

# 基于改进遗传算法的家电回收车辆路径规划方法

黄新林, 张隆飞, 唐小伟

(同济大学 电子与信息工程学院, 上海 201804)

**摘要:** 为了提高家电回收效率以及降低回收成本, 提出了一种基于改进遗传算法(GA)的家电回收车辆路径优化方法。将家电回收车辆路径规划问题建模为一个变体的旅行商问题(TSP)以最小化运输成本, 但该问题难以在多项式时间内进行求解。提出了一种基于高斯矩阵变异(GMM)算子的改进遗传算法, 利用原始站点数据信息中隐含的站点位序分布特性建立高斯概率矩阵, 并采用轮盘赌选择法将高斯概率矩阵作用于个体基因突变, 在保证种群基因多样性的同时, 引导种群向高适应度方向进化。最后, 采用上海地区的家电回收点实际数据开展实验仿真以验证所提出算法的有效性, 并与其他算法进行对比。结果表明, 与传统遗传算法相比, 在将求解精度差保持在1%以内的情况下, 所提出改进遗传算法的平均收敛速度可以提升50%~60%, 算法耗时降低48%。

**关键词:** 家电回收; 旅行商问题(TSP); 遗传算法(GA); 高斯矩阵变异(GMM)算子

中图分类号: TP301.6

文献标志码: A

## Path Planning Method for Household Appliance Recycling Vehicle Based on Improved Genetic Algorithm

HUANG Xinlin, ZHANG Longfei, TANG Xiaowei

(College of Electronic and Information Engineering, Tongji University, Shanghai 201804, China)

**Abstract:** In order to improve the recycling efficiency and reduce the cost of recycling disused products, a path optimization method for recycling vehicles based on a modified genetic algorithm (GA) was proposed. Firstly, the path planning problem for the recycling vehicle was modeled as a variant of the traveling salesman problem (TSP), aiming at minimizing transportation costs, which, however, is an NP-hard problem. Then, an improved genetic algorithm based on the Gaussian matrix mutation (GMM) operator was put forward. The algorithm

leveraged the site order distribution characteristics hidden behind the original station data information to establish a Gaussian probability matrix. The Gaussian probability matrix was then applied to individual gene mutations combined with the roulette selection method, so as to guide the population to evolve towards high fitness while maintaining the genetic diversity. Finally, comprehensive simulations were carried out using the actual data collected from recycling sites in Shanghai to validate the proposed algorithm, and compared with other algorithms. The simulation results show that the proposed algorithm can increase the average convergence speed by 50%~60% and reduce the time consumption by 48% compared with the traditional genetic algorithm, under the precision gap within 1%.

**Keywords:** household appliance recycling; traveling salesman problem (TSP); genetic algorithm (GA); Gaussian matrix mutation (GMM) operator

随着居民生活水平的提高, 家用电器的更新换代速度不断加快, 报废家电数量持续增长。废旧家电中包含众多可供回收再利用的生产原料, 依据生产者责任延伸制, 家电生产企业有责任构建和完善废旧家电的“回收、拆解、再利用”循环体系, 让废旧家电得到妥善处理。作为整个循环体系的第一步, 回收过程需要为回收车辆规划合理的运输路径, 从而达到提高运输效率以及降低运输成本的目的。

家电回收车辆路径规划问题本质上是旅行商问题(TSP)的变体问题。TSP是经典的组合优化问题, 描述了一名旅行商从起点出发, 遍历给定的所有城市, 最终返回起点, 求解最短路径的问题<sup>[1]</sup>。家电回收车辆路径规划问题与TSP类似, 考虑单个车辆行驶途中从各个回收点装载废旧家电并运输至最终

收稿日期: 2023-01-15

基金项目: 国家重点研发计划(2022YFB3305801)

第一作者: 黄新林, 教授, 博士生导师, 工学博士, 主要研究方向为机器学习。

E-mail: xlhuang@tongji.edu.cn

通信作者: 唐小伟, 工学博士, 主要研究方向为路径优化。E-mail: xwtang@tongji.edu.cn



论文  
拓展  
介绍

回收点,寻找运输成本最小的行驶路径,本质上是沿途取货的 TSP。随着 TSP 的提出,逐渐产生了众多分支问题,如带提货和送货的旅行商问题(TSPPD)<sup>[2]</sup>、概率旅行商问题(PTSP)<sup>[3]</sup>、多旅行商问题(MTSP)<sup>[4]</sup>和带酒店选择的旅行商问题(TSPHS)<sup>[5]</sup>等。众多学者针对这些分支问题提出了不同的求解算法。Palacio 等<sup>[6]</sup>针对单商品送取货问题提出了一种基于多起点进化局部搜索和变邻域下降的混合启发式算法。王勇臻等<sup>[7]</sup>针对 MTSP 提出了一种改进的分组遗传算法。Sousa 等<sup>[8]</sup>针对 TSPHS 提出了一种变邻域搜索算法。在提出的众多 TSP 变体问题中,尚未有文献针对家电回收问题中的车辆路径规划进行系统研究。

学者们对于 TSP 及其变体问题提出了相应的优化求解算法。Ekmekci<sup>[9]</sup>提出了一种基于蚁群优化(ACO)算法的改进算法,通过记忆算法将每次迭代轨迹的成本信息存储下来,并根据这种信息对轨迹上的信息素进行更新。Phu-Ang<sup>[10]</sup>提出了一种改进的人工免疫算法,通过引入双分裂和 K-means 聚类算法进行求解算法优化。也有学者将不同的启发式算法进行混合,以发挥各自优势。Wu 等<sup>[11]</sup>提出了一种遗传算法、贪婪算法和麻雀搜索算法相结合的算法,在求解精度、求解速度和鲁棒性上均有所提高。Akter 等<sup>[12]</sup>针对染色体交叉过程提出了一种新的交叉方法,在进行交叉操作前对交叉后的成本进行比较来决定是否进行交叉操作。Bao 等<sup>[13]</sup>将部分映射交叉算子与校正机制相结合,提高了路径规划算法的性能。Prasad 等<sup>[14]</sup>提出了一种结合贪婪算法的交叉算子,得到的结果优于传统的交叉方法。

在上述算法中,遗传算法较早被用于求解 TSP,在解决复杂类型的 TSP 时仍有不错的求解效果。因此,采用遗传算法对家电回收车辆路径规划问题进行求解,在现有研究的基础上,提出了一种改进的遗传算法。通过引入高斯矩阵变异(GMM)算子,将回收点的距离及回收点在可行解中的隐藏位序分布特性作为遗传算法中个体基因突变的引导因子,引导种群向高适应度方向进化。上海家电回收点的真实数据仿真结果表明,所提算法能够在保证求解精度的同时显著提高算法收敛速度,减小算法耗时。

## 1 问题描述与模型建立

### 1.1 问题描述

家电回收车辆路径规划问题描述如下:给定一

辆车的行驶起点、终点以及若干需要服务的客户。由于客户数量较多且零散分布,因此可以通过设立回收点提高回收效率,简化回收流程。每个回收点为一定区域范围内客户提供上门回收服务,并将该区域内的废旧家电进行集中整理,等待回收车辆统一回收。回收车辆从起点出发,按照路线依次访问各个回收点完成废旧家电回收,遍历  $N$  个回收点后,车辆行驶至终点,完成回收任务。在建立模型之前首先要划分客户区域,从而选取回收点。对于客户区域的划分可参考 Lyu 等<sup>[15]</sup>提出的无人机移动基站螺旋选址算法,其基本思想是回收点拥有相同的服务范围。设立第一个回收点,优先保证当前处于未服务区域边界的某一客户处于回收点的服务范围,调整回收点的位置,使边界及边界内侧的客户尽可能多地被覆盖,之后重新划分未服务区域边界,逆时针选取下一边界客户,重复上述过程,直至所有客户均有回收点为其提供服务。注意,在所考虑的模型中,起点可以是车辆运输公司,终点可以是废旧家电拆解中心。车辆运输途中需要考虑由载重而产生的运输成本,具体地,车辆载重越大,油耗越高,运输成本也越高。假设回收车辆的最大载重能满足  $N$  个回收点的回收需求。因此,需要在考虑下述约束条件的情况下最小化运输成本:

(1) 车辆路线约束。车辆须从起点出发,访问所有回收点后须抵达终点。

(2) 车辆访问约束。每个回收点必须且仅被回收车辆访问一次。

### 1.2 数学模型

在集合  $K = \{0, 1, 2, 3, \dots, N, N+1\}$  中, 0 表示出发点的编号,  $N+1$  表示终点的编号,  $1 \sim N$  分别表示  $N$  个回收点的编号。回收点分布情况如图 1 所示, 每个回收点对应的回收量为  $e_i (i \in K/\{0, N+1\})$  (回收量指家电质量)。假设任意 2 个站点间的距离为  $d_{ij} (i, j \in K)$ , 则  $(N+2)$  个站点两两之间的距离可构成一个  $(N+2)(N+2)$  维的矩阵  $D$ 。单位距离运输成本与车辆载重有关, 考虑车辆自重为  $O$ 。将路线规划得到的可行解记为  $v = (v_0, v_1, \dots, v_N, v_{N+1}) (v_n \in K)$ ,  $v$  为  $K$  中元素的有序排列。将所有可行解构成的可行解集记为  $\Omega = (v_1, v_2, \dots, v_M)$ , 其中  $M = N!$ 。用  $C_{v_i v_{i+1}}$  表示回收车辆在站点  $v_i$  和  $v_{i+1}$  之间行驶时单位距离的运输成本, 具体表达式如下所示:

$$C_{v_i v_{i+1}} = \alpha_1 O + \alpha_2 Q_{v_i v_{i+1}}, \forall i \in K \quad (1)$$

式中:  $\alpha_1$  表示车辆空载时单位距离的运输成本系数;

$\alpha_2$ 表示货物单位距离的运输成本系数; $Q_{v_i v_{i+1}}$ 表示车辆在站点 $v_i$ 和 $v_{i+1}$ 之间行驶时的载重。 $Q_{v_i v_{i+1}}$ 的具体表达式如下所示:

$$Q_{v_i v_{i+1}} = \sum_{k=1}^i e_{v_k}, \forall i \in K \quad (2)$$

如上所述,回收车辆在行驶过程中的总成本可以表示为如下形式:

$$\begin{aligned} \sum_{i=0}^N d_{v_i v_{i+1}} C_{v_i v_{i+1}} &= \sum_{i=0}^N d_{v_i v_{i+1}} (\alpha_1 O + \alpha_2 Q_{v_i v_{i+1}}) = \\ &\theta \sum_{i=0}^N d_{v_i v_{i+1}} + \mu \sum_{i=0}^N d_{v_i v_{i+1}} Q_{v_i v_{i+1}} \end{aligned} \quad (3)$$

式中: $\theta \sum_{i=0}^N d_{v_i v_{i+1}}$ 表示车辆自重运输过程中产生的运输成本; $\mu \sum_{i=0}^N d_{v_i v_{i+1}} Q_{v_i v_{i+1}}$ 表示回收家电在运输过程中产生的运输成本。 $\theta = \alpha_1 O$ 和 $\mu = \alpha_2$ 分别表示两部分运输成本对应的权重系数。通过调整 $\theta$ 和 $\mu$ 可使得模型适用于更多的场景。

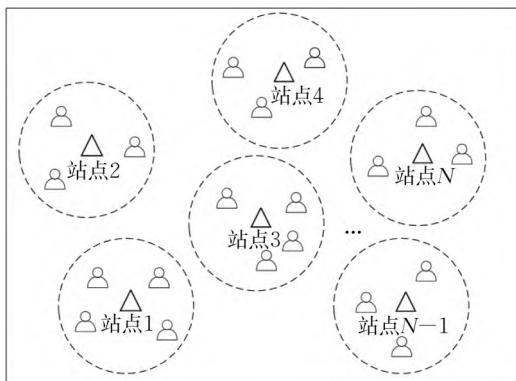


图1 回收点分布示意图

Fig.1 Schematic diagram of recycling point distribution

根据以上表述,成本最小化的回收车辆路径规划问题可以表示为

$$\min_v \sum_{i=0}^N d_{v_i v_{i+1}} C_{v_i v_{i+1}} \quad (4)$$

$$\text{s.t. } v_0 = 0, v_{N+1} = N+1 \quad (5)$$

$$v_i \neq v_j, \forall i, j \in K, i \neq j \quad (6)$$

式(4)为优化目标函数,表示车辆访问 $N$ 个回收点的过程中产生的运输费用总和;式(5)为车辆路线初始站点约束条件;式(6)为车辆访问约束条件。上述优化问题实际上是TSP的变体形式,既包含原始TSP,同时也包含变体的TSP。车辆自重的运输成本最小化问题属于原始的TSP,在确定回收点访问顺序时只需考虑回收点之间的距离。货物的运输成

本最小化问题则属于变体的TSP,在确定回收点访问顺序时除了需要考虑回收点之间的距离外,还需要考虑各回收点的家电回收数量。直观上,具有更重货物的站点应该越靠后访问,这样可以减少货物由于先被装载而在运输过程中产生的额外开销。事实上,有可能出现货物重量较大的回收点离起点较近的情况,如果按照回收点货物重量越大越晚访问的策略进行回收,就可能导致回收车辆出现“走弯路”的情况。因此,需要设计一种合理的路径规划算法来权衡由于车辆自重和装载货物产生的运输成本。

## 2 改进的遗传算法

### 2.1 经典的遗传算法

遗传算法最早由美国的Holland教授提出<sup>[16]</sup>,是一种模拟生物在自然界中遗传和进化过程的自适应全局优化搜索算法。借鉴自然界中种群个体的遗传过程,通过个体选择、基因交叉和基因变异等机制,提高了后代个体的适应度<sup>[17]</sup>。遗传算法以决策变量的编码作为操作对象,个体的染色体与问题的可行解 $v$ 相对应,种群构成整个可行解集 $\Omega$ 。在本研究问题中,可以将目标函数理解为遗传算法中的适应度函数 $f(v)$ 。遗传算法的大致流程可以概括为以下6个步骤,对应的算法流程如图2所示。

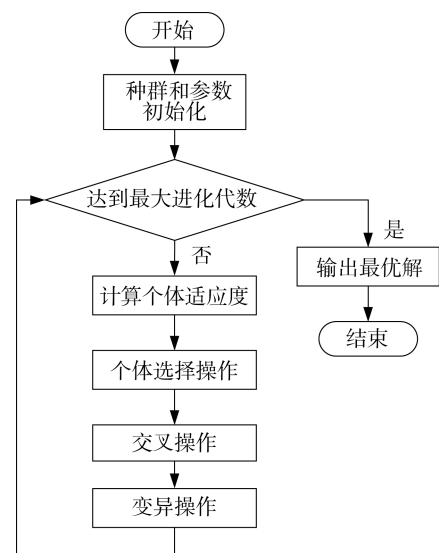


图2 遗传算法流程

Fig.2 Flow chart of genetic algorithm

(1) 种群初始化。设置进化代数计数器 $t$ 、最大进化代数 $T$ ,随机生成初始种群。

(2) 个体评价。按照适应度函数 $f(v)$ 对所有个



体进行评价。

(3) 个体选择。将选择算子作用于现有种群个体,将携带有优良基因的个体保留至下一代。

(4) 交叉操作。基于交叉概率 $P_c$ 从现有种群中成对选择个体,使用交叉算子选择基因进行交叉操作,生成新的个体。

(5) 变异操作。基于变异概率 $P_m$ 从现有种群中选择个体,使用变异算子对个体基因进行变异处理,突变成其他编码值的等位基因。

(6) 终止进化判断。更新进化代数计数器 $t$ ,判断是否达到最大进化代数 $T$ 。若 $t \leq T$ ,则重复步骤(2)至步骤(5);若 $t > T$ ,则终止进化。种群中适应度最高的个体即为求得的最优解。

如何在加快算法收敛速度的同时,保持种群的多样性,一直都是遗传算法中较难解决的问题。本研究针对经典遗传算法在TSP求解中搜索速度慢、难以兼顾种群多样性的问题进行优化。

## 2.2 高斯矩阵变异算子

变异操作是遗传算法的关键步骤,变异操作能够保持种群个体基因的多样性,防止算法陷入局部最优,提高算法的全局搜索能力,避免早熟收敛。传统的变异算子难以兼顾收敛性与全局寻优能力。因此,提出了基于站点位置关系和需求信息的高斯矩阵变异算子,在保持种群多样性的同时,能够将种群变异先验信息作为突变方向的引导因子,加快种群向高适应度方向进化,提高算法收敛速度。

高斯矩阵变异算子基于各回收点与起点、终点的距离关系以及回收点的回收需求构建高斯概率矩阵。将各回收点与起点间的距离表示为 $D_s = (d_{12}, \dots, d_{1(N+1)})$ ,将回收点编号按照与起点距离升序排列得到向量 $\mathbf{a}$ ;将各回收点与终点间的距离表示为 $D_e = (d_{(N+2)2}, \dots, d_{(N+2)(N+1)})$ ,将回收点编号按照与终点距离降序排列得到向量 $\mathbf{b}$ ;各回收点编号按照回收量 $e_i$ 升序排列得到向量 $\mathbf{u}$ 。由此可构建经过排序的站点基础序列矩阵 $H$ ,表达式如下所示:

$$H = (\mathbf{a}, \mathbf{b}, \mathbf{u}) = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ \vdots & \vdots & \vdots \\ h_{N1} & h_{N2} & h_{N3} \end{bmatrix} \quad (7)$$

$H$ 的列向量均为回收点集合的有序排列,由此可以得到回收点 $w (w \in K/\{0, N+1\})$ 在 $H$ 中的由行序号组成的向量 $\beta_w = (n_w, m_w, k_w)$ (这里 $w$ 表示回收点的原始编号)。 $\beta_w$ 反映了各回收点在 $H$ 中的分布特性,也包含了各回收点在个体进化过程中于可行解 $v$ 上的隐含的位序分布特性。

采用高斯分布对隐含的位序分布特性进行拟合。假设回收点 $w$ 在 $H$ 中隐含的位序分布特性 $X_w (X_w \in K/\{0, N+1\})$ 可近似由高斯分布 $N(\mu_w, \sigma_w^2)$ 来拟合,其中参数 $\mu_w$ 和 $\sigma_w^2$ 可分别由下式得到:

$$\mu_w = \frac{(n_w + m_w + k_w)}{3} \quad (8)$$

$$\sigma_w^2 = \frac{(\beta_w - \mu_w)(\beta_w - \mu_w)^T}{3} \quad (9)$$

由于位序分布特性 $X_w$ 只能取有限正整数,因此需要对得到的高斯分布 $N(\mu_w, \sigma_w^2)$ 进行截断处理和离散化。由截断高斯分布定义可知,若 $X \sim N(\mu, \sigma^2)$ ,当 $X \in (c, d) (-\infty < c, d < +\infty)$ 时,则截断高斯分布的概率密度函数以及对应的分布函数如下所示:

$$\phi(\mu, \sigma, c, d; x) = \begin{cases} 0, & x \leq c \\ \frac{\phi(\mu, \sigma^2; x)}{\Phi(\mu, \sigma^2; d) - \Phi(\mu, \sigma^2; c)}, & c < x < d \\ 0, & x \geq d \end{cases} \quad (10)$$

$$\Psi(\mu, \sigma, c, d; x) = \begin{cases} 0, & x \leq c \\ \frac{\Phi(\mu, \sigma^2; x) - \Phi(\mu, \sigma^2; c)}{\Phi(\mu, \sigma^2; d) - \Phi(\mu, \sigma^2; c)}, & c < x < d \\ 1, & x \geq d \end{cases} \quad (11)$$

式中: $\phi(\cdot)$ 和 $\Phi(\cdot)$ 分别表示标准正态分布的概率密度函数和分布函数。对得到的截断高斯分布进行离散化处理,将其收缩到 $K/\{0, N+1\}$ 中的整数值,得到在 $K/\{0, N+1\}$ 中整数值上的概率分布。回收点 $w$ 在 $v$ 中位序为 $i$ 的概率可通过对 $\phi(\mu, \sigma, c, d; x)$ 在区间 $[i-0.5, i+0.5]$ 上计算得到, $c, d$ 的取值分别为 $0.5, N+0.5$ 。因此, $N(\mu_w, \sigma_w^2)$ 经截断处理后的概率密度函数为 $\phi(\mu_w, \sigma_w, 0.5, N+0.5; x)$ 、分布函数为 $\Psi(\mu_w, \sigma_w, 0.5, N+0.5; x)$ 。进一步地,回收点 $w$ 在 $v$ 中位序为 $i$ 的概率 $P_w(i)$ 可以表示为

$$\begin{aligned} P_w(i) &= \Psi(\mu_w, \sigma_w, 0.5, N+0.5; i+0.5) - \\ &\quad \Psi(\mu_w, \sigma_w, 0.5, N+0.5; i-0.5) = \\ &\quad \frac{\Phi(\mu_w, \sigma_w^2; i+0.5) - \Phi(\mu_w, \sigma_w^2; i-0.5)}{\Phi(\mu_w, \sigma_w^2; N+0.5) - \Phi(\mu_w, \sigma_w^2; 0.5)} \end{aligned} \quad (12)$$

由式(12)可得到高斯概率矩阵 $P_G$ ,如下所示:

$$P_G = [P_{w,i}]_{N \times N}, P_{w,i} = P_w(i) \quad (13)$$

在变异操作时,以高斯概率矩阵  $P_G$  为突变影响因子,结合轮盘赌选择法产生个体的基因突变。以变异概率  $P_m$  从种群中随机选择待变异个体,设选中的某一个体基因为  $v_i = (v_{i,1}, v_{i,2}, \dots, v_{i,N+2})$ , 随机选择  $v_{i,i}$  为待突变基因。计算  $P_G$  第  $v_{i,i}$  行的累积概率  $q(j) (j \in K/\{0, N+1\})$ , 如下所示:

$$q(j) = \sum_{m=1}^j P_{v_{i,i}}(m) \quad (14)$$

在  $(0, 1)$  内生成随机数  $r$ , 则  $r$  总能位于某一区间  $(q(j), q(j+1)]$  内, 由此生成了  $v_{i,i}$  基因在  $v_i$  中新的一位序  $j$ , 将基因  $v_{i,i}$  和  $v_{i,j}$  交换位置, 即完成了突变操作。高斯概率矩阵生成算法及其应用于变异算子算法的伪代码如图3和图4所示。

算法1 高斯概率矩阵生成算法

输入: 回收点距离矩阵  $D$ , 各回收点回收量构成的向量  $e$   
输出: 高斯概率矩阵  $M$   
1: 回收点按与起点距离升序排序后的编号序列  $a$ :  
     $a = \text{sort\_ascend}(D(1, 2:(N+1)))$ ;  
2: 回收点按与终点距离降序排序后的编号序列  $b$ :  
     $b = \text{sort\_descend}(D(N+2, 2:(N+1)))$ ;  
3: 回收点按回收量升序排序后的编号序列  $u$ :  
     $u = \text{sort\_ascend}(e)$ ;  
4: for  $i = 1: \text{size}(e, 2)$   
5: 求基础序列矩阵  $H$  的第  $i$  行元素:  
     $H(i, :) = [\text{find}(a, i), \text{find}(b, i), \text{find}(u, i)]$ ;  
6: 计算高斯分布参数  $\mu_i$  和  $\sigma_i$ :  
     $\mu_i = \text{mean}(H(i, :))$ ;  
7:  $\sigma_i = \text{standard\_dev}(H(i, :))$ ;  
8: 计算截断高斯分布  $\psi(\mu_i, \sigma_i, c, d; x)$  和  $\Psi(\mu_i, \sigma_i, c, d; x)$ :  
     $\psi(\mu_i, \sigma_i, c, d; x) = \phi(\mu_i, \sigma_i^2; x) / (\Phi(\mu_i, \sigma_i^2; d) - \Phi(\mu_i, \sigma_i^2; c))$ ;  
9:  $\Psi(\mu_i, \sigma_i, c, d; x) = (\Phi(\mu_i, \sigma_i^2; x) - \Phi(\mu_i, \sigma_i^2; c)) / (\Phi(\mu_i, \sigma_i^2; d) - \Phi(\mu_i, \sigma_i^2; c))$ ;  
10: 离散化处理得到矩阵  $M$ :  
    for  $j = 1: \text{size}(e, 2)$   
11:  $M(i, j) = \Psi(\mu_i, \sigma_i, 0.5, N+0.5, j+0.5) - \Psi(\mu_i, \sigma_i, 0.5, N+0.5, j-0.5)$ ;  
12: end  
13: end

图3 高斯概率矩阵生成算法

Fig.3 Gaussian probability matrix generation algorithm

### 3 实验验证

#### 3.1 实验数据准备

为了验证算法在实际问题中的适用性,采用上海地区回收点的实际数据开展实验仿真。实验所需的地点在上海市区地图上随机产生。随机选取20个坐标点作为废旧家电回收点,选取一汽车运输公司为回收车辆起点,一家电回收市场为终点。选取点如图5所示。图5对应地址如表1所示。任意两

算法2 高斯矩阵变异算子

输入: 高斯概率矩阵  $M$ , 种群  $\Omega$   
输出: 加入变异个体的新种群  $\Omega'$   
1: for  $j = 3: \text{size}(\Omega, 1)$   
2: 初始化  $v$ 、 $\Omega'$  和  $n$ :  
     $v = \Omega(j, :)$ ,  $\Omega' = \Omega$ ;  
3:  $n = \text{size}(v, 2)$ ;  
4: 生成随机数向量  $r$ :  
     $r = \text{rand}(1, n)$ ;  
5: 计算变异基因位序序列  $\pi$ :  
     $\pi = \text{find}(r < P_m)$ ;  
6: for  $i = 1: \text{size}(\pi, 2)$   
7:  $v' = v$ ;  
8: 选取待变异基因站点  $c$ :  
     $c = v'(\pi(i))$ ;  
9: 计算矩阵  $M$  的第  $c$  行的累积概率:  
     $q = \text{cumsum}(M(c, :))$ ;  
10: 选择站点  $c$  新的位序  $\gamma$ :  
     $\gamma = \text{find\_first}(q \geq \text{rand})$ ;  
11: 完成突变并将新的个体加入种群:  
     $v'(\pi(i)) = v'(\gamma)$ ;  $v'(\gamma) = c$   
12:  $\Omega' = [\Omega'; v']$ ;  
13: end  
14: end

图4 高斯矩阵变异算子对应算法

Fig.4 Algorithm of Gaussian matrix mutation operator

地址之间的距离采用地图导航平台提供的道路行驶路线距离,通过 Python 程序调用开放地图平台(API)进行批量计算。各回收点在其服务区域内回收的废旧家电量如表2所示。

#### 3.2 基于上海地区实际数据的算法性能分析

实验仿真所采用的计算机配置为 Intel®Core™i5-8250U@1.60GHz,内存为8GB。程序由 MatlabR2019b 进行编译。分别采用基于高斯矩阵变异算子的改进遗传算法(GA-GMM)、经典遗传算法(GA)<sup>[18]</sup>和ACO算法(下文简称ACO)<sup>[19]</sup>对实例数据进行求解,比较求解性能。

GA-GMM和GA的参数设置如表3所示,ACO的参数设置如表4所示。由于GA-GMM、GA和ACO 3种算法均为基于概率的启发式算法,单次求解结果无法准确反映算法性能,因此每种算法最大迭代次数  $T$  均设为200,每种算法重复运行200次,对200次的运行结果进行统计分析,结果如表5所示。GA-GMM求解得到的回收路径为  $0 \rightarrow 7 \rightarrow 10 \rightarrow 18 \rightarrow 12 \rightarrow 19 \rightarrow 13 \rightarrow 15 \rightarrow 11 \rightarrow 3 \rightarrow 14 \rightarrow 2 \rightarrow 5 \rightarrow 1 \rightarrow 8 \rightarrow 4 \rightarrow 6 \rightarrow 9 \rightarrow 17 \rightarrow 16 \rightarrow 20 \rightarrow 21$  (数字代表表1中的地址编号)。从表5可知,GA-GMM和GA均可以得到合理的路径规划方案且求解精度相差较小。GA的均值最小,GA-GMM与GA的均值相差0.69%;ACO求得的解较差,ACO均值与GA-GMM的存在

表 1 各回收点、起点及终点地址  
Tab.1 Recycling points, start and end addresses

编号	地址	属性	编号	地址	属性
0	上海市嘉定区园大路 55 号	起点	11	上海市嘉定区临夏路 1000 弄 77 号	回收点
1	上海市长宁区长宁路 1958 号	回收点	12	上海市浦东新区桃林路 608 号	回收点
2	上海市普陀区大渡河路 1262 弄 22 号	回收点	13	上海市长宁区剑河路 838 号	回收点
3	上海市普陀区靖边路 200 号	回收点	14	上海市普陀区同普路 1778 号	回收点
4	上海市静安区海防路 520 号	回收点	15	上海市闵行区北翟路 1444 弄 518 号	回收点
5	上海市长宁区古北路 17 号	回收点	16	上海市徐汇区茶陵路 225 弄 9 号	回收点
6	上海市静安区胶州路 669 号	回收点	17	上海市黄浦区雁荡路 78 号	回收点
7	上海市普陀区陕西北路 1712 号	回收点	18	上海市杨浦区控江路 1525 弄	回收点
8	上海市长宁区武夷路 728 号	回收点	19	上海市徐汇区钦江路 123 号	回收点
9	上海市静安区南京西路 1795 号	回收点	20	上海市浦东新区龙阳路 588 号	回收点
10	上海市虹口区丰镇路 151 弄 10 号	回收点	21	上海市浦东新区华中路金茂商务楼	终点

表 2 各回收点对应回收量  
Tab.2 Recycled mass of each collection point

回收点编号	回收量/t	回收点编号	回收量/t
1	0.2	11	0.7
2	0.3	12	0.5
3	0.3	13	0.3
4	0.5	14	0.2
5	0.3	15	0.5
6	0.9	16	0.6
7	0.1	17	0.3
8	0.2	18	0.4
9	0.4	19	0.5
10	0.3	20	0.2



图 5 各回收点分布

Fig.5 Distribution of recycling points

6.50% 的差距。变异系数(CV)又称为标准差率,为标准差与平均值之比,用以比较均值不等变量的离散程度。从表 5 可知,GA-GMM 相较于 GA 变异系数仅增加了 0.40%,ACO 的变异系数值最小,求解效果整体较差。3 种算法的耗时如表 6 所示。ACO 运行 200 次的耗时( $t_{200}$ )最短,其次是 GA-GMM,GA 耗时最长,GA-GMM 相较于 GA 平均单次耗时( $t_1$ )减少了 48.74%。实验结果表明:3 种算法中 ACO 的求解效果最差;GA-GMM 和 GA 能够求得相同的最优解,GA-GMM 在平均值偏差和变异系数 2 个指标上与 GA 仅存在 0.69% 和 0.40% 的差距,均在 1% 以内。考虑到解的结果具有一定的随机性,GA-

GMM 具有较高的求解精度,相较于 GA,运行速度提升明显。

表 3 GA-GMM、GA 参数设置  
Tab.3 Parameter setting of GA-GMM and GA

种群规模	交叉概率	变异概率	迭代次数
80	0.9	0.5	200

表 4 ACO 参数设置  
Tab.4 Parameter setting of ACO

蚁群规模	信息素启发因子	期望启发因子	信息素挥发因子	迭代次数
80	0.9	0.3	200	100

表 5 GA-GMM、GA 和 ACO 结果对比  
Tab.5 Comparison of calculation results among GA-GMM, GA and ACO

算法	运输成本最小值/最大值	运输成本平均值	变异系数/%	平均值偏差 $\Delta\tau$ /%
GA-GMM	576.39/681.22	596.10	3.48	0.69
GA	576.39/667.24	592.00	3.08	0
ACO	603.21/671.11	630.50	1.54	6.50

最大值、最小值及平均值单位均为元。

表 6 算法耗时比较  
Tab.6 Comparison of time-consuming between different algorithms

算法	$t_{200}/s$	$t_1/s$	$\frac{t_{1(GA)} - t_{1(GA-GMM或ACO)}}{t_{1(GA)}}/\%$
GA-GMM	1 386.80	6.93	48.74
GA	2 703.28	13.52	0
ACO	1 325.28	6.63	50.96

图 6 为 3 种算法重复运行 200 次的平均收敛曲线,曲线每一点对应算法 200 次重复运行中当前迭代次数的均值。从图 6 可以看出,GA-GMM 的收敛速度明显快于 GA 和 ACO,在 40 次迭代时运输成本已经接近收敛值,而 GA 和 ACO 在快速收敛阶段之后,仍需经过缓慢收敛才能得到最终稳定解。表 7



列出了运输成本为600、605、610、615和620时的平均迭代次数。GA-GMM相较于GA,收敛速度分别提高了58.62%、62.32%、58.93%、55.56%和52.50%,具有更好的搜索能力。GA-GMM相较于GA能在保持较高求解质量的同时,加快收敛速度,降低算法运行时间,提高求解效率。

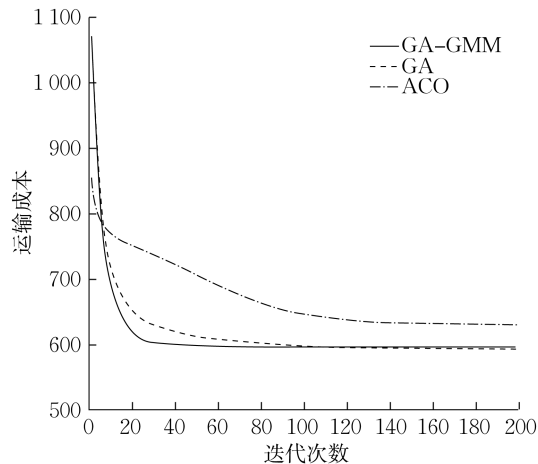


图6 算法收敛曲线

Fig.6 Convergence curves of three algorithms

表7 算法收敛性比较

Tab.7 Comparison of algorithm convergence

运输成本	各算法收敛速度提高率/%		
	GA	GA-GMM	ACO
600	0	58.62	—
605	0	62.32	—
610	0	58.93	—
615	0	55.56	—
620	0	52.50	—

注:ACO未收敛至表中运输成本值。

在上述实验的基础上,针对GA-GMM在不同变异概率情况下的求解性能进行进一步研究。实验中除变异概率外其他参数与上述实验保持一致,取0.1~0.8间以0.1为间隔的8个变异概率值,得到的平均成本收敛曲线如图7所示。

从图7可以看出,在快速收敛阶段,随着 $P_m$ 的增加,算法的收敛速度逐渐加快,但在 $P_m > 0.4$ 时,收敛速度加快的趋势明显下降,继续增大 $P_m$ 对算法收敛速度影响较小。在迭代次数接近200次时,算法收敛值基本稳定, $P_m = 0.1$ 时稳定收敛值最大, $P_m = 0.5$ 时稳定收敛值最小,最大值与最小值之间相差1.33%,差距较小。过高的变异概率对算法的性能提升并没有太大作用,相反还会由于变异次数的增加而导致算法耗时更长,适当的变异概率选择对算法的求解性能有重要的影响。

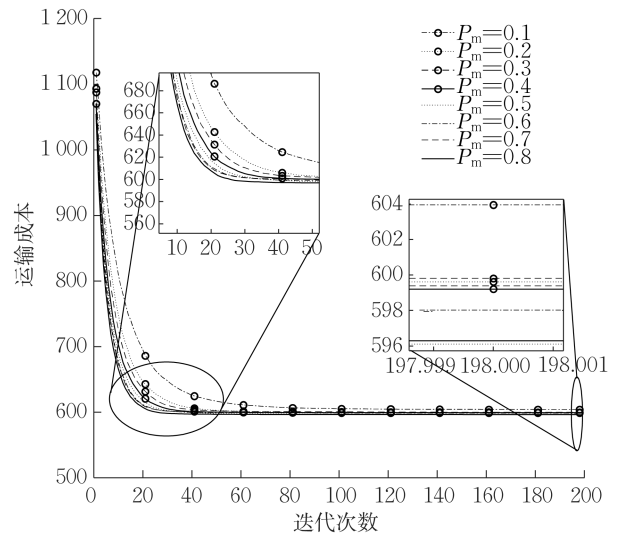


图7 不同变异概率下GA-GMM平均成本收敛曲线

Fig.7 Convergence curves of average cost of GA-GMM under different mutation probabilities

## 4 结语与展望

针对废旧家电回收车辆路径优化问题的研究,提出了一种基于高斯矩阵变异算子的遗传算法。基于原始的回收点距离及回收量信息构建包含回收点位序分布特性的高斯概率矩阵模型,并通过轮盘赌选择法将高斯概率矩阵作用于遗传算法的染色体基因突变,实现了在保持种群基因多样性的同时引导种群向高适应度方向变异。选取了上海地区家电回收点实际数据进行求解,对算法的性能进行了对比和分析。仿真结果表明,基于高斯矩阵变异算子的改进遗传算法在保持较高的求解质量的同时,在收敛速度、搜索能力和算法耗时等方面均优于传统的遗传算法和蚁群优化算法,同时变异概率的选择也会在一定程度上影响求解性能。

在后续的研究中,将结合实际的运输情况,在建立模型时考虑回收的货物特征、载物平台物理特征和道路特征等以及由道路拥堵情况、装载货物方式等产生的时效及运输成本,进一步优化现有算法,实现对更加复杂优化问题的求解。同时,在求解相似类型的问题时,也可以将高斯矩阵变异算子与其他智能优化算法相结合,实现算法优势互补。

### 作者贡献声明:

黄新林:提供整体思路,论文修订。

张隆飞:算法设计,论文撰写。

唐小伟:算法设计指导,论文修订。

## 参考文献:

- [1] DANTZIG G, FULKERSON R, JOHNSON S. Solution of a large-scale traveling-salesman problem [J]. Journal of the Operations Research Society of America, 1954, 2(4): 393.
- [2] HERNANDEZ-PEREZ H, SALAZAR-GONZALEZ J J. The one-commodity pickup-and-delivery travelling salesman problem [M]. Berlin: Springer, 2003.
- [3] JAILLET P. A priori solution of a traveling salesman problem in which a random subset of the customers are visited [J]. Operations Research, 1988, 36(6): 929.
- [4] BEKTAS T. The multiple traveling salesman problem: an overview of formulations and solution procedures [J]. Omega, 2006, 34(3): 209.
- [5] VANSTEENWEGEN P, SOUFFRIAU W, SORENSEN K. The travelling salesperson problem with hotel selection [J]. Journal of the Operational Research Society, 2012, 63(2): 207.
- [6] PALACIO J D, RIVERA J C. A multi-start evolutionary local search for the one-commodity pickup and delivery traveling salesman problem [J]. Annals of Operations Research, 2020, 316(3): 1.
- [7] 王勇臻, 陈燕, 于莹莹. 求解多旅行商问题的改进分组遗传算法 [J]. 电子与信息学报, 2017, 39(1): 198.  
WANG Yongzhen, CHEN Yan, YU Yingying. Improved grouping genetic algorithm for solving multiple traveling salesman problem [J]. Journal of Electronics & Information Technology, 2017, 39(1): 198.
- [8] SOUSA M M, OCHI L S, COELHO I M, *et al.* A variable neighborhood search heuristic for the traveling salesman problem with hotel selection [C]//2015 Latin American Computing Conference (CLEI). Arequipa: IEEE, 2015: 1-12.
- [9] EKMEKCI D. An ant colony optimization memorizing better solutions (ACO-MBS) for traveling salesman problem [C]//3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT). Ankara: IEEE, 2019: 1-5.
- [10] PHU-ANG A. An improve artificial immune algorithm for solving the travelling salesman problem [C]//Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering. Cha-am: IEEE, 2021: 261-264.
- [11] WU C, FU X, PEI J, *et al.* A novel sparrow search algorithm for the traveling salesman problem [J]. IEEE Access, 2021, 9: 153456.
- [12] AKTER S, NAHAR N, SHAHADATHOSSAIN M, *et al.* A new crossover technique to improve genetic algorithm and its application to TSP [C]//International Conference on Electrical, Computer and Communication Engineering (ECCE). Cox's Bazar: IEEE, 2019: 1-6.
- [13] BAO C, YANG Q, GAO X D, *et al.* Genetic algorithm with adapted crossover operators for multiple traveling salesmen problem with visiting constraints [C]//IEEE International Conference on Systems, Man, and Cybernetics (SMC). Prague: IEEE, 2022: 3033-3039.
- [14] PRASAD J S, JAIN V. An optimized algorithm for solving travelling salesman problem using greedy cross over operator [C]//3rd International Conference on Computing for Sustainable Global Development (INDIACom). New Delhi: IEEE, 2016: 2981-2984.
- [15] LYU J, ZENG Y, ZHANG R, *et al.* Placement optimization of UAV-mounted mobile base stations [J]. IEEE Communications Letters, 2016, 21(3): 604.
- [16] HOLLAND J H. Genetic algorithms and the optimal allocation of trials [J]. SIAM Journal on Computing, 1973, 2(2): 88.
- [17] 包子阳, 余继周, 杨杉. 智能优化算法及其 MATLAB 实例 [M]. 北京: 电子工业出版社, 2016.  
BAO Ziyang, YU Jizhou, YANG Shan. Intelligent optimization algorithm and MATLAB example [M]. Beijing: Publishing House of Electronics Industry, 2016.
- [18] HARIYADI P M, NGUYEN P T, ISWANTO I, *et al.* Traveling salesman problem solution using genetic algorithm [J]. Journal of Critical Reviews, 2020, 7(1): 56.
- [19] CUI S, HAN S. Ant colony algorithm and its application in solving the traveling salesman problem [C]//3rd International Conference on Instrumentation, Measurement, Computer, Communication and Control. Shenyang: IEEE, 2013: 1200-1203.