

中图分类号: F252  
学科分类号: 120100

论文编号: 1028709 12-s064

# 硕士学位论文

## 随机需求同时取送货车车辆路径问题建模及优化研究

研究生姓名	刘 晴
学科、专业	管理科学与工程
研究方向	物流管理
指导教师	刘 文 杰 副教授

南京航空航天大学

研究生院 经济与管理学院  
二零一二年三月

Nanjing University of Aeronautics and Astronautics

The Graduate School

College of Economics and Management

**Study on the Modeling and Optimization of  
Stochastic Demand Vehicle Routing Problem  
with Simultaneous Delivery and Pick-Up**

A Thesis in

Management Science and Engineering

by

Liu Qing

Advised by

Associate Prof. Liu Wen-jie

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Master of Management

March, 2012

# 承诺书

本人声明所呈交的硕士学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得南京航空航天大学或其他教育机构的学位或证书而使用过的材料。

本人授权南京航空航天大学可以将学位论文的全部或部分内 容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本承诺书）

作者签名：\_\_\_\_\_

日 期：\_\_\_\_\_

## 摘 要

随着资源短缺和环境污染问题的日益加重，废旧产品的回收再制造越来越受到我国政府的高度重视，国内许多企业开展了逆向物流工作。为了避免单独实施正向物流或逆向物流造成的车辆运输能力的无谓浪费，有效降低配送成本，逆向物流实施企业通常考虑同时实施新产品配送和废旧产品回收，形成了同时取送货车辆路径问题（VRPSDP）。在现有的同时取送货车辆路径问题研究过程中，通常假设客户取货需求和送货需求均是已知的。然而，现实生活中同时取送货车辆路径问题要复杂的多，产品取货需求通常具有一定的随机性，属于随机需求同时取送货问题（SDVRPSDP）。由于随机因素的引入，该问题求解变得十分复杂，成为困扰企业实施同时取送货的难点。因此，深入研究随机需求同时取送货车辆路径问题，成为逆向物流实施企业必须面对的一项十分紧迫的难题。

本文针对具有随机需求的同时取送货车辆路径问题开展研究，主要研究内容如下：

首先，简要介绍了研究的背景与意义，分析了国内外的研究现状以及研究存在的不足，阐述了论文的主要研究内容与创新点，提出了研究的主要框架。

其次，详细描述了同时取送货车辆路径问题VRPSDP的定义与分类，介绍和分析了目前VRPSDP问题的常用求解算法，重点介绍了本论文采用的蚁群系统算法，为研究奠定了理论基础。

第三，研究了随机需求同时取送货车辆路径问题（简称随机需求 VRPSDP 问题），构建了送货需求已知、取货需求随机的同时取送货车辆路径问题数学模型，提出以蚁群系统算法为主体、路径模拟抽样法与局部搜索为补充的混合蚁群系统优化算法（ACS-RSM）。通过案例仿真，对比分析了该算法与其它启发式算法，验证了算法的有效性和合理性，有效解决了随机需求 VRPSDP 问题的求解问题，获得较为理想的车辆路径优化方案。

第四，研究了带时间窗的随机需求 VRPSDP 问题，构建了送货需求已知、取货需求随机且带时间窗的同时取送货车辆路径问题数学模型。根据该问题中客户具有配送时间要求的重要特点，提出了改进的混合蚁群系统优化算法，设计了新的路径构建策略，解决了带时间窗的随机需求 VRPSDP 问题的求解问题。

**关键词：**同时取送货车辆路径问题； 蚁群系统； 随机需求同时取送货车辆路径问题；  
混合蚁群系统优化算法 ACS-RSM； 带时间窗的随机需求 VRPSDP 问题

## ABSTRACT

With the shortage of resources and environment pollution problem getting worse, our government pays more attention to the reuse of the used products, and many domestic enterprises develop the reverse logistics system. In order to avoid the waste of vehicle transportation capacity caused by the separate implementation logistics or reverse logistics, and to reduce the transportation cost, the enterprises often consider implementing the new products distribution and the used products recycling at the same time, thus formed vehicle routing problem with simultaneous delivery and pick-up(which is abbreviated VRPSDP). In existing research of VRPSDP, it is described that the customer's demands for pick-up and delivery are known. However, the pickup demand of the customers is usually stochastic in real world, which formed the stochastic demand vehicle routing problem with simultaneous delivery and pick-up(which is abbreviated SDVRPSDP). Due to random factors, how to solve the SDVRPSDP becomes very complex. Therefore, the study on the model and the algorithm of SDVRPSDP becomes a problem which the reverse logistics enterprises have to face.

The stochastic demand vehicle routing problem with simultaneous delivery and pickup is studies in this paper. The main contents are listed below:

Firstly, the research background and research signification of VRPSDP are introduced. Its current researches are analyzed and some existing problems are pointed out. The main contributes and creative points of this paper are elaborated and the research framework is presented.

Secondly, the general definition and classification of VRPSDP are described in detail. The classical algorithms to solve the problem are analyzed. The Ant Colony System which is used in this paper are introduced that lays a theoretical foundation for this research.

Thirdly, the stochastic demand vehicle routing problem with simultaneous delivery and pick-up is discussed and a mixed integer programming model is constructed. In order to solve this problem effectively, an Ant Colony System algorithm combining with Route Simulation Method and Local Search Method (which is abbreviated ACS-RSM) is proposed. The availability and effectiveness of this algorithm is proved by the case simulation in which the new algorithm and other algorithms are used.

Fourthly, the stochastic demand vehicle routing problem with simultaneous delivery and pick-up and time window is analyzed and a mixed integer programming model is constructed. Considering the characteristics that the vehicles need to service the customers in the given time, an improved ACS-RSM algorithm is established, in which the route construction rules are redesign. The problem is solved successfully.

**Keywords:** Vehicle Routing Problem with simultaneous Delivery and Pick-up; Ant Colony System; Stochastic Demand Vehicle Routing Problem with simultaneous Delivery and Pick-up; An Ant Colony System - Route Simulation Method; Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pick-Up with Time Windows

# 目 录

第一章 绪论 .....	1
1.1 研究背景和研究意义.....	1
1.2 国内外研究现状及述评.....	2
1.3 主要研究内容及创新点.....	4
1.3.1 主要研究内容.....	4
1.3.2 本文的创新点.....	4
1.4 论文的研究框架.....	5
1.5 本章小结 .....	6
第二章 同时取送货车车辆路径问题概述.....	7
2.1 同时取送货车车辆路径问题的定义.....	7
2.2 同时取送货车车辆路径问题的主要分类.....	7
2.3 同时取送货车车辆路径问题的常见求解算法.....	9
2.4 本章小结 .....	14
第三章 随机需求 VRPSDP 问题建模及优化研究.....	15
3.1 引言 .....	15
3.2 随机需求 VRPSDP 问题建模.....	15
3.2.1 随机需求 VRPSDP 问题的描述.....	16
3.2.2 基本假设.....	17
3.2.3 数学模型.....	17
3.3 混合蚁群系统优化算法 ACS-RSM 设计 .....	18
3.3.1 ACS-RSM 算法设计的难点及解决方案 .....	18
3.3.2 ACS-RSM 算法框架的组成及其功能 .....	20
3.3.3 ACS-RSM 算法框架整体运行思路及算法基本流程.....	22
3.4 算例分析 .....	25
3.4.1 随机需求 VRP 问题对比分析 .....	25
3.4.2 随机需求 VRPSDP 问题算例仿真.....	26
3.5 本章小结 .....	27
第四章 带时间窗的随机需求 VRPSDP 问题建模及优化研究.....	28
4.1 引言 .....	28
4.2 带时间窗的随机需求 VRPSDP 问题建模.....	28
4.2.1 带时间窗的随机需求 VRPSDP 问题描述.....	28

4.2.2 数学模型.....	28
4.3 改进的混合蚁群系统优化算法 ACS-RSM 设计 .....	30
4.3.1 路径构建规则相关因子设计.....	30
4.3.2 路径构建规则构建.....	31
4.4 算例仿真分析.....	32
4.4.1 带硬时间窗的 VRPSDP 问题算例对比分析.....	32
4.4.2 带软时间窗的 VRPSDP 问题的对比分析.....	33
4.4.3 带时间窗的随机需求 VRPSDP 问题.....	36
4.5 本章小结 .....	37
第五章 总结与展望 .....	38
5.1 全文总结 .....	38
5.2 研究展望 .....	38
参考文献 .....	39
致    谢 .....	41
在学期间的研究成果及发表的学术论文.....	42
附录 .....	43

## 图清单

图 1.1 VRPB、VRPBM 以及 VRPSDP 示意图 .....	2
图 1.2 研究技术路线图.....	5
图 2.1 同时取送货车辆路径问题示意图 .....	7
图 2.2 蚂蚁搜索食物过程.....	11
图 3.1 产品回收再利用示意图.....	16
图 3.2 RSM 算法的基本思想 .....	20
图 3.3 ACS-RSM 算法构架图 .....	20
图 3.4 路径内 2-opt 交换邻域示意图 .....	22
图 3.5 ACS-RSM 算法架构的具体算法流程图 .....	24
图 3.6 随机需求 VRPSDP 问题算例路径分布 .....	27
图 4.1 收敛结果曲线图.....	34
图 4.2 车辆路径方案图.....	35
图 4.3 最小路径收敛结果.....	36



## 表清单

表 3.1 客户的单组模拟抽样取、送货需求.....	20
表 3.2 算法在不同参数组合下的实验结果.....	25
表 3.3 不同算法对 50 个客户的随机需求 VRP 问题的寻优计算结果 .....	26
表 3.4 ACS-RSM 算法对随机需求 VRPSDP 问题的求解结果 .....	26
表 4.1 10 次随机实验的计算结果.....	33
表 4.2 不同的算法的结果比较.....	33
表 4.3 带软时间窗的 VRP 的客户点坐标、时间窗及送货需求客户点 .....	33
表 4.4 不同算法的结果比较.....	34
表 4.5 10 次实验的计算结果.....	35
表 4.6 10 次随机实验的计算结果.....	37
表 4.7 路径方案 .....	37

# 第一章 绪论

## 1.1 研究背景和研究意义

随着人们对环境保护及可持续发展认识的逐步深入，废旧产品的重新利用问题越来越受到重视。世界各国加大了有关废旧产品的立法力度，规定生产商和经销商负责回收其生产或销售的产品。目前，我国把发展循环经济作为实现可持续发展战略的重要途径和实现形式。为了解决经济发展和环境保护之间的矛盾，我国政府高度重视产品回收再利用工作并相继出台鼓励和促进产品回收再利用的法律、法规，加之产品回收再利用丰厚利润的诱惑，越来越多的企业开始进行产品回收业务。通过回收、翻新、维修等活动，企业可以很大程度的降低生产成本。因此，国内许多诸如汽车、电子等行业的制造商已经认识到回收逆向物流的经济效应和社会效应，并开始纷纷致力于如何优化自身的回收逆向物流系统以及正向物流系统以面对激烈的竞争环境。

回收逆向物流问题主要研究如何有效实现废旧产品的回收再利用，它与正向物流共同构成闭环物流，是实现循环经济的主要途径之一。物流运输是物流系统中的重要环节，其中，运输路线是否合理直接影响到配送成本和服务质量。实施产品回收的企业不仅要考虑产品的配送运输问题还要考虑废旧产品回收运输问题，除此之外，回收逆向物流比正向物流要复杂的多，具有不确定性和随机性等正向物流不具有的特征，回收企业往往无法预知产品回收的具体数量。企业若是沿用传统的物流配送模式，不仅反应速度慢导致服务质量的下降和物流运输成本的增加，同时也会增加城市交通负担和加重城市污染。在这样的情况下，需要研究一套新的与确定需求具有同时取送货的车辆路径问题（Vehicle Routing Problem with Simultaneous Delivery and Pick-up, VRPSDP）相对应的随机需求同时取送货的车辆路径问题（Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pick-Up, SDVRPSDP）的理论与方法。企业只有提高配送车辆的运输效率和保证服务水平才能提高企业的竞争力，因此，研究能够优质有效的为企业提供回收车辆路径优化方案具有现实意义和应用价值。

本文重点研究随机需求 VRPSDP 问题和带时间窗的随机需求 VRPSDP 问题（Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pick-up and Time Window, SDVRPSDPTW）。根据问题的特点，分别构建满足企业需求的数学规划模型，设计满足要求的混合蚁群系统优化算法和改进的混合蚁群系统优化算法，为企业提供优良的回收车辆路径优化方案，有效降低企业的运输成本，提高服务质量。该研究能够有效扩展车辆路径问题的研究领域及求解算法空间。

# 1.2 国内外研究现状及述评

逆向物流实施企业在回收产品时，为了降低配送成本，提高企业运作效率，通常统筹考虑新产品的配送及废旧产品回收问题（例如富士施乐爱科制造有限公司主要利用自建专营店或特约经销商实现新产品的销售与废旧产品回收），从而避免了单独运作正向物流或逆向物流造成车辆能力的无畏浪费。因此，基于正向物流与逆向物流的不同关系，延伸出了三种形式的回收车辆路径问题：回程取货车辆路径问题 VRPB（Vehicle Routing Problem with Backhauls）、混合取送货车辆路径问题 VRPBM（Vehicle Routing Problem with Backhauls of Mixed Loads）以及同时取货和送货的车辆路径问题 VRPSDP（Vehicle Routing Problem with Simultaneous Delivery and Pick-up），其示意图如图 1.1 所示。VRPB 和 VRPBM 的典型特征是客户分为送货客户和取货客户两类，VRPB 研究中取货总是发生在所有送货完成之后，VRPBM 研究中取货可以发生在所有送货完成之前；VRPSDP 则是客户可能同时具有取货和送货需求，取货和送货同时发生。

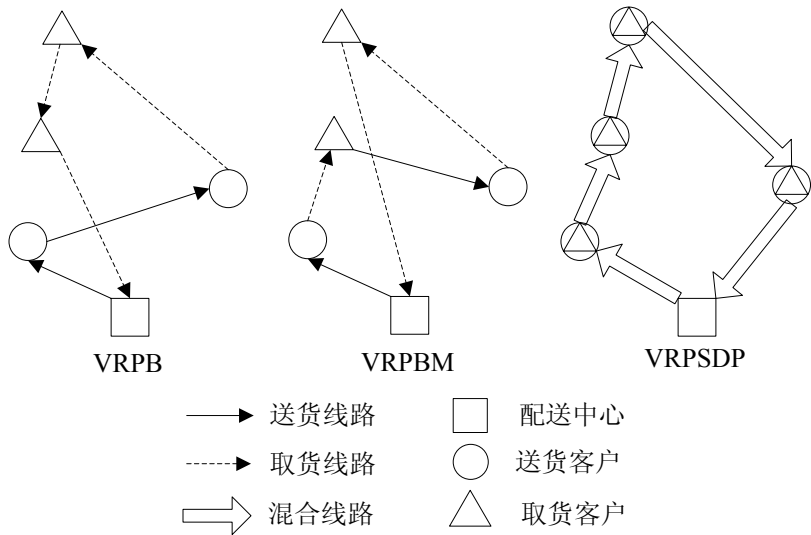


图 1.1 VRPB、VRPBM 以及 VRPSDP 示意图

## （1）VRPB 和 VRPBM 问题研究

目前，回收车辆路径问题研究主要集中在 VRPB 和 VRPBM 两个方面。依据解决方案精确程度的不同，VRPB 问题的研究方法主要划分为精确求解算法<sup>[1-3]</sup>和启发式算法<sup>[4-11]</sup>，其中文献<sup>[1-3]</sup>分别采用集合覆盖算法、基于拉各朗日松弛与割平面的分枝界定法和线性规划松弛算法研究了小规模 VRPB 问题的精确解；启发式算法主要包括构造式启发<sup>[4,5]</sup>（如插入法、插入与委派结合法等）、元启发<sup>[6-9]</sup>（如禁忌搜索、局部搜索等）以及两阶段启发<sup>[10,11]</sup>（如先聚类客户后安排配送路线法、先采用 K 树拉各朗日松弛后 3-opt 改进法以及先遗传算法后整数规划法等）三种形式，利用启发式规则通常获得大规模 VRPB 问题的近似优化解。针对 VRPBM 问题，文献<sup>[12-15]</sup>主要采用构造式启发算法进行了研究，其研究思路为：首先，采用传统 VRP 求解算法获得仅有送货

客户的配送路线；然后，基于一定的插入准则将取货客户逐一插入到已有的配送路线中，在插入过程中利用惩罚函数确保取货客户尽量安排在配送线路的后端。

## （2）VRPSDP 问题研究

VRPSDP 问题由 Min<sup>[16]</sup>于 1989 年首次提出，此后十几年未有学者对其加以研究。直到进入 21 世纪，各国政府纷纷颁布延伸生产商责任的相关法令，供应新产品并同时取回废旧产品成为一种新的要求，VRPSDP 问题才重新引起学者的关注。Dethloff<sup>[17]</sup>从回程概念出发构建了 VRPSDP 问题的数学模型，并提出了基于旅行距离、剩余装载能力、径向距离惩罚以及综合剩余装载能力四种插入准则的构造式启发算法；Salhi 和 Nagy<sup>[18]</sup>从路径可行性程度的角度出发，采用局部搜索元启发式算法求解 VRPSDP 问题；Tang & Galvão<sup>[19]</sup>采用禁忌搜索元启发式算法求解了 TSPSPD 问题(该问题是 VRPSDP 在旅行商问题 TSP 的特殊表现形式)；Yuvraj 和 Prakash<sup>[20]</sup>则采用蚁群算法探讨了 VRPSDP 的求解问题。我国的 VRPSDP 问题研究起步较晚，2004 年广州大学谢如鹤<sup>[21]</sup>在国内首次研究了该问题，提出了基于剩余装载能力插入准则的构造式启发算法，实现了车辆剩余装载能力与车辆行驶距离的紧密有效的结合。此后，研究得到不断深入。陆琳和谭清美<sup>[22]</sup>提出了感应因子、期望程度因子、距离性比因子以及加速因子的概念，构建了自感应蚁群算法求解回收车辆路径问题 VRPSDP，并通过仿真实验证明了自感应蚁群算法的有效性；张涛和田文馨等<sup>[23]</sup>则针对具有车辆最大行驶距离限制的 VRPSDP 问题，建立了混合整数规划模型，设计了考虑车辆负载使用率的启发式因子，并采用结合排序蚂蚁系统和最大最小蚂蚁系统的改进蚁群算法求解该回收车辆路径问题。此外，文献<sup>[24-27]</sup>主要采用具有自适应性的混合遗传算法研究了 VRPSDP 问题，文献<sup>[28]</sup>进一步设计了粗粒度并行遗传算法求解 VRPSDP 问题，文献<sup>[29]</sup>则采用协同粒子群-模拟退火算法求解 VRPSDP 问题。

## （3）VRPSDPTW 问题研究

在现实生活中会经常出现带时间窗的同时取送货车辆路径问题 (Vehicle Routing Problem with Simultaneous Delivery and Pick-up and Time Window, VRPSDPTW)，但有关 VRPSDPTW 的研究才刚刚起步。Angeles<sup>[30]</sup>等人采用运用定界法求解了带时间窗的送货和取货车辆路径问题 (VRPSDPTW)；Ropke<sup>[31,32]</sup>等人运用精确算法和邻域搜索算法求解 VRPSDPTW。目前国内有关 VRPSDPTW 的研究很少，DENG Ai-Min<sup>[33]</sup>采用改进的模拟退火算法来求解带软时间窗的同时取送货车辆路径问题 (VRPSDPTW)；曹二保和赖明勇<sup>[34]</sup>针对 VRPSDPTW 设计一种随进化代数自动更新的交叉率，采用改进的差分进化算法来求解该回收车辆路径问题。

## （4）相关文献述评

总结国内外的研究现状，可以得出在 VRPSDP 问题以及 VRPSDPTW 问题研究方面，国内外学者取得了一定的研究成果，为本课题的研究提供了基础和借鉴。然而，相对于颇具研究成果的 VRPB 和 VRPBM 问题，目前国内外有关 VRPSDP 问题和 VRPSDPTW 问题的研究相对较少且研

究不够深入，具体表现为：在构建VRPSDP问题和VRPSDPTW问题的数学模型时，现有研究均假设取货数量信息已知。然而，在现实生活中由于产品回收本身具有分散性与复杂性，客户的废旧产品取货需求呈现一定的随机性。只有当车辆抵达时才能获知确切的取货数量，这经常导致路径失败Route Failure的发生，从而增加无效的车辆行驶距离以及配送时间。面对随机需求，如何制定合理的车辆路径方案成为困扰回收企业的难题。

VRPSDP问题本身就是NP难的问题，而随机性又增加了决策的复杂性和难度。目前，尚未发现研究随机需求的VRPSDP问题的相关报道。为了切实解决逆向物流实施企业面临的难题，构建新的VRPSDP模型并提出高效的优化算法必将成为未来研究的一项重要工作。

## 1.3 主要研究内容及创新点

### 1.3.1 主要研究内容

本文的第一章阐述了本文的研究背景和研究意义，介绍了关于同时取送货车辆路径问题的国内外研究现状和研究思路，以及本文各章的主要研究内容。

第二章对具有同时取送货车辆路径问题进行了概述。描述了同时取送货车辆路径问题的定义以及介绍了同时取送货车辆路径问题的分类，为本文的进一步研究打下了理论基础。并对求解同时取送货车辆路径问题的常见方法进行了较为详细的归纳和总结。本文特别对蚁群系统算法进行了较为详细的介绍，为新算法的设计提供了理论基础。

第三章对随机需求VRPSDP问题进行建模研究，构建了该类问题的数学模型，设计了混合蚁群系统优化算法ACS-RSM，并通过MATLAB对算法进行仿真实验，验证所建数学模型及新算法的有效性和合理性。

第四章对带时间窗的随机需求VRPSDP问题进行研究，在合理的问题目标和假设条件的基础上，建立带时间窗的随机需求VRPSDP数学模型，并根据该问题的特点设计了更适合于求解该类问题的路径构建规则，最后通过MATLAB对算法进行实验仿真，获得相应的回收路径优化方案。

第五章对本文所做的研究工作进行简要的归纳和总结，指出了本文研究所存在的不足，并提出今后所需进一步研究的方向和问题。

### 1.3.2 本文的创新点

本文针对随机需求VRPSDP问题设计了以蚁群系统算法ACS为主体、路径模拟抽样法RSM和局部搜索为补充的混合蚁群系统优化算法ACS-RSM。该算法的特点是：采用路径模拟抽样法RSM合理估算车辆路径方案的期望行驶总距离，有效的解决客户较多且客户需求随机这种复杂环境下期望行驶总距离难以计算的问题。

### 1.4 论文的研究框架

研究技术路线如下图所示。

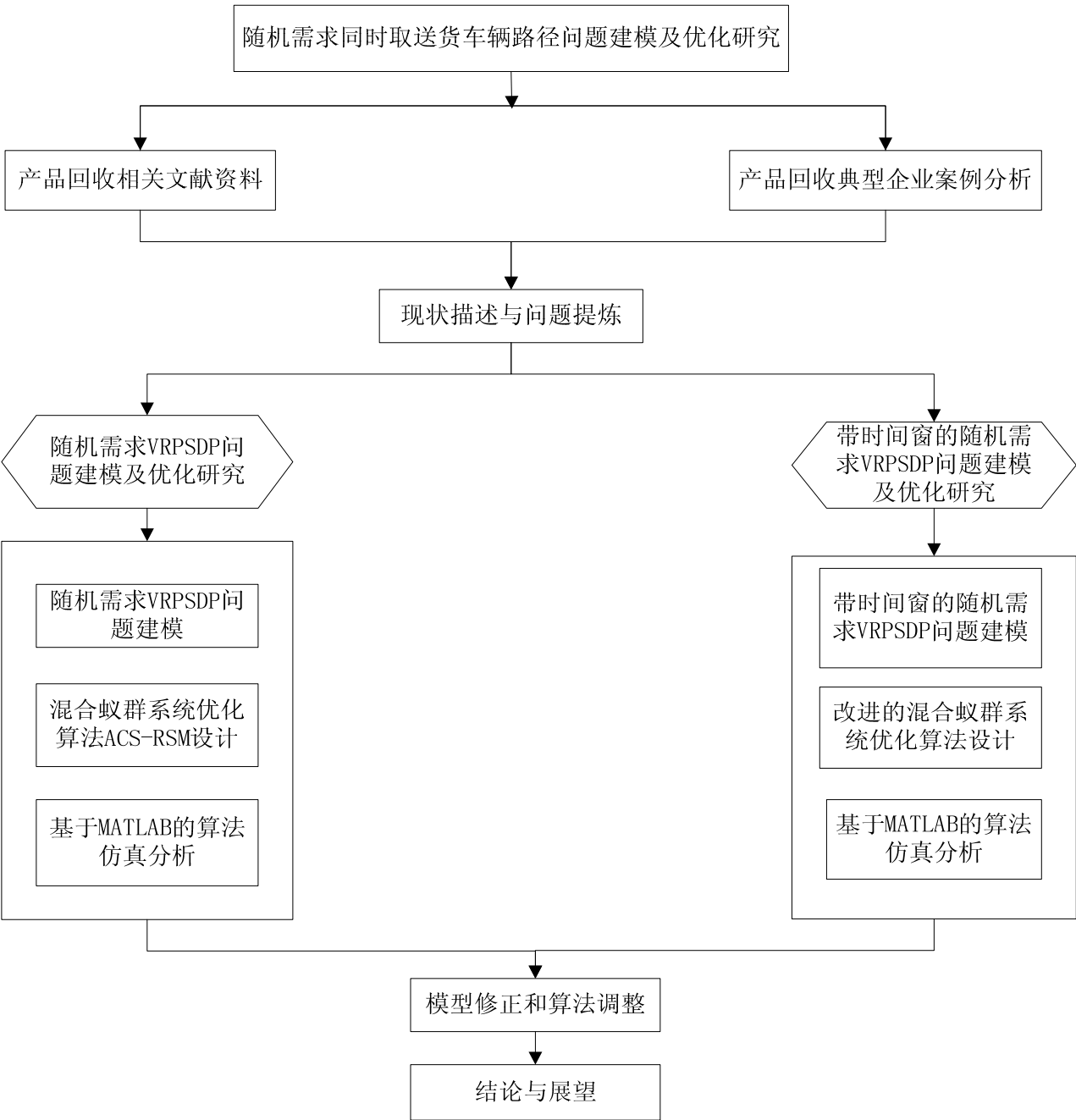


图 1.2 研究技术路线图

本文的研究思路如图1.2所示，基本思路是：首先，在收集国内外产品回收相关文献资料的基础上，结合国内产品回收企业的典型案例，凝练课题研究的两类车辆路径主要问题，即随机需求VRPSDP问题和带时间窗的随机需求VRPSDP问题。其次，针对提出的两类车辆路径问题，分别做出合理的假设，构建相应的数学模型并设计合理的混合蚁群系统优化算法。最后，以此

为基础使用MATLAB软件实现编程，对企业的案例进行仿真求解，通过仿真实验来验证本文所提出算法的有效性，进而对课题模型修正以及进行算法调整。

## 1.5 本章小结

本文首先阐述了该文章的研究背景及研究意义，介绍了国内外有关同时取送货车辆路径问题的研究现状，根据同时取送货车辆路径问题的研究现状引出本文的研究对象。最后，概述了本论文的主要研究内容、创新点以及论文框架。

## 第二章 同时取送货车车辆路径问题概述

### 2.1 同时取送货车车辆路径问题的定义

同时取送货车车辆路径问题 VRPSDP 是运筹学中一类经典的优化组合问题。该问题由 Min.H 于 1989 年首次提出，主要是在配送车辆数确定和车辆最大负载能力等有限约束条件下，研究 1 个图书中心与 22 个地方图书馆之间图书同时发送与回库的车辆路径方案优化问题，目标是寻求配送成本的最小化。此后的十多年间，针对该问题的研究相对较少。直到进入 21 世纪，随着产品回收问题越来越受到企业和政府的关注，对该问题的研究才重新获得专家学者和企业管理者的高度重视，成为运筹学和组合优化领域的前沿和研究热点问题。

同时取送货车车辆路径问题 VRPSDP 一般可以定义为：对一系列给定的同时具有取货和送货需求的客户（当然取货需求或送货需求可以为零），在满足一定的约束条件下（例如配送车辆数量、配送车辆最大行驶里程、车辆最大负载能力以及客户服务时间等约束），确定优化的配送车辆行驶路径方案，使配送车辆从配送中心出发依次服务各个客户节点，最终返回配送中心，使预期配送目标最优化（譬如说车辆行驶里程最短、车辆配送成本最小以及配送时间最短等）。同时取送货车车辆路径问题 VRPSDP 的示意图如图 2.1 所示。

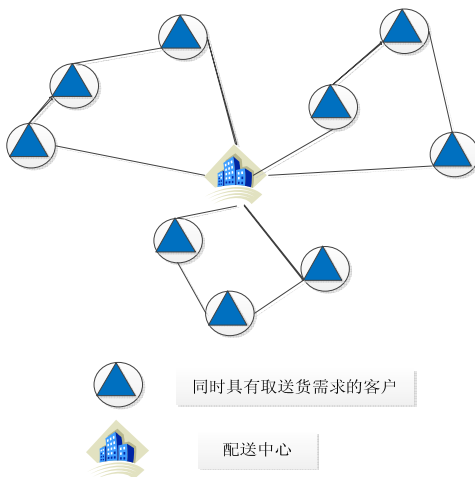


图 2.1 同时取送货车车辆路径问题示意图

### 2.2 同时取送货车车辆路径问题的主要分类

自上个世纪 80 年代末同时取送货车车辆路径问题 VRPSDP 被提出以来，由于企业实际发展的需要，该问题扩展形成了许多不同形态。根据客户信息是否确定、是否带时间窗以及配送车辆与配送中心关系等分类标准的不同，VRPSDP 问题分为以下几类：



### (1) 确定性 VRPSDP 问题和随机性 VRPSDP 问题

根据客户配送需求数量、客户配送所需时间等信息是否事先预知, 可以将同时取送货车辆路径问题分为确定性同时取送货车辆路径问题和随机性同时取送货车辆路径问题。对于随机性同时取送货车辆路径问题来说, 由于在构造车辆路径之前, 无法完全提前得知部分相关信息, 所以问题的优化目标通常为期望总成本最小。相对于确定性同时取送货车辆路径问题, 随机同时取送货车辆路径问题要复杂的多, 求解更加困难。目前, 随机性 VRPSDP 问题主要分为以下三类:

#### 1) 随机需求 VRPSDP 问题

随机需求 VRPSDP 问题 (Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pick-up, 简称 SDVRPSDP) 是指已知需要服务的客户的具体位置, 但是无法提前得知其准确的需求量, 仅知道顾客的需求量服从某种已知的概率分布。随机需求同时取送货车辆路径问题是随机同时取送货车辆路径问题当中研究最为广泛且深入的问题, 这类问题在日常活动中有着广泛的应用, 如啤酒和饮料配送、邮局向具有“随机需求”的用户分发货币等。

#### 2) 随机顾客 VRPSDP 问题

随机顾客同时取送货车辆路径问题 SCVRPSDP (Stochastic Customers Vehicle Routing Problem with Simultaneous Delivery and Pick-Up, 简称 SCVRPSDP) 是指在一个计划期内, 并不需要服务所有的潜在客户, 而是根据由统计资料得到的概率服务某些客户对象, 这类问题经常出现与消防、安全、医疗等公共事业单位和邮局、运输等服务行业。

#### 3) 随机旅行时间 VRPSDP 问题

由于道路的拥挤程度、天气变化、交通事故等情况经常导致车辆行驶速度的变化, 进而导致车辆在两客户间的路段上的行驶时间是随机的, 在这样的情况下, 各客户节点之间可以考虑为具有某种统计规律的随机变量, 这样的问题称为随机旅行时间的同时取送货车辆路径问题 STVRPSDP (Stochastic Time Vehicle Routing Problem with Simultaneous Delivery and Pick-Up, 简称 STVRPSDP)。

### (2) 带时间窗的 VRPSDP 问题和无时间窗的 VRPSDP 问题

根据客户服务有无时间限制, 可以将同时取送货车辆路径问题分为无时间窗的同时取送货车辆路径问题和带时间窗的同时取送货车辆路径问题 VRPSDPTW (Vehicle Routing Problem with Simultaneous Delivery and Pick-Up with Time Windows, 简称 VRPSDPTW)。VRPSDPTW 是同时取送货车辆路径问题 VRPSDP 的一个重要的拓展, 在这类问题中, 每个客户都有一个与之相对应的时间窗, 车辆需要在顾客点的时间窗口内完成服务任务。带时间窗的同时取送货车辆路径问题又分为带硬时间窗的 VRPSDP 问题 (Vehicle Routing Problem with Simultaneous Delivery and Pick-Up with Hard Time Windows, 简称 VRPSDPHTW) 和带软时间窗的 VRPSDP 问

题 (Vehicle Routing Problem with Simultaneous Delivery and Pick-Up with Soft Time Windows, 简称 VRPSDPSTW) 两类, 其中 VRPSDPHTW 问题是指车辆必须在客户相应的时间窗内完成任务, 客户拒绝接受时间窗之外提供的服务; 而 VRPSDPSTW 问题则不同, 是指车辆可以不在时间窗口内完成任务, 若超出客户的服务时间窗口, 则对目标函数设置一定的惩罚。

### (3) 开放式 VRPSDP 问题和封闭式 VRPSDP 问题

根据车辆对车场的所属关系, 同时取送货车辆路径问题分为开放式同时取送货车辆路径问题 (Open Vehicle Routing Problem with Simultaneous Delivery and Pick-Up, 简称 OVPRSDP) 和封闭式同时取送货车辆路径问题。开放式同时取送货车辆路径问题是指不要求车辆返回车场, 或者如果返回车场, 车辆则按原路返回。封闭式同时取送货车辆路径问题 OVPRSDP 是指要求车辆在完成配送任务后必须返回其发出车场。

根据上述分类标准进行组合, 同时取送货车辆路径问题 VRPSDP 可以演变为不同的形态。但考虑到目前我国产品回收的实际状况, 问题主要表现为客户取货需求具有一定随机性、客户具有服务时间要求和通常采用一个配送中心的客观情况。因此, 本论文主要研究随机需求 VRPSDP 问题 (Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pick-Up, 简称 SDVRPSDP) 和带时间窗的随机需求 VRPSDP 问题 (Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pick-Up with Time Windows, 简称 SDVRPSDPSTW) 两类问题, 这两类问题均属于封闭式 VRPSDP 问题。

## 2.3 同时取送货车辆路径问题的常见求解算法

VRPSDP 问题本身属于 NP 难问题。为了在有效的时间内获得较为满意的车辆路径优化方案, 目前国内外学者主要采用以下启发式算法进行求解:

### (1) 遗传算法

遗传算法 (Genetic Algorithm, GA) 是于上个世纪 60 年代年由美国 J.Holland 教授首先提出来的, 是模拟达尔文生物进化论的自然选择和遗传学机理的生物进化过程的计算模拟, 通过模拟自然进化过程搜索最优解的随机化搜索方法。在遗传算法中, 通过编码组成初始群体后, 遗传操作的任务就是对群体的个体按照它们对环境适应度 (适应度评估) 施加一定的操作, 从而实现优胜劣汰的进化过程, 最终得到满足某种收敛条件的最适应环境的一个群体, 从而获得问题的最优解。目前, 国内不少学者采用遗传算法及其改进算法求解了 VRPSDP 问题, 参见文献 [25-29]。

### (2) 禁忌搜索算法

禁忌搜索算法 (Tabu Search, TS) 的思想最早由 Fred Glover 教授提出的, 是对人类强大记

忆功能的一种模拟。其依赖系统化地使用“记忆存储”来指引整个搜索过程，对已经进行的优化迭代过程进行选择和记录，从而对下一步的搜索方向进行指导，以上所述就是建立 Tabu 表的过程。在 Tabu 表中，保存了最近若干次迭代过程中所进行的移动的反方向移动，但凡 Tabu 表中记录的移动过程，在当前的迭代过程中是不可以再次进行的，因为这样可以防止算法重复选择那些最近迭代过程中已访问过的解集，从而避免算法得到局部最优解。然而，这样也可能会妨碍算法朝着有吸引力的、还未进行访问的解移动。为了避免出现这种不理想的情况，算法使用一个渴望标准来覆盖某些移动的禁忌状态。

### （3）模拟退火算法

模拟退火算法(simulated annealing, SA)是 Kirkpatrick 等人于 1983 年成功应用于组合优化领域的一种亚启发式算法，其是对固体（晶体）的物理退火进行模拟来探索所求问题的最优解。其出发点是基于物理中固体物质的退火过程与一般组合优化问题之间的相似性。在物理上的退火过程中，一个固体通常首先被加热熔化，然后随着温度慢慢地冷却，从而获得一个处于最小内能状态的晶格结构。模拟退火算法的思想就是将这个物理过程转化为针对组合优化问题的求解算法，将问题的解看成是粒子状态，问题的目标函数对应为粒子的能量函数，因此对问题的求解过程就是获得粒子能量最小的过程，可有效地防止陷入局部最优并最终趋于全局最优的优化算法，允许在一定环境下接受较差解。

### （4）蚁群系统算法

蚁群算法是由意大利学者 M.Dorigo 等人于 1991 年首先提出的一种新型仿生行为算法，是在真实蚂蚁觅食行为的启发下提出的一种元启发式算法。它利用一群人工蚂蚁的协作来寻找好的解，是一种并行元启发式算法，属于随机搜索算法。在短时间内，该算法得到各国学者的不断改进和补充，逐渐形成一套成熟的算法框架，并成功应用于车辆路径优化问题。蚁群系统算法是一种特殊的蚁群算法，具有分布性好、鲁棒性强、计算效率高、可实现正反馈以及易于与其它启发式算法相结合的突出优点，成为解决随机需求 VRPSDP 等组合优化问题的一种理想算法。本论文主要采用蚁群系统算法求解随机需求 VRPSDP 问题。

#### 1) 蚁群系统算法的基本原理

蚁群系统算法是对蚂蚁群体觅食的生态行为进行研究，发现蚂蚁在觅食的过程中会在其经过的路径上释放出一种特殊的挥发性分泌物——信息素，使得一定范围内的蚂蚁能够感知到信息素的存在及其强度，并以此指导自己的运动行为。蚂蚁倾向于沿着信息素强度高的方向移动，当某条路径上经过的蚂蚁越多，该路径上的信息素也就越多，后面的蚂蚁选择该条路径的概率就越高，从而更增加了该路径上信息素的强度。因此，由大量蚂蚁组成的蚁群的集体行为表现为一种正反馈现象，从而使得蚁群找到最优路径。

下面以一个形象化的图示来说明蚂蚁的搜索过程。

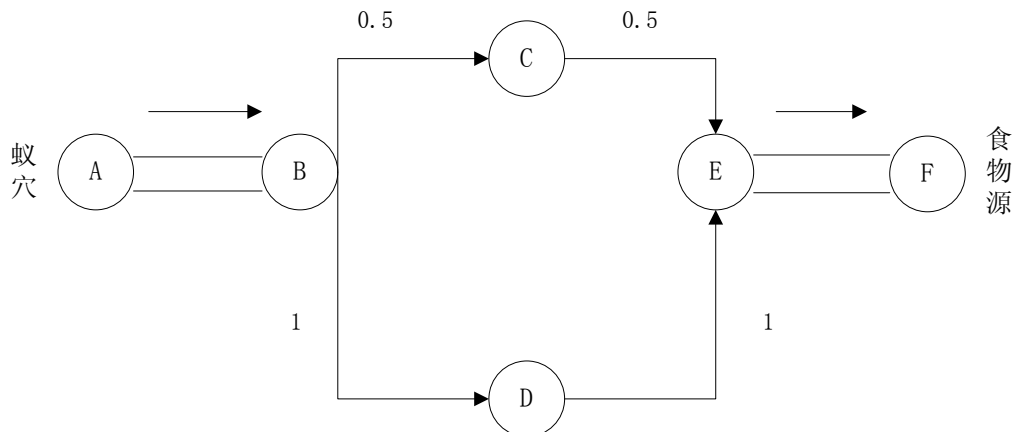


图 2.2 蚂蚁搜索食物过程

假设 A、F 两点分别是蚁穴和食物源，从 A 点到达 F 点，有两条路径 A-B-C-E-F 和 A-B-D-E-F，其中 B-C 和 C-E 之间的距离均为 0.5m，B-D 和 D-E 之间的距离均为 1m。蚂蚁离开蚁穴 A 探索食物，它们到达了一个 B 点（决策点），进行路径选择。一开始蚂蚁会以相同概率随机选择两条路径中的一条并释放信息素，由于两条路径的长度不同，选择较短路径的蚂蚁会首先到达食物源 F，并开始返回它们的蚁穴。当返回的蚂蚁需要再次在 E 点进行抉择，路径上的信息素浓度将会影响它们的决定。显然短分支上的信息素累积速度要比长分支的快，所以短分支上的信息素浓度要高于长分支上的信息素浓度。而高浓度的信息素又会导致更多的蚂蚁再次选择短分支，这个过程会一直进行直到所有的蚂蚁都集中到 A-B-C-E-F 上。这种自组织行为称为正反馈过程：某分支走过的蚂蚁越多，其信息素浓度就越大，而后来的蚂蚁选择该分支的概率就越高。

## 2) 蚁群系统算法数学模型

蚁群系统算法 ACS 是一种针对难解的离散优化问题的元启发式算法。VRPSDP 问题可以用一个带权的完全图  $G = (V, A)$  来描述，其中  $V = \{v_1, v_2, \dots, v_n\}$ ，A 是所有边的集合。 $d_{ij}$  是城市  $v_i$  和城市  $v_j$  之间的距离，其中  $v_i, v_j \in V$ 。目标是寻找一条访问图 G 中每一个结点一次的最小路径程度的闭合路径。 $\tau_{ij}$  是城市  $v_i$  与城市  $v_j$  连线之间的信息素浓度，初始时刻各分支上的信息素浓度  $\tau_{ij}(0) = C$ 。 $\eta_{ij}$  是从城市  $v_i$  到城市  $v_j$  的启发式期望值。蚂蚁在寻找路径的过程中，依据信息素和启发式期望值来选择路径。

蚁群在寻找最优路径的过程中，需要根据某些规则构造路径，当蚂蚁构建了一条完整路径方案后，它们就会在走过的路径上释放信息素。所以，构造蚁群系统算法的数学模型时主要包含路径构建和信息素更新两项关键环节。

### ① 路径构建

相对于基本蚁群算法，蚁群系统算法 ACS 采用了一种积极的路径构建选择规则，从而能更好地开发利用蚂蚁所积累的搜索经验。在蚁群系统算法 ACS 中，位于客户  $i$  的蚂蚁  $k$  根据公式

(2.1) 所示的路径构建规则来选择下一个访问的客户  $j$ 。

$$j = \begin{cases} \arg \max_{l \in N_i^k} \{\tau_{il} \cdot [\eta_{il}]^\beta\}, & q \leq q_0 \\ p_{ij}^k, & q > q_0 \end{cases} \quad (2.1)$$

$$p_{ij}^k = \frac{([\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta)}{(\sum_{i \in N_i^k} [\tau_{il}]^\alpha \cdot [\eta_{il}]^\beta)}, \quad j \in N_i^k \quad (2.2)$$

其中,  $q$  是均匀分布在区间 $[0,1]$ 中的一个随机变量;  $q_0$  是一个确定参数值, 满足  $0 \leq q_0 \leq 1$ ;  $N_i^k$  代表了位于客户  $i$  的蚂蚁  $k$  可以直接到达的且还未被蚂蚁  $k$  访问的相邻客户集合;  $\eta_{ij}$  为从客户  $i$  访问客户  $j$  的启发式信息,  $\eta_{ij} = 1/d_{ij}$ ;  $\tau_{ij}$  为从客户  $i$  访问客户  $j$  的信息素;  $p_{ij}^k$  表示蚂蚁  $k$  由当前客户  $i$  转移到客户  $j$  的概率, 其根据公式 (2.2) 给出的概率分布产生出来的一个随机变量。

路径构建规则也可以理解为, 蚂蚁选择当前可能的最优移动方式的概率是  $q_0$ , 这种最优的移动方式是根据信息素的积累量和启发式信息值求出的。同时, 蚂蚁以  $(1-q_0)$  的概率有偏向性地探索各条边。通过调整参数  $q_0$ , 可以调节算法对新路径的探索度, 从而决定算法是应该集中搜索至今最优路径附近的区域, 还是应该探索其他区域。

## ② 信息素更新

蚁群系统算法 ACS 的信息素更新主要包括局部信息素更新和全局信息素更新, 具体情况如下:

### (a) 局部信息素更新

在路径构建过程中, 蚂蚁每经过一条边  $(i, j)$ , 都立刻采用下式更新该边上的信息素:

$$\tau_{ij} = (1-\rho) \cdot \tau(i, j) + \rho \cdot \tau_0 \quad (2.4)$$

其中  $c_{mn}$  是该车辆路径方案的车辆行驶总行程;  $\tau_0$  表示信息素量的初始值,  $\tau_0 = 1/n \cdot c_{mn}$ ;  $\tau(i, j)$  表示客户  $i$  与客户  $j$  之间的信息素量;  $\rho$  代表信息素的挥发系数。

局部更新的作用在于, 蚂蚁每一次经过边  $(i, j)$ , 该边的信息素  $\tau(i, j)$  将会减少, 从而使得其他蚂蚁选中该边的概率相对减少。也就是说, 这将增加探索未使用过的边的机会, 使得算法不会陷入停滞状态。

### (b) 全局信息素更新

在蚁群系统算法 ACS 中, 只有至今最优蚂蚁被允许在每一次迭代之后释放信息素。全局信息素更新规则如下式所示:

$$\tau_{ij} = (1-\rho) \cdot \tau(i, j) + \rho \cdot \Delta \tau_{ij}^{bs}, \quad \forall (i, j) \in T^{bs} \quad (2.3)$$

其中  $\Delta\tau_{ij}^{bs}$  表示本次迭代新释放的信息素量,  $\Delta\tau_{ij}^{bs} = 1/C^{bs}$ ;  $C^{bs}$  代表至今最优路径方案的车辆行驶总距离;  $T^{bs}$  表示至今最优路径; 参数  $\rho$  代表信息素蒸发速率。

在蚁群系统算法ACS的信息素更新规则中, 无论是信息素蒸发还是信息素释放动作都只在  $T^{bs}$  的边上执行, 而不是像蚁群算法的其它扩展算法那样应用到整个系统的所有边上。因此信息素的计算复杂度从  $O(n^2)$  降到  $O(n)$ , 其中  $n$  代表待解问题的规模。

### 3) 蚂蚁系统算法 ACS 的主要优点

蚁群系统算法 ACS 具有以下优点:

① 蚁群系统算法是一种自组织算法。在算法开始之初, 单个蚂蚁无序的寻找问题的解, 算法经过一段时间的迭代演化, 蚂蚁通过感知信息素浓度, 自动的逐渐探索到接近于最优解的解, 这体现了一个从无序演化到有序的过程, 突出了自组织的特点。

② 蚁群系统算法在本质上是一种并行式算法。每只蚂蚁彼此独立地进行搜索, 只通过信息素来进行信息传递。因此, 蚁群系统算法可以看作是一个分布式的系统, 它可以同时在问题的多点进行独立的解搜索, 不但加强了算法的可靠性, 而且使得算法具有较强的搜索最优解的能力。

③ 蚁群系统算法是具有正反馈的算法。通过对真实蚂蚁觅食过程的观察, 可以看出, 蚂蚁最终之所以可以找到最短路径, 是因为蚂蚁选择路径是直接依赖于信息素的累积。信息素的累积过程是一个正反馈的过程, 会在较优的路径上释放更多的信息素, 而更多的信息素又会吸引更多的蚂蚁, 这个正反馈的过程引导整个系统朝着最优解的方向进化。对于蚁群系统算法来说, 即使在初始时刻信息素的浓度是相同的, 也会由于随机波动的出现, 使得各边上的信息素浓度不相同, 最终蚂蚁构造出的解就有了优劣之分。因此, 正反馈是蚁群算法的重要特征, 它使得算法演化过程得以进行。

④ 蚁群系统算法具有较强的鲁棒性。相比于其它算法, 蚁群系统算法对初始路线要求不高, 即蚁群系统算法的求解结果不依赖于初始路线的选择, 而且在搜索过程中不需要进行人工的调整。其次蚁群系统算法的参数数目少, 设置简单, 易于蚁群系统算法应用到其它组合优化问题的求解。

蚁群系统算法 ACS 是最积极的一种蚁群算法, 它可以在较短的计算时间内获得较优解, 并且在各种蚁群算法的比较中, 蚁群系统算法 ACS 具有最优的求解性能, 其所能找到的最优解比其余蚁群算法所找到的最优解都好。所以本文将基于蚁群系统算法 ACS 设计混合蚁群系统优化算法求解随机需求 VRPSDP 数学模型, 并进而对混合蚁群系统算法进行改进来求解带时间窗的随机需求 VRPSDP 数学模型。

## 2.4 本章小结

本章首先介绍了同时取送货车辆路径问题的定义和分类，总结了同时取送货车辆路径问题的求解算法，为本文求解随机需求的同时取送货车辆路径问题提供了参考和依据。本章对求解同时取送货车辆路径问题的遗传算法、禁忌算法、模拟退火算法、蚁群系统算法依次进行了阐述，并详尽地介绍了蚁群系统算法在同时取送货车辆路径问题中的应用发展以及运作流程，为处理 NP-难的随机需求 VRPSDP 问题提供了必要的数学求解工具。

## 第三章 随机需求 VRPSDP 问题建模及优化研究

### 3.1 引言

目前,国内外企业将其关注重点主要集中在产品生产及销售上,而将废弃产品的回收再利用排除在企业经营战略之外。这样,产品历经设计→制造→消费→废弃四个阶段,形成了从摇篮到坟墓的开环经济发展模式。该模式的突出弊端是产生数目惊人的废弃产品。据联合国环境计划署报告显示,全球每年大约有 5000 万辆~6000 万辆汽车报废,而每年废旧电器电子产品的数量则高达 2000 万吨~5000 万吨。我国的情况也不容乐观,据保守估计,自 2004 年起,我国每年至少有 400 万台冰箱、500 万台电视机以及 600 万台洗衣机报废。产品废弃的直接后果是造成巨大资源浪费和严重环境污染,整个人类的生存环境面临严峻的考验。为了解决经济发展和环境保护之间的矛盾,世界各国政府纷纷颁布法令来规范产品回收处理问题,其中比较知名的法案有:2001 年日本政府实施的《家用电器再生利用法》,2003 年欧盟出台的《报废电子电气设备指令》,美国政府主张延伸生产商责任制度 EPR(Extensible Producer Responsibility),上述法案都主张由制造商(或经销商)负责废旧产品的回收再利用。因此,自 20 世纪 90 年代以来,美国福特、惠普、荷兰飞利浦以及德国大众等许多全球知名企业都建立了专门的产品回收系统。我国政府也高度重视该项工作,在《国民经济和社会发展第十一个五年规划纲要》中明确提出建立废旧家电回收利用示范基地、再制造示范企业等循环经济示范点工程;2008 年,国家发改委颁布了《汽车零部件再制造试点方案》,鼓励汽车整车生产企业通过售后服务体系回收废旧汽车零部件用于再制造;2009 年 1 月正式实施的《中华人民共和国循环经济促进法》,明确规定生产商必须负责回收或委托其它机构回收列入回收目录的废旧产品。同时,该法规大力支持企业开展废旧汽车零部件以及废旧电器电子产品回收再利用,对从事该业务的企业给予财政支持和税收优惠。

产品回收是逆向物流的首要环节(见图 3.1),其实施效果直接关系到后续拆解与检测、再加工处理等环节能否顺利实施。随着国家相继出台鼓励和促进产品回收再利用的政策与法规,越来越多的企业将投入到产品回收再利用这一领域。例如,上海大众联合发展有限公司利用其在国内的 4S 店对桑塔纳以及帕萨特系列的废旧发动机开展回收工作;富士施乐爱科制造(苏州)有限公司则主要利用自建专营店或特约经销商,实现打印机、复印机以及硒鼓等新产品的销售与废旧产品的回收;苏宁电器、国美电器等大型连锁家电企业则针对国家倡导实施的“家电以旧换新”活动,全面开展电视、冰箱、空调等废旧家电产品的回收工作。在实施废旧机电产品回收过程中,为了有效降低车辆的配送运输成本、提高客户的服务满意度,上述回收企业通常采用新产品配送和废旧产品回收同时实施的方式。然而在现实生活中,由于产品使用环境



与条件、产品本身质量性能以及消费者习惯等多种因素影响，废旧产品在回收数量方面呈现较强的随机性（即取货需求具有随机性），从而形成随机需求 VRPSDP 问题(Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pick-Up，简称 SDVRPSDP)。目前，国内外学者在研究同时取送货车车辆路径问题 VRPSDP 时，往往假设客户的取货需求和送货需求均是已知的，而对具有随机取货需求的废旧产品回收问题研究相对较少，这严重影响了企业产品回收工作的有效开展。如何有效解决随机需求 VRPSDP 问题的优化求解问题，成为一项各逆向物流实施企业亟待解决的问题。

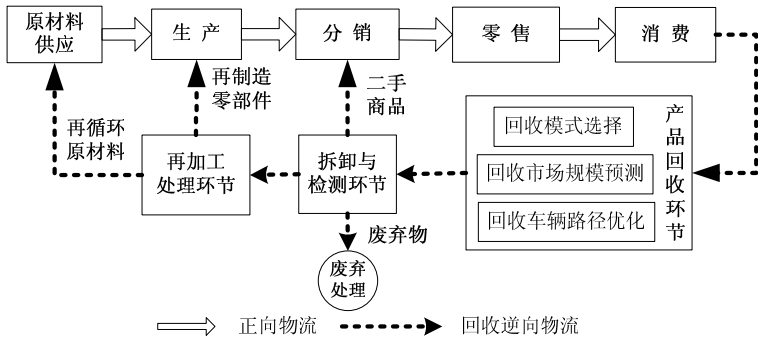


图3.1 产品回收再利用示意图

本章针对送货需求已知、取货需求随机的 VRPSDP 问题，提出了以蚁群系统算法为主体、路径模拟抽样法与局部搜索为补充的混合蚁群系统优化算法，高效解决随机需求 VRPSDP 问题的求解问题，从而为国内逆向物流实施企业开展废旧产品回收提供较为理想的车辆路径优化方案，有效降低企业的物流运作成本。

### 3.2 随机需求 VRPSDP 问题建模

#### 3.2.1 随机需求 VRPSDP 问题的描述

本文研究的随机需求 VRPSDP 问题可以描述为：给定一个配送中心和多台配送车辆，每个客户点的位置坐标是已知的，多台完全相同的车辆从配送中心出发，将客户需要的货物从配送中心运送到各个客户，同时将具有取货需求的客户的货物运回配送中心。车辆的最大行驶距离和车辆负载能力相同，各个客户的送货需求已知，取货需求相互独立且服从某种分布。该模型的最终目标是：在满足客户取送货需求、车辆负载能力以及车辆最大行驶距离等诸多现实约束条件下，合理安排车辆行驶路线，使得期望行驶总距离最小化。

在该问题中，车辆从配送中心出发，为每个客户点提供取货和送货服务，并最终返回服务中心。由于各个客户的取货需求是不可预计的随机变量，会出现当车辆到达某一客户时该客户的取货量大于车辆的剩余空间的情况，这种情况称为路径失败。当发生路径失败时，采取部分服务策略，即先满足该客户的部分取货需求，回到配送中心卸下货物恢复容量后再返回该客户

点继续原路线的服务。

### 3.2.2 基本假设

为了将现实生活中随机需求 VRPSDP 问题抽象为数学模型，建立如下假设：

- (1) 只有一个配送中心，所有车辆从配送中心出发，并最终完成任务返回到该中心
- (2) 配送和回收的货物可以混装
- (3) 每个客户和配送中心的具体位置已知，即各客户节点之间以及客户与配送中心之间的距离已知
- (4) 每个客户只能由同一辆车进行服务，每个客户最多被服务 2 次
- (5) 每辆车具有相同的最大行驶距离和最大承载能力

### 3.2.3 数学模型

模型参数说明：

$n$ ：客户数量；

$C$ ：客户集合， $C = \{1, 2, 3, \dots, n\}$ ；

$H$ ：车辆集合， $H = \{1, 2, 3, \dots, m\}$ ；

$V$ ：所有顶点的集合， $V = \{0\} \cup C$ ，其中 0 代表仓库；

$Q$ ：车辆的最大载货能力；

$L$ ：车辆的最大行驶距离；

$d_{ij}$ ：客户  $i$  与客户  $j$  之间的距离，其中  $i \in V, j \in V, d_{ii} = \infty$ ；

$s_i$ ：客户  $i$  的送货需求量；

$q_i$ ：客户  $i$  的取货需求量，其服从某种分布；

$w_{ijk}$ ：车辆  $k$  访问客户  $i$  后而在访问  $j$  前的载货量；

$x_{ijk}$ ：若车辆  $k$  从客户  $i$  到客户  $j$ ，则  $x_{ijk} = 1$ ；否则  $x_{ijk} = 0$ 。

$$\min E\left(\sum_{k \in H} \sum_{i \in V} \sum_{j \in V} d_{ij} x_{ijk}\right) \quad (3.1)$$

Subject to:

$$\sum_{k \in H} \sum_{i \in V} x_{ijk} \leq 2 \quad \forall j \in C \quad (3.2)$$

$$\sum_{i \in V} \sum_{j \in V} d_{ij} x_{ijk} \leq L \quad \forall k \in H \quad (3.3)$$

$$\sum_{i \in V} x_{ijk} - \sum_{i \in V} x_{jik} = 0 \quad \forall j \in V, \forall k \in H \quad (3.4)$$

$$\sum_{k \in H} \sum_{i \in V} w_{ijk} - s_j = \sum_{k \in H} \sum_{i \in V} w_{jik} - q_j \quad \forall j \in V \quad (3.5)$$

$$w_{ijk} \leq Q \quad \forall i, j \in V, \forall k \in H \quad (3.6)$$

$$w_{ijk} \geq 0 \quad \forall i, j \in V, \forall k \in H \quad (3.7)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j \in V, \forall k \in H \quad (3.8)$$

其中：目标函数（3.1）为最小化车辆期望行驶总距离；

约束（3.2）确保每个客户仅由一辆车提供服务且至多被服务两次；

约束（3.3）保证每辆车不能超过其最大行驶距离；

约束（3.4）确保每个客户只能由同一辆车提供服务；

约束（3.5）确保车辆在服务客户  $j$  的前后路径上的载货量的变化等式；

约束（3.6）确保每辆车的载货量不能超过车辆的最大载货能力；

约束（3.7）保证每辆车在任何两节点间的载货量为非负数

### 3.3 混合蚁群系统优化算法 ACS-RSM 设计

相对于确定性 VRPSDP 问题，随机需求 VRPSDP 问题的复杂性程度更高。为了能在合理时间范围内获得较为理想的车辆路径方案，寻求一种新的求解算法成为一项重要的工作。蚁群系统算法是一种并行元启发式算法，具有分布性好、鲁棒性强、可实现正反馈以及易于与其它启发式算法结合的突出优点，成为解决该问题的一种理想算法。为此，本文提出了以蚁群系统算法为主体、路径模拟抽样法与局部搜索为补充的混合蚁群系统优化算法 ACS-RSM（An Ant Colony System - Route Simulation Method），实现随机需求 VRPSDP 问题的求解。

#### 3.3.1 ACS-RSM 算法设计的难点及解决方案

##### （1）ACS-RSM 算法设计的难点

在蚁群系统算法的每次循环过程中，需要计算当次循环获得的车辆路径方案的行驶总距离，进而利用其完成算法中的局部信息素更新和全局信息素更新。然而，在随机需求 VRPSDP 问题中，由于客户取货需求具有随机性，每台配送车辆只有抵达该客户时才能获得客户的真正取货需求信息。因此，在实际执行车辆路径方案时，会出现车辆到达某客户而该客户的取货需求量大于配送车辆剩余负载能力的情况，即配送车辆剩余负载能力无法满足客户取货需求的情况，这个情况称为“路径失败问题”。当配送车辆在某客户处发生路径失败时，将采取部分服务策略，即车辆需要返回配送中心卸载，恢复负载能力后继续到该客户处提供取货服务，这样势必会增加配送车辆的行驶距离。由此可见，客户取货需求的随机性导致了配送车辆的车辆行驶总距离也同样具有随机性，即意味着配送车辆在真正执行路径方案之前，是无法确定该车辆路径方案

的行驶总距离。因此，如何计算蚁群系统算法每次循环获得的车辆路径方案的车辆行驶总距离，成为算法必须解决的一个突出难题。

## (2) ACS-RSM 算法难点的解决方案

文献[35]通过数学推导证明获得以下研究结论：针对随机性问题，在保证抽样数量达到一定程度的情况下，可以利用模拟抽样数据取代随机性数据，并能够保证相应的精度。基于该研究理论，本文引入了路径模拟抽样 RSM 解决算法每次循环获得的车辆路径方案的行驶总距离计算难题，其基本思路是：

首先，针对 ACS-RSM 算法每次循环获得的车辆路径方案，根据每位客户的随机取货需求概率分布函数，随机模拟抽样生成  $V$  组（数量需达到一定程度）客户取、送货需求信息；

然后，针对每组模拟抽样获得的确定性取、送货需求信息，计算在该车辆路径方案（仅代表车辆访问客户的先后顺序）下的行驶总距离。下面以一个形象化图示说明在一组模拟抽样情况下行驶总距离的计算问题。

图 3.2 展示了一条车辆路径方案（配送的先后顺序为配送中心  $0 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow$  配送中心  $0$ ），单台车辆的最大承载能力为 15 台且任意两客户之间的距离为 1。客户的送货需求已知，根据每位客户的随机取货需求模拟抽样生成一组取货需求信息（该信息只有当车辆到达客户时才能够获知），形成如表 3.1 所示的取、送需求。车辆第一次离开配送中心 0 抵达客户 2，在满足该客户取、送货需求后离开客户 2，此时车辆能够满足后续客户 3、1 的送货需求为 10 台，车辆剩余装载能力为 0 台，车辆行驶距离为 1；接着车辆到达客户 3，在满足该客户 6 台送货需求之后，车辆能够满足后续客户 1 的送货需求为 4 台，车辆剩余装载能力为 6，即仅能完成 6 台取货需求。而客户 3 的取货需求为 9 台，此时出现“路径失败问题”，于是车辆完成该客户的 6 台取货需求并返回到配送中心 0。在配送中心 0 卸下 11 台取货需求（此时车辆能够满足后续客户 1 的送货需求仍为 4 台），并根据客户 1 的送货需求装上 1 台新产品，随后再返回客户 3 进行服务（图 3.2 中采用虚线箭头代表该部分服务策略），车辆行驶距离为 3；在客户 3 处完成剩余的 3 台取货需求，此时车辆能够满足客户 1 的送货需求为 5 台，剩余装载能力为 7 台；之后车辆到达客户 1，完成该客户的取送货需求并返回配送中心，车辆行驶距离为 2。因此，针对该组模拟抽样的行驶总距离为 6。

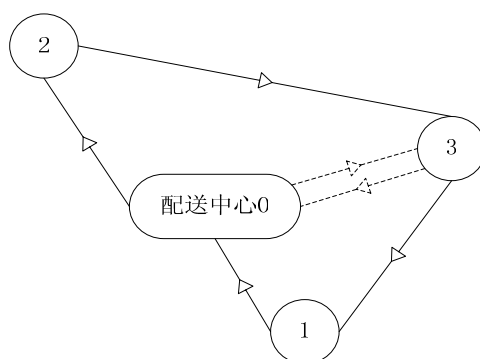


图 3.2 RSM 算法的基本思想

表 3.1 客户的单组模拟抽样取、送货需求

客户	送货需求量	取货需求量
1	5 台	6 台
2	5 台	5 台
3	6 台	9 台

最后，针对计算获得的  $V$  组行驶总距离，采用其期望值作为算法本次循环获得的车辆路径方案的行驶总距离。

### 3.3.2 ACS-RSM 算法框架的组成及其功能

本文构建的混合蚁群系统优化算法 ACS-RSM 架构如下图 3.3 所示。该算法架构主要包括蚁群系统算法 ACS 模块和路径模拟抽样 RSM 模块，其中路径模拟抽样 RSM 模块内含局部搜索子模块和车辆路径评价子模块。各模块的基本功能如下：

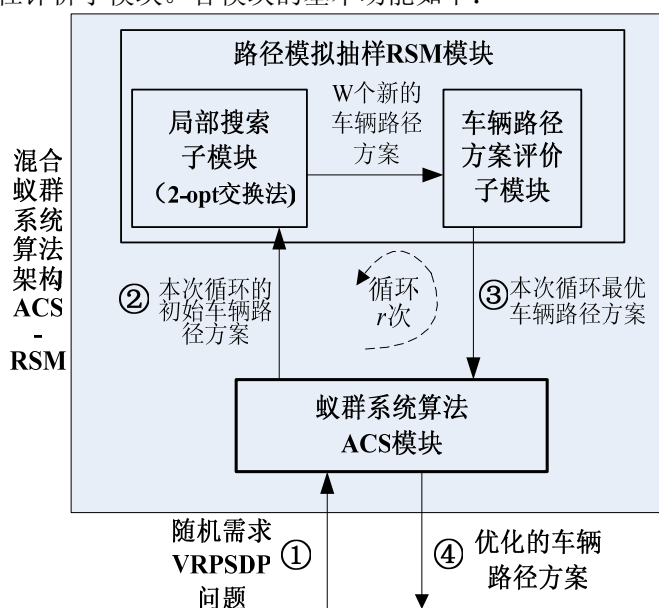


图 3.3 ACS-RSM 算法构架图

(1) 蚁群系统算法 ACS 模块的基本功能

蚁群系统算法 ACS 模块的主要功能是：利用蚁群系统算法，生成算法各次循环的初始车辆路径方案，并将其传送给路径模拟抽样 RSM 模块，用于 RSM 模块对其进行相应处理生成本次循环的最优车辆路径方案（RSM 模块的具体功能详见下文）。

由于篇幅有限，蚁群系统算法 ACS 模块的算法流程详见附录。ACS 模块实现其基本功能的关键在于制定合理的路径构建规则和信息素更新策略，其具体实现方式如下：

### 1) 路径构建规则

在蚁群系统算法 ACS 中，位于客户  $i$  的蚂蚁  $k$  根据公式 (3.9) 来选择下一个访问的客户  $j$ 。

$$j = \begin{cases} \arg \max_{l \in N_i^k} \{\tau_{ij} \cdot [\eta_{il}]^\beta\}, & q \leq q_0 \\ p_{ij}^k, & q > q_0 \end{cases} \quad (3.9)$$

$$p_{ij}^k = \frac{([\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta)}{(\sum_{i \in N_i^k} [\tau_{il}]^\alpha \cdot [\eta_{il}]^\beta)}, \quad j \in N_i^k \quad (3.10)$$

其中： $q$  是均匀分布在  $[0,1]$  中的一个随机变量； $q_0$  是一个确定参数值，满足  $0 \leq q_0 \leq 1$ ；

$N_i^k$  代表了位于客户  $i$  的蚂蚁  $k$  可以直接到达的且还未被蚂蚁  $k$  访问的相邻客户集合；

$\eta_{ij}$  为从客户  $i$  访问客户  $j$  的启发式信息，取  $\eta_{ij} = 1/d_{ij}$ ；

$\tau_{ij}$  为从客户  $i$  访问客户  $j$  的信息素；

$p_{ij}^k$  表示蚂蚁  $k$  由当前客户  $i$  转移到客户  $j$  的概率，其根据公式 (3.10) 给出的概率分布产生出来的一个随机变量。

### 2) 信息素更新准则

**局部信息素更新**: 蚂蚁在构造解的过程中，每经过一条边  $(i, j)$  都按以下公式进行信息素局部更新：

$$\tau_{ij} = (1 - \rho) \cdot \tau(i, j) + \rho \cdot \tau_0 \quad (3.11)$$

其中： $\tau_0$  表示路径上信息素量的初始值， $\tau_0 = 1/n \cdot c_{mn}$ ；

$c_{mn}$  是为按照最近邻启发式方法构造的车辆路径方案的行驶总行程；

$\tau(i, j)$  表示客户  $i$  与客户  $j$  之间的信息素量； $\rho$  代表信息素挥发系数

**全局信息素更新**: 在蚁群系统算法 ACS 中，只有至今最优路径上的信息素被允许进行全局更新。算法按照公式 (3.12) 对至今最优路径上所有的边进行全局信息素更新。

$$\tau_{ij} = (1 - \rho) \cdot \tau(i, j) + \rho \cdot \Delta \tau_{ij}^{bs}, \quad \forall (i, j) \in T^{bs} \quad (3.12)$$

其中： $\Delta \tau_{ij}^{bs}$  表示本次循环新释放的信息素量， $\Delta \tau_{ij}^{bs} = 1/C^{bs}$ ；

$C^{bs}$  代表至今最优车辆路径方案的车辆行驶总距离； $T^{bs}$  表示至今最优路径。

## (2) 路径模拟抽样 RSM 模块的基本功能

路径模拟抽样 RSM 模块包括局部搜索子模块和车辆路径评价子模块。两模块相互配合，共同完成寻求 ACS-RSM 算法本次循环最优车辆路径方案的主要功能。两子模块的具体功能如下：

### 1) 局部搜索子模块

局部搜索子模块主要功能是：采用 2-opt 交换方法，对蚁群系统算法 ACS 模块生成的本次循环初始车辆路径方案（该方案包括多条车辆路径）进行随机的局部交换，形成多个新的车辆路径方案，并传送给车辆路径方案评价子模块进行路径模拟抽样。

采用 2-opt 交换的例子如下图所示：

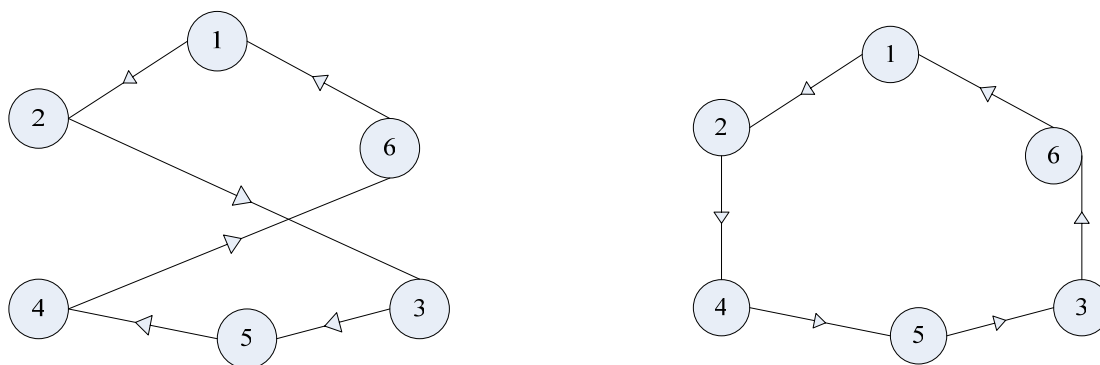


图 3.4 路径内 2-opt 交换邻域示意图

在图 3.4 中，车辆路径方案  $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 4 \rightarrow 6 \rightarrow 1$ ，其中，边(2,3)和边(4,6)由边(2,4)和(3,6)代替后，得到车辆路径方案  $1 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 3 \rightarrow 6 \rightarrow 1$ 。

### 2) 车辆方案评价子模块

车辆方案评价子模块的主要功能是：针对局部搜索子模块提供的每个新车辆路径方案，该模块模拟客户随机取货需求，抽样生成多组确定性客户需求。分别计算每组确定需求下的车辆路径行驶距离，并采用其均值作为该新车辆路径方案的期望行驶总距离。比较全部新车辆路径方案的期望行驶总距离，实现车辆路径方案优劣评价，获得本次循环的最优车辆路径方案，并传送给 ACS 模块用于蚁群系统算法进行信息素更新。

## 3.3.3 ACS-RSM 算法框架整体运行思路及算法基本流程

### (1) ACS-RSM 算法框架整体运行思路

ACS-RSM 算法框架整体运行思路如下：

首先，调用蚁群系统算法 ACS 模块，根据随机客户需求函数产生  $r$  组（ $r$  足够大）确定性

客户需求。针对各组确定客户需求,采用 ACS 算法分别求解车辆路径方案,统计各路径的使用概率,根据统计概率大小关系,形成初始车辆路径方案,并发送给路径模拟抽样 RSM 模块;

其次,针对初始车辆路径方案,调用局部搜索子模块,生成  $W$  个新的车辆路径方案,对每个新的车辆路径方案,调用车辆路径方案评价模块,抽样生成  $V$  组确定性客户需求,分别计算每组确定需求下的车辆路径行驶距离,并采用其均值作为每个新的车辆路径方案的期望行驶总距离;

最后,对  $W$  个新的车辆路径方案进行期望行驶总距离优劣评价,得到本次循环的最优车辆路径方案,并实现信息素更新。

## (2) ACS-RSM 算法的基本流程

采用 ACS-RSM 算法框架求解随机需求 VRPSDP 问题的流程如图 3.5 所示。具体算法流程如下:

Step 1: 系统参数初始化。设置客户数量、客户间距离、车辆台数、车辆最大载重量与最大行驶距离、客户取货随机需求函数、参数  $\varepsilon$ , 并初始化客户之间路径的信息素浓度;

Step 2: 采用混合蚁群系统优化算法 ACS-RSM 构造优化的路径方案。调用蚁群系统算法 ACS 模块,根据客户需求函数生成  $r$  组(够大)确定性客户需求,针对各组确定客户需求,采用蚁群系统算法 ACS 分别求解车辆路径方案,统计各路径的使用概率,根据统计概率大小关系,形成初始车辆路径方案;

Step 3: 针对初始车辆路径方案,调用局部搜索子模块,生成  $W$  个新的车辆路径方案,针对每个新的车辆路径方案,调用车辆路径方案评价子模块采取路径模拟抽样法随机生成  $V$  组客户需求,分别计算每组确定需求下的车辆行驶总距离,并采用其均值作为每个新的车辆路径方案的期望车辆行驶总距离;

Step 4: 对车辆路径方案进行优劣评价,保留本循环最优车辆路径方案及其期望行驶总距离,转到 Step 5;

Step 5: 判断系统终止条件。若本循环所得解的期望行驶距离  $l_1$  与之前循环所得解的期望行驶距离  $l_0$  之差绝对值小于  $\varepsilon$ , 则获得最优路径方案,转到 Step 6; 否则,更新信息素,继续执行 Step 2-Step 4;

Step 6: 系统运行结束,系统参数归零。



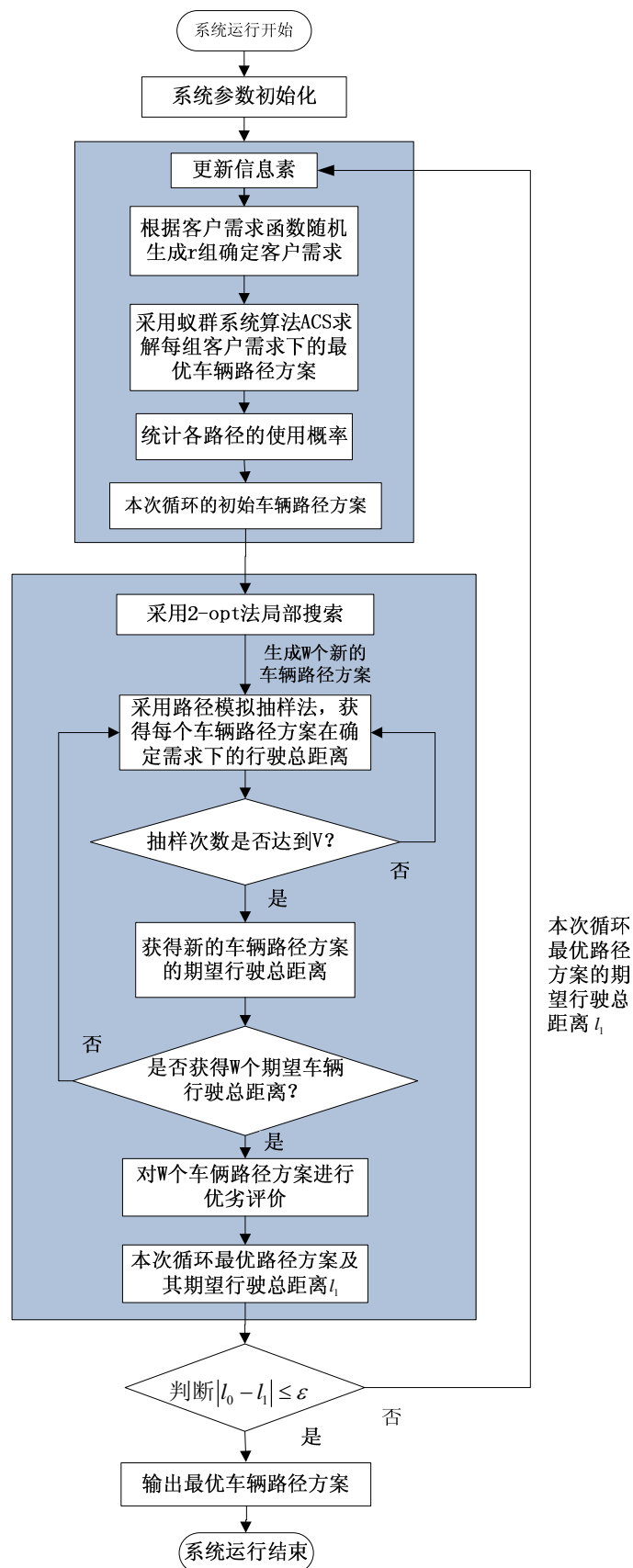


图 3.5 ACS-RSM 算法架构的具体算法流程图

### 3.4 算例分析

对于随机需求 VRPSDP 问题,目前的国内外研究相对很少,尚未提出比较理想的求解算法。为了验证混合蚁群系统优化算法 ACS-RSM 的可行性和有效性,研究首先利用具有随机送货需求的 VRP 问题(随机需求 VRPSDP 问题的特例,只有送货需求,没有取货需求)进行算法求解质量的对比分析,以验证算法的合理性及准确性。然后,针对随机需求 VRPSDP 问题仿真算例,采用 ACS-RSM 算法进行求解,从而获得期望车辆行驶总距离最小化的车辆路径方案。

#### 3.4.1 随机需求 VRP 问题对比分析

研究利用文献[36]中随机需求 VRP 问题的相关数据进行算法的对比分析,具体数据如下:拥有 1 个配送中心,50 个客户随机分布在边长为 100 的正方形区域。车辆数目为 4 辆,每辆车的最大承载力为 20、最大行驶距离为 200。每个客户的送货需求量满足[0,4]上的二项分布随机分布。研究采用 MATLAB7.0 实现混合蚁群系统优化算法 ACS-RSM 编程,表 3.2 是在不同参数组合情况下得到的数据结果。

表 3.2 算法在不同参数组合下的实验结果

参数组合编号	$\alpha$	$\beta$	$q_0$	车辆的期望行驶距离	车辆数
1	1	3	0.9	811.20	4
2	1	6	0.9	945.8	4
3	3	3	0.9	987.23	4
4	1	3	0.45	904.91	4
5	3	3	0.45	865.46	4

通过分析和对比表 3.2 中在不同参数组合下得到的目标函数值,可以得到以下结论:作为信息素、启发式因子分别在总信息中所占权重,当  $\alpha$ 、 $\beta$  的取值分别在[1,3]、[2,6]内取值时,蚂蚁可以在构造解的过程中较好的利用启发信息来缩短总行程,且当  $\alpha$ 、 $\beta$  相对较小时,更有利于蚁群获得近优解。 $q_0$  的大小影响收敛的速度, $q_0$  越小,蚂蚁在选择下一个节点时随机选择可行节点的概率就越大,增强了全局性搜索解的能力,但不容易找到希望的解空间进行深入搜索,容易降低收敛速度。通过对比,第一组参数可以找到较好解,对于解决随机需求 VRPSDP 问题的混合蚁群系统优化算法是比较合适的。

表 3.3 给出了本文算法在第一组参数设置下所得寻优结果以及文献[36]算法的求解结果。通过表 3.3 的结果可以看出,混合蚁群系统优化算法 ACS-RSM 算法求解结果明显优于文献[36],其车辆路径缩短 20.52%。由此可见,在合理的时间范围内,ACS-RSM 算法能够获得高质量的车辆路径方案,从而验证了算法的可行性和有效性。

表 3.3 不同算法对 50 个客户的随机需求 VRP 问题的寻优计算结果

算法	最终车辆路径方案	期望车辆行驶总距离
文献[36]算法	R1/0 3 9 50 11 17 20 25 38 28 4 39 47 24 22 0/ R2/0 45 19 2 42 40 8 29 46 48 5 21 27 0/ R3/0 12 13 6 37 15 33 1 7 44 10 35 30 43 31 0/ R4/0 36 49 34 41 18 26 32 14 23 16 0/	1020.60
ACS-RSM 算法	R1/0 7 19 28 37 13 24 25 33 22 49 32 15 47 0/ R2/0 30 16 42 31 50 4 41 27 14 6 34 5 9 0/ R3/0 39 36 35 23 3 29 2 40 20 17 18 43 0/ R4/0 26 44 1 38 8 46 10 21 45 48 12 11 0/	811.20

### 3.4.2 随机需求 VRPSDP 问题算例仿真

针对随机需求 VRPSDP 问题, 假设有 1 个配送中心, 50 个客户随机分布在边长为 100 的正方形区域, 顾客的送货需求为[0,4]之间的随机数, 取货需求满足[0,4]上的二项式分布。配送中心拥有 4 辆货车, 车辆的最大负载能力为 20, 最大行驶距离为 200。设定随机产生 1000 组数据, 即  $r=1000$ , 从而获得最理想的车辆路径方案, 研究采用 MATLAB7.0 实现混合蚁群系统算法 ACS-RSM 编程, 进行实验的计算结果具体如表 3.4 所示。

表 3.4 ACS-RSM 算法对随机需求 VRPSDP 问题的求解结果

算法	最终车辆路径方案	期望车辆行驶总距离
ACS-RSM 算法	R1/0 29 16 24 27 21 38 26 19 42 13 49 31 25 0/ R2/0 15 32 4 8 40 5 20 48 3 18 50 2 23 0/ R3/0 33 47 7 45 37 35 12 34 22 11 10 0/ R4/0 9 30 14 36 6 1 44 41 28 46 17 39 43 0/	989.12

该问题的车辆路径方案如下图 3.6 所示。

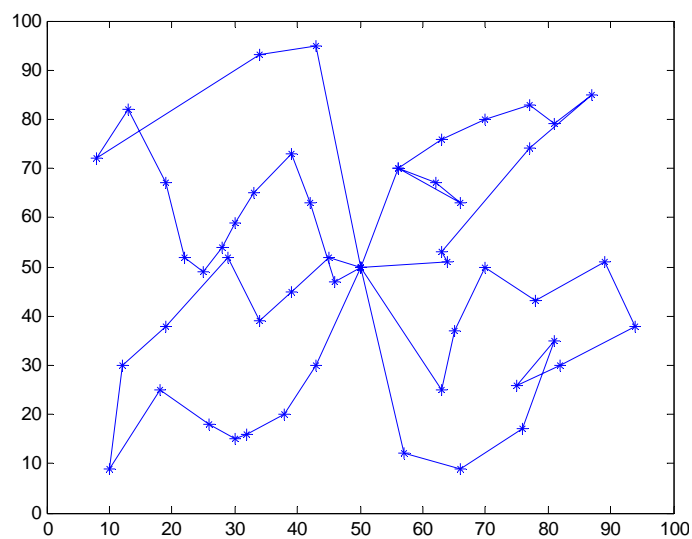


图 3.6 随机需求 VRPSDP 问题算例路径分布

### 3.5 本章小结

随机需求 VRPSDP 问题是当今逆向物流研究必须面对的一个新的难题问题。本章构建了随机需求 VRPSDP 问题的混合整数规划数学模型，提出了以蚁群系统算法 ACS 为主体、路径模拟抽样和局部搜索为补充的混合蚁群系统优化算法 ACS-RSM，并通过对随机需求 VRP 问题(随机需求 VRPSDP 问题的特例)求解的对比分析，验证了算法的可行性和有效性。研究可以有效解决企业面临的随机需求 VRPSDP 难题，有助于降低企业的物流运作成本。

## 第四章 带时间窗的随机需求 VRPSDP 问题建模及优化研究

### 4.1 引言

近年来,随着逆向物流的快速发展,越来越多的客户根据自身经营情况,要求配送车辆在规定的时间内为自己同时提供送货服务和取货服务。例如,在开展废旧发动机回收以及新发动机配送的业务时,很多客户会要求配送车辆在每天上午(或下午)的某一具体时间段内为其提供取、送货服务,譬如在下午 16:30-17:30 这一客户业务量相对较少、时间较宽裕的时间段内提供取送货服务。为了迎合客户的特殊时间要求,逆向物流实施企业在规划车辆行驶路径方案时,不仅需要考虑车辆的最大承载能力、最大行驶距离限制,同时还需要考虑客户的时间窗要求,这样就形成了带时间窗的随机需求 VRPSDP 问题(Stochastic Demand Vehicle Routing Problem with Simultaneous Delivery and Pickup with Time Windows, SDVRPSDPTW)。面对如此复杂的配送环境,如何有效解决带时间窗的随机需求 VRPSDP 问题的优化求解问题,成为各逆向物流实施企业不得不解决的另一项现实问题。

本章针对带时间窗的随机需求 VRPSDP 问题,提出基于 ACS-RSM 的改进混合蚁群系统优化算法。该改进优化算法通过在 ACS-RSM 算法框架的路径转移规则中引入时间窗跨度因子、服务等待因子、紧迫性因子和路径值节约因子,实现对带时间窗的随机需求 VRPSDP 问题的优化求解。最后,通过案例分析验证了该改进优化算法的可行性和有效性。

### 4.2 带时间窗的随机需求 VRPSDP 问题建模

#### 4.2.1 带时间窗的随机需求 VRPSDP 问题描述

本章节研究的带时间窗的随机需求 VRPSDP 问题可以描述为:给定一个配送中心和多台配送车辆,多台车辆从配送中心出发,将客户需要的货物从配送中心运送到各个客户,同时将具有取货需求的客户的货物运回配送中心。每个客户的所要求的时间窗 $[a_i, b_i]$ 已知,车辆的最大行驶距离和车辆负载能力相同,单位距离成本为 1。各个客户的送货需求已知、取货需求相互独立且服从某种随机分布。该模型的最终目标是:在满足客户取送货需求、车辆负载能力以及车辆最大行驶距离等诸多现实约束条件下,寻求期望总费用的最小化。

#### 4.2.2 数学模型

模型参数说明:

$n$ : 客户数量;

$C$ : 客户集合,  $C = \{1, 2, 3, \dots, n\}$ ;

$H$ : 车辆集合,  $H = \{1, 2, 3, \dots, m\}$ ;  
 $V$ : 所有顶点集合,  $V = \{0\} \cup C$ , 其中 0 代表仓库;  
 $Q$ : 车辆的最大装载能力;  
 $L$ : 车辆的最大行驶距离;  
 $d_{ij}$ : 客户  $i$  与客户  $j$  之间的距离, 其中  $i \in V, j \in V, d_{ii} = \infty$ ;  
 $s_i$ : 客户  $i$  的送货需求量;  
 $q_i$ : 客户  $i$  的取货需求量, 服从某种分布;  
 $t_{ij}$ : 车辆从客户  $i$  到客户  $j$  的行驶时间;  
 $[a_i, b_i]$ : 客户  $i$  的时间窗;  
 $t_i$ : 为客户  $i$  的服务时间;  
 $s_{ik}$ : 车辆  $k$  到达客户  $i$  的时间;  
 $c_1$ : 等待惩罚系数, 如果车辆到达  $i$  的时间早于  $a_i$ ;  
 $c_2$ : 迟到惩罚系数, 如果车辆到达  $i$  的时间晚于  $b_i$ ;  
 $w_{ijk}$ : 车辆  $k$  访问客户  $i$  后而在访问  $j$  前的载货量;  
 $x_{ijk}$ : 若车辆  $k$  从客户  $i$  到客户  $j$ , 则  $x_{ijk} = 1$ ; 否则  $x_{ijk} = 0$ .

$$\min E \left( \sum_{k \in H} \sum_{i \in V} \sum_{j \in V} d_{ij} \cdot x_{ijk} + c_1 \sum_{j \in V} \max(a_j - s_{jk}, 0) + c_2 \sum_{j \in V} \max(s_{jk} - b_j, 0) \right) \quad (4.1)$$

Subject to:

$$\sum_{k \in H} \sum_{i \in V} x_{ijk} \leq 2 \quad \forall j \in C \quad (4.2)$$

$$\sum_{i \in V} \sum_{j \in V} d_{ij} x_{ijk} \leq L \quad \forall k \in H \quad (4.3)$$

$$\sum_{i \in V} x_{ijk} - \sum_{i \in V} x_{jik} = 0 \quad \forall j \in V, \forall k \in H \quad (4.4)$$

$$\sum_{k \in H} \sum_{i \in V} w_{ijk} - s_j = \sum_{k \in H} \sum_{i \in V} w_{jik} - q_j \quad \forall j \in V \quad (4.5)$$

$$x_{ijk}(s_{ik} + t_i + t_{ij} - s_{jk}) \leq 0 \quad \forall j \in V, \forall k \in H \quad (4.6)$$

$$a_i - \varepsilon \leq s_{ik} \leq b_i - \varepsilon \quad \forall i, j \in V, \forall k \in H \quad (4.7)$$

$$w_{ijk} \leq Q \quad \forall i, j \in V, \forall k \in H \quad (4.8)$$

$$w_{ijk} \geq 0 \quad \forall i, j \in V, \forall k \in H \quad (4.9)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i, j \in V, \forall k \in H \quad (4.10)$$

其中, 目标函数 (4.1) 为车辆的总费用最小化约束; 约束 (4.2) 每个客户仅由同一辆车提

供服务且至多被服务 2 次；约束（4.3）保证每辆车不能超过其最大行驶距离；约束（4.4）确保每个客户只能由同一辆车提供服务；约束（4.5）确保车辆在服务客户  $j$  前后路段上的载货量的变化等式；约束（4.6）确保车辆  $k$  从客户  $i$  向客户  $j$  行驶的过程中，在时间  $(s_{ik} + t_i + t_{ij})$  之前不能对客户  $j$  进行服务；约束（4.7）为时间窗约束条件；约束（4.8）确保每辆车的载货量不能超过车辆的最大负载能力；约束（4.9）保证每辆车在任何两节点间的载货量为非负数。

客户点的时间窗约束分为硬时间窗约束和软时间窗约束两种，前者要求车辆必须在时间窗内到达，若车辆到达客户  $i$  的时间早于  $a_i$ ，则配送车辆需要在  $i$  处等待，而到达时间不能晚于  $b_i$ ，即约束（4.7）中  $\varepsilon = 0$ ，且目标函数(4.1)中的  $c_1 = \infty$ ， $c_2 = \infty$ ；后者不要求配送车辆一定要在时间窗内到达，但是若其到达客户  $i$  的时间早于  $a_i$  或晚于  $b_i$ ，需要支付一定的惩罚费用，即约束（4.7）中  $\varepsilon = \infty$ 。

### 4.3 改进的混合蚁群系统优化算法 ACS-RSM 设计

随机需求 VRPSDP 问题已经被证明是一个 NP 难题。在带时间窗的随机需求 VRPSDP 问题中，由于每个客户都要求配送车辆要在一定的时间内对其进行服务，导致问题的求解难度进一步增加。因此，为了能在合理时间范围内获得最为理想的车辆路径方案，需要对第三章提出的混合蚁群系统优化算法 ACS-RSM 进行相应改进，从而获得适用于带时间窗的随机需求 VRPSDP 问题的改进混合蚁群系统优化算法 ACS-RSM 算法。

鉴于该改进的混合蚁群系统优化算法在算法框架组成及功能、算法整体运行思路与算法基本流程等方面与第三章的混合蚁群系统优化算法基本相同，其主要区别在于时间窗的引入改变了算法中的路径构建规则。因此，本部分主要介绍该改进算法的路径构建规则设计问题。

#### 4.3.1 路径构建规则相关因子设计

路径构建规则直接影响着带时间窗的随机需求 VRPSDP 问题的求解速度和质量，是改进混合蚁群系统算法设计的核心内容。在构造路径构建规则的过程中，如果只是单纯的考虑启发式信息  $\eta_{ij} = 1/d_{ij}$  和信息素  $\tau_{ij}$  两个因素，难以适用于带时间窗的随机需求 VRPSDP 问题。配送车辆会因为未满足客户的时间要求而导致总运输成本的增加，因此，本文考虑将时间窗跨度因子、服务等待因子、紧迫性因子和路径值节约因子引入到路径构建规则中，从而重新设计算法的路径构建规则。

##### （1）时间窗跨度因子

在带时间窗的随机需求 VRPSDP 问题中，每个客户的时间窗跨度不同，故而客户的服务紧迫性也有所不同。时间窗跨度越小，则该客户点更具紧迫性。因此在选择下一个客户进行访问时，应优先服务时间窗跨度较小的客户，这样会提高车辆后续配送的服务效率。令时间窗跨度因子如式（4.11）所示，显然  $\mu_j$  越大，选择客户  $j$  越合理。

$$\mu_j = \frac{1}{b_j - a_j} \quad (4.11)$$

## (2) 服务等待因子

车辆  $k$  从  $i$  客户出发到达  $j$  的时间为  $s_{jk}$ ，而车辆  $k$  对客户  $j$  开始的服务时间必须在  $[a_j, b_j]$  内。如果  $s_{jk} < a_j$ ，则车辆  $k$  需要在客户  $j$  处等待到  $a_j$  才可以对  $j$  进行取送货服务，如果等待的时间过长会影响车辆的配送效率，为了提高配送车辆的服务效率，车辆在构造路径的过程中会优先考虑等待时间短的客户。因此，结合这个特点，本文在路径构建规则中引入等待时间因子  $\lambda_j$ ，定义变量  $\lambda_j$ ：

$$\lambda_j = a_j - s_{jk}, \quad s_{jk} < a_j \quad (4.12)$$

## (3) 紧迫性因子

车辆  $k$  从  $i$  客户出发到达  $j$  的时间为  $s_{jk}$ ，而车辆  $k$  对客户  $j$  开始的服务时间必须在  $[a_j, b_j]$  内。因此，如果  $s_{jk} > b_j$ ，则车辆  $k$  没有及时地满足客户的取货和送货需求，会导致客户的满意度受损，为了降低车辆的配送成本、提高配送效率，车辆在构造路径的过程中会优先考虑与时间窗上限时间差少的客户。因此，结合以上特点，本文在路径构建公式中引入紧迫性因子  $\omega_{ij}$ ，定义变量  $\omega_{ij}$ ：

$$\omega_{ij} = \begin{cases} b_j - s_{jk}, & a_j < s_{jk} < b_j \\ x(s_{jk} - b_j), & s_{jk} > b_j \end{cases} \quad (4.13)$$

$$s_{jk} = s_{ik} + t_{ij} + t_i \quad (4.14)$$

在 (4.13) 式中，若带时间窗的随机需求 VRPSDP 问题为软时间窗约束的 VRPSDP 问题，则  $x = 1$ ；若为硬时间窗约束的 VRPSDP 问题，则  $x = \infty$ 。

## (4) 路径值节约因子

由于配送中心的存在，所以不仅需要考虑节点之间的距离，还需要考虑到节点与配送中心的间距情况。因此，本文吸收了节约法<sup>[37]</sup>的思想，引入路径值节约因子：

$$\delta_{ij} = d_{i0} + d_{0j} - d_{ij} \quad (4.15)$$

### 4.3.2 路径构建规则构建

综合考虑上述四个因子，设计该改进混合蚁群系统优化算法的路径构建规则。在该算法中，位于客户  $i$  的蚂蚁  $k$  根据如公式 (4.16) 所示的路径构建规则来选择下一个访问的客户  $j$ 。



$$j = \begin{cases} \arg \max_{j \in N_i^k} \left\{ [\tau_{ij}] [\eta_{ij}]^\beta [\mu_j]^\gamma \left[ \frac{1}{\omega_{ij}} \right]^\varepsilon \left[ \frac{1}{\lambda_j} \right]^\kappa \left[ \frac{1}{\delta_{ij}} \right]^\alpha \right\}, & q \leq q_0 \\ p_{ij}^k, & q > q_0 \end{cases} \quad (4.16)$$

$$p_{ij}^k = \frac{[\tau_{ij}] [\eta_{ij}]^\beta [\mu_j]^\gamma \left[ \frac{1}{\omega_{ij}} \right]^\varepsilon \left[ \frac{1}{\lambda_j} \right]^\kappa \left[ \frac{1}{\delta_{ij}} \right]^\alpha}{\sum_{i \in N_i^k} [\tau_{ij}] [\eta_{ij}]^\beta [\mu_j]^\gamma \left[ \frac{1}{\omega_{ij}} \right]^\varepsilon \left[ \frac{1}{\lambda_j} \right]^\kappa \left[ \frac{1}{\delta_{ij}} \right]^\alpha}, \quad j \in N_i^k \quad (4.17)$$

其中,  $q$  是均匀分布在区间 $[0,1]$ 中的一个随机变量;  $q_0$  是一个确定参数值, 满足  $0 \leq q_0 \leq 1$ ;  $N_i^k$  代表了位于客户  $i$  的蚂蚁  $k$  可以直接到达的且还未被蚂蚁  $k$  访问的相邻客户集合;  $\eta_{ij}$  为从客户  $i$  访问客户  $j$  的启发式信息,  $\eta_{ij} = 1/d_{ij}$ ;  $\tau_{ij}$  为从客户  $i$  访问客户  $j$  的信息素;  $p_{ij}^k$  表示蚂蚁  $k$  由当前客户  $i$  转移到客户  $j$  的概率。

#### 4.4 算例仿真分析

为了更好地验证本文所提出算法对带时间窗的随机需求 VRPSDP 问题求解的有效性和可行性, 进行了仿真计算, 研究首先利用带硬时间窗的 VRPSDP 问题以及带软时间窗的 VRP (带软时间窗的 VRPSDP 问题的特例, 没有取货需求)、带软时间窗的 VRPSDP 进行算法求解质量的对比分析以及仿真, 以验证算法的合理性及准确性。然后, 针对带时间窗的随机需求 VRPSDP 问题仿真算例, 采用基于 ACS-RSM 算法的改进混合蚁群系统优化算法进行求解, 从而获得期望车辆行驶总距离最小化的车辆路径方案。

##### 4.4.1 带硬时间窗的 VRPSDP 问题算例对比分析

本算例采用文献[34]的数据进行算法比较, 具体数据如下所示: 拥有 1 个配送中心, 40 个客户, 客户和配送中心的坐标随机分布在边长为 100 的正方形区域。每个客户的送货需求量  $s_i$  为 $[0,100]$ 的随机数, 取货量  $q_i = (0.5 + r_i)s_i$ , 其中  $r_i$  为在 $[0,1]$ 内产生的均匀随机数。配送中心

的车辆数目  $m=3$  辆, 每辆车的最大承载力为  $Q = \sum_{i=1}^{40} s_i / m$ 、最大行驶距离为 400。客户要求的

时间窗的下界  $a_i$  为 $[0,10]$ 内产生的均匀随机数, 上界  $b_i = (2 + \lambda_i)a_i$ , 其中  $\lambda_i$  为在 $[0,1]$ 内产生的均匀随机数, 每个客户的服务时间为 $[0,1]$ 内产生的均匀随机数, 车辆的行驶速度为 50km/h。规定最大迭代次数为 500, 研究采用 MATLAB7.0 实现改进的混合蚁群系统优化算法编程, 进行

10 次实验的计算结果如表 4.1 所示。

表 4.1 10 次随机实验的计算结果

实验编号	1	2	3	4	5
路径长度	1415.1	1572.2	1543.7	1643.3	1668
实验编号	6	7	8	9	10
路径长度	1501.3	1552.7	1678.3	1443.2	1682.1

通过文献[34]算法和改进的混合蚁群系统优化算法获得的求解对比结果如表 4.2 所示。

表 4.2 不同算法的结果比较

算法	平均路径长度	平均计算时间
文献[34]算法	1824.5	66.2329
改进的混合蚁群优化系统算法	1570	16.1214

通过表 4.2 的结果可以看出，本文提出的改进混合蚁群系统优化算法求解结果优于文献 [34]，其平均车辆路径缩短 254.5。

#### 4.4.2 带软时间窗的 VRPSDP 问题的对比分析

对于带软时间窗的 VRPSDP 问题，目前的国内外研究相对很少，为了验证基于 ACS-RSM 的改进混合蚁群系统优化算法的可行性和有效性，本小节首先利用带软时间窗的 VRP 问题进行算法求解质量的对比分析，以验证算法的合理性及准确性。然后，针对带软时间窗的 VRPSDP 问题仿真算例，采用改进的混合蚁群系统优化算法进行求解，从而获得总费用最小化的车辆路径方案。

##### (1) 带软时间窗的 VRP 问题对比分析

基于文献[38]中实验 2 的数据：拥有一个配送中心，15 个客户。客户和配送中心的坐标位置、送货量以及时间窗如表 4.3 所示。每辆车的最大承载能力为 5，行驶速度为 1。等待惩罚系数  $c_1 = 0.5$ ，迟到惩罚系数  $c_2 = 2$ ，忽略在送货点的装货时间。规定最大迭代次数为 500，研究采用 MATLAB7.0 实现改进的混合蚁群系统优化算法编程，进行实验的具体计算结果如表 4.4 所示，图 4.1 是算法的迭代收敛结果曲线图。

表 4.3 带软时间窗的 VRP 的客户点坐标、时间窗及送货需求客户点

	配送中心和客户点							
	0	1	2	3	4	5	6	7
X	50	19	33	35	53	70	27	10
Y	50	0	3	21	19	94	44	69
$s_i$	0	1.0	1.8	1.1	0.6	1.9	1.4	1.2

续表 4.3 带软时间窗的 VRP 的客户点坐标、时间窗及送货需求客户点

$[a_i, b_i]$	$[0, \infty]$	$[74, 144]$	$[58, 128]$	$[15, 85]$	$[96, 166]$	$[47, 117]$	$[85, 155]$	$[21, 19]$
	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>
X	56	16	68	41	83	25	73	70
Y	4	81	76	10	43	91	29	18
$s_i$	0.2	1.7	0.8	0.9	0.8	1.9	1.6	0.9
$[a_i, b_i]$	$[9, 79]$	$[37, 107]$	$[21, 121]$	$[74, 174]$	$[58, 158]$	$[15, 125]$	$[56, 156]$	$[87, 187]$

表 4.4 不同算法的结果比较

	总费用	路径长度	惩罚费用	路径方案
文献[38]算法	570	557.67	12.33	0-3-8-1-2-4-0 0-10-5-13-0 0-9-7-6-0 0-12-14-15-11-0
本文算法	509.7635	509.7635	0	0-7-9-13-5-10-0 0-1-2-11-8-0 0-12-4-15-0 0-6-3-4-0

从表 4.4 中可以看出, 本文的算法求解带软时间窗的 VRP 是有效的, 求解结果明显优于文献[38]。相比于文献[38], 本文算法所得总费用和路径长度分别减少 9.24% 和 7.07%。

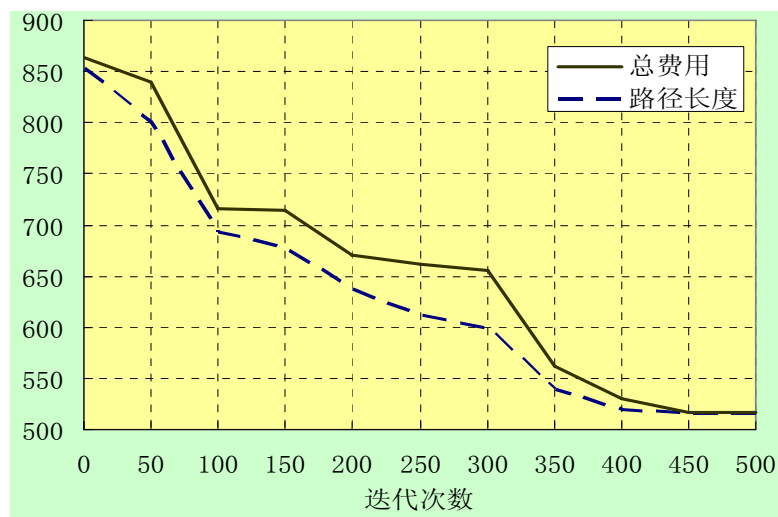


图 4.1 收敛结果曲线图

本文算法所求的车辆路径方案如下图 4.2 所示。

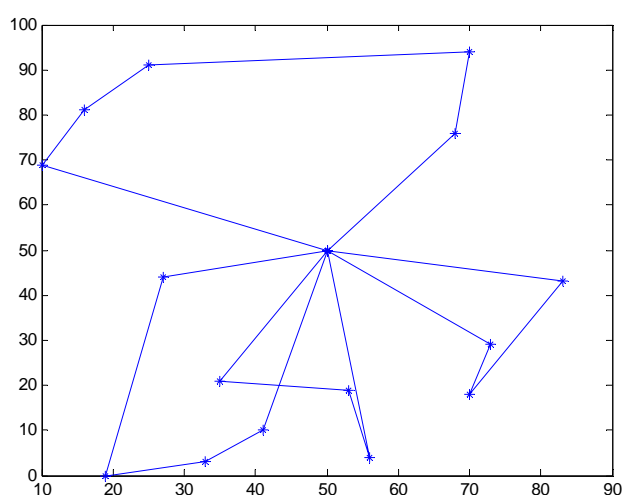


图 4.2 车辆路径方案图

## (2) 带软时间窗的 VRPSDP 问题算例仿真

假设拥有 1 个配送中心, 40 个客户, 客户和配送中心的坐标随机分布在边长为 100 的正方形区域。每个客户的送货需求量  $s_i$  为[0,100]的随机数, 取货量  $q_i = (0.5 + r_i)s_i$ , 其中  $r_i$  为在[0,1]

内产生的均匀随机数。配送中心的车辆数目  $m=3$  辆, 每辆车的最大承载力为  $Q = \sum_{i=1}^{40} s_i / m$ 、最

大行驶距离为 400。车辆从配送中心出发时, 已知客户的取送货需求量。客户要求的时间窗的下界  $a_i$  为[0,10]内产生的均匀随机数, 上界  $b_i = (2 + \lambda_i)a_i$ , 其中  $\lambda_i$  为在[0,1]内产生的均匀随机数, 每个客户的服务时间为[0,1]内产生的均匀随机数, 车辆的行驶速度为 50km/h。等待惩罚系数  $c_1 = 0.5$ , 迟到惩罚系数  $c_2 = 2$ 。规定最大迭代次数为 500, 研究采用 MATLAB7.0 实现改进的混合蚁群系统优化算法编程, 10 次实验的计算结果如下表 4.5 所示, 其中, 最小总费用(1584)的算法收敛效果如图 4.3 所示。

表 4.5 10 次实验的计算结果

实验编号	1	2	3	4	5
路径长度	1637.6	1614.1	1487.5	1598.1	1599
总费用	1804.9	1748.8	1692.4	1773.1	1806
惩罚费用	131.3	134.7	204.9	175	207
实验编号	6	7	8	9	10
路径长度	1506.8	1316.9	1516	1645.1	1483.3
总费用	1675.3	1444.8	1584.7	1776.6	1584
惩罚费用	168.5	127.9	68.7	131.5	100.7

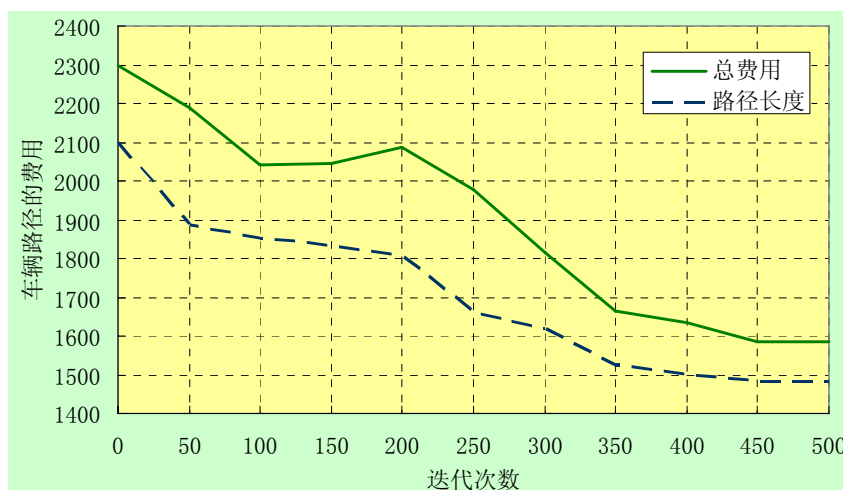


图 4.3 最小路径收敛结果

从以上实验结果可以看出，本文设计的算法可以在合理的时间范围内求解带软时间窗的 VRPSDP 问题和带硬时间窗的 VRPSDP 问题，从而验证了本文算法的可行性和有效性。

#### 4.4.3 带时间窗的随机需求 VRPSDP 问题

针对带时间窗的随机需求 VRPSDP 问题，假设拥有 1 个配送中心，40 个客户，客户和配送中心的坐标随机分布在边长为 100 的正方形区域。每个客户的送货需求量  $s_i$  为  $[0,100]$  的随机数，取货量  $q_i = (0.5 + r_i)s_i$ ，其中  $r_i$  为在  $[0,1]$  内产生的均匀随机数，车辆在离开配送中心时已经知道每个客户的送货需求量，但取货量未知。配送中心的车辆数目  $m=3$  辆，每辆车的最大承载力为  $Q = \sum_{i=1}^{40} s_i / m$ 、最大行驶距离为 400。客户要求的时间窗的下界  $a_i$  为  $[0,10]$  内产生的均匀随机数，上界  $b_i = (2 + \lambda_i)a_i$ ，其中  $\lambda_i$  为在  $[0,1]$  内产生的均匀随机数，每个客户的服务时间为  $[0,1]$  内产生的均匀随机数，车辆的行驶速度为 50km/h，研究采用 MATLAB7.0 实现改进的混合蚁群系统优化算法编程。

表 4.6 给出了 10 次实验的计算结果，其中等待惩罚系数  $c_1 = 0.5$ ，迟到惩罚系数  $c_2 = 2$ 。从表 4.6 中，可以看出本文提出的改进混合蚁群系统优化算法可以有效的解决带软时间窗的随机需求 VRPSDP 问题，并且获得 10 组实验结果，这 10 组数据的平均路径长度为 1846.8、平均总费用为 2069.99、平均惩罚费用为 223.12。通过这些实验，可以看出本文设计的改进混合蚁群系统优化算法对于求解带软时间窗的随机需求 VRPSDP 问题是有效的，可以找到满意解。

表 4.6 10 次随机实验的计算结果

实验编号	1	2	3	4	5
路径长度	1840.4	1873.3	1960.9	1741.6	1835.1
总费用	2072.8	2197.9	2199.6	1944.5	2021.8
惩罚费用	232.4	324.6	238.7	202.9	186.7
实验编号	6	7	8	9	10
路径长度	1914.3	1814.9	1897.3	1776.7	1814.2
总费用	2117.8	2032.5	2079.5	2044	1989.5
惩罚费用	203.5	217.6	182.2	267.3	175.3

其中第4组实验的期望总费用是1944.5，其对应的路径方案如表4.7所示。

表 4.7 路径方案

车辆 1	0-21-1-17-3-16-32-25-31-13-24-12-36-20-27-0
车辆 2	0-39-5-29-26-22-8-15-14-2-9-30-11-6-0
车辆 3	0-4-33-23-35-7-28-18-10-37-40-34-19-38-0

由表4.6和表4.7可以看出，本文所提出的改进混合蚁群系统优化算法可以有效的解决带软时间窗的随机需求VRPSDP问题，也从而证明了改进的混合蚁群系统优化算法的合理性。

## 4.5 本章小结

本章以带时间窗的随机需求 VRPSDP 问题为研究对象，构建了带时间窗的随机需求 VRPSDP 问题的混合整数规划数学模型，在路径构建规则中引入时间窗跨度因子、服务等待因子、紧迫性因子和路径值节约因子，并通过对带软时间窗的 VRPSDP 问题和带硬时间窗的 VRPSDP 求解的对比分析，有效解决企业面临的带时间窗的随机需求 VRPSDP 问题，验证了算法的可行性和有效性，有助于降低企业的物流运作成本、提高车辆的配送效率。

## 第五章 总结与展望

### 5.1 全文总结

本文重点研究了随机需求同时取送货车辆路径问题以及带时间窗的随机需求同时取送货车辆路径问题，主要研究内容如下：

第一章，简要介绍了研究的背景与意义，较全面地分析了国内外对于同时取送货车辆路径问题的研究现状以及研究成果存在的不足，阐述了论文的主要研究内容与创新点，提出了研究的主要框架。

第二章，详细描述了同时取送货车辆路径问题VRPSDP的定义与分类，介绍和分析了目前VRPSDP问题的常用求解算法，重点介绍了本文采用的蚁群系统算法，为研究奠定了理论基础。

第三章，研究了随机需求VRPSDP问题，构建了送货需求已知、取货需求的随机的同时取送货车辆路径问题SDVRPSDP数学模型，提出以蚁群系统算法为主体、路径模拟抽样法与局部搜索为补充的混合蚁群系统优化算法ACS-RSM。通过案例仿真试验，对比分析了该算法与其它启发式算法，验证了算法的有效性和合理性，有效解决了随机需求VRPSDP问题的求解问题，获得较为理想的车辆路径优化方案。

第四章，研究了带时间窗的随机需求VRPSDP问题，构建了送货需求已知、取货需求随机且带时间窗的同时取送货车辆路径问题数学模型。根据该问题中客户具有配送时间要求的重要特点，提出了改进的混合蚁群系统优化算法ACS-RSM，设计了新的路径转移规则，解决了带时间窗的随机需求VRPSDP问题的求解问题。

### 5.2 研究展望

根据客户需求是否具有随机性、客户是否动态出现等不同条件，同时取送货车辆路径问题具有不同的表现形式，因此未来研究可以在以下方面展开：

(1) 取货需求和送货需求均随机的同时取送货车辆路径问题研究。本文仅对送货需求已知、取货需求未知的同时取送货车辆路径问题进行了研究，然而车辆路径问题中还存在取货需求和送货需求均随机的同时取送货车辆路径问题，需要开展进一步的研究。

(2) 具有动态客户的随机需求同时取送货车辆路径问题研究。本文所研究的随机需求VRPSDP问题和带时间窗的随机需求VRPSDP问题，属于静态车辆路径问题范畴。而现实生活中，在车辆路径方案执行过程中，仍会接收到新客户的取送货请求，从而形成具有动态客户的随机需求同时取送货车辆路径问题。如何寻求该问题的车辆路径优化方案成为未来研究的一个重要方向。

## 参考文献

- [1] Yano C., T. Chan, et al. Vehicle Routing at Quality Stores[J]. Interfaces, 1987, 17:52~63.
- [2] Toth, P., D.Vigo. An exact algorithm for the vehicle routing problem with backhauls[J]. Transportation Science, 1997, 31(4):372~385.
- [3] Mingozzi A., Giorgi S., et al. An exact method for the vehicle routing problem with backhauls[J]. Transportation Research A, 1999, 33(3): 315~329.
- [4] Golden, B., E. Baker, et al. The vehicle routing problem with backhauling: two approaches [J]. Proceedings of the XXI Annual Meeting of S. E. Tims, Martle Beach, 1988: 90~92.
- [5] S.R. Thangiah, J.Y. Potvin, T. Sun. Heuristic approaches to vehicle routing with backhauls and time windows[J]. Computers & Operations Research, 1996, 23(11): 1043~1057.
- [6] C.Duhamel, J.Y. Potvin, J.M. Rousseau. A tabu search heuristic for the vehicle routing problem with back-hauls windows[J]. Journal of the Operational Research Society, 2001, 52: 928~936.
- [7] J. Brandao. A new tabu search algorithm for the vehicle routing problem with backhauls[J]. European Journal of Operational Research, 2006, 173: 540~555.
- [8] Y. Zhong, M.H. Cole. A vehicle routing problems with backhauls and time windows: a guided local search solution[J]. Transportation Research, 2005, 41: 131~144.
- [9] R.T Moghaddam, A.R. Saremi, M.S. Ziaee. A memetic algorithm for a vehicle routing problem with backhauls [J]. Applied Mathematics and Computation, 2006, 181: 1049~1060.
- [10] Goetschalckx M., C. Jacobs-Blecha. The vehicle routing problem with backhauls[J]. European Journal of Operational Research, 1989, 42: 39~51.
- [11] Toth P., D. Vigo. A heuristic algorithm for the vehicle routing problem with backhauls[C]. Advanced Methods in Transportation Analysis, Berlin, Springer, 1996: 585~608.
- [12] Deif I. , L. Bodin. Extension of the Clarke and Wright algorithm for solving the vehicle routing problem with backhauling[C]. Proceedings of the Babson Conference on Software Uses in Transportation and Logistic Management, Babson Park, 1984: 75~96.
- [13] Casco O, Golden B, Wasil E. Vehicle routing with backhauls: models, algorithms and case studies [C]. Vehicle routing: methods and studies, Amsterdam: North Holland, 1988, 127~147.
- [14] Salhi S, Nagy G. A cluster insertion heuristic for the single and multiple depot vehicle routing problems with backhauls. Journal of the Operational Research Society, 1999, 50 (10): 1034~1042.
- [15] A.C. Wade, S. Salhi. An investigation into a new class of vehicle routing problem with backhauls[J]. Omega, 2002, 30: 479~487.
- [16] Min.H. The multiple vehicle routing problem with simultaneous delivery and pick-up points. Transportation research, 1989, 23: 377~386.
- [17] J.Dethloff. Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up [J]. OR Spektrum, 2001, 23: 79~96.
- [18] S.Salhi, G Nagy. Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries. European Journal of Operational research, 2005, 162 (1): 126~141.
- [19] F.A.Tang, R.D.Galvão. A tabu search algorithm for the vehicle routing problem with



- simultaneous pick-up and delivery service [J]. *Computers & Operations Research*, 2006, 33: 595~619.
- [20] Yuvraj G, Prakash A. An ant colony system (ACS) for vehicle routing problem with simultaneous delivery and pick up[J]. *Computers & Operations Research*, 2009, 36: 3215~3223.
- [21] 谢如鹤, 刘霆, 邱祝强. 基于剩余装载能力的逆向物流车辆路径问题. *系统工程*, 2004, 22(10): 20~23.
- [22] 陆琳, 谭清美. 基于自感应蚁群算法的 VRPSDP 问题研究 [J]. *中国管理科学*, 2007, 15(2): 97~102.
- [23] 张涛, 田文馨. 带车辆行程约束的VRPSDP问题的改进蚁群算法[J]. *系统工程理论与实践*, 2008, 1: 132~140.
- [24] 孙小年, 陈幼林, 杨东援. 装卸一体化车辆路径问题的遗传算法研究[J]. *系统工程理论与实践*, 2007, 2: 149~152.
- [25] 张建勇, 李军. 具有同时配送和回收需求的车辆路径问题的混合遗传算法[J]. *中国公路学报*, 2006, 19(4): 118~122.
- [26] 龙磊, 陈秋双. 具有同时集送货需求的车辆路径问题的自适应混合遗传算法[J]. *计算机集成制造系统*, 2008, 14(3): 548~556.
- [27] 彭春林, 梁春华, 周泓. 求解同时取货和送货车辆路径问题的改进遗传算法[J]. *系统仿真学报*, 2008, 20(9): 2266~2270.
- [28] 龙磊, 陈秋双. 具有同时集送货需求的车辆路径问题的粗粒度并行遗传算法[J]. *系统仿真学报*, 2009, 21(7): 1962~1973.
- [29] 张涛, 张春梅, 张玥杰. 协同粒子群-模拟退火算法求解 VRPSDP 问题[J]. *系统管理学报*, 2009, 18(6): 681~685.
- [30] Angelelli E, Mansini R. The vehicle routing problem with time windows and simultaneous pickup and delivery[A]. In: Speranza MG, Klose A, Van Wassenhove LN (eds) *Quantitative approaches to distribution logistics and supply chain management*[C]. Springer, Berlin, 2002. 249~267.
- [31] Ropke S, Pisinger D. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time window [J]. *Transport Science*, 2006, 40(4): 455~472.
- [32] Ropke S, Cordeau JF, Laporte G. Models and a branch-and-cut algorithm for pickup and delivery problems with time window [J]. *Networks*, 2007, 49(4): 258~272.
- [33] Deng Ai-min, Mao-Chao, Zhou Yan-ting. Optimizing Research of an Improved Simulated Annealing Algorithm to Soft Time Windows Vehicle Routing Problem with Pick-up and Delivery[J]. *Systems Engineering—Theory & Practice*, 2009, 29(5): 186~192.
- [34] 曹二保, 赖明勇. 基于改进差分进化算法的 VRP-SDPTW 研究[J]. *管理工程学报*, 2009, 2: 80~84.
- [35] Alexander Shapiro. Monte Carlo Sampling Methods[M]. *Handbooks in OR&MS*, 2003.
- [36] Teodorovic D., Pavkovic G. A simulated annealing technique approach to the vehicle routing problem in the case of stochastic demand[J]. *Transportation Planning and Technology*, 1992, 16: 261~273.
- [37] Clarke G, Wright J. Scheduling of vehicles from a central depot to a number of delivery points [J]. *Operations Research*, 1964, 12(4): 568~581.
- [38] 李琳, 刘士新. 改进的蚁群算法求解带时间窗的车辆路径问题[J]. *控制与决策*, 2010, 9(25): 1380~1383.

## 致 谢

本论文是在我的导师刘文杰副教授的悉心指导下，确定主题开展研究完成的。刘文杰老师对论文逐字逐句的修改、提出意见和建议，为论文的最终完成倾注了大量的智慧和心血。在攻读硕士研究生二年多的时间里，无论是生活上，还是在学习上，刘文杰老师都给予我真切的关怀。导师渊博的知识、严谨的治学、认真的教学态度，都令我由衷的敬佩，他谦和温婉的处事风范也让我感触良深、受益匪浅。再次对我的导师致以崇高的敬意和深深的感激之情！

值此论文完成之际，向所有给予我关心和帮助的老师 and 同学致以最诚挚的谢意和最美好的祝愿，特别是 50402 室所有的姐妹们，和她们一起生活和学习很快乐。

最后我还要向我的父母和弟弟献上我最衷心的感谢，正是他们自始至终的支持和鼓励，才使得我能够顺利完成学业。感谢父母这么多年来对我的养育和教导，在今后的工作中我一定不会辜负他们的期望，再接再厉。

## 在学期间的研究成果及发表的学术论文

### 攻读硕士学位期间录用论文情况

1. Liu Qing, Liu Wen-jie. Study on Modeling and Optimization of SDVRPSDP based on ACS-RSM Algorithm. 2011 年 IEEE 灰色系统与智能服务国际会议.
2. 刘晴, 刘文杰. 基于 ACS-RSM 的随机 VRPSDP 问题建模及优化研究. 2010 年江苏省管理学研究生学术论文, 获得硕士研究生组优胜奖.

### 攻读硕士学位期间参加科研项目情况

1. 南京航空航天大学哲学社会科学基金(V0987-093): 随机环境下 VRPSDP 问题建模及优化研究, 2009-2011.

## 附录

蚁群算法系统 ACS 子模块的基本算法流程如下:

Step 1: 初始化系统参数, 生成初始解  $\pi_0$ , 设置迭代次数  $H$ ;

Step 2:  $load \leftarrow start\_load$ ,  $cur\_load \leftarrow start\_load$ ,  $veh\_n \leftarrow 1$ ,  $length \leftarrow 0$ , 其中  $load$  为车辆当前送货载货量,  $cur\_load$  为车辆当前的载货量,  $start\_load$  为车辆离开仓库时的送货载货量,  $length$  为车辆当前已行驶的距离, 将初始到达的点放于当前的解中;

Step 3: 判断每个蚂蚁下一个要访问的客户是否满足以下约束条件:

- 1) 未被提供服务过;
- 2) 可以满足送货需求量:  $load > s_j$ ;
- 3) 满足车辆最大承载能力限制:  $cur\_load - s_j + q_i < Q$ ;
- 4) 满足车辆最大行驶距离限制:  $length + d_{ij} + d_{j0} < L$ .

若有满足上述约束条件的可行节点, 则蚂蚁按路径规则公式(3.9)选择访问节点  $j$ , 并设置  $load \leftarrow load - s_j$ ,  $cur\_load \leftarrow load - s_j + q_j$ ,  $length \leftarrow length + d_{ij}$ . 将所选的客户点  $j$  置于解集中, 并按公式(3.11)对边  $(i, j)$  进行局部更新, 重复 Step 3, 否则转到 Step 4;

Step 4: 若所有客户点都已置于问题的解集中, 则计算目标函数值, 转到 Step 6. 否则, 转到 Step 5;

Step 5: 若没有满足约束条件的可行节点, 但仍有未被提供服务的客户点, 则车辆需返回配送中心, 置  $load \leftarrow start\_load$ ,  $cur\_load \leftarrow start\_load$ ,  $veh\_n \leftarrow veh\_n + 1$ , 转到 Step 3;

Step 6: 按公式(3.12)进行全局信息素更新,  $H \leftarrow H + 1$ ;

Step 7: 若  $H$  小于预定的次数, 则转到 Step 2;

Step 8: 运行结束, 输出优化的车辆路径方案.