

**毕业设计（论文）**

基于区域聚类的多无人机与物流车路径优化算法研究

专 业 计算机科学与技术

学　　 号 1143730210

学 生 周昊天

指 导 教 师 曲明成

答 辩 日 期 2018/6/16

**摘 要**

车辆路径问题自1959年提出以来，由于其应用的广泛性和[经济](http://wiki.mbalib.com/wiki/%E7%BB%8F%E6%B5%8E" \o "经济)上的重大价值，一直受到国内外学者的广泛关注。带容量的车辆路径问题（CVRP）作为传统车辆路径问题中的一个重要分支，经过几十年的研究已经有了较为成熟的启发式算法优化方案。随着无人机技术的逐渐成熟以及在物流运输中的出众表现，带无人机的路径优化问题（UVAVRP）成为当前学术和工程界亟待解决的崭新问题。本文在考虑无人机与物流车协同配送这一现实场景下，对带无人机的CVRP进行了创新性的研究，结合使用改进后的聚类算法和禁忌搜索算法，对复杂问题进行预处理、全局CVRP求解、局部CVRP求解这三个阶段的分解，最终构建出用于解决该问题的三阶段混合算法。经过详细的实验证明，本算法通过合理规划物流车与无人机的协同配送方案，有效地降低了运行成本和时间成本，从理论层面上解决了物流车与无人机协同配送这一难题。

关键词：带容量的车辆路径问题 ； 协同运输 ；三阶段混合算法 ；聚类算法 ； 禁忌搜索算法

**Abstract**

Since its introduction in 1959, the vehicle routing problem has attracted widespread attention from scholars at home and abroad due to its extensive application and significant economic value. The capacity-carrying vehicle routing problem (CVRP) is an important branch of the traditional vehicle routing problem. After decades of research, there have been more mature heuristic algorithm optimization solutions. With the gradual maturity of UAV technology and outstanding performance in logistics and transportation, the path optimization problem (UVAVRP) with drones has become a new problem to be solved in the current academic and engineering world. In this paper, considering the cooperative scenario of unmanned aerial vehicle and logistics vehicle coordinated delivery, an innovative research is conducted on CVRP with drones, and the improved clustering algorithm and tabu search algorithm are combined to preprocess complex problems. , Global CVRP solution, local CVRP solution to these three stages of decomposition, the final construction of a three-stage hybrid algorithm to solve the problem of the scene, through the rational planning of logistics vehicles and unmanned aerial vehicles cooperative distribution, effectively reducing the operating costs and Time cost, from the theoretical level, innovatively solved the problem of collaborative delivery of logistics vehicles and drones.

**Keywords:** CVRP , UAV , cooperative transportation , three-stage hybrid algorithm , clustering algorithm , tabu search algorithm

目录

[第一章 绪 论](#_Toc14429_WPSOffice_Level1) [1](#_Toc14429_WPSOffice_Level1)

[1.1. 课题背景及研究的目的和意义](#_Toc9681_WPSOffice_Level2) [1](#_Toc9681_WPSOffice_Level2)

[1.1.1. 课题背景](#_Toc9681_WPSOffice_Level3) [1](#_Toc9681_WPSOffice_Level3)

[1.1.2. 研究的目的和意义](#_Toc16317_WPSOffice_Level3) [1](#_Toc16317_WPSOffice_Level3)

[1.2. 路径规划问题及其相关理论的发展概况](#_Toc16317_WPSOffice_Level2) [2](#_Toc16317_WPSOffice_Level2)

[1.2.1. 路径规划算法的发展](#_Toc13770_WPSOffice_Level3) [2](#_Toc13770_WPSOffice_Level3)

[1.2.2. 聚类算法与路径规划算法结合先例](#_Toc3107_WPSOffice_Level3) [4](#_Toc3107_WPSOffice_Level3)

[1.2.3. 无人机配送技术的应用](#_Toc5065_WPSOffice_Level3) [4](#_Toc5065_WPSOffice_Level3)

[1.3. 本文研究的主要内容](#_Toc13770_WPSOffice_Level2) [4](#_Toc13770_WPSOffice_Level2)

[1.3.1. 应用场景描述](#_Toc49_WPSOffice_Level3) [4](#_Toc49_WPSOffice_Level3)

[1.3.2. 算法思想综述](#_Toc1692_WPSOffice_Level3) [5](#_Toc1692_WPSOffice_Level3)

[1.3.3. 算法创新点](#_Toc20324_WPSOffice_Level3) [6](#_Toc20324_WPSOffice_Level3)

[第二章 问题模型构建](#_Toc9681_WPSOffice_Level1) [8](#_Toc9681_WPSOffice_Level1)

[2.1. 问题场景描述](#_Toc3107_WPSOffice_Level2) [8](#_Toc3107_WPSOffice_Level2)

[2.2. 数学符号定义](#_Toc5065_WPSOffice_Level2) [8](#_Toc5065_WPSOffice_Level2)

[2.3. 数学模型构建](#_Toc49_WPSOffice_Level2) [9](#_Toc49_WPSOffice_Level2)

[第三章 问题模型求解](#_Toc16317_WPSOffice_Level1) [11](#_Toc16317_WPSOffice_Level1)

[3.1. 算法思想综述](#_Toc1692_WPSOffice_Level2) [11](#_Toc1692_WPSOffice_Level2)

[3.2. 各模块功能详解](#_Toc20324_WPSOffice_Level2) [11](#_Toc20324_WPSOffice_Level2)

[3.2.1. 预处理模块](#_Toc25639_WPSOffice_Level3) [12](#_Toc25639_WPSOffice_Level3)

[3.2.2. CVRP全局解模块](#_Toc3837_WPSOffice_Level3) [14](#_Toc3837_WPSOffice_Level3)

[3.2.3. CVRP局部解模块](#_Toc28620_WPSOffice_Level3) [15](#_Toc28620_WPSOffice_Level3)

[3.2.4. 显示模块](#_Toc4954_WPSOffice_Level3) [17](#_Toc4954_WPSOffice_Level3)

[3.2.5. 代价计算模块](#_Toc22767_WPSOffice_Level3) [19](#_Toc22767_WPSOffice_Level3)

[3.3. 核心算法解析](#_Toc25639_WPSOffice_Level2) [20](#_Toc25639_WPSOffice_Level2)

[3.3.1. 带限制条件的的迭代求解K-means聚类算法](#_Toc12515_WPSOffice_Level3) [20](#_Toc12515_WPSOffice_Level3)

[3.3.2. Tabu-CVRP算法](#_Toc17058_WPSOffice_Level3) [23](#_Toc17058_WPSOffice_Level3)

[3.3.3. 局部类内分治规划算法](#_Toc17791_WPSOffice_Level3) [30](#_Toc17791_WPSOffice_Level3)

[第四章 实验及结果分析](#_Toc13770_WPSOffice_Level1) [32](#_Toc13770_WPSOffice_Level1)

[4.1. 实验简介](#_Toc3837_WPSOffice_Level2) [32](#_Toc3837_WPSOffice_Level2)

[4.2. 实验过程](#_Toc28620_WPSOffice_Level2) [33](#_Toc28620_WPSOffice_Level2)

[(1) 与传统物流算法对比，展示新算法优势](#_Toc17865_WPSOffice_Level3) [33](#_Toc17865_WPSOffice_Level3)

[(2) 算法自身测试，性能影响因素研究](#_Toc22201_WPSOffice_Level3) [33](#_Toc22201_WPSOffice_Level3)

[4.2.1. 模拟数据生成模块](#_Toc10541_WPSOffice_Level3) [33](#_Toc10541_WPSOffice_Level3)

[4.2.2. 样本数量测试](#_Toc12791_WPSOffice_Level3) [36](#_Toc12791_WPSOffice_Level3)

[4.2.3. 轻重件比例测试](#_Toc10072_WPSOffice_Level3) [37](#_Toc10072_WPSOffice_Level3)

[4.2.4. 无人车载重测试](#_Toc10092_WPSOffice_Level3) [39](#_Toc10092_WPSOffice_Level3)

[4.2.5 无人机数量测试](#_Toc6198_WPSOffice_Level3) [40](#_Toc6198_WPSOffice_Level3)

[4.2.6 无人机速度测试](#_Toc32183_WPSOffice_Level3) [41](#_Toc32183_WPSOffice_Level3)

[4.3. 实验结果及分析](#_Toc4954_WPSOffice_Level2) [41](#_Toc4954_WPSOffice_Level2)

[结 论](#_Toc3107_WPSOffice_Level1) [43](#_Toc3107_WPSOffice_Level1)

[(1) 聚类思想在本场景下的适用性改进](#_Toc5065_WPSOffice_Level1) [43](#_Toc5065_WPSOffice_Level1)

[(2) 三阶段混合算法对复杂问题的分解与整合](#_Toc49_WPSOffice_Level1) [43](#_Toc49_WPSOffice_Level1)

[(3) 引入无人机后的物流车与无人机协同配送方式](#_Toc1692_WPSOffice_Level1) [44](#_Toc1692_WPSOffice_Level1)

[参考文献](#_Toc20324_WPSOffice_Level1) [45](#_Toc20324_WPSOffice_Level1)

[哈尔滨工业大学本科毕业设计（论文）原创性声明](#_Toc25639_WPSOffice_Level1) [47](#_Toc25639_WPSOffice_Level1)

[致 谢](#_Toc3837_WPSOffice_Level1) [48](#_Toc3837_WPSOffice_Level1)

# 绪 论

## 课题背景及研究的目的和意义

### 课题背景

9月5日，2017中国快递“最后一公里”峰会在北京举行，峰会指出中国快递业从2010年的10亿件发展到现在的400亿件。在快递业务量飞速增长，快递物流在人们的日常生活中扮演着十分重要的角色。

而快递运输中的“最后一公里”相关问题尤其突出：“最后一公里物流”是物流配送的最后一个环节，是指客户寄出的包裹，被运输到配送点后，由物流企业通过一定的运输工具将货物从最近的分拣中心送到客户手中，实现门到门的服务。物流成本是一种重要的生产经营要素，为了提高“最后一公里”的配送效率，降低配送成本是至关重要的。早在2010年中国物流与采购联合会发布的数据显示，中国物流总费用占国内生产总值比重约18%左右，其中“最后一公里”占到整个物流配送成本30%以上。刘强东在接受美国财经新闻网站CNBC的独家采访时提到，京东就雇佣了七万名快递员，如果能够用机器人（无人机）来送包裹，最后一公里配送的成本将会很低。

其次，最后一公里配送是城市配送环节中与客户直接沟通和接触的环节，这个环节的质量和效率很大程度上决定了客户的满意程度。配送员的众多不确定的个人因素很有可能成为影响客户体验的重要因素。使用无人车与无人机进行准时精确的配送，将毫无疑问提升用户满意度。

### 研究的目的和意义

本课题组通过研究无人机控制与协同作业管理系统，要实现物流车与无人机协同作业的最后一公里配送自动化系统。该系统控制区域的分拣中心自动分配货物，发出携带无人机的物流车，实现物流车、无人机的协同作业完成区域快件的配送。该系统的完成将极大降低配送成本，提升区域快递最后一公里配送的效率并且提升用户满意度。

我在课题组中负责研究单派送中心多物流车多无人机情况下的路径规划算法的设计，该算法需要在综合考虑快件重量限制、无人车无人机协同工作等复杂条件下，规划出让综合行驶代价和总时间代价尽可能低的快件配送路径。

## 路径规划问题及其相关理论的发展概况

物流配送是物流系统的关键功能之一，涉及从制造工厂或配送中心通过运输网络到消费者的产品流动。运筹学研究文献称之为车辆路径问题（VRP）。 VRP是一种通用名称，指的是一类组合优化问题。其中客户将由多辆车辆提供服务。车辆离开仓库，为客户提供网络服务，并在完成路线后返回仓库。每个客户都有一定的需求。这个问题在Dantzig和Ramser（1959）[1]的文献中首次提出。  
 VRP是NP难题（NP-hard） ，迄今为止，这类问题中没有一个找到有效算法。学术界目前倾向于接受[NP完全问题](https://baike.baidu.com/item/NP%E5%AE%8C%E5%85%A8%E9%97%AE%E9%A2%98" \t "https://baike.baidu.com/item/NP-hard/_blank)（NP-Complete或NPC）和NP难题（NP-Hard或NPH）不存在有效算法这一猜想，认为这类问题的大型实例不能用[精确算法](https://baike.baidu.com/item/%E7%B2%BE%E7%A1%AE%E7%AE%97%E6%B3%95" \t "https://baike.baidu.com/item/NP-hard/_blank)求解，必须寻求这类问题的有效的[近似算法](https://baike.baidu.com/item/%E8%BF%91%E4%BC%BC%E7%AE%97%E6%B3%95" \t "https://baike.baidu.com/item/NP-hard/_blank)。因此，由于商业求解者需要解决的真实世界VRP的大小和频率，他们倾向于使用启发式算法。   
VRP在工业界有许多显著的应用。因为运输通常是产品成本的重要组成部分（10％)[2] ，VRP优化算法的使用通常可以为大多公司节省5％的成本。事实上，交通运输行业占10％欧盟国内生产总值的百分比[3]。因此，VRP节省的成本，即使不到5％也是非常重要的。

### 路径规划算法的发展

在形式上，经典车辆路径问题（VRP）由有向图G(E，V)表示，其中V = {0,1，...，n}表示节点集合，E是弧。 仓库被为节点j = 0，并且客户端是节点j = 1,2，...，n 每个节点需求dj> 0。每个弧表示从节点i到节点j的路线。 每个弧Cij> 0的权重对应于从节点i到节点j的成本（时间或偶数距离）。 如果Cij = Cji，那么我们正面对对称的VRP，否则问题是不对称的。 从复杂性的角度来看，经典的VRP是NP-hard已知的，因为它推广了旅行商问题（TSP）和Bin包装问题（BPP），它们都是众所周知的NP难题。

车辆路径（VRP）问题的研究起步早，经过50多年的研究，成果丰富。车辆路径问题最早由Dantzig and Ramser[1]提出，之后大量影响因素在这个问题中被考虑,例如：路径长度[4]，距离时间限制[5]，收取与分发、时间窗问题[6]等等。

车辆路径问题又根据研究方向不同衍生出例如：

The capacitated VRP(CVRP)[7][8]

Split Delivery vehicle routing problem(SDVRP)[9]

The vehicle routing problem with multiple depots(MDVRP)[10] 等问题。

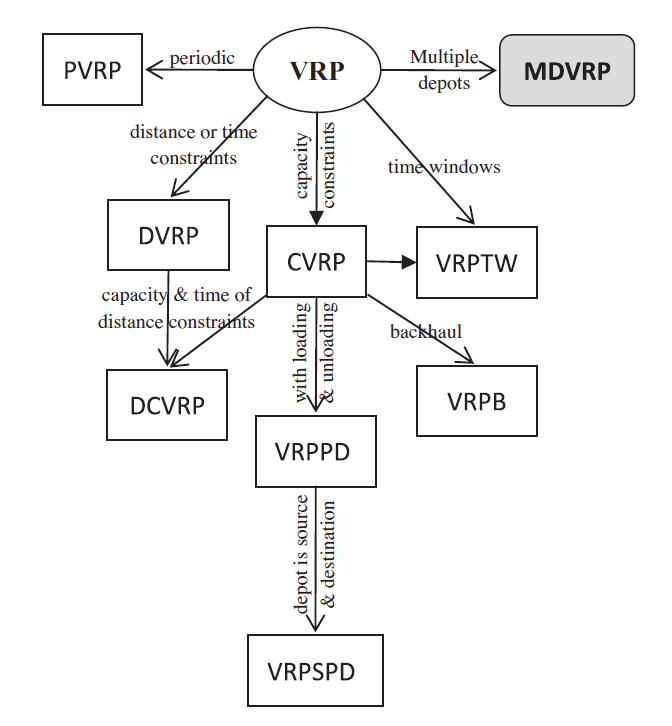


图 1-2-1 现有的VRP问题研究方向

虽然VRP问题又来已久，但是涉及两级车（two-echelon vehicle）的VRP问题却是一个崭新的领域，直到今年才刚开始出现与无人机与运输车辆协同工作相关的研究[11][12]。然而这些文献的主要针对目标在于完成对区域内点的访问任务，并没有考虑无人机的承载能力和货物运输能力,因此物流车无人机进行物流协同配送完成解决最后一公里配送的相关问题还有待解决。

### 聚类算法与路径规划算法结合先例

先聚类再规划路径可以通过将快件点预先分类再将每个聚类块视作整体进行路径规划。这种思想可以通过减少每次算法处理的点数来有效降低问题的复杂程度。近年来涌现了不少的将聚类算法与路径规划相结合的文献。

Gillett (1974)[13]是第一批采用先聚类再规划路径方法来解决VRP的作者。 他们开发了基于扫描的启发式算法，客户根据坐标以及容量大小确定不同的组别。每组数据当作TSP来解决。

Fisher (1981)[14]提出了解决VRP的两阶段算法：第一个阶段中，根据特定标准筛选出每条路径中的快件点，接着第二阶段进行一个路径优化算法。

Qi (2012)[15]发表的论文中综合考虑了空间和时间两个维度的信息对快件点进行聚类。他将代表时间和空间代价放到同一个坐标空间进行考虑，并且研究出一种测量两个快件点间时空间综合距离的方法。

### 无人机配送技术的应用

随着无人技术的日益发展，众多国内外物流公司开始将无人技术应用到运输行业中。

亚马逊谷歌等公司首先将提出研发无人机空管系统的计划[16]。并于2016年利用无人机首次完成货物送达任务[17]。而在2017年首次提出利用地面车辆与空运无人机协同运输的货物运输方式[18]。然而随着无人驾驶技术日益发展和无人机在快递行业逐渐应用[19],之前对于车辆路径问题的求解方法显得有些不足，无人技术使得车辆路径问题有了更好的求解方法：物流车与无人机的协同配送

## 本文研究的主要内容

### 应用场景描述

快件点（轻件点、重件点）：初始生成的快件点只有x坐标，y坐标和重量三个属性，根据无人机的载重进行划分，低于重量阈值的视为轻件点，大于重量阈值的视为重件点，轻件点可以由无人机或者物流车送达，而重件点只能由物流车送达。

模拟真实物流配送场景： 真实的物流配送过程中，快件点的产生总是成局部集聚的趋势（比如在城市中住宅区和商业中心快件点分布密集而公园、广场以及告诉公路沿路快件点稀疏甚至没有），以这个场景为前提，本文实现了模拟真实环境快件点分布的高斯函数模型模拟出了在一片区域中出现数个快件点呈正态分布的集聚中心，这些中心的边缘区域零散分布着少许的快件点。

单快件分发中心： 一片待处理的快件配送区域内只有一个快件分发中心，该中心存有多辆物流车可以并行执行配送任务。

物流车：每辆物流车带有容量限制，每次进行配送任务时能携带的快件重量之和不能超过物流车容量限制。同时每辆车配备多台无人机可以帮助物流车进行协同配送。

无人机：每架无人机也有着容量限制，每次飞行可以携带的轻件重件之和不能超过无人机容量限制。无人机有着作业半径限制，进行轻件点划分的时候将考虑到无人机的作业半径大小。

物流车有着载重量大的优点，然而配送速度慢、配送成本高并且受道路交通限制很多区域无法送达或者配送代价很大。而无人机虽然容量小，但是速度快运输成本低，不受地形与交通情况限制。

目前的传统物流配送算法都是只单纯基于车辆的路径规划，如果结合两者的优点，考虑物流车与无人机的协同配送情况，就产生了全新的问题领域：带无人机的路径规划问题（UAV-VRP），本文从这一全新领域入手，创新性地提出了一种基于区域聚类的物流车与无人机协同配送三阶段混合优化算法，经大量实验证明相比较传统物流配送算法，能够同时有效降低行驶成本与时间成本。

### 算法思想综述

本算法要解决的是基于上述应用场景抽象出的单快件中分发中心，多物流车多无人机进行协同工作完成快件配送任务规划的难题。

算法的核心是一个三阶段的混合算法：

1. 数据预处理阶段：

轻重件分离；轻件聚类；选取聚类中心；重件点归类

1. CVRP全局解阶段：

所有重件点等价重件点进行UGV-Tabu-CVRP求解，规划出物流车配送路径

1. CVRP局部解阶段：

对每个类的不同情况进行三大类七小类的划分分别进行对应处理；每个类内运行UAV-Tabu-CVRP求解

三个阶段分别独立设计出三个模块，三个不同模块通过生成的中间数据文件连结,前一个阶段的输出转为下一个阶段的输入，最终输出计算好的物流车路径和无人机路径文件。同时设计好的显示模块可以分别显示物流车路径图与无人机路径图，代价计算模块可以根据不同的要求计算出路径代价和时间代价。

### 算法创新点

1. 聚类思想在本场景下的适用性改进

综合研究已有的先聚类后路径规划的算法思想之后，我发现由于都是应用在纯物流车配送的场景中，这些算法思想大多是划分完成运送区域后对区域内的所有点进行规划，再整体合并。这种先对小范围区域进行处理的方法，对聚类算法就有着较高的要求，聚类阶段就必须要在考虑到下一阶段整体规划的前提下进行合理划分，否则，先在局部完成该区域的路径规划很有可能会到导致整体性能的下降。

而在我们的应用场景中重件点与轻件点混杂，轻件点尽量使用无人机配送会大大降低配送成本。因此我们场景中的聚类对象为轻件点，聚类限制条件应为无人机覆盖半径。同时也要根据无人机数量对聚类区域的点数进行适当的初步限制。

而在聚类与路径规划的配合方面，与之前的大多数算法思想不同，本算法将采用先整体后局部的方式进行规划：首先对所有轻件点进行聚类，聚类完成后选取聚类中心，把原本的那个轻件点转换成这个类的等价重件点，附上整个类的快件重量。这样生成的等价重件点加入其余重件点中，视为重件点先用改进过的UGV-Tabu-CVRP算法进行一次CVRP求全局最优解，解出每辆物流车应该走的路线，然后再对每个区域内的轻件点进行UAV-Tabu-CVRP求解，解出局部无人机飞行路径的最优解。这样从整体细化到局部，首先保证了物流车代价的最小，其次，完美安排了无人机的最优配送路径，由于无人机配送速度快，区域内可与物流策划并行工作，时间代价主要就是物流车的配送时间。通过本文提出的先整体再局部的聚类-路径规划思想，理论层面上可以将综合代价降低至最小。

1. 三阶段混合算法对复杂问题的分解与整合

本文待解决的应用场景涉及各物流车任务分配、无人机任务分配、物流车无人机容量限制以及物流车无人机协同工作等问题。算法的输入只有所有快件点的坐标和重量，要从简单的输入中提取信息，进行处理从而得到整体的路径规划，就需要对该复杂问题进行分解。我将整体算法主体分为主要三个阶段：数据预处理阶段、CVRP全局解阶段以及CVRP局部解阶段。

数据预处理阶段中，将输入的只有坐标和重量的原始数据先进行轻重件分离，再用改进的K-means聚类算法对所有轻件点进行聚类，并生成等价重件点，最后将在轻件点类内范围内的重件点归入该类。

CVRP全局解阶段中，将数据预处理阶段生成的等价重件点和其他重件点一起导入改进的UGV-Tabu-CVRP解决器中，输出每辆物流车分配的快件点路径。

CVRP局部解阶段中，将之前已经安排好的等价重件点还原成轻件点类，通过路径前后的类内重件点分布进行分类讨论处理先确定起飞点和着陆点，然后根据类内点数决定类内是进行单飞还是进行局部的UAV-Tabu-CVRP，最后输出每个类内的无人机规划好的路径。

1. 引入无人机后的物流车与无人机协同配送方式

目前的无人机配送方式仅是定点起飞定点着陆进行配送，在无人机进行配送的过程中，物流车是停滞不动的，因此浪费了大量等待时间。但是考虑到安全因素以及技术条件限制，移动中无人机的起飞会合是不合理的。因此，受[XX]启发，设计出制定起飞点和着陆回合点的方式，这样能在物流车进行运输的时候尽可能并行派送无人机从而减少总代价。同时改进Tabu-CVRP算法，使其适用于非单点环路（双点单向路：起飞点和着陆点）的情况，实现每架无人机能在载重范围内携带多件快件的UAV-Tabu-CVRP算法。

# 问题模型构建

## 问题场景描述

本文的研究场景可以概括为：在仅有一个快件分发中心的区域中，有多辆（数量充足充足）容量为car\_w的物流车对m个快件收货点进行快递分发，且m个快件收货点上需要送达的总重量不同；其中总重量大于给定阈值K的快件收货点可划分为重件点，否则划分为轻件点，统称为快件点。

物流车携带分配给其的快件和n()架无人机从快件分发中心出发，尽可能以最小的行驶代价和最小的总时间成本，使物流车与无人机进行协同运输，将分配的所有轻重件点全部送达，最后回到快件分发中心。

无人机的配送范围要在其最大作业半径R以内，在单次随车配送过程中假设无人机电量充足，且能在物流车上实时充电。

## 数学符号定义

用到的概念转换为数学符号：

|  |  |
| --- | --- |
| 符号 | 描述 |
| car\_w | 无人车最大载重 |
| car\_p | 无人车单位距离行驶成本 |
| car\_s | 无人车速度 |
| disij | 第i个快件点到第j个快件点的距离 |
| K | 无人机最大载重量（轻重件划分阈值） |
| class\_num | 类数量 |
| starti | 类i中的起飞点 |
| endi | 类i中的着陆点 |
| flight\_s | 无人机飞行速度 |
| fi | 无人机在类i中的路径数 |
| flight\_dij | 无人机在类i中第j条路径的飞行的距离 |
| flight\_wij | 无人机在类i中第j条路径上的快件点载重 |
| m | 快件点总数 |
| car\_ri | 给第i辆车分配的快件点集 |
| wi | 快件点i的快件重量 |
| yij | 物流车直接从快件点i运送到快件点j时置1 |
| cpi | 快件点i由物流车送达时置1 |
| opi | 快件点i由无人机送达时置1 |

表 2-2 数学符号定义

## 数学模型构建

根据2.1中的问题场景描述，运用2.2中给出的符号定义建立本文要解决问题的数学模型：





 (1)

 (2)

 (3)

 (4)

Travel\_cost为物流车与无人机行驶代价之和即总行驶代价，通过将计算出的物流车与无人机路径长度总和分别乘以相应代价系数再求和就能得到总体的代价大小。

Time\_cost为物流车与无人机运行完所有快件所需的时间代价，计算方法为将物流车行驶需要的时间减去起飞点到着陆点间物流车行驶需要花费的时间再加上这两点间无人机最长路径需要花费的时间和物流车行驶时间的最大值。

本算法的优化目标就是在加入无人机的情况下尽可能降低这两者的大小。

约束条件(1)表明默认无人机配送速度比物流车快，且物流车配送代价比无人机高；

约束条件(2)表明每架无人机配送任务携带的快件总重必须不大于无人机最大载重量K；

约束条件(3)表明每辆物流车携带的快件总重量必须下雨不大于物流车最大载重量car\_w；

约束条件(4)表明每个快件点都必须送到且只由物流车或者无人机访问一次。

通过构建上述数学模型，将真实应用场景中的复杂问题抽象为数学问题清晰展示出来，应用场景中的约束条件也都悉数加以考虑，本文算法优化的目标就清晰地转换为对Travel\_cost与Time\_cost尽可能求最小值。

# 问题模型求解

## 算法思想综述

本算法解决了单快件中分发中心，多物流车多无人机进行协同工作完成快件配送任务规划的难题。本算法的核心是一个三阶段的混合算法，三个阶段分别独立设计出三个模块，三个不同模块通过生成的中间数据文件连结,前一个阶段的输出转为下一个阶段的输入，最终输出计算好的物流车路径和无人机路径文件。同时设计好的显示模块可以分别显示物流车路径图与无人机路径图，代价计算模块可以根据不同的要求计算出路径代价和时间代价。

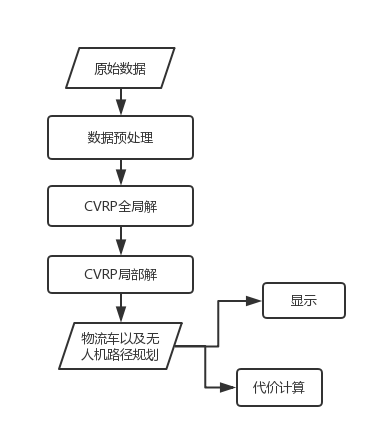


图 3-1-1 算法总流程图

## 各模块功能详解

算法总共分为：预处理模块、CVRP全局解模块、CVRP局部解模块、显示模块和代价计算模块。接下来分别对各个模块进行详细地介绍

### 预处理模块

预处理模块的功能总述：

先将输入的只有坐标和重量的快件点文件(Datasetx,xlsx)拆分成重件点（HP.xlsx）和轻件点文件(LP.xlse)并分别添加新的初始属性数据保存,然后运行计算路网矩阵的函数先生成轻件点路网矩阵；

接着进入此模块的核心部分将所有轻件点数据以及轻件点路网矩阵导入改进后的K-means聚类函数中，聚类函数对轻件点数据进行迭代处理直到每个类内数据都满足问题模型的限制要求，更新轻件点表的数据；

然后运行生成聚类中心的函数，在这个函数中，分别遍历每个类中的所有点选出到其他点距离的加权平均和最低的那个点设为该类的聚类中心即等价重件点，标记后添加到重件点表中，同时在遍历类内轻件点过程中判断上下左右边界，将类边界内的所有重件点都归为同一类并标记，更新重件点表；

然后生成现在的轻件-重件路网代价矩阵和重件点代价矩阵。

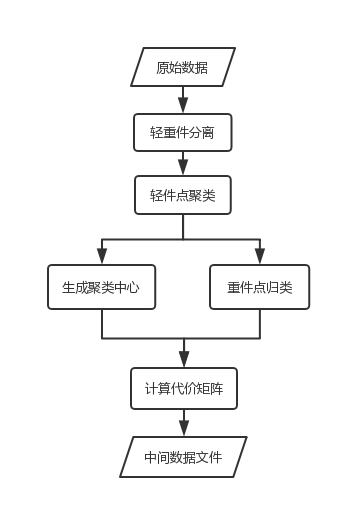


图3-2-1-1 数据预处理模块流程图

轻重件分离函数（Split\_LH）：

对于原始数据，轻重件分离函数将每个数据的重量属性与阈值K比较，不大于K的为轻件点，否则划分为重件点，分别添加type,class属性，并存入对应数据文件中。

轻件点路网代价生成函数（Generate\_LPCostmatrix，Generate\_LHCostmatrix，Generate\_HPCostmatrix）：

本文中的路网代价都是用欧式距离直接表示的，路网代价生成函数中， 对每两个点都进行遍历，计算出他们之间的欧式距离，并保存到路网代价矩阵中。

改进的K-means聚类函数（LP\_Cluster）：

这部分为预处理模块的核心算法，经过多种聚类算法的编写与测试，最后设计出了一种基于K-means聚类算法的带限制的迭代求解K-means聚类算法（将在3.3中详细解析），输入分离后的所有轻件点数据，聚类算法对轻件点进行迭代分类，每次迭代增加类的个数，直到所有类都满足聚类半径小于无人机工作半径且类内聚类点个数不超过给定值时退出迭代，根据分类结果更新轻件点表。

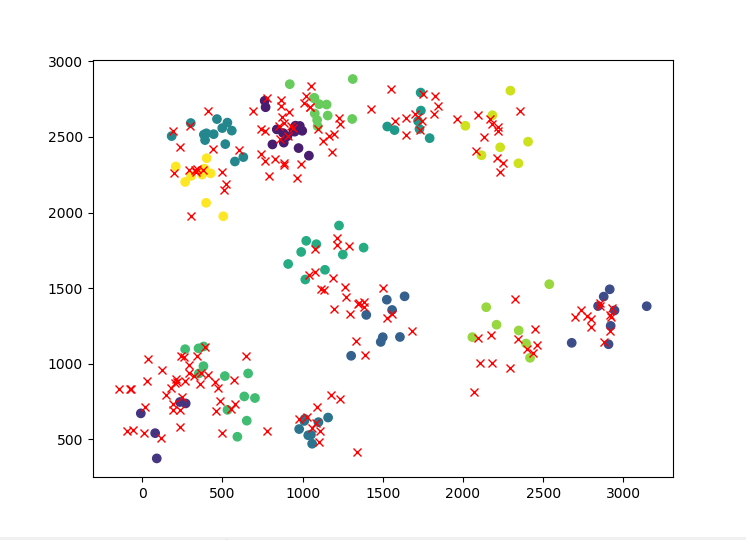


图 3-2-1-2 聚类模块效果图

生成聚类中心函数（Generate\_Center）：

首先，这个函数将划分好的每个类中所有点计算该点到其余所有点的加权平均和，取该值最小的点作为等价重件点会使该点到其余类内点的距离尽可能小且均匀。等价重件点确定后进行重新赋重（将整个类中轻件点的重量赋给该点）并添加相应的type，class属性并加入重件表中。

然后每类的边界也同时进行了判断，取类边界的方法也经过多次测试最终决定采用边界点左边进行矩形划分（使用以聚类中心为圆心的方案会导致大量点重叠，无法有效分类处理），类边界内包含的重件点将在类属性中相应地改变值，更新重件表。

最后就是导出初始化的Classtype表，这个表是为了CVRP局部解时记录每个类采用的无人机配送方式以及起飞着陆点而存在的。

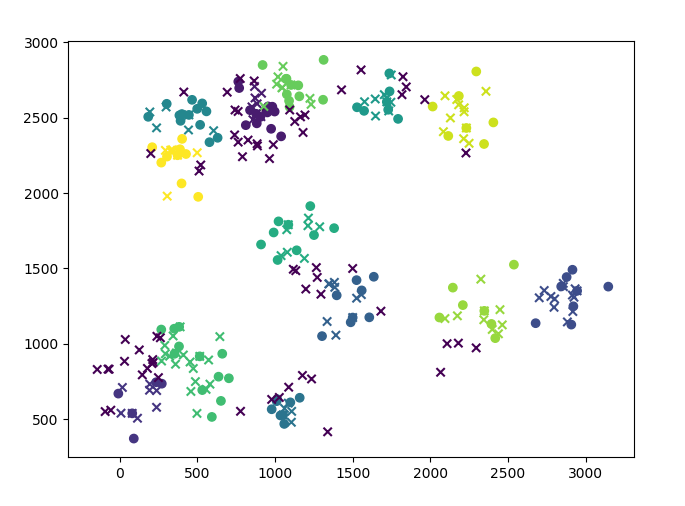


图 3-2-1-3 重件归类效果图

### CVRP全局解模块

CVRP全局解模块用的是改进的禁忌搜索算法，禁忌搜索算法是启发式算法的一种也是本文的核心算法之一，将在3.3中对其进行详细介绍，本文先基本实现单程的禁忌搜索VRP解决器，再参考[]改进为CVRP问题解决器，由于禁忌搜索算法受初始解影响较大，再加上了多初始解并行运行取最优的机制，最终完成了UGV-Tabu-CVRP算法。该模块读取预处理模块处理好的带有等价重件点的重件表，对所有重件进行了CVRP规划，生成物流车的初步路径规划结果。

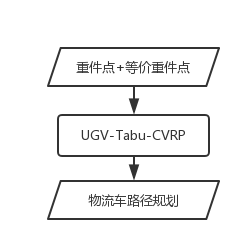


图 3-2-2-1 CVRP全局解模块流程图

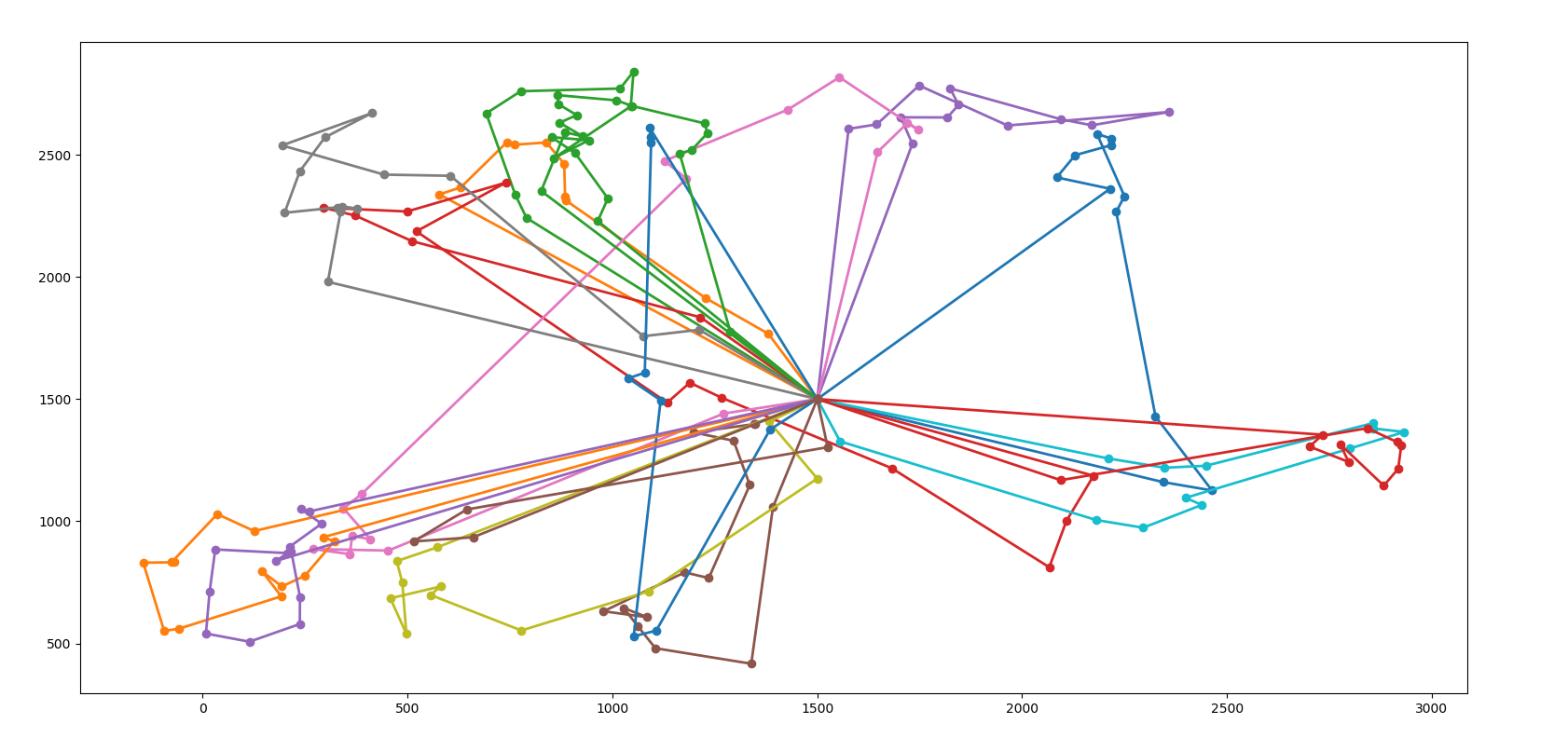


图 3-2-2-2 CVRP全局解模块规划效果图

### CVRP局部解模块

CVRP局部解模块里由两个核心算法，一个是根据无人机派送情景改进的两点单向的UAV-Tabu-CVRP算法；另一个是对每个轻件点类内出现的所有的三大种、七小种情况进行了全面的分治，通过对所有情况的分类考虑，在类内点较多超过无人机数目时进行UAV-Tabu-CVRP求解尽可能降低无人机配送的行驶成本，在类内点较少时进行无人机直飞配送规划，使配送时间最短。（核心算法都将在3.3进行更细致的讨论）。

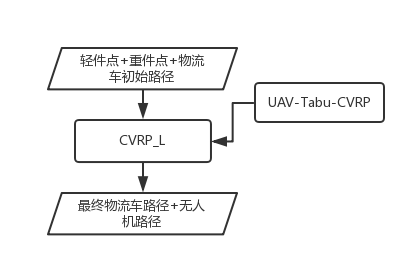


图 3-2-3-1 CVRP局部解流程图

UAV-Tabu-CVRP 算法：

由于应用场景和之前的全局CVRP情景不同（之前是单快件分发中心，规划出的每条结果都呈环路；而现在是从指定起飞点到终点的单向路径）数据结构与过程计算都要进行适用性大改。

CVRP\_L 分治算法：

由于轻重件点分布的随机性、初步路径规划结果的不可预测性以及尽可能减少总行驶代价的考量，根据每个类的具体情况对每个类的无人机规划策略进行划分。

从等价重件点出发向所在路径两边延伸，直到遇到的重件点不是该类类内重件点。

1. 两边都有类内重件点

两边的端点入口处设为起飞点，出口处设为着陆点

1. 轻件点数量小于无人机数量：不进行UAV-Tabu-CVRP局部求解每个点直飞配送
2. 轻件点数量大于无人机数量：进行UAV-Tabu-CVRP局部求解，输出路径结果
3. 两边都没有类内重件点
4. 类内轻件点小于等于无人机数量：单定点往返直飞配送
5. 类内轻件点大于无人机数量：

两两遍历所有类内点，找出两个轻件点使与路径前后两个最近的重件点路径长度之和最低（这样就能最大限度地降低物流车的行驶代价），转换为等价重件点，加入路径中，更新重件点表。

两边的入口处等价重件点设为起飞点，出口处等价重件点设为着陆点

1. 轻件点数量小于无人机数量：不进行UAV-Tabu-CVRP局部求解每个点直飞配送
2. 轻件点数量大于无人机数量：进行UAV-Tabu-CVRP局部求解，输出路径结果
3. 一边有类内重件点一边没有

有类内重件点的那头的重件点和等价重件点两者构成起飞和着陆点

1. 轻件点数量小于无人机数量：不进行UAV-Tabu-CVRP局部求解每个点直飞配送
2. 轻件点数量大于无人机数量：进行UAV-Tabu-CVRP局部求解，输出路径结果

### 显示模块

显示模块分为物流车路径显示模块和无人机路径显示模块。

两个模块都是先读取前面程序运行过程中生成的物流车和无人机路径结果文件New\_routes.csv或UAV\_routes.csv（通过python csv库里的reader函数逐行读取）

最后逐行存入数据矩阵中,再调用matplotlib里的绘图函数对每条路径进行绘制。效果如图。

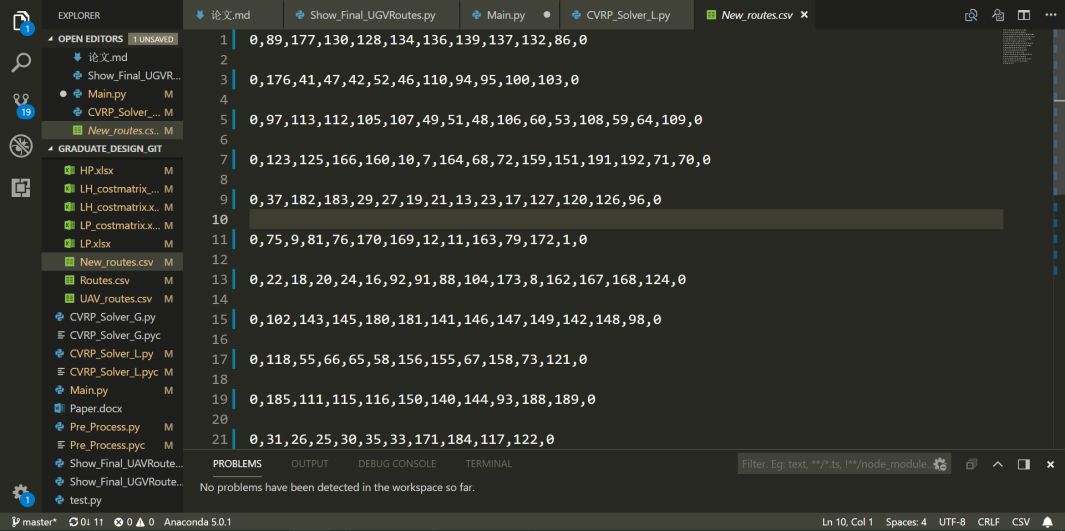


图 3-2-4-1 物流车路径文件

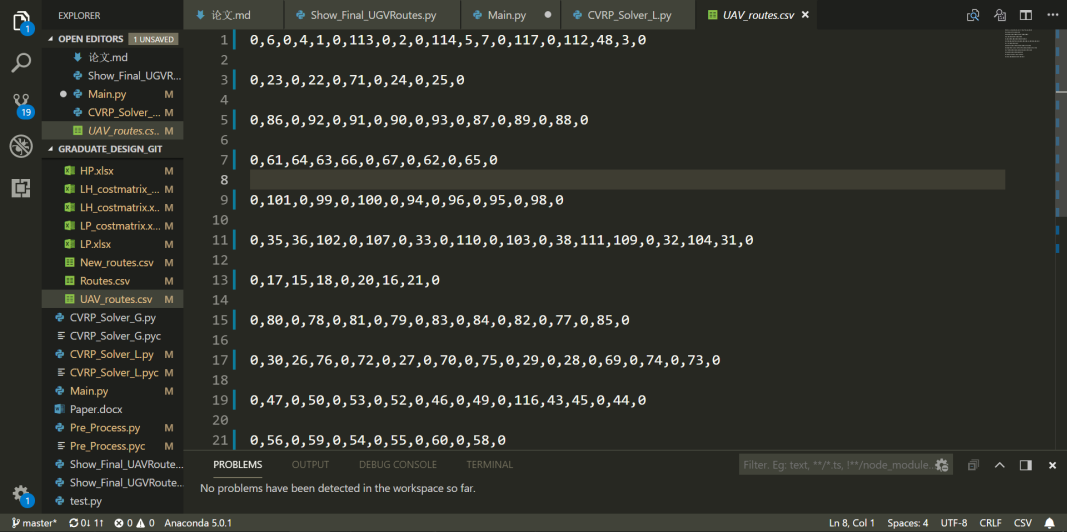


图 3-2-4-2 局部无人机路径文件

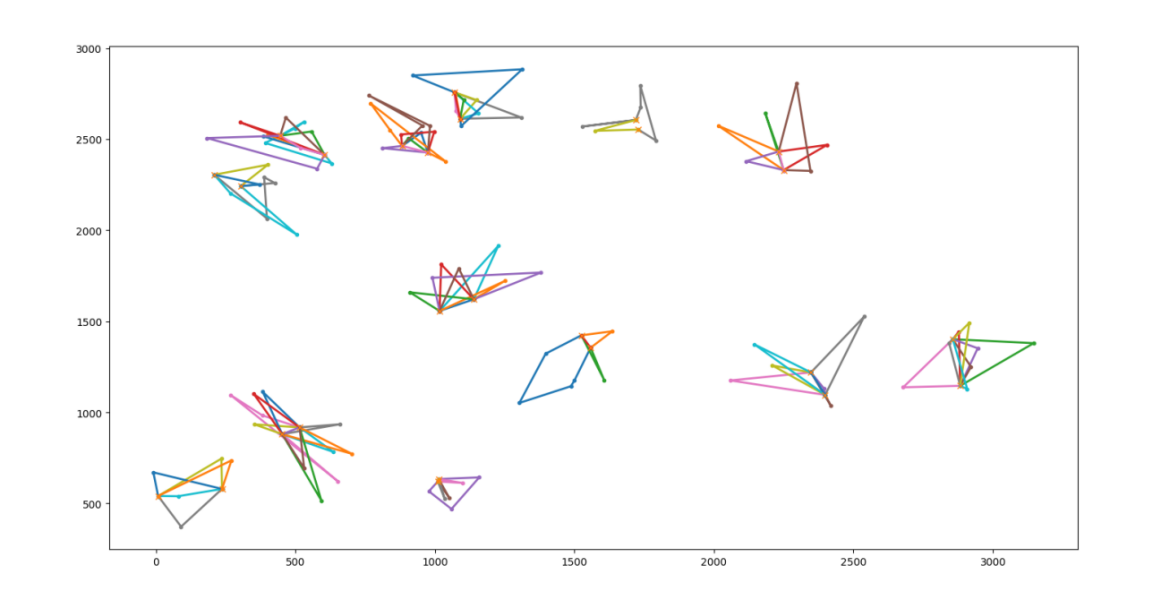


图 3-2-4-3 局部无人机路径规划结果图

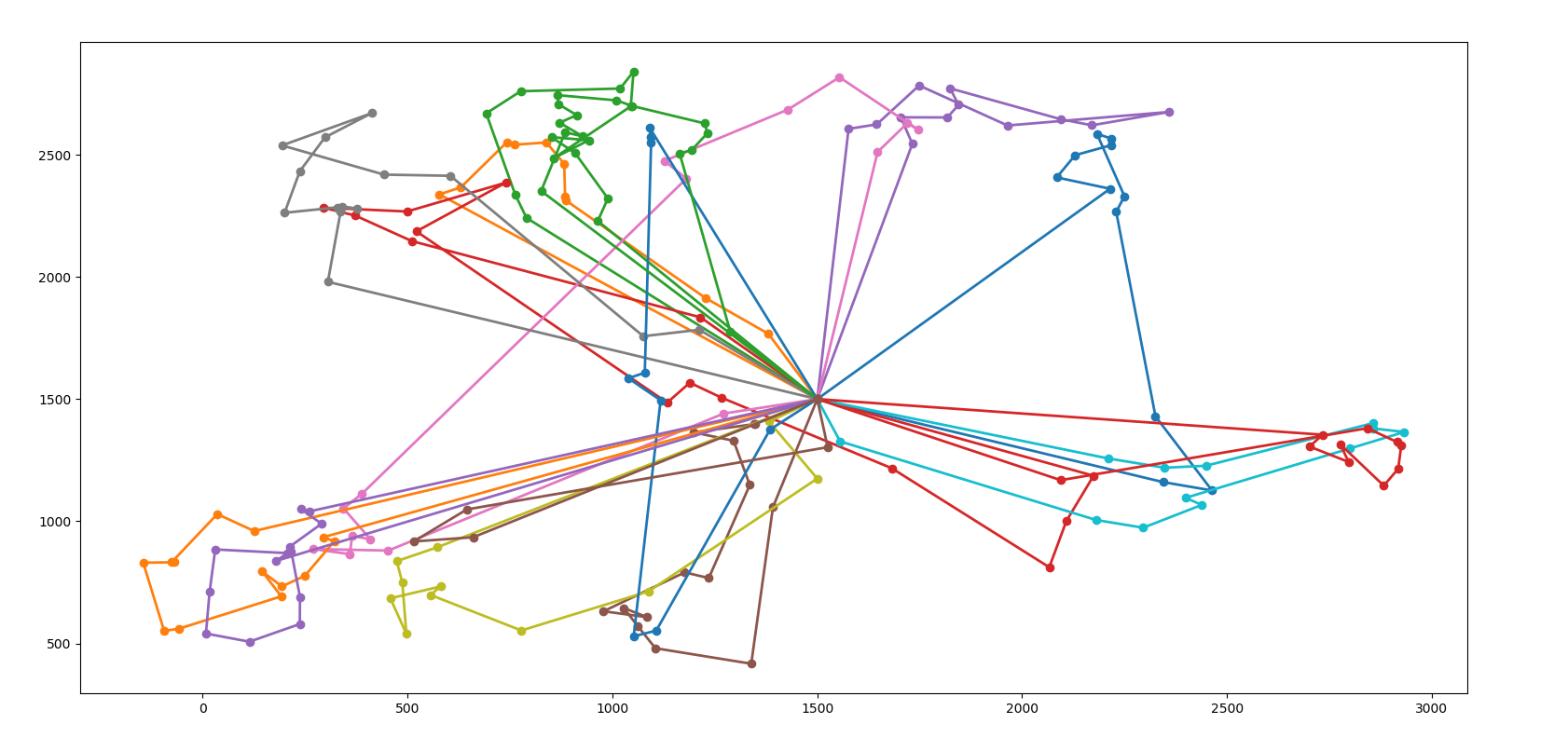


图 3-2-4-4 更新后的物流车路径图

### 代价计算模块

无人机个数的限制在代价计算模块中进行考虑。在进行UAV-Tabu-CVRP规划之后得到的路径数量很有可能大于无人机数量，因此需要进行处理。首先对所有路径的代价进行计算，进行从大到小的排序。保留无人机数量个代价较大的路径，其余点从就近的起飞降落点进行来回直飞。

代价计算模块也是先读取前面程序运行过程中生成的物流车和无人机路径结果文件New\_routes.csv或UAV\_routes.csv（通过python csv库里的reader函数逐行读取），先计算所有路径长度之和，分别得到物流车行驶路径总长与无人机行驶路径总长，再根据问题模型求解里给出的公式，分别对两个计算到的路径之和乘以相应的代价系数，最终得到总的行驶代价。



时间代价计算比行驶代价计算要复杂，首先需要计算出物流车行驶的总时间（总路径长度除以物流车速度），然后减去起飞点和着陆点之间物流车行驶需要花费的时间，然后挑选出的每个类内无人机飞行的最远路径长度，计算出每个类内无人机路径最长的那条路径需要花费的时间（无人机路径长度除以无人机飞行速度）和物流车从每个类内起飞点着陆点花费时间的最大值。



## 核心算法解析

### 带限制条件的的迭代求解K-means聚类算法

本文中使用的聚类算法经过两次编写尝试。

#### 初版分裂法聚类算法

初版的聚类算法，我在研究了各种聚类算法思想[20]之后选择了分裂法聚类。分裂法聚类先将所有样本归为一个大的初始类，然后在不满足条件时反复分裂，直到满足限制条件要求时停止。由于我们场景中需要判断的限制条件是聚类半径大小和类内点的个数，这些条件可以在迭代分裂过程中轻易检测出来，这就使得聚类算法速度很快，但是却没有考虑我们聚类算法应用场景是局部点聚集的情况。分裂法聚类算法在样本分布均匀的情况下效果很好，但是在局部点聚集情况下却十分不理想。

分裂法聚类伪代码：

1. 将样本集中的所有的样本归为一个初始类簇c0；

Repeat：

1. 在同一个类簇（计为c）中计算两两样本之间的距离，找出距离最远的两个样本a, b
2. 将样本a，b分配到不同的类簇c1和c2中
3. 计算原类簇（c）中剩余的其他样本点和a，b的距离，若是dis(a)<dis(b)，则将样本点归到c1中，否则归到c2中；

Until：类内最远距离大于阈值

对初版的聚类算法进行测试，测试结果如图3-3-1-1所示：

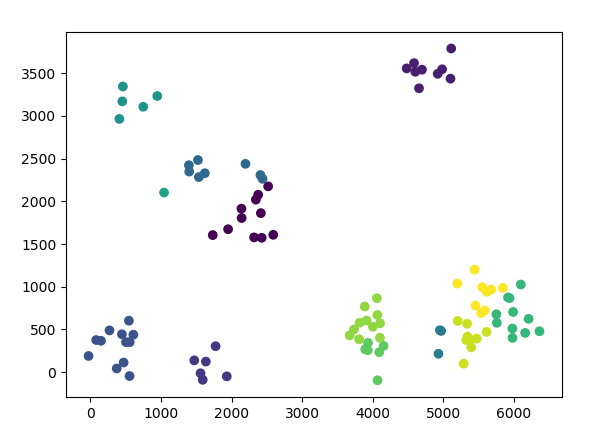


图3-3-1-1 分裂法聚类结果图

由于每次只是简单将点归于当前类的最远两个端点，首先只选取两个端点作为聚类标准就十分不科学，其次当出现两个最远点两有一侧点数量十分稀疏的时候，这个区域内的点，即使各自分隔稀疏，这个类的分裂也会结束，因此聚类的效果就很差，经过仔细思考后判断分裂法聚类不适用于本算法的场景。图中也可以看出即使点很少的情况下聚类结果还有明显的不合理点。

#### 带限制的K-means迭代聚类算法

在经历了第一次聚类算法设计失败后，查阅了各种对于局部聚集情况聚类效果出色的聚类算法，最后决定了以K-means聚类算法为基础进行适用于本算法的改造。

K-means聚类伪代码：

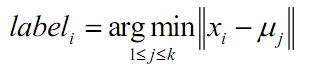
1. 从数据中随机抽取k个点作为初始聚类中心

Repeat:

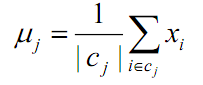
1. 计算各个点到k个中心的距离，将其归到最近的聚类里
2. 调整聚类中心，移到当前类的几何中心

Until:聚类中心不再移动

对于每个样本xi，将其标记为距离类别中心μj最近的类别，即：



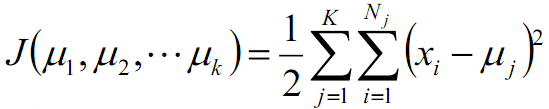
将每个类别中心更新为隶属该类别的所有样本(cj为j类中所有轻件点个数)的均值



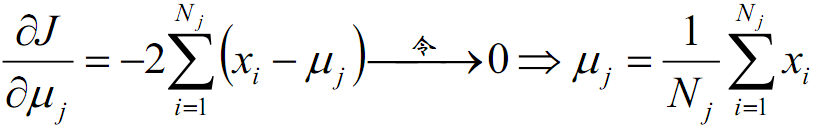
K-means算法公式化解释：

  记k个簇中心分别为μ1,μ2，μ3……μk,每个簇的样本数目为N1、N2……Nk。

使用平方误差做为误差函数，得：



将该函数做为目标函数，求解该函数的最小值。可以使用梯度下降法求，该函数为凸函数，驻点为：



可以看到，要想使损失函数最小，聚类中心要为各簇中样本点的平均值。由此可以看出，K-means算法在每次迭代更新时使用各簇中样本点的平均值为聚类中心是有道理的。

通过重复迭代K-means算法，逐渐增加类的个数直到满足限制条件（每个类的点数不大于预先设置的阈值以及聚类半径不大于无人机的工作半径。因为K-means聚类算法从聚类中心出发对所有点进行划分，并且在每次划分结束后进行聚类中心的重新选取，因此K-means算法在对集聚型数据进行处理时十分有效。K-means迭代算法实现效果如图3-3-1-2

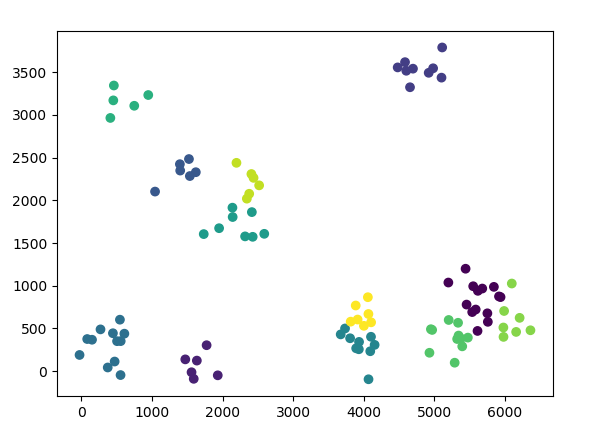


图3-3-1-2 限制条件为15点&1200聚类半径的K-means聚类迭代

图中可以看出每个类内的点都被很好的进行了划分并且都满足限制条件要求。

### Tabu-CVRP算法

本文算法设计的另一个核心是以禁忌搜索（Tabu-Search）算法为基础的VRP解决算法。

在本算法设计过程中经历了

1. 用禁忌搜索算法实现单环路车辆路径规划算法（Tabu-VRP）
2. 用禁忌搜索算法实现带容量限制的车辆路径规划算法（Tabu-CVRP）
3. 改进禁忌搜索算法，最终得到适用于物流车配送的带容量限制的车辆路径规划算法（UGV-Tabu-CVRP）
4. 根据无人机场景中的起飞点着陆点不同，改造前面得到的Tabu-CVRP算法的数据结构得到适用于无人机配送的带容量限制的载具路径规划算法（UGV-Tabu-CVRP）

下面进行详细介绍：

#### Tabu-VRP

禁忌搜索算法伪代码：

1. 随机生成一个初始解X0设置为当前解Xnow，建立空的禁忌表

Repeat:

1. Xnow生成若干邻域解（2-交换、不能在禁忌表中出现）
2. 在Xnow领域中找出代价最小的解更新Xnow，更新禁忌表

Until:满足要求或者达到固定迭代次数

禁忌搜索（Tabu Search或Taboo Search，简称TS）的思想最早由Glover(1986)提出，它是对局部领域搜索的一种扩展，是一种全局逐步寻优算法，是对人类智力过程的一种模拟。其特点是采用禁忌技术，即用一个禁忌表记录下已经到达过的局部最优点，在下一次搜索中，利用禁忌表中的信息不再或有选择地搜索这些点，以此来跳出局部最优点。

在禁忌搜索算法中，首先按照随机方法产生一个初始解作为当前解，然后在当前解的邻域中搜索若干个解，取其中的最好解作为新的当前解。为了避免陷入局部最优解，这种优化方法允许一定的下山操作（使解的质量变差）。另外，为了避免对已搜索过的局部最优解的重复，禁忌搜索算法使用禁忌表记录已搜索的局部最优解的历史信息，这可在一定程度上使搜索过程避开局部极值点，从而开辟新的搜索区域。

禁忌搜索最重要的思想是标记对应已搜索的局部最优解的一些对象，并在进一步的迭代搜索中尽量避开这些对象（而不是绝对禁止循环），从而保证对不同的有效搜索途径的探索。禁忌搜索涉及到临域（neighborhood）、禁忌表（tabu list）、禁忌长度（tabu length）、候选解（candidate）、藐视准则（aspiration criterion）等概念

禁忌算法其主要特点是在搜索开始阶段，解的质量提高很快，随着搜索过程的继续，解的质量的提高速度逐渐放缓，甚至在很长的搜索阶段内解的质量没有太大提高，适合中小规模的NP问题求解，整体效率比较均衡。

禁忌搜索算法特点主要体现在如下四个方面

1. 基本思想——避免在搜索过程中陷入局部最优
2. 只进不退的原则,通过禁忌表实现
3. 不以局部最优作为停止准则
4. 邻域选优的规则模拟了人类的记忆功能

根据上述禁忌搜索算法的思想编写出本文实现的初版禁忌搜索算法代码，初版算法流程如下图：

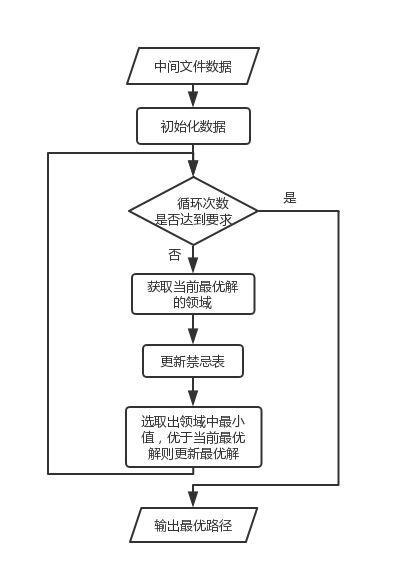


图 3-3-2-1 禁忌搜索算法流程图

中间文件的数据导入到禁忌搜索算法中后，算法先进行初始化，将所有算法要用到的全局变量赋初值；然后开始进入循环部分，每次循环中先生成当前的全局最优解（初始化中赋值为随机乱序的路径串）的领域解（采用随机2-opt交换方式）；然后取领域中的最优解与当前的全局最优解做比较，如果优于当前全局最优解的话，就用这个领域中的最优解更新全局最优解，然后继续开始下一次循环，直到循环次数满足要求。

禁忌搜索算法中的每一个解都代表着一种路径规划方案，循环结束得到算法计算出的全局最优解的时候，得到的解也就是单环路的路径规划结果。

实现基本的禁忌搜索算法后，运行TSPLIB att48测试集：

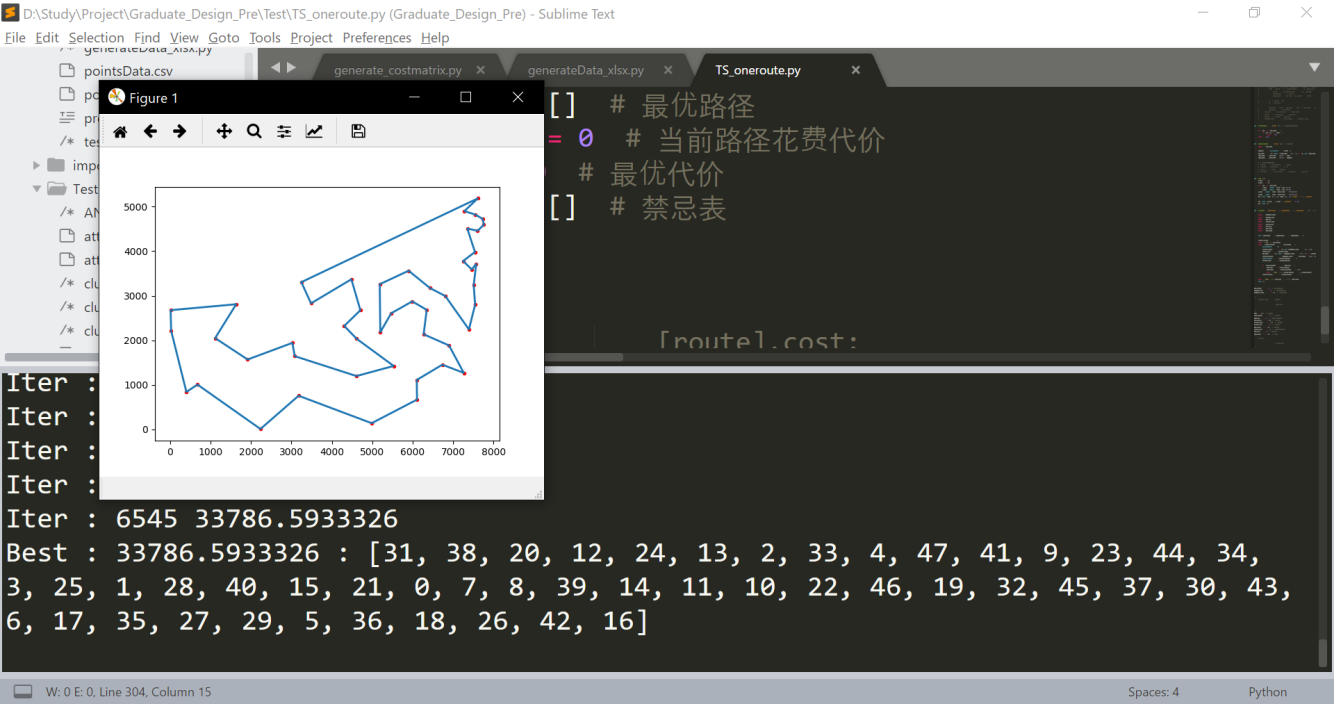


图3-3-2-2 单路程禁忌搜索算法 结果

Att48测试集的最优解是33522，实现的初版禁忌搜索算法最好能够达到33786的效果，已经很令人满意了。

#### Tabu-CVRP

如何将单路程的禁忌搜索算法改进为CVRP，是接下来需要考虑的问题。在单路程禁忌搜索算法中，每个解用一个不出现重复数的长数组List表示。

如何进行改造使Tabu-VRP算法也能应用于CVRP问题呢，查阅了多篇禁忌搜索算法在CVRP问题上的应用相关的文献[21][22]之后我有了改造的思路，我决定采用类似的策略，即通过在每次生成解的时候检索数字串，在当前扫描过的数字串所代表的快件点重量之和大于物流车容量的时候插入代表物流中心的0数字，扫描完成后该数列代表的真实路径规划就产生出来。然后再计算真实路径对应的代价。

这时编写算法改动的地方就是

1. 快件中心0加入重件点表
2. 编写生成真实路径函数，输入数字串后自动生成由0分隔开的真实路径
3. 生成领域函数中的生成代价函数改为计算由快件中心0分隔开的真实路径代价

改写Tabu-VRP算法，得到的新Tabu-CVRP算法运行结果如图所示

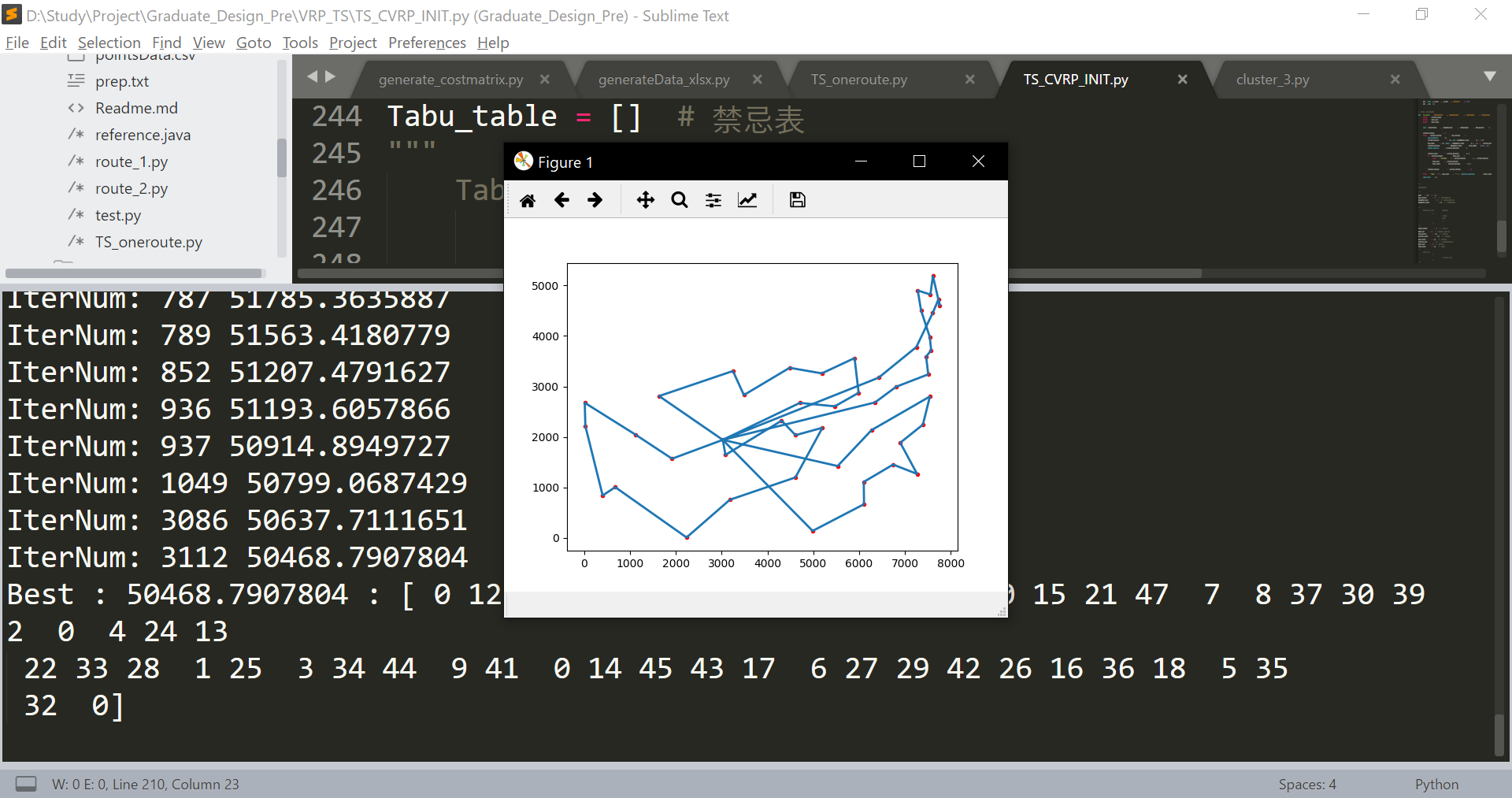


图3-3-2-3 Tabu-CVRP 结果图

规划后的路径已经出现了清晰的环路，总代价为50000以上，路径规划效果还行。但是算法很容易陷入局部最优，导致代价没有缩到最小，接下来就对这方面进行优化。

#### UGV-Tabu-CVRP

由于禁忌搜索是以初始解为根开始进行迭代的，算法得出的结果受初始解影响较大，不同的初始解可能导向不同的局部最优解。上面的Tabu-CVRP算法中采用的初始解就是一个随机生成路径序列转换成的真实路径，当算法开始运行的时候就注定了最终结果会过早地收敛到局部最优。因此，为了使算法结果代价尽可能的小，所以我决定使用多初始解，也就是随机生成多个初始解，同时进行迭代，迭代到一定次数后再选出当前最优秀的路径进行接下来的迭代，同时改进了路径显示函数，将每条路径分别导出再进行绘制，使结果更加清晰，同时加上路径上色，效果如图所示：

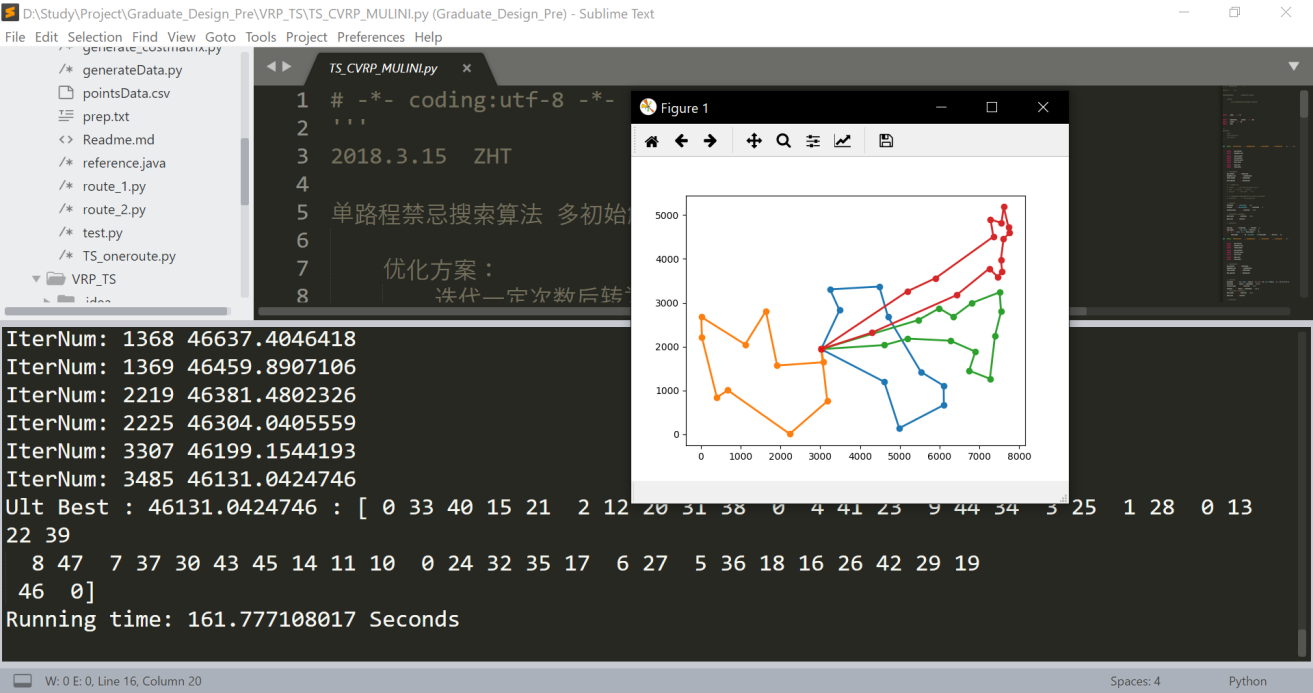


图3-3-2-4 UGV-Tabu-CVRP 效果图

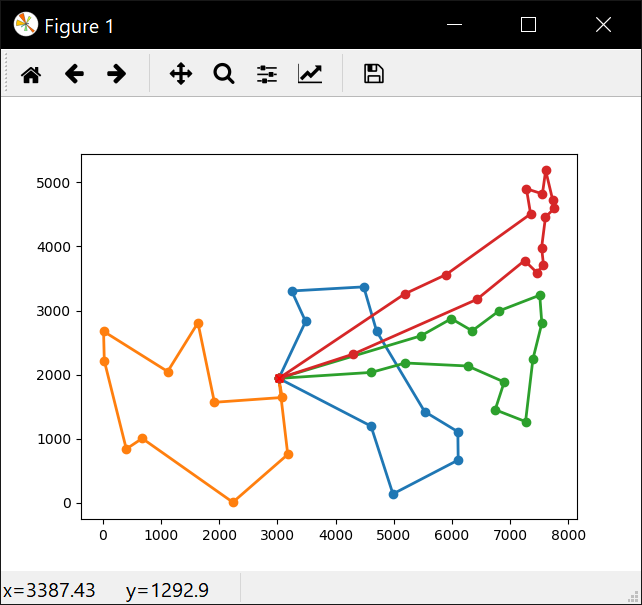


图3-3-2-5路径效果图放大

优化后的算法在运行同一测试集时代价降低到了46131，总体能比上个版本节约3000--4000的路径代价，规划出的路径也是十分清晰的环路，达到了预期效果。

#### UAV-Tabu-CVRP

最初设想无人机都是单定点起飞降落，但是这样会导致物流车在无人机作业过程中一直处于等待状态，浪费大量的时间，在参考了Zhihao Luo的关于物流车无人机协同工作完成区域覆盖的论文[23]后，我将先前的思路改进为设置起飞点和着陆点，无人机在物流车到达类内起飞点后全部放出进行并行配送作业，同时物流车开往着陆点，并在着陆点等待与无人机会合。

大不相同的情景理应导致数据结构乃至算法的大改动，但是经过仔细地思考之后，我参考了先前Tabu-CVRP的解决方案，思考出了新的类似的无人机CVRP模块的解决方案:

同样在Tabu-VRP的基础上设置生成真实路径的函数，同样是当重量达限制条件后添加快件分发中心0，在将数字串生成真实路径之后代入新的计算代价函数中。

新的类内无人机路径代价计算函数:

1. 将每个0分隔出的中间数列分别取出，各自计算代价
2. 将每个中间数列都加上头尾轻件点分别到起飞点和着陆点的距离
3. 全部求和得到正确的当前路径规划解的总代价

其余算法部分与Tabu-VRP相同。

对新的算法进行测试，结果图如下。

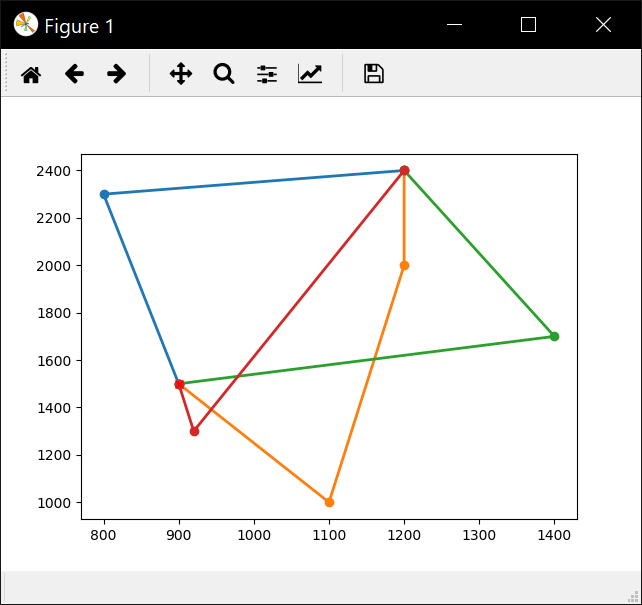


图3-3-2-6 无人机局部路径规划

### 局部类内分治规划算法

在算法的CVRP\_L模块中，进行由于轻重件点分布的随机性、初步路径规划结果的不可预测性以及尽可能减少总行驶代价的考量，根据每个类的具体情况对每个类的无人机规划策略进行划分。

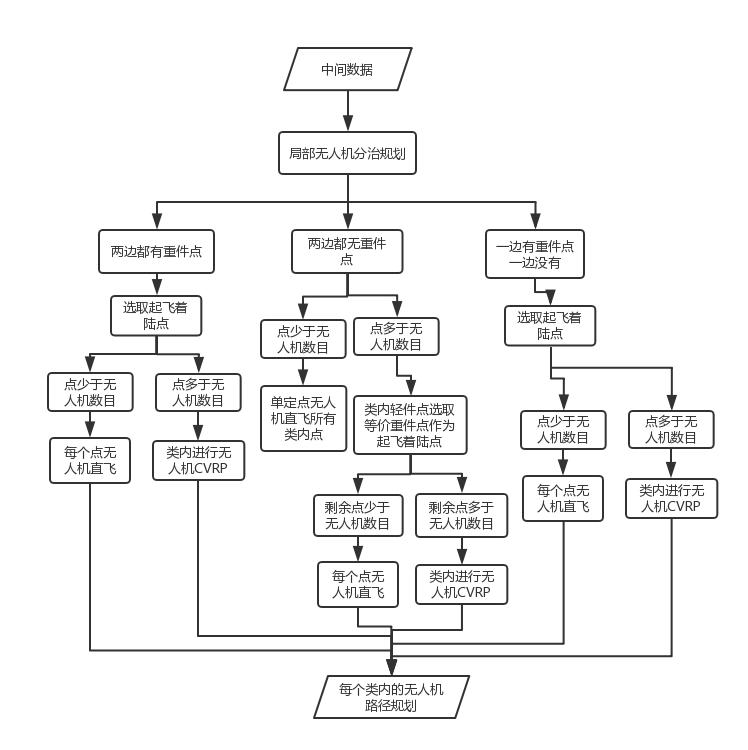


图 3-3-3-1 局部类内分治规划算法流程图

从等价重件点出发向所在路径两边延伸，直到遇到的重件点不是该类的类内重件点。

1. 两边都有类内重件点

两边的端点入口处设为起飞点，出口处设为着陆点

* + - 1. 轻件点数量小于无人机数量：不进行UAV-Tabu-CVRP局部求解每个点直飞配送
      2. 轻件点数量大于无人机数量：进行UAV-Tabu-CVRP局部求解，输出路径结果

1. 两边都没有类内重件点
   1. 类内轻件点小于等于无人机数量：单定点往返直飞配送
   2. 类内轻件点大于无人机数量：

两两遍历所有类内点，找出两个轻件点使与路径前后两个最近的重件点路径长度之和最低（这样就能最大限度地降低物流车的行驶代价），转换为等价重件点，加入路径中，更新重件点表。

两边的入口处等价重件点设为起飞点，出口处等价重件点设为着陆点

* + - 1. 轻件点数量小于无人机数量：不进行UAV-Tabu-CVRP局部求解每个点直飞配送
      2. 轻件点数量大于无人机数量：进行UAV-Tabu-CVRP局部求解，输出路径结果

1. 一边有类内重件点一边没有

有类内重件点的那头的重件点和等价重件点两者构成起飞和着陆点

* + - 1. 轻件点数量小于无人机数量：不进行UAV-Tabu-CVRP局部求解每个点直飞配送
      2. 轻件点数量大于无人机数量：进行UAV-Tabu-CVRP局部求解，输出路径结果

经过这三大类7小类讨论，基本上所有情况已经都涵盖进去，运行完此算法后每个类内无人机的路径规划就都得以确定。

# 实验及结果分析

## 实验简介

首先为了模拟本文算法所应用的局部集聚的真实情况，首先我设计出了模拟数据生成模块，该模块通过应用了高斯函数变化，可以自行设置大致的点聚类类数，集聚程度，快件重量范围，从而大致控制输出的局部集聚的模拟数据的分布情况。

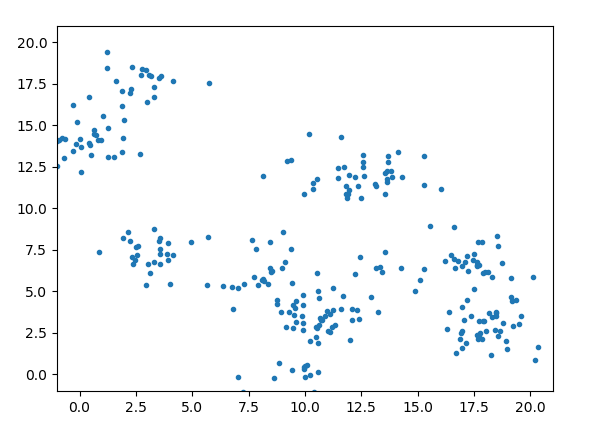


图 4-1-1 模拟数据生成模块效果图

测试阶段：

由于启发式算法受数据个数影响较大，作为程序的底层实现思想，有必要对不同数量的点数进行测试。设计的测试集分为50，150，250，300个点左右的生成的模拟数据。

其次是轻重件比例，本文研究的是物流车与无人机的协同 运输，轻件比例不同会很大程度上影响到包括聚类算法、局部无人机分治规划算法等一系列算法的性能，因此对其进行测试很有必要。

除了上面提到的两点，无人车载重、无人机数量以及无人机速度也对本文算法的运行结果有影响吗，都需要进行具体测试。

## 实验过程

实验平台介绍：

DELL XPS15 9550 CPU : Intel i7-6700HQ 2.6GHz

实验目的 :

1. 与传统物流算法对比，展示新算法优势
2. 算法自身测试，性能影响因素研究

实验过程设计：

无人机与物流车的行驶代价比例为0.2 ：1

1. 固定轻件占比40%左右，无人机数量为4，无人机速度为3，物流车载重 为500。分别生成50点左右，100点左右、200点左右以及300点左右集 进行测试。同时这里在传统算法里运行测试集
2. 固定总件数为100件左右，无人机数量为4，无人机速度为3，物流车载 重为500。调整轻件占比分别为20%，40% ，60% ，80%。
3. 固定总件数为100件左右，轻件占比40%左右，无人机数量为4，无人机 速度为3。物流车载重分别为200，300，400，500。
4. 固定总件数为100件，轻件占比40%，物流车载重为500，无人机数量为 4。调整无人机速度为2，3，4....测试代价随无人机速度上升的变化。
5. 固定总件数为100件，轻件占比40%，物流车载重为500，无人机速度为 3。调整无人机数量为1，2，3，4，5测试代价随无人机数量上升的变化。

### 模拟数据生成模块

局部集聚的样本数据生成模块主要由两个函数构成：

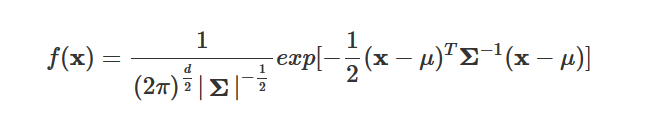
1. isDistinct

主要通过np.linalg.norm函数计算类间的二范数与给出的标准做比较，判断类与类之间的距离是否符合标准，是就返回True否则返回False。

1. generateData

生成数据模块中，首先根据给定的类数量随机生成多个类的中心点，同时用isDistinct函数判断避免靠的太近，然后根据设置好的高斯分布协方差在每个类中心点的周围生成样本数据。

多维高斯分布 ：



协方差矩阵的对角线元素为x和y轴的方差

反斜对角线上的两个值为协方差，表明x和y的线性相关程度（正值时：x增大，y也随之增大；负值时：x增大，y随之减小）

高斯分布的协方差矩阵是一个对称矩阵，该矩阵决定了二维高斯分布的形状，数值的大小决定了高斯分布的收敛程度。如图：

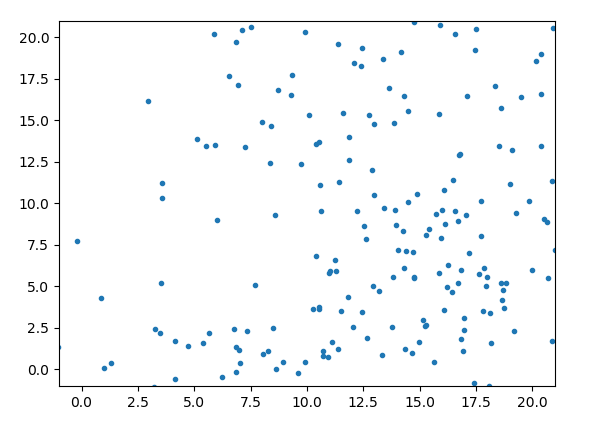


图 4-2-1-1高斯分布协方差[[10, 0], [0, 10]]

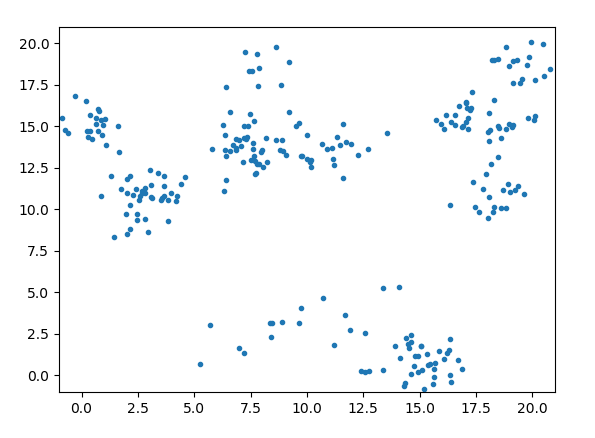


图 4-2-1-2高斯分布协方差[[1, 0], [0, 1]]

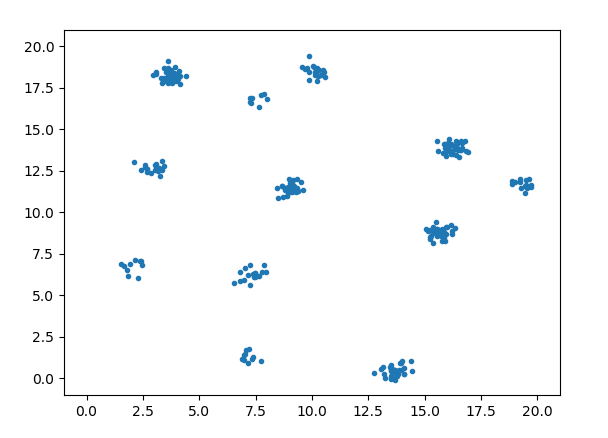


图 4-2-1-3高斯分布协方差[[0.1, 0], [0,0.1]]

图中可以清晰地看出对高斯协方差大小的调整可以有效改变点的聚集程度。真实场景中的快件点分布应如图4-2-1-2所示，在局部集聚的基础上有着交集，也可能出现离散点的情况。生成模拟数据的时候要不断根据样本范围调整高斯协方差大小，从而达到想要得到的效果。

### 样本数量测试

在样本数量环节中，我们固定轻件占比40%左右，无人机数量为4，无人机速度为3，物流车载重为500。分别生成50点左右，100点左右、200点左右以及300点左右的样本集进行测试。

图 4-2-2-1 路径长度对比

如上图所示，就行驶距离而言，随着点数的上涨，无论是传统算法还是新算法，物流车、无人机的行驶距离都会随着点数的上升而上升，并且上升幅度越来越大

图4-2-2-2 行驶代价对比

图4-2-2-3 时间代价对比

如上面两张图所示，当点数较少时，新算法与传统性能相近，随着点数的上涨，两种算法行驶代价之间的差距在不断扩大，从相差500左右扩大到后来的13000。而时间代价的差别更大，从相差2000一直扩大到17000。由此可见新算法在点数较少时性能不输传统算法，在点数较大时对于传统算法有着很大的优势。

### 轻重件比例测试

图4-2-3-1 无人机物流车行驶路径长度随轻件点占比升高变化

随着轻件比例的上升，物理车承担的派件任务逐渐减少，无人机承担的派送任务逐渐上升。由于轻件点较多的情况下会有部分轻件点转换为等价重件点，所以理论上讲无人机行驶路径长度的增加幅度应该大于物流车。从测试结果中可以看出，无人机配送行驶路径长度的上升幅度比物流车下降幅度要大，符合预期。

图4-2-3-2 总行驶代价随轻件点占比升高变化

图4-2-3-3 总时间代价随轻件点占比升高变化

从上面两组测试数据中可以看出，无论是行驶代价还是时间代价，随着轻件点占比的提升，都有了明显的降低（接近线性）。在轻件点比例较大的情况下性能更优。

### 无人车载重测试

图4-2-4-1 综合行驶代价随物流车载重增加变化趋势

图4-2-4-1 总时间代价随物流车载重增加变化趋势

物流车容量上升过程中，单词配送能完成更多的配送任务，从而减少了物流车的总配送次数，降低了往返成本，因此理论上随着物流车载重提升两种代价都将有效降低。如图所示，验证了前面的假设。

### 无人机数量测试

图4-2-5-1 综合行驶代价随无人机数量增加变化趋势

图4-2-5-2 总时间代价随无人机数量增加变化趋势

增加无人数量情况下，在CVRP\_L中会有更多的类采用各无人机单件派送的情况，会减少时间代价，同时在代价计算模块中，由于无人机数量的提升，更少的点需要在起飞或者着陆点进行定点配送，因此也会降低行驶代价和时间代价。

但是无人机数量增加到一定数量后，局部配送能力就趋于饱和，因此再增加无人机数量带来的收益将会很小。从图中可以看出降低幅度明显由初期的明显下降变为逐渐平缓。

### 无人机速度测试

图4-2-6-1 总时间代价随无人机速度增加变化趋势

无人机的速度影响的只有时间代价模块，无人机的速度越快类内配送完成速度越快，物流车的停车等待时间越短。因此可以预计当无人机速度逐渐增大到某固定值时，停车等待时间将会被消除，总时间代价将会等于物流车的配送时间。

图中也可以看出随着无人机速度的增加总时间代价先是明显降低，然后降低速度放缓。并逼近物流车配送时间18415.2。

## 实验结果及分析

经过详细的实验，本文提出的新的算法的有效性得到了验证，相比于传统算法，样本数量较少时新算法有着微小的优势，但一旦样本数量扩到到100、200甚至是300点时新算法的优越性一下子体现出来：无论是行驶代价还是时间代价，代价差从2000左右逐渐扩大至15000以上。

而对算法本身参数的测试也验证了之前假设的各种影响：

1. 两种代价都会随轻件比例上升而稳定降低，降低速度接近线性。
2. 两种代价都会随无人车载重上升而降低，降低速度先快后慢，逐渐趋于平缓
3. 两种代价都会随无人机数量的上升而降低，降低速度先快后慢，逐渐趋于平缓
4. 总时间代价会随无人机速度的提升而降低，降低速度先快后慢，逐渐趋于平缓，逐渐逼近物流车配送时间。

# 结 论

本文通对物流车与无人机的协同配送算法的研究，提出了基于区域聚类的多无人机与物流车路径三阶段混合优化算法。有效地降低了运行成本和时间成本，从理论层面上创新性地解决了物流车与无人机协同配送这一难题。

本文的创新点有如下几点：

1. 聚类思想在本场景下的适用性改进

综合研究已有的先聚类后路径规划的算法思想之后，我发现由于都是应用在纯物流车配送的场景中，这些算法思想大多是划分完成运送区域后对区域内的所有点进行规划，再整体合并。这种先对小范围区域进行处理的方法，对聚类算法就有着较高的要求，聚类阶段就必须要在考虑到下一阶段整体规划的前提下进行合理划分，否则，先在局部完成该区域的路径规划很有可能会到导致整体性能的下降。

而在我们的应用场景中重件点与轻件点混杂，轻件点尽量使用无人机配送会大大降低配送成本。因此我们场景中的聚类对象为轻件点，聚类限制条件应为无人机覆盖半径。同时也要根据无人机数量对聚类区域的点数进行适当的初步限制。

而在聚类与路径规划的配合方面，与之前的大多数算法思想不同，本算法将采用先整体后局部的方式进行规划：首先对所有轻件点进行聚类，聚类完成后选取聚类中心，把原本的那个轻件点转换成这个类的等价重件点，附上整个类的快件重量。这样生成的等价重件点加入其余重件点中，视为重件点先用改进过的UGV-Tabu-CVRP算法进行一次CVRP求全局最优解，解出每辆物流车应该走的路线，然后再对每个区域内的轻件点进行UAV-Tabu-CVRP求解，解出局部无人机飞行路径的最优解。这样从整体细化到局部，首先保证了物流车代价的最小，其次，完美安排了无人机的最优配送路径，由于无人机配送速度快，区域内可与物流策划并行工作，时间代价主要就是物流车的配送时间。通过本文提出的先整体再局部的聚类-路径规划思想，理论层面上可以将综合代价降低至最小。

1. 三阶段混合算法对复杂问题的分解与整合

本文待解决的应用场景涉及各物流车任务分配、无人机任务分配、物流车无人机容量限制以及物流车无人机协同工作等问题。算法的输入只有所有快件点的坐标和重量，要从简单的输入中提取信息，进行处理从而得到整体的路径规划，就需要对该复杂问题进行分解。我将整体算法主体分为主要三个阶段：数据预处理阶段、CVRP全局解阶段以及CVRP局部解阶段。

数据预处理阶段中，将输入的只有坐标和重量的原始数据先进行轻重件分离，再用改进的K-means聚类算法对所有轻件点进行聚类，并生成等价重件点，最后将在轻件点类内范围内的重件点归入该类。

CVRP全局解阶段中，将数据预处理阶段生成的等价重件点和其他重件点一起导入改进的UGV-Tabu-CVRP解决器中，输出每辆物流车分配的快件点路径。

CVRP局部解阶段中，将之前已经安排好的等价重件点还原成轻件点类，通过路径前后的类内重件点分布进行分类讨论处理先确定起飞点和着陆点，然后根据类内点数决定类内是进行单飞还是进行局部的UAV-Tabu-CVRP，最后输出每个类内的无人机规划好的路径。

1. 引入无人机后的物流车与无人机协同配送方式

目前的无人机配送方式仅是定点起飞定点着陆进行配送，在无人机进行配送的过程中，物流车是停滞不动的，因此浪费了大量等待时间。但是考虑到安全因素以及技术条件限制，移动中无人机的起飞会合是不合理的。因此，受[XX]启发，设计出制定起飞点和着陆回合点的方式，这样能在物流车进行运输的时候尽可能并行派送无人机从而减少总代价。同时改进Tabu-CVRP算法，使其适用于非单点环路（双点单向路：起飞点和着陆点）的情况，实现每架无人机能在载重范围内携带多件快件的UAV-Tabu-CVRP算法。

今后进一步在本研究方向进行研究工作的展望与设想：

1. 由于本文侧重研究的是物流车与无人的协同工作，因此仅是对现有的Tabu-CVRP进行了实现和简单的优化，作为本算法基础之一吗，对启发式CVRP算法的优化将是之后研究的一个重要内容。
2. 在局部无人机路径规划算法中，任无法避免地存在着物流车停车等待无人机配送完成的情况，新的更有效的局部规划算法的提出也能很大提升算法性能
3. 目前本算法考虑的是静态条件下进行规划，今后研究方向可以是研究动态变化的情况。

# 参考文献

1. Dantzig, George Bernard; Ramser, John Hubert (October 1959). ["The Truck Dispatching Problem"](http://andresjaquep.files.wordpress.com/2008/10/2627477-clasico-dantzig.pdf) Management Science. 6 (1): 80–91
2. Comtois, Claude; Slack, Brian; Rodrigue, Jean-Paul (2013). The geography of transport systems (Third edition. ed.). London: Routledge, Taylor & Francis Group
3. Geir Hasle, Knut-Andreas Lie, Ewald Quak,; Kloster, O (2007). Geometric modelling, numerical simulation, and optimization applied mathematics at SINTEF ([Online-Ausg.]. ed.). Berlin: Springer Verlag
4. Cordeau, Gendreau, Hertz, Laporte, & Sormany New heuristics for the vehicle routing problem in Logistics Systems: Design and Optimization 2005 pp.279-298; New York: Springer
5. Juan, Faulin, Adelanteado, Grasman, & Montoya Torres Solving the capacitated vehicle routing problem maximum traveling distance and service time requirements: An approach based on Monte Carlo In M. D. Rossetti, R. R. Hill, B. Johanson, A. Dunkin, & R. G. Ingalls (Eds.) Proceedings of the 2009 winter simulation conference 2009 pp. 2467–2475
6. Lopez-Castro & Montoya-Torres Vehicle routing problem with fuzzy time windowsIn Proceedings of the 2011 IEEE workshop on computational intelligence in production and logistics systems 2011 ,pp. 39–46 IEEE Publishing.
7. Baldacci, R., Toth, P., & Vigo, D. Exact algorithms for routing problems under vehicle capacity constraints in Annals of Operations Research, 2010 , 175, pp.213–245.
8. Cordeau, J. F., Laporte, G., Savelsbergh, M. W. P., & Vigo, D. Vehicle routing. In C. Barnhart & G. Laporte (Eds.).in Transportation, handbooks in operations research and management science (2007 Vol. 14, pp. 367–428). Amsterdam: North-Holland
9. Archetti, C., & Speranza, M. G. The split delivery vehicle routing problem: a survey. In The vehicle routing problem: latest advances and new challenges. Operations research/computer science interfaces Series 2008 Vol. 43. Part I (pp. 103–122).
10. Jairo R. Montoya-Torres, Julián López Franco, Santiago Nieto Isaza, Heriberto Felizzola Jiméne,& Nilson Herazo-Padilla A literature review on the vehicle routing problem with multiple depots in Computers & Industrial Engineering （2015 79 pp.115–129）
11. Luo Z, Liu Z, Shi J. A Two-Echelon Cooperated Routing Problem for a Ground Vehicle and Its Carried Unmanned Aerial Vehicle:[J]. Sensors, 2017
12. Yu K, Budhiraja A K, Tokekar P. Algorithms for Routing of Unmanned Aerial Vehicles with Mobile Recharging Stations[J]. 2017
13. Billy E Gillett and Leland R Miller. “A heuristical gorithm for the vehicle-dispatch problem” In: Operations research 22.2(1974),pp.340–349.
14. Marshall L Fisher and Ramchandran Jaikumar . “A generalized assignment heuristic for vehicle routing”. In:Networks11.2(1981) , pp.109–124.
15. Mingyao Qi et al. “A spatiotemporal partitioning approach for large-scale vehicle routing problems with time windows”. In: Transportation Research Part E:Logistics and Transportation Review48.1(2012), pp.248–257
16. <http://tech.163.com/18/0313/14/DCPK7UCO00097U7T.html>
17. <https://www.ifanr.com/762191>
18. <http://digi.tech.qq.com/a/20180211/022705.htm>
19. [https://view.inews.qq.com/a/20180228A0I7SG00?fro](https://view.inews.qq.com/a/20180228A0I7SG00?from=timeline)
20. Sergejs Pugacs “A Clustering Approach for Vehicle Routing Problems with Hard Time Windows” , universidade nova June ,2014
21. 钟石泉，贺国光 《单车场复杂情况下的车辆调度》 System Engineering. May,2005 pp.29-32
22. 张思伟 《单车场多送货点车辆调度优化的一种改进禁忌算法》 Industrial Engineering Journal , May 2006 pp55-60
23. Zhihao Luo, Zhong Liu and Jianmai Shi “A Two-Echelon Cooperated Routing Problem for a Ground Vehicle and Its Carried Unmanned Aerial Vehicle” In:sensors 2017,17,1114

# 哈尔滨工业大学本科毕业设计（论文）原创性声明

本人郑重声明：在哈尔滨工业大学攻读学士学位期间，所提交的毕业设计（论文）《基于区域聚类的多无人机与物流车路径优化算法研究》，是本人在导师指导下独立进行研究工作所取得的成果。对本文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明，其它未注明部分不包含他人已发表或撰写过的研究成果，不存在购买、由他人代写、剽窃和伪造数据等作假行为。

本人愿为此声明承担法律责任。

作者签名： 日期： 年 月 日

# 致 谢

衷心感谢导师曲明成老师对本人的精心指导。我得以顺利完成毕业设计以及论文，感谢曲老师给我提供了综合楼607智能软件技术研究中心这个学术氛围浓厚的研究环境。从我的课题选取、方案论证到编程实现、实验验证，无不凝聚着曲老师的心血和汗水，他的言传身教将使我终生受益，他科学严谨、刻苦钻研的作风值得我学习。

此外还要感谢智能软件技术研究中心的各位老师以及中心学长们的热情帮助和支持！有了你们为我热情答疑解惑，一起探讨问题场景、解决方案我才能顺利完成毕业设计，完整完成毕业论文的创作。

最后我要感谢哈工大计算机学院全体老师，没有你们的教育我不会有今天这样扎实的专业基础；我要感谢哈工大所有的教职员工，有了你们的付出我们才拥有了四年美好的大学生活，才能有工大良好的学术气氛，才能催我上进奋发图强。

谢谢！

# 附录

## 车辆路径问题的改进禁忌搜索算法

作者：Hongmei Jia , Yang Li, Bo Dong, Hongying Ya

## 摘要

车辆路线问题在配送管理领域有非常重要的应用。 VRP既具有理论上的实际意义（由于其真实世界的应用），这解释了过去几年研究人员对VRP的关注程度，也因为VRP是NP-Hard问题。 在本文中，我们通过引入变异和混合局部搜索策略来克服当前TS的弱点，设计并实现了一种新的禁忌搜索。 这里我们关心算法策略和参数以及它们如何影响设计的禁忌搜索的性能。 将改进的禁忌搜索与其他算法进行比较后，显示了改进的禁忌搜索的优异性能。 首先，无论问题的规模大小，改进的禁忌搜索解决方案的质量都很好; 其次，改进的禁忌搜索算法非常稳定; 最后，收敛速度快，计算效率高。

## 正文

1. **车辆路径问题介绍**

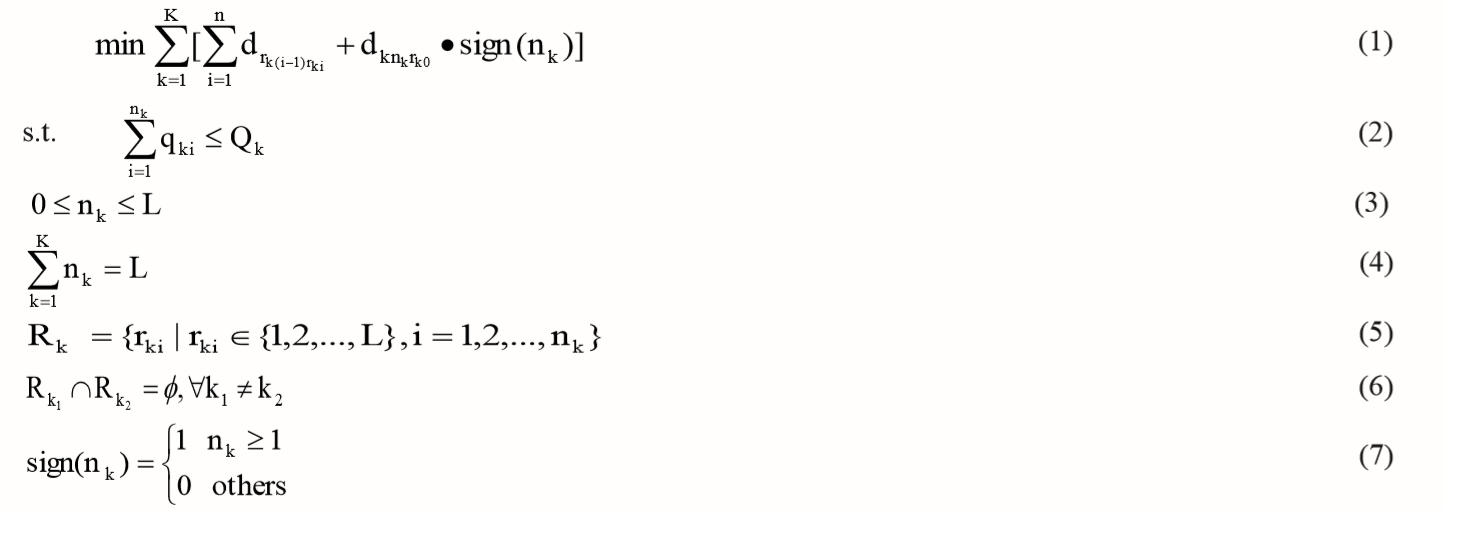
Dantzig和Ramser在1959年提出的车辆路径问题在配送管理领域有着非常重要的应用。 因此，他们已经成为组合优化文献中研究最多的问题之一，大量的论文和书籍涉及解决这些问题的众多程序。 车辆路径问题（VRP）是一个复杂的组合优化问题，它可以被看作是两个众所周知的问题的融合：旅行销售人员问题（TSP）和箱包装问题（BPP）。 它可以描述如下：给定一个具有统一容量的车队，一个共同的车厂和几个客户需求，找到具有总体最小路线成本的一组路线，以满足所有需求。

1. **文献回顾**

VRP是NP难题，因此难以解决。 VRP既具有理论研究的意义又有现实应用的重要性（因为其真实世界的应用），这一事实解释了过去时间里中研究人员对VRP的重视程度。许多研究人员在过去几十年一直在研究车辆路径问题。 Bodin，Golden（1983）等人在他们的论文中列举了700多篇文献。 Christofides（1985），Golden和Assad（1988）的论文集以及Altinkemer和Gavish（1991），Laporte（1992），Salhi（1993）的论文摘要详细介绍了该领域的研究。该领域的主要研究人员有Bodin，Christofides，Golden，Assad，Ball，Laporte，Rinnooy Kan，Lenstra，Desrosiers，Desrochers等。有很多研究人员研究了车辆路径问题发生后的复杂性，这是解决算法研究的基础。 Lenstra和Rinnooy Kan在他们的论文《车辆路线和调度问题的复杂性》中证明这个问题是NP难以解决的。 Dantzig和Fuikerson（1954）证明了VRP的复杂性O（n'）。 Savelsbergh（1985）和Solomon（1986）提出了带时间窗的VRP，其比标准的VRP更复杂; Lenstra和Rinnoopy Kan证明，大多数VRP都是NP难题。作为NP难题，物流配送车辆路线问题的配送路线项目会随着客户的增加而呈指数增长。这里有用于解决车辆路径问题的技术。其中几乎所有的都是启发式和元启发式，因为当城市（客户）数量很大时，没有确切的算法可以保证在合理的计算时间内找到最佳旅程。这是因为问题是NP难度问题所决定的。

1. **本文中VRP问题的数学公式**

从图论的角度来看，VRP可以表述如下：令G=(C,L)是一个具有节点集C=(c0,c1,c2,c3,...,cn)和弧集的完全图。在这个图模型中，c0是仓库，其他节点是要服务的客户，每个节点与待交付货物的固定重量qi相关联，对于每一个弧线(ci,cj)与表示ci和cj之间的行驶距离的值dij相关联。



目标函数(1)表示总行程距离。 约束条件(2)确保每辆卡车的总载重不超过卡车容量。 约束条件(3)确保客户数量不超过客户总数。 约束条件(4)保证所有客户都被访问。 约束条件（5）是每条路线的客户。 约束条件（6）保证每个顾客只被一辆卡车访问。 约束条件（7）表示：如果由卡车服务的客户数量不止一个就置一，否则置0。

1. **改进的禁忌搜索算法解决VRP的设计**

作为局部搜索的扩展，禁忌搜索涉及决定算法性能的解决方案指示方法，解决方案赋值，局部搜索策略和终止准则等关键要素。 此外，我们需要一些元素，如tabus，禁忌长度，候选集等。、

* 1. **解决指示方法**

当我们用启发式算法求解VRP时，解决指示方法是决定算法性能的关键工作。在我们的搜索中，我们使用客户和虚拟仓库阵列作为解决方案指示方法。我们通过添加虚拟仓库将车辆路径问题转换为旅行推销员问题。基于上述思想提出了该解决方案的指示方法。设置0作为仓库，1...L为客户，有K辆车，所以最多有K条路径。每条路线从仓库开始并且终止于仓库。所以这K-1个数的一个排列就是一种规划解。例如，有3个卡车服务的客户有7个，我们可以用129638547（8,9是虚拟仓库）的随机数组表示分配方案：卡车路线1：9（0）0-1-2-，卡车路线2：8（0）-6-3-9（0），卡车路线3：-5 -4-7-08（0），有三条路线，三辆卡车分配喜悦; Solution573894216表示配送方案：卡车路线1：8（0）0-5-7-3路线3：-4-2-1-6-0 9（0），有两条路线，两辆卡车配送。分配方案固定为路线数少于或等于可用卡车的数量。

* 1. **解决方案评估**

基于对VRP公式的描述，我们通过以下方式对解决方案的质量进行阀门调整：首先，找到解决方案对应的分配方案，然后证明方案是否满足约束条件，同时计算目标函数越大，分布方案越好，求解质量越好，目标函数的阀值越好，并且解的满足约束条件。客户和虚拟仓库同时获得的解决方案和相应的分销方案满足了每个客户的约束，并且仅由一辆卡车提供服务。对于每个分配方案，我们需要证明每条路线的总负荷是正确的，并确保任何路线的总负荷不超过卡车容量。如果解决方案不满足约束条件，则设置方案是不可行的方案，但仍需计算目标值。对于一个解决方案，设置不可行路线的数量为M（M 0，该解为可行解），设目标函数的阀值为Z，惩罚功率为W P（功率设为适当的整数记录为目标函数的值空间），然后通过公式（8）计算求解值E（解决方案的质量随着E越小越好）。解法赋值法体现了惩罚函数处理约束的思想。

E=Z+M\*Pw （8）

* 1. **局部搜索策略**

在我们的研究中，计算机生成一个0到3之间的随机数，然后根据生成的随机数以相同的可能性采用三个n中的一个。 在运营商之后，我们必须确保试验解决方案的邻居仍然是解决方案空间的组成部分，意味着解决方案是有效的。 本地搜索策略与解决方案指示方法有关。 我们同时使用客户和虚拟仓库阵列作为解决方案指示方法。

* 1. **终止准则**

作为启发式算法，禁忌搜索在迭代中需要终止标准以便在合理的时间内得到解决方案。 显然，在实践中，搜索必须在某个时候停止。 TS中有四种最常用的停止标准。 首先是在固定数量的迭代（或固定的CPU时间）之后停止; 第二种是在没有改进目标函数值（大多数实现中使用的标准）的情况下经过若干次迭代后停止; 第三种是当目标达到预定的阈值时停止; 第四种是通常在完成一系列阶段后停止搜索，每个阶段的持续时间由上述标准之一确定。 在本文中，我们采用第二个标准。

* 1. **禁忌表**

与LS相比，Tabus是TS中独特的元素之一。 Tabus用于防止通过非改进移动从本地最优方案移开时的骑行。 禁忌（禁止）移动，扭转最近移动的效果，以防止搜索将其步骤追溯到其来源。 在我们的研究中，我们采用tabus来解决向量的变化。 例如，解决方案123456789S，操作员交换。 交换点为点4和点7，运营商试运行后的邻居：123756489 S。 作为点4和点5之间交换之后的解决方案的S可以是局部最优的。 为了禁止局部最优化，Tabus是在迭代的接下来的几个步骤中禁止点4和点5之间的交换。 Tabus为：如果新解比S更好，则选择邻居中的新解，如果S是局部最优，则选择邻域中的劣解。

* 1. **禁忌长度**

Tabu长度是tabus的步数。 给tabusxa numberl（tabu length）意味着tabusxis在nextliteration步骤中被禁止，l list（x）tabu。 l-1列表（x）tabu。 经过一步迭代，tabusxis发布到0列表（x）tabu。 以下是选择禁忌长度的几种方法：l是一个常量，例如10 l，l n（n是邻域中邻居的数量）。 这种方法容易实现; ，l] [最小ll min。 l是一个取决于目标函数和邻域结构的阀门的变量。 min max l，l是常数，取决于问题的大小，限制变化空间b）a]（0N，b N [a;改变空间b）a]（0N，b N [a也由数 邻居邻居n禁忌长度的选择与实际问题的特点以及算法设计者的经验密切相关，禁忌长度太短导致循环，禁忌长度过长导致计算时间增加。

* 1. **可能性禁忌搜索和候选集**

在常规TS中，必须评估当前解决方案的邻域N（S）的每个元素的目标。 从计算角度来看，这可能证明是非常昂贵的。 另一种方法是只考虑N（S）的随机样本（S）N'，从而显着减少计算负担。 替代方案的另一个有吸引力的特征是增加的随机性可以充当反循环机制; 这使得人们可以使用较短的禁忌清单，如果对邻里进行全面的勘探，那么这将是必要的。 一个消极的一面，必须指出的是，在这种情况下，人们可能会错过优秀的解决方案。 概率也可以应用于激活禁忌标准。 另一种控制移动次数的方法是通过候选列表策略，它提供了更有战略意义的方法来生成一个有用的子集（S）

* 1. **愿望准则**

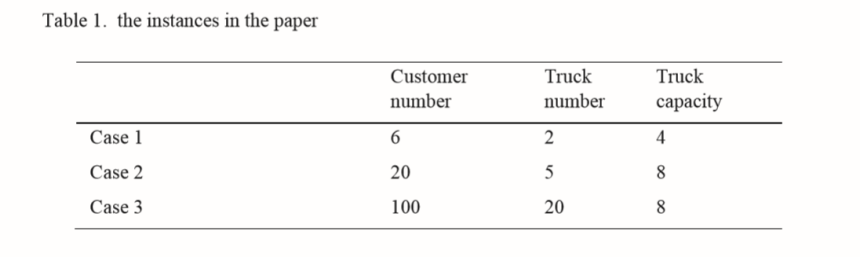
虽然TS是核心，但tabus有时过于强大：即使没有骑自行车的危险，它们也可能禁止有吸引力的举动，或者可能导致搜索过程全面停滞。 这些被称为渴望标准。 最简单和最常用的期望准则（几乎在所有TS实现中都可以找到）包含允许移动，即使它是禁忌，如果它导致目标值优于当前最知名解决方案的解决方案。

* 1. **突变因子**

与传统TS不同的是，变异策略是改进禁忌搜索中非常重要的一步，决定了算法性能的改善程度。 变异算子之后，必须保证试验解的邻居仍然是解空间的组成部分，意味着解的有效性。 当生成的概率小于突变概率m P时，将发生突变。 如果发生变异算子，则生成交换次数J，交换选定元素的次序J次（变异点由计算机随机产生）。

1. **实验和分析**

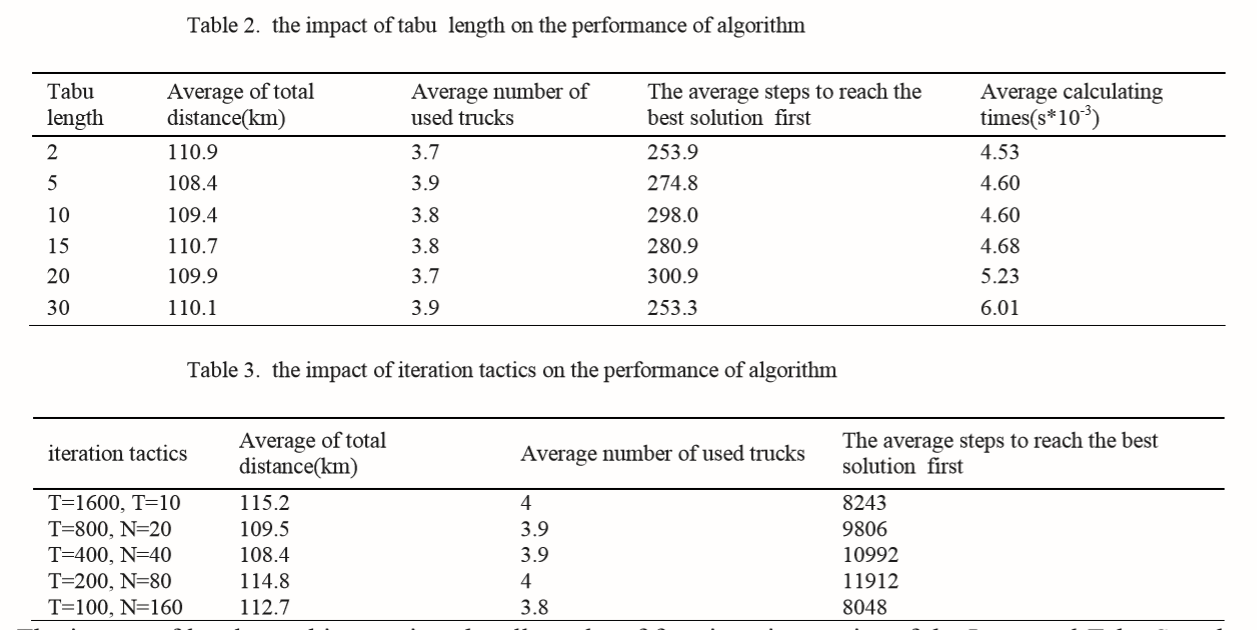
为了比较改进的禁忌搜索和其他启发式算法的性能，我们设计了3个实例来分析和评估改进的禁忌搜索的性能，实例如下。



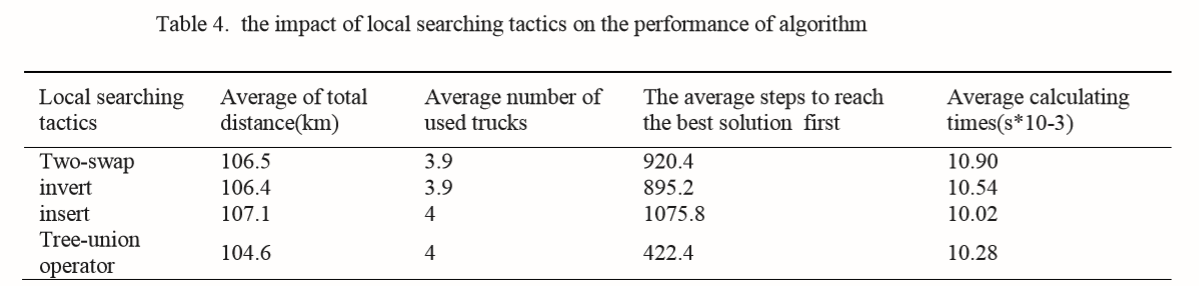
* 1. **算法策略和参数对禁忌搜索性能的影响**

作为启发式算法，Tabu Search的性能体现在两个方面：一是算法的效果，即解的质量; 其次，算法的效率，即在合理的计算时间内得到最优解的能力还是很好的解决方案。 性能不仅与算法策略有关，而且与算法的参数有关。 现在我们将研究一些算法策略和参数对Tabu Search对VRP性能的影响。 Tabu长度的影响：如表2所示，禁忌搜索的平均距离在五个禁忌长度中非常相似。 当禁忌长度增加到20和30时，计算时间比禁忌长度为15的时间长，但结果更差。

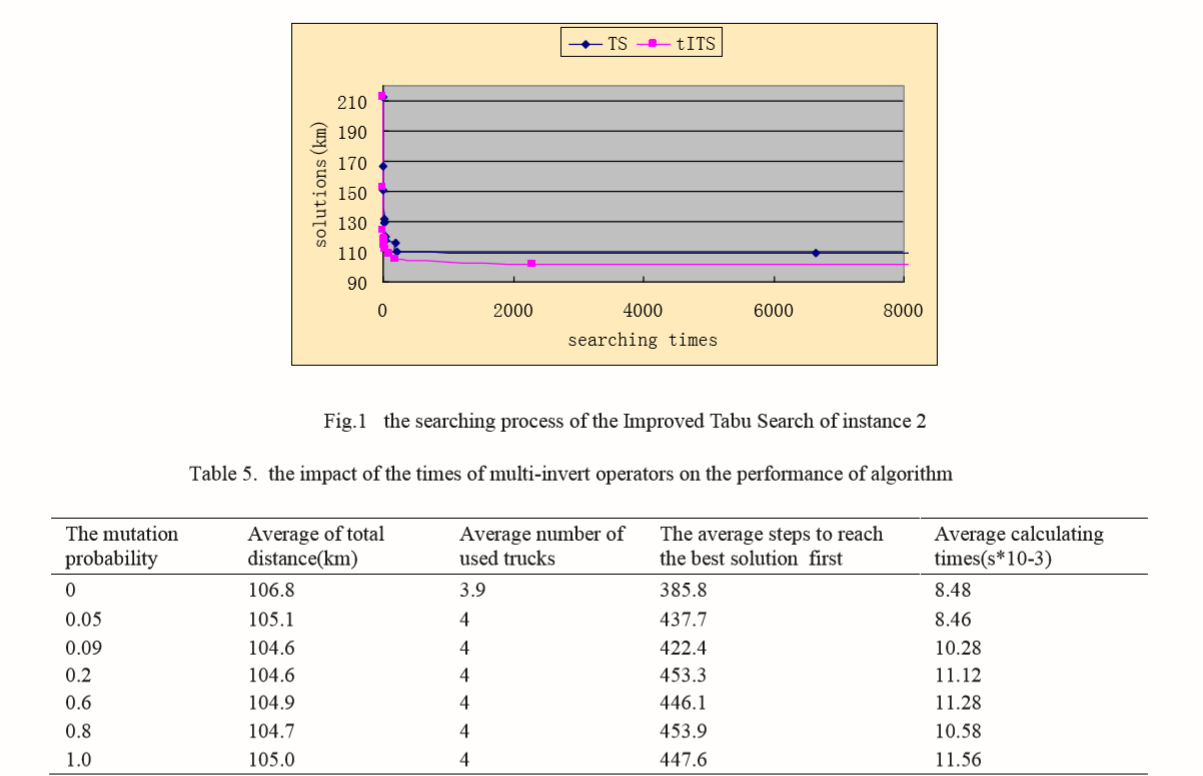
迭代策略的影响：在表3中：在五种迭代策略中，T = 800，N = 20和T = 400，N = 40的结果比其他结果更好。 所以我们可以看到不同的迭代策略对性能有影响。 当搜索时间为常量时，邻居N的数量太大或太小以至于不利于算法的性能。 对于实例3.2，当N在20〜40之间，搜索时间为16000时，可以得到更好的解。



局部搜索策略的影响：由于采用了变异策略，改进禁忌搜索的四种迭代策略的所有结果均优于TS。 在四种迭代策略中，Treeunion算子的结果明显优于其他策略，四种算子的计算时间非常接近。 所以我们可以看到不同的迭代策略对性能有一定的影响。 三联算子效果较好的原因是搜索空间扩大，解决方案的多样性由于更多的算子而增加



Tabu搜索的搜索过程如图1所示，在迭代开始（0〜190）时，解的质量提高非常快，总距离从212.44km减少到104.6km。 在迭代阶段190〜2300，解的质量是停滞的。 现在全局最优的解决方案的质量在步骤2300达到最高。在整个过程中，我们可以看到tITS的收敛速度比TS的速度快，解决方案的质量好于TS的质量 在同一步骤。 变异概率的影响：表5中，随着变异概率的增加，解的质量增加，但当变异概率大于0.09时，随着变异概率的增加，解的质量停止增加，甚至质量有时会减少。 所以我们可以看到，变异算子有助于提高解决方案的质量，同时增加解决方案和方法的多样性。



通过研究搜索过程可以总结出算法的以下特征：1通过迭代步骤，解决方案的质量得到改善。 迭代开始阶段，解决方案质量的提高非常快。 随着迭代步骤的增加，改进速度越来越慢; 2考虑到解决方案的质量和收敛速度，改进的禁忌搜索的性能要优于TS的性能。 证明了改进的禁忌搜索算法对于全局最优搜索能力比禁忌搜索更有效。

1. **结论**

在本文中，我们通过引入变异和混合局部搜索策略来克服当前TS的弱点，设计并实现了一种新的禁忌搜索，这里我们关注算法策略和参数如何影响设计禁忌搜索的性能。 将改进的禁忌搜索与其他算法进行比较后，显示了改进的禁忌搜索的优良性能。 首先，无论问题的规模大小，改进禁忌搜索解决方案的质量都非常好; 其次，改进的禁忌搜索算法非常稳定; 最后，收敛速度快，计算效率高。