码的基因与之互换。通过比较我们发现首位上的基因为 4, 不是 0 代码。于是我们随机将染色体中间的 0 代码与首位上的基因互换, 假设我们将第一个 0 代码和代码 4 互换, 则子体 B1 为 0450123060。这样 B1 则成为合法的子代个体, 可以进行下一步的操作。在保证染色体首尾的基因都为 0 代码的前提下, 若得到的子体中间出现连续两个 0 代码的现象, 则将其中一个 0 代码与任意一个位置上的非 0 代码互换, 多次执行该步骤, 直到新的基因串成为合法的子代个体。

这种经过改进的新的交叉算子避免了产生不可行解的情况, 在一定程度上保持了种群的多样性, 避免陷入局部最优解。

### ③变异算子

物种变异的可能性很小, 所以在遗传算法中变异操作只能起辅助作用。对每代种群以变异概率进行染色体变异。本文采用的是一种逆转变异<sup>[20]</sup>。具体过程是: 随机产生一染色体的两变异点, 将变异段进行逆转得到新的个体。例如: 032 | 046701 | 50 (“|”表示变异位), 变异后得到的新个体是 03210764050。同样的, 若变异后得到的新个体中出现连续两个 0 代码的情况, 则将其中一个 0 代码与任意位置上的非 0 代码互换, 多次执行该步骤, 直到新基因串成为合法的子代个体。例如: 0320 | 4670 | 150, 变异后得到的新个体为 03200764150, 此时出现两个 0 代码相连的情况, 则将其中一个 0 代码与任意一个位置上的非 0 代码互换, 多次执行该步骤, 直到新的基因串成为合法的子代个体。

以上是主要的遗传算法设计。本文采用 MATLAB 软件进行编程实现此算法。编程计算的主要步骤为 (gen 表示当前代数, i 表示个体计数器):

Step 1: 使用整数编码, 构造表示可行线路的染色体;

Step 2: 设置控制参数 (交叉率  $p_c$ , 变异率  $p_m$  和群体规模  $N$ )

Step 3:  $gen=0$ , 随即产生初始群体  $p(0)$ , 群体中包含  $N$  个染色体, 每个染色体表示一个路径方案;

Step 4: 若满足算法终止条件, 则输出结果, 停止, 否则继续;

Step 5:  $i=0$ ;

Step 6: 计算适应度;

Step 7:  $i=i+1$ ;

Step 8: 若  $i \leq N$ , 回到 Step 5, 否则转 Step 9;

Step 9: 根据适应度按赌盘法复制下一代染色体;

Step 10: 进行类 PMX 交叉、逆转变异;

Step 11:  $gen=gen+1$ ;

Step 12: 若满足算法终止程序, 则停止, 否则转 Step4。

用框图表示如图 5-2。

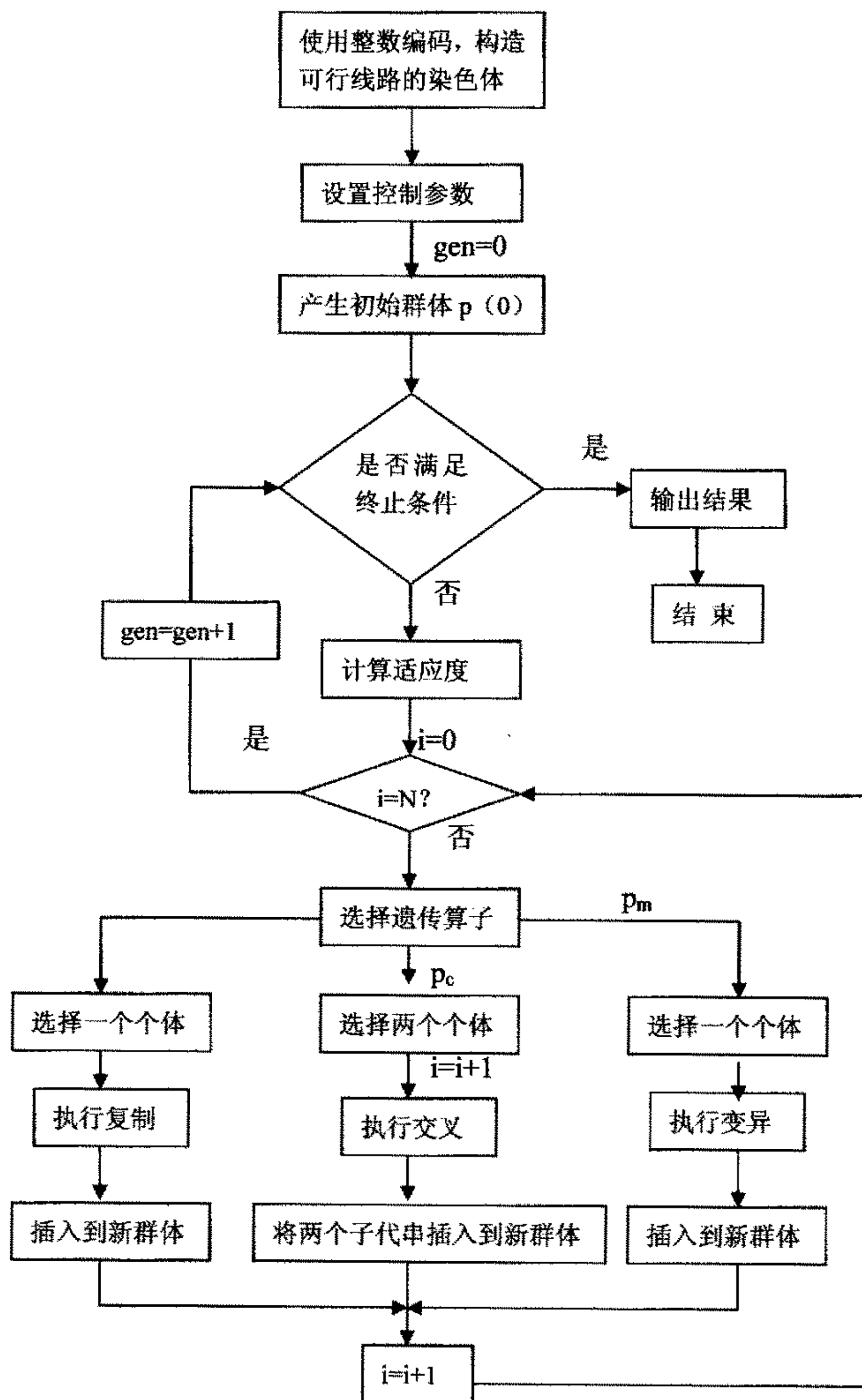


图 5-2 改进遗传算法计算流程

### 5.3 实证分析

某公司（用 0 表示）对公司在某一地区的整体销售量进行了统计分析，在对大约 100 多家客户通过排列评优、综合评定选出 10 家大客户（依次用 1, 2, …, 10 表示），它们之间的距离（km）如表 5-1 所示，各用户的需求量（kg）如表 5-2 所示。现要在保证车辆不超载，并且保证每家客户的送货量的前提下，找出对这 10 家客户进行送货的最短路径。

表 5-1 配送中心与客户之间以及各客户之间的距离（km）

Cij	0	1	2	3	4	5	6	7	8	9	10
0	0	3.5	19	5	4	25	12	7.5	25	17	11.5
1	3.5	0	18.5	3	3	27	13	9	24	17	18.5
2	19	18.5	0	19	18	14.5	16	11	12.5	16.5	7
3	5	3	19	0	1.5	28.5	17	11	23	22	23
4	4	3	18	1.5	0	26.5	16	11.3	22	21	18.5
5	25	27	14.5	28.5	26.5	0	15	17	12	10	2.5
6	12	13	16	17	16	15	0	14.5	14	5	6.5
7	7.5	9	11	11	11.3	17	14.5	0	17	10	8.5
8	25	24	12.5	23	22	12	14	17	0	15.5	9
9	17	17	16.5	22	21	10	5	10	15.5	0	4
10	11.5	18.5	7	23	18.5	2.5	6.5	8.5	9	4	0

表 5-2 各客户的需求量 g（kg）

客户 配送点	1	2	3	4	5	6	7	8	9	10
g	133	310	358	125	306	332	334	300	179	273

已知配送车辆的最大承载量为 1 吨。为了安排路线，需要先对车辆数  $m$  进行评估。在 5.1 节中已经说明如何对  $m$  的评估，按照此方法，我们评估的  $m = [\sum g_i / (\alpha q)] + 1 = [2316 / (0.85 * 1000)] + 1 = 3$ 。

本文的遗传算法参数设置为：选择整数编码，种群中的个体数目为 10，最大代数数为 100，交叉概率为 0.9，变异概率为 0.04。算法主要程序见附录。

最后得到车辆的行驶路线为 0—4—3—1—7—0；0—2—8—5—0；0—6—9—10—0，总的行驶距离为 126km。经检验，此路径安排完全满足各用户的需求量约束和车辆的承载量约束，是此问题的一个较优的可行解。

## 5.4 本章小结

本章针对 VRP，对遗传算法进行了改进。具体的改进措施有：对染色体进行整数编码、引入罚函数对车辆容量约束进行处理、对类 PMX 交叉算子进行了改进、采用逆转变异算子并对由此操作产生的不合法个体进行相关操作，直到得到合法的子代个体才能进行下一次迭代。这样避免了产生不可行解的可能性，增加了群体的多样性。利用 MATLAB 进行编程计算，以一个具体实例说明了该算法的可行性。



## 第 6 章 全文总结与研究展望

### 6.1 本文研究工作总结

本文结合物流配送系统中对车辆调度问题的要求，提出了本文的研究对象：物流配送中的车辆路径问题（VRP）。在进行本文的研究时，作者收集了国内外专家学者关于 VRP 研究的相关文献资料，进行分类、整理，作为研究的基础；通过介绍物流配送在整个物流系统中的意义以及我国物流配送的现状与发展，对整个配送系统进行了初步的规划和设计，同时说明了解决 VRP 的必要性和现实意义；然后针对 VRP 并考虑各种约束条件进行了数学建模；在求解 VRP 的方法上，采用了整数编码构造可行线路的染色体、对适值函数加入惩罚项解决约束问题、对类 PMX 交叉算子进行了改进、采用了逆转变异算子等方法对传统的遗传算法进行了改进。并用 MATLAB 编写程序实现了本文提出的算法，实证分析了该算法的可行性。

### 6.2 本文的创新点

（1）针对 VRP 的目标及其约束条件建立了数学模型；

（2）对传统的遗传算法进行了改进用于求解 VRP，主要对其中的交叉算子进行了改进，选取了适当的变异算子，用 MATLAB 编程实现了此算法。以一个具体实例作为实证分析的材料，结果显示了本文提出的算法的可行性，得到了满足各约束条件的路径。说明改进后的算法具有较好的全局寻优性能，能寻求到较优的可行解。

### 6.3 进一步研究的方向

由于时间有限，本文仅给出了简单 VRP 的算法研究，没有考虑一些带有更加复杂约束条件的 VRP。在以后的工作中，可以从以下几个方面完善对 VRP 的研究。

### (1) 同一用户的需求可由多辆车同时满足的情况

在实际配送中,可能会发生  $n$  个用户的需求综合恰好等于  $m$  辆车的车辆承载量之和,但由于每辆车的承载量和每个用户的需求量是不同的,这是在车辆承载量的限制下,可能需要  $m$  辆以上的车辆配送。如果一个用户允许一次被两辆或两辆以上的车访问,就可能减少车辆数。显然,一辆车的固定费用很大,可能比允许一个客户一次被多辆车访问少走路程所节省下来的费用还要大。目前在问题的研究中,还没有考虑到这一情况,而在实际配送中,特别在客户数量较多的情况下,这显示对车辆优化调度有着比较大的影响。

### (2) 随机型运输线路的优化

到目前为止,VRP 的研究大多都局限于确定型 VRP。而实际配送活动中,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等。因此,对于随机型的车辆调度问题的研究,也是非常重要的。

另外实现计算机配送管理系统的开发也是另一个研究的重点。

## 参 考 文 献

- [1] Dantzig, Ramser. The truck dispatching problem. Management Science, 1959. 6: 81~89
- [2] 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.
- [3] Balinski M, Quand R. On an integer program for a delivery problem [J]. Operations Research, 1962. 12: 300~304
- [4] Eilon S, Watson-Gandy CDT, Christofides N. Distribution management: mathematical modeling and practical analysis [M]. London: Griffin, 1971.
- [5] Gillett B, Miller L. A heuristic algorithm for the vehicle dispatch problem[J]. Operations Research, 1974. 22: 340~349
- [6] Christofides N, Mingozzi A, Toth P. Exact algorithms for the vehicle routing problem, based on spanning the shortest path relaxation [J]. Mathematical programming, 1981. 20: 255~282
- [7] Fisher M L. Optimal solution of vehicle routing problems using minimum k-trees [J]. Operation Research, 1994. 42 (4): 626~624
- [8] Gendreau M, Hertz A, Laporte G. A tabu search heuristic for the vehicle routing problem [M]. Montreal: Publication#777, Centre de recherche sur les transports, 1991.
- [9] Taillard E. Parallel interactive search method for vehicle routing problems [J]. Networks, 1993. 23: 661~673
- [10] Lawrence S, Mohammad A. Parametric experimentation with a genetic algorithmic configuration for solving the vehicle routing problem[A]. Proceedings-Annual Meeting of the Decision sciences Institute [C]. Decis Scil Inst. 1996. 488~490
- [11] Nicolas B, Pascal B. Optimization by hybridization of a genetic algorithm with constraint satisfaction techniques [A]. Proceedings of the IEEE Conference on Evolutionary Computation [C]. 1998
- [12] 张涛, 王梦光. 遗传算法和 3-OPT 结合求解带有能力约束的 VRP [J]. 东北大学学报 (自然科学版), 1999. 20 (3): 253~256
- [13] 肖鹏等. 车辆路径问题的单亲遗传算法 [J]. 计算技术与自动化, 2000. 19 (1): 26~30

- [14] 李嘉等. 一类特殊车辆路径问题 (VRP) [J]. 东北大学学报 (自然科学版), 2001. 22 (3): 245~248
- [15] 方霞等. 基于免疫遗传算法的物流配送车辆路径优化问题研究 [J]. 土木工程学报, 2003. 36 (7): 43~46
- [16] 纪寿文等. 货运车辆优化调度方法 [J]. 公路交通科技, 2003. 20 (6): 109~112
- [17] 王正彬, 杜文. 考虑线路安排的物流配送方案模型及其算法研究. 技术交流, 2003 (12): 72~73
- [18] 陈湘洲, 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. 25 (1): 36~41
- [19] 崔雪丽, 马良, 范炳权. 车辆路径问题 (VRP) 的蚂蚁搜索算法 [J]. 系统工程学报, 2004 (4)
- [20] 张兢, 周泉. 基于免疫遗传算法的物流配送车辆路径优化研究 [J]. 湖南大学学报 (自然科学版), 2004. 31 (5): 54~58
- [21] 杨善林, 凌海峰, 刘业政. 基于蚁群算法的混合方法求解车辆路径问题 [J]. 福州大学学报, 2005 (3): 14~17
- [22] 魏俊华, 王安麟. 基于分段遗传编码的配送路径优化 [J]. 工业工程与管理, 2005 (4): 85~88
- [23] 尹晓峰, 杜艳萍. 车辆路径问题的蚁群算法研究 [J]. 太原科技大学学报, 2005. 26 (4): 279~283
- [24] 邢文训, 谢金星编著. 现代优化计算方法 [M]. 北京: 清华大学出版社, 1999. 140
- [25] 王小平, 曹立明. 遗传算法—理论、应用与软件实现 [M]. 西安: 西安交通大学出版社, 2002.
- [26] 周明, 孙树栋. 遗传算法原理及应用 [M]. 北京: 国防工业出版社, 1996.
- [27] 汪寿阳, 赵秋红, 夏国平. 集成物流管理系统中的定位——运输路线安排问题的研究 [J]. 管理科学学报, 2000. 3 (6): 69~75
- [28] 李军, 谢秉磊, 郭耀煌. 非满载车辆调度问题的遗传算法. 系统工程理论与实践, 2000. 20 (3): 235~239
- [29] 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. 26 (12): 100~101
- [30] 卜心怡, 于涛. 物流配送路线模型优化及应用 [J]. 杭州电子科技大学学报, 2005. 25 (1): 93

- [31] 张智星. MATLAB 程序设计与应用 [M]. 北京: 清华大学出版社, 2002.
- [32] 郝渊晓主编. 现代物流配送 [M]. 广州: 中山大学出版社, 2001.
- [33][美] Z. 米凯利维茨. 演化程序——遗传算法与数据编码的结合(周家驹, 何险峰). 北京: 科学出版社, 2000.
- [34] 郎茂祥. 物流配送车辆调度问题的模型和算法研究 [D]. 北京: 北方交通大学, 2002
- [35] 张潜, 高立群等. 物流配送路径多目标的聚类—改进遗传算法. 控制与决策, 2003. 18 (4): 418~422
- [36] 李军, 谢秉磊, 郭耀煌. 基于自然数编码的模式理论研究. 西南交通大学学报, 2000. 35 (1)
- [37] 王小平, 曹立明. 遗传算法理论、应用与软件实现 [M]. 西安: 西安交通大学出版社, 2002.
- [38] 王沫然. MATLAB 与科学计算 [M]. 北京: 电子工业出版社, 2003.
- [39] 孟小平. 物流配送及其运输调度优化研究 [D]. 大连: 大连海事大学, 2001
- [40] 谢秉磊. 随机车辆路径问题研究 [D]. 成都: 西南交通大学, 2003
- [41] Arunkumar S, Chocklingam T. Genetic search algorithms and their randomized operators. Computers Math, 1996. 25 (12): 91~100
- [42] Agnetis A. Planning the routing to minimize total transportation time. International Journal of Flexible Manufacturing Systems, 1996. 16 (2): 131~157
- [43] 张晓贵, 方浩. 遗传算法的编码机制研究. 信息与控制, 1997. 26 (2): 134~139
- [44] 任庆生, 叶中行, 曾进. 交叉算子的搜索能力. 计算机研究与发展, 1999. 12 (11): 22~25
- [45] 席裕根, 柴天佑, 恽为民. 遗传算法综述. 控制理论与应用, 1996 (6): 26~28

## 致 谢

本篇论文是在郭燕导师的指导下完成的。论文从选题、文章构架、研究方法以及论文的若干次修改直到最后完稿，都得到了郭老师的悉心指导和大力支持，凝聚了郭老师大量心血。在此向郭老师表示衷心的感谢！

感谢室友杜娟对我学习和生活上的帮助！

感谢我的父母和亲人给予我强大的物质及精神上的支持！

在以后的学习和工作中我会加倍努力，回报恩师的教诲和亲友们的关心和厚爱！



## 附 录

### 1. 计算适应度的 MATLAB 代码:

```
function [sol,eval]=fitness(sol,options)
```

```
%距离矩阵
```

```
d=[0    3.5  19   5   4   25   12   7.5  25   17   11.5;
    3.5  0    18.5 3    3    27   13   9    24   17   18.5;
    19   18.5 0    19   18   14.5 16   11   12.5 16.5 7;
    5    3    19   0    1.5 28.5 17   11   23   22   23;
    4    3    18   1.5 0    26.5 16   11.3 22   21   18.5;
    25   27   14.5 28.5 26.5 0    15   17   12   10   2.5;
    12   13   16   17   16   15   0    14.5 14   5    6.5;
    7.5  9    11   11   11.3 17   14.5 0    17   10   8.5;
    25   24   12.5 23   22   12   14   17   0    15.5 9;
    17   17   16.5 22   21   10   5    10   15.5 0    4;
    11.5 18.5 7    23   18.5 2.5 6.5 8.5 9    4    0];
```

```
%需求矩阵
```

```
G=[0.133 0.310 0.358 0.125 0.306 0.322 0.334 0.300 0.179 0.273];
```

```
aaa=0;
```

```
bbb=0;
```

```
ccc=0;
```

```
for k=1:3
```

```
    for i=1:10
```

```
        for j=1:10
```

```
            if(aaa<=1)    %每辆车的总装载量小于1t（车的额定载重量）
```

```
                xijk=1;    %可以走i到j这条路
```

```
            else
```

```
                xijk=0;    %不可以走i到j这条路
```

```
            end
```

```

        bbb=bbb+xi jk;
        if(bbb==1)
            ykj=1;          %第j个客户由k车完成
        else
            ykj=0;
        end
        aaa=aaa+G(j)*ykj;%第k辆车的载重加上第j个客户的需求等于其
                           %新的载重
        ccc=ccc+ykj;
    end
end
end

if ((aaa-1<=0)&(ccc==1))
    fsum=0;
    for i=1:10
        for j=1:10
            fsum=fsum+d(i,j)*xi jk;
        end
    end
    eval=fsum;
else
    eval=-10000;          %惩罚
end
eval=-eval;

```

## 2. 初始化函数

```

function [pop]=initializega(num, bounds, evalFN, evalOps, options)
% initializega函数产生一个随机矩阵，矩阵的行数等于种群规模，此函数用
% 来产生初始群体
% 输入参数：
% num——种群中的个体数目

```

```
% bounds——代表变量上下界的矩阵
% evalFN——适应度函数
% evalOps——传递给适应度函数的参数
% options——选择编码形式
% 输出参数:
% pop——生成的初始群体

if nargin<5
    options=[1e-6 1];
end
if nargin<4
    evalOps=[];
end

if any(evalFN<48)
    if options(2)==1
        estr=[' x=pop(i,1); pop(i,xZomeLength)=', evalFN ';''];
    else
        estr=[' x=b2f(pop(i,:), bounds, bits); pop(i,xZomeLength)=', ...
            evalFN ';''];
    end
else
    if options(2)==1
        estr=[' [ pop(i,:) pop(i,xZomeLength)]=', evalFN '(pop(i,:), [0...
            evalOps]);''];
    else
        estr=[' x=b2f(pop(i,:), bounds, bits); [x v]=', evalFN ...
            '(x, [0 evalOps]); pop(i,:)=[f2b(x, bounds, bits) v];''];
    end
end
end
```

```

numVars      = size(bounds,1);          %变量的数量
rng          = (bounds(:,2)-bounds(:,1))'; %变量的范围

if options(2)==1
    xZomeLength = numVars+1;
    pop         = zeros(num, xZomeLength);
    pop(:,1:numVars)=(ones(num,1)*rng).*(rand(num, numVars))+...
        (ones(num,1)*bounds(:,1))';
else
    bits=calcbits(bounds,options(1));
    xZomeLength = sum(bits)+1;
    pop = round(rand(num, sum(bits)+1));
end

for i=1:num
    eval(estr);
end

```

### 3. 终止函数代码

```

function [done] = maxGenTerm(ops, bPop, endPop)
% 此函数若返回1, 代表达到最大迭代次数, 中止GA计算
% 输入参数:
% ops——选择向量[当前代数 最大迭代次数]
% bPop——最优解矩阵 [generation_found solution_string]
% endPop——当前解的代数

```

```

currentGen = ops(1);
maxGen     = ops(2);
done       = currentGen >= maxGen;

```

### 4. 用遗传算法求解的代码

%距离矩阵

```

d=[0    3.5  19   5    4    25   12   7.5  25   17   11.5;

```

```

3.5 0 18.5 3 3 27 13 9 24 17 18.5;
19 18.5 0 19 18 14.5 16 11 12.5 16.5 7;
5 3 19 0 1.5 28.5 17 11 23 22 23;
4 3 18 1.5 0 26.5 16 11.3 22 21 18.5;
25 27 14.5 28.5 26.5 0 15 17 12 10 2.5;
12 13 16 17 16 15 0 14.5 14 5 6.5;
7.5 9 11 11 11.3 17 14.5 0 17 10 8.5;
25 24 12.5 23 22 12 14 17 0 15.5 9;
17 17 16.5 22 21 10 5 10 15.5 0 4;
11.5 18.5 7 23 18.5 2.5 6.5 8.5 9 4 0];
%需求向量
G=[0.133 0.310 0.358 0.125 0.306 0.322 0.334 0.300 0.179 0.273];
bounds=[0 2;0 2;0 2;0 2;0 2;0 2;0 2;0 2;0 2;0 2];
flag=0;
while flag<10
    init=initializega(1,bounds,'fitness');
    aaa=0;
    bbb=0;
    ccc=0;
    for k=1:3
        for i=1:11
            for j=1:11
                if(sum2<=1) % 总重量小于1t
                    xijk=1; % 可以走这条路
                else
                    xijk=0; % 不可以走这条路
                end
                bbb=bbb+xijk;
                if (sum3==1)
                    ykj=1; % 第j个客户由k车完成
                else

```

```

        ykj=0;
    end
    aaa=aaa+G(j)*ykj; % 第k辆车的载重加上第j个客户的需求等于其新
                    % 的载重
    ccc=ccc+ykj;
end
end
end
if ((aaa-1<=0)&(ccc==1))
    flag=flag+1;
    initPop(flag,:)=init;
else
    continue
end
end
[p]=ga(bounds,'fitness',[],initPop,[1e-5 1 1],... 'maxGenTerm',100,...
    'normGeomSelect',[0.08],['arithXover'],[20], 'nonUnifMutation',[2 1
3])

```



## 攻读学位期间公开发表的论文

- [1] 盛倩蓉, 郭燕. 我国整车物流发展现状与发展策略[J]. 物流科技, 2005. 28(10)
- [2] 盛倩蓉. 电子商务的发展对物流配送的影响[J]. 物流科技, 已录用.