

DOI: 10.3969/j.issn.1009-9492.2023.10.012

刘婷, 王孙康宏, 陈壮耿, 等. 基于群体智能算法的通勤车辆路径优化问题 [J]. 机电工程技术, 2023, 52 (10): 72-78.

基于群体智能算法的通勤车辆路径优化问题*

刘 婷, 王孙康宏, 陈壮耿, 魏丽军*

(广东工业大学 机电工程学院, 广州 510006)

摘要: 为解决员工通勤难的问题, 面向大型企业为员工提供统一接送通勤服务, 研究多目的地和多车型的車輛路径问题, 并建立相应的数学模型, 提出群体智能算法对其进行求解。基于现实生活中通勤服务车辆实际约束, 采用混合整数规划方法, 以最小化运营成本为目标构建多目的地和多车型的車輛路径问题的数学模型。提出一种基于S-N链的解表示方法以及对应的解码过程和评价准则, 并采用群体智能算法中的蜘蛛猴优化算法对问题进行求解。为验证蜘蛛猴优化算法的有效性, 将其与粒子群优化算法进行比较。结果表明, 在相同求解时间下, 蜘蛛猴优化算法求解此问题的性能更优。对17组随机算例进行测试, 验证所提数学模型和蜘蛛猴优化算法能够有效解决多目的地和多车型的車輛路径问题。

关键词: 多目的地多车型車輛路径规划; 取送货问题; 群体智能算法; 蜘蛛猴优化算法

中图分类号: U492.2

文献标志码: A

文章编号: 1009-9492(2023)10-0072-07

Swarm Intelligence Algorithm for the Multi Destination Vehicle Routing Problem

Liu Ting, Wang Sunkanghong, Chen Zhuangeng, Wei Lijun

(School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: To solve the problem of employee commutes laborious, a multi-destination and multi-type vehicle routing problem is studied for large enterprises to provide uniform transfer and commuting services for employees. The corresponding mathematical model is built, and the swarm intelligence algorithm is used to solve the optimal solution. Based on the actual constraints of commuting service vehicles in real life, a mathematical model of multi-destination and multi-vehicle routing problem is constructed by using mixed integer programming method to minimize operating costs. A solution representation method based on S-N chain, the corresponding decoding process and evaluation criteria are developed, and the spider monkey optimization algorithm in swarm intelligence algorithm is used to solve the problem. In order to verify the effectiveness of spider monkey optimization algorithm, it is compared with particle swarm optimization algorithm. The results show that spider monkey optimization algorithm has better performance in solving this problem under the same solving time. 17 groups of multi-group random examples are tested to verify that the mathematical model, by using the spider monkey optimization algorithm, the multi-destination and multi-vehicle routing problems can be effectively solved.

Key words: multi-destination vehicle routing problem; pickup and delivery problem; swarm intelligence algorithm; spider monkey optimization algorithm

0 引言

通勤服务企业或大型企业自身的通勤服务团队为员工提供统一接送服务, 通勤团队首先设定多个乘车点, 乘客可就近选择上下车站点, 每个站点会有前往多个目的地或厂区的乘客, 通勤团队需合理进行车辆、车型和运行路线规划, 保证把乘客送达目的地, 在保障运营工作的同时实现运营企业全年运营成本最优。

大型企业为员工提供统一接送通勤服务相当于多目的地多车型車輛路径规划问题, 是取送货車輛路径规划问题 (Pickup and Delivery Problem, PDP) 的延伸, 扩展了PDP的应用场景。近年来, 国内外学者针对不同的业务场景和用户规模进行了大量的研究和实践, 提出了诸多策略, 已取得显著成果。王科峰等^[1]从精确算法、并

行算法、构造型启发式算法和现代启发式算法4个方面对同时取送货車輛路径问题进行了详细的评述。徐宁等^[2]为提高大规模同时取送货車輛路径优化问题的求解效率和精度, 基于先分解-再求解的解决思路, 提出了两阶段混合算法RPCHVNS求解该类问题。裴颂文等^[3]在逆向物流的路径规划问题上展开了研究, 针对逆向物流需求的不确定性提出了KMG模型, 该模型融合了带权值的拓展性k-means++聚类算法和遗传算法, 有效解决了包含逆向物流的无人机调度问题。吴腾宇等^[4]面向外卖的及时配送需求, 将配送車輛返回原点取货的约束融入在线旅行商问题中, 基于固定取货点的思想设计了TAIB算法和IGNORE算法对该问题进行求解。陈雪等^[5]以最小化碳排放成本为优化目标, 提出了一种学习型蚁群优化

基金项目: 国家自然科学基金资助项目 (71871070)

收稿日期: 2023-02-02

算法 LACO, 用于求解带同时取送货的绿色两级车辆路径问题, 通过仿真实验对比, 验证 LACO 算法的有效性。包胜男等^[6]基于同城速运背景, 考虑订单的时间窗要求构建 VRPSPDTW 模型, 采用改进的遗传算法求解了车辆最优配送路径。李博威等^[7]基于软时间窗和同时取送货的双重需求, 同时考虑行驶里程、车辆数目、客户满意度等因素构建了 MINLP 模型, 采用理想点法对多目标优化问题进行简化求解。李志涛等^[8]采用基于硬时间窗约束对需要二次清运的回收车辆路径问题进行研究, 构建硬时间窗约束的 VRP 模型对问题进行求解。Belgin 等^[9]研究了同时取送货的两级车辆路径问题, 开发了一种基于可变领域下降和局部搜索的混合启发式算法求解此问题, 并有效解决了中大规模配送问题。Majidi 等^[10]针对同时取送货的车辆排污路径问题, 构建了一种非线性混合整数规划模型, 提出用自适应大邻域搜索算法求解此问题, 通过对比两类基准实例验证算法的有效性。Lagos 等^[11]考虑时间窗口约束, 采用改进的粒子群优化算法求解了同时取送货和时间窗的车辆路径问题。Chentli 等^[12]研究了固定数量同时取送货的车辆路径问题, 提出用自适应大邻域搜索算法求解最优行驶路线, 构建了多个算例验证算法有效性。Qiu 等^[13]面向逆向物流中生产工艺路线问题, 提出了一种分支定界引导搜索算法的求解方法, 此算法在求解大规模算例时具有良好的性能。Zhao 等^[14]对同时取送货的选址与路径问题展开了研究, 提出了一种超启发式算法框架对其进行优化求解, 并开发了 4 种选择机制和 5 种激活策略检验框架的性能。Ma 等^[15]针对考虑带时间窗和多决策者的同时取送货的车辆路径问题, 设计了一种混合优先级遗传算法对此问题进行求解, 并通过多个不同规模的算例验证算法的适用性。Afra 等^[16]研究了考虑启动成本和环保因素的取送货的车辆路径问题, 采用了拉格朗日松弛算法求解此问题。Park 等^[17]针对同时取送货车辆路径问题提出一种等待策略, 并开发了一种遗传算法解决大规模问题。

综上所述, 国内外学者们针对 PDP 延伸问题的研究, 主要集中在搭建问题求解模型和算法框架, 并通过大量实例验证算法的有效性, 为本文研究的多目的地多车型车辆路径规划问题的求解提供了参考。本文考虑实际应用场景, 以通勤服务企业运营成本最小为目标, 依据实际约束搭建了相应的数学模型, 并采用群体智能算法对问题进行优化处理, 通过不同规模的算例验证算法的有效性。

1 多目的地多车型车辆路径规划问题描述与数学模型

1.1 问题描述

对于运营服务企业而言, 运营成本与车辆数量、车型、行驶路径有关, 因此需要通过计算获取不同车型车

辆数量和不同车辆的运行路线, 使车辆数量尽可能少且总里程尽可能短, 保障运营成本总能耗最小。

本文的假设条件如下: (1) 通勤团队设定多个乘车点, 所有站点位置信息已知 (包含车辆统一停靠区域、目的地、上下车站点); (2) 每个乘车点有前往多个目的地的乘客, 且每个乘车点对应的不同目的地的乘客数量已知; (3) 里程用欧氏距离表示; (4) 只考虑单程, 即车辆返回统一放置点的路程不考虑; (5) 不同类型车辆的最大乘客承运量已知, 车辆运营过程中不得超载; (6) 不考虑运营车辆数不足和车辆需要充电、加油的情况; (7) 车辆第 2 次空车后不允许再搭载乘客 (第 1 次空车是出发时)。

多目的地多车型车辆路径规划问题的最终目标为找出一个路径集合, 对于每个路径的每个站点需要安排多少车辆搭载多少数量的乘客、搭载去往什么目的地的乘客, 并为每个路径分配合适的车型, 使得路径集合的总能耗最低。

1.2 数学模型

多目的地多车型车辆路径规划问题的特点是每个站点的搭载乘客数是变量, 搭载乘客的类型也是变量 (以目的地区分乘客类型), 问题的变量较多, 比较复杂。基于多目的地多车型车辆路径规划问题与 PDP 相似, 利用求解 PDP 的一些算法来对该问题进行启发式求解。建立 PD (Pickup and Delivery, PD) 模型求解多目的地多车型车辆路径规划问题, 下面是对 PD 模型的介绍。

设 0 为车辆统一放置中心; T 为车型数目; Q_k 为车型 k 的最大载客量; p_j 为第 j 个站点的乘客数量; c_{ij}^k 为第 k 种车型从站点 i 行驶至站点 j 的能耗 ($c_{ij}^k = c_{ji}^k$); d_{ij} 为站点 i 到站点 j 的欧氏距离; δ^k 为第 k 种车型的单位能耗; y_{ij} 为站点 i 到站点 j 的乘客装载量。此外, 决策变量取值规则如下。

$$x_{ij}^k = \begin{cases} 1 & \text{第 } k \text{ 种车型从站点 } i \text{ 行驶至站点 } j \\ 0 & \text{其他情况} \end{cases} \quad (1)$$

综上, 具体数学模型可表示如下。

$$\min \sum_{k=1}^T \sum_{i=0}^n \sum_{j=0}^n c_{ij}^k x_{ij}^k \quad (2)$$

$$\text{s.t.} \sum_{k=1}^T \sum_{i=0}^n x_{ij}^k = 1 \quad j = 1, \dots, n \quad (3)$$

$$\sum_{i=0}^n x_{ij}^k - \sum_{l=0}^n x_{jl}^k = 0 \quad j = 0, \dots, n, \quad k = 1, \dots, T \quad (4)$$

$$\sum_{i=0}^n y_{ij} - \sum_{l=0}^n y_{jl} = p_j \quad j = 1, \dots, n \quad (5)$$

$$y_{0j} \leq \sum_{k=1}^T Q_k x_{0j}^k \quad j = 1, \dots, n \quad (6)$$

$$y_{ij} \leq M \sum_{k=1}^T x_{ij}^k \quad i \neq j = 0, \dots, n, \quad M = \infty \quad (7)$$

$$c_{ij}^k = d_{ij} \delta^k \quad (8)$$

$$y_{ij} \geq 0 \quad i \neq j = 0, \dots, n \quad (9)$$

$$x_{ij}^k \in \{0, 1\} \quad i \neq j = 0, \dots, n, \quad k = 1, \dots, T \quad (10)$$

其中, 目标函数(2)表示最小化总能耗; 约束(3)和(4)保证车辆到达目的地后乘客的下车; 约束(5)表示乘客的移动, 确保每个乘客都能被送往目的地; 约束(6)确保车辆在行驶过程中搭载的乘客数量不超过其最大载客量; 约束(7)表示如果没有车辆从站点*i*行驶至站点*j*, 则从站点*i*到站点*j*就没有人员流动。

PD模型支持一辆汽车在到达站点时搭载任意数量的乘客(小于等于车辆的最大乘客承运量即可), 并且支持一条路径内包含多目的地, 有概率得到更好的解; 缺点是解的搜索空间巨大, 需要搜索很久才能够得到较好的解。

2 群体智能算法求解PD模型

Kachitvichyanukul等^[18]提出了两种求解具有多取多送的广义多车辆段车辆路径问题(GVRP-MDMPDR)的求解方法, 该方法可以与GLNPSO结合使用, 初步结果表明, 他们所提出的方法能够为大多数测试问题提供良好的解决方案。基于此, 针对多目的地多车型车辆路径规划问题开发了一种基于S-N链的解表示方法、解码过程和评价规则, 可以很好地结合群体智能算法框架对本问题进行求解。群体智能算法求解PD模型流程如图1所示。

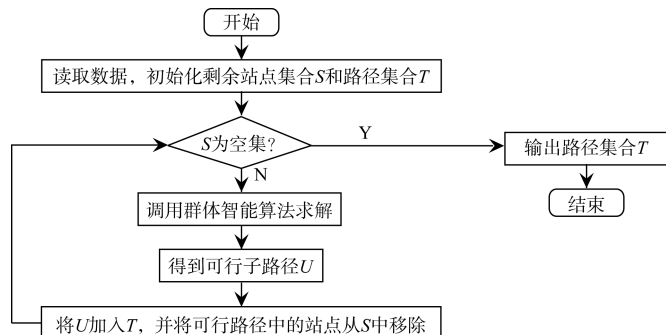


图1 群体智能算法求解PD模型流程

2.1 基于S-N链的解表示方法

群体智能算法的核心是智能体, 每一个智能体都有一定维度数的自变量数组, 解的表示方法就是基于自变量数组的表示方法。

为表示解, 本文给每个智能体定义了两条自变量编码链(S-N链), 一条用来控制站点访问顺序, 称之为S链, 位于S链中的自变量定义域为[0, 1]; 另一条用来控制每个站点的接送人数, 称之为N链, 位于N链的自变量的定义域为[0, $p_j + 1$], p_j 为第*j*个位置元素对应乘车点及对应目的地的乘客人数。假设只有3个乘客点和3个目的地, 则基于S-N链的解表示方法如图2所示。

2.2 解码过程

解码过程即根据S-N链的信息, 通过一定规则获取一个可行路径的过程, 具体为以下7个步骤。

	目的地A:			目的地B:			目的地C:		
S链:	0.12	0.67	0.16	0.57	0.09	0.78	0.54	0.95	0.36
N链:	0.71	8.64	7.32	10.84	1.27	3.11	4.77	6.42	5.75

图2 基于S-N链的解表示方法

步骤1: 按照S链进行不减排序, 然后将S链替换为标号, 并对N链进行向下取整;

步骤2: 初始化人数记录数组*P*, 当前载客量*N*=0, 历史载质量*N'*=0, 当前指针*i*=1, 路径记录集合*L*=*S*;

步骤3: 如果当前指针超出编码链长度或者当前载客量为0且历史载质量不为0, 则程序结束; 如果当前指针所指元素的目的地已经在集合*L*中, 则*i*=*i*+1, 返回步骤3; 如果上面条件都不符合则判断指针所指的乘客点编号是否位于集合*L*中, 如果位于集合*L*中, 则转步骤4, 否则转步骤5;

步骤4: 判断不等式 $p_{l_i} + N' \leq Q$ 是否成立, 如果成立, 则转步骤6, 否则转步骤7;

步骤5: 判断不等式 $p_{l_i} + N \leq Q$ 是否成立, 如果成立, 则将当前指针对应乘客点编号加入集合*L*, 转步骤6, 否则转步骤7;

步骤6: 按照当前指针对应目的地更新人数记录数组 $P_{l_i} = P_{l_i} + p_{l_i}$, 更新当前载客量和历史载质量 $N = N + p_{l_i}$, $N' = N' + p_{l_i}$, *i*=*i*+1转步骤3;

步骤7: 设集合*L*的最后一个非目的地且乘客数非0的元素*J*, 则选择与乘客点*J*最接近的且不存在于集合*L*中的目的地*M*加入集合*L*, 更新人数记录数组和当前载客量 $P_{l_i} = P_{l_i} - p_{l_i}$, $N = N - p_{l_i}$, $\forall i \in$ 目的地*M*, *i*=*i*+1转步骤3。

假设此路径车辆的最大载客量为15, 基于所提S-N链的解码过程如图3所示。

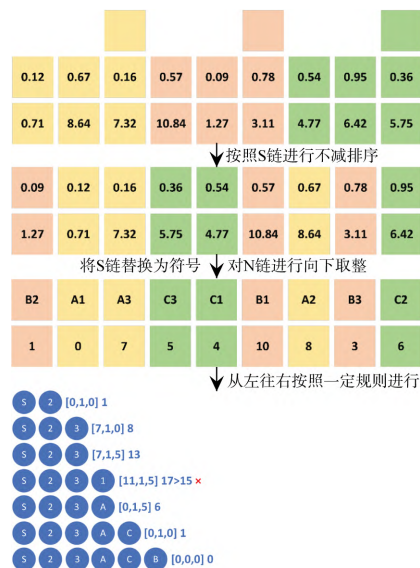


图3 解码过程示意

2.3 评价规则

如图3所示, 最终解码得到的一条可行路径为S→2→3→A→C→B, 其中在乘车点2上车人数为[0, 1, 0], 在乘车点3上车人数为[7, 0, 5], 这条路径的总载客量 $T=1+7+5=13$ 。设 L 为该路径总长度, 设 C 为当前车型的百公里能耗, 则该可行路径的评价值可以由式(11)表示。

$$Z = \begin{cases} \infty & T = 0 \\ \frac{LC}{T^{1.2}} & T \neq 0 \end{cases} \quad (11)$$

同样的S链和N链, 采用不同车型解码得到的路径都不一定相同, 故在评价时, 需要遍历每一种车型解码, 获得 k 个评价值 Z 和 k 条可行路径, 从中选取 Z 值最小的可行路径作为最终解码的结果返回。

2.4 群体智能算法

蜘蛛猴优化算法 (Spider Monkey Optimization algorithm, SMO) 是一种元启发式算法, 是 Bansal 等^[19]根据自然界中蜘蛛猴的智能觅食行为提出的一种新型群体智能算法。SMO 算法主要通过6个阶段实现优化问题的解决, 具体如下。

(1) 初始化阶段: 随机在 D 维优化求解空间中生成一个由 N 个蜘蛛猴构成的均匀分布的初始种群, $E_{sm,i}$ 为种群中的第 i 个蜘蛛猴, 每个 $E_{sm,i}$ 进行初始化公式为:

$$E_{sm,ij} = E_{sm,minj} + U(0,1) \times (E_{sm,maxj} - E_{sm,minj}) \quad (12)$$

式中: $E_{sm,minj}$ 、 $E_{sm,maxj}$ 分别为搜索空间中第 j 维的下限和上限; $U(0,1)$ 为 $(0,1)$ 中的随机数。

(2) 局部领导者阶段: 种群中的蜘蛛猴基于其局部领导者和小组成员的经验对自己的位置进行更新, 并在更新位置计算其适应度, 若适应度值低于其原位置, 则不更新, 否则更新。蜘蛛猴位置更新的适应度公式为:

$$E_{sm,newij} = E_{sm,ij} + U(0,1) \times (L_{L,kj} - E_{sm,ij}) + U(-1,1) \times (E_{sm,rj} - E_{sm,ij}) \quad (13)$$

式中: $E_{sm,ij}$ 为第 i 个蜘蛛猴的第 j 维变量; $L_{L,kj}$ 为第 k 组局部领导者的第 j 维变量; $E_{sm,rj}$ 为从第 r 组中随机选择的蜘蛛猴的第 j 维变量 ($r \neq i$); $U(-1,1)$ 为 $(-1,1)$ 中的随机数。

(3) 全局领导者阶段: 种群中的蜘蛛猴基于其全局领导者和局部小组成员的经验对自己的位置进行更新, 同时基于某一选择概率进行位置的更新。第 i 个蜘蛛猴适应度值 F_i 可依据目标函数 f_i 进行计算。

$$F_i = \begin{cases} \frac{1}{1+f_i} & f_i \geq 0 \\ 1 + \text{abs}(f_i) & f_i < 0 \end{cases} \quad (14)$$

选择概率 P_i 基于轮盘赌选择确定, 蜘蛛猴在全局领导者阶段被选中的概率为:

$$P_i = \frac{F_i}{\sum_{i=1}^N F_i} \text{ 或 } P_i = 0.9 \times \frac{F_i}{F_{\max}} + 0.1 \quad (15)$$

则全局领导者阶段蜘蛛猴位置更新方程为:

$$E_{sm,newij} = E_{sm,ij} + U(0,1) \times (G_{L,j} - E_{sm,ij}) + U(-1,1) \times (E_{sm,rj} - E_{sm,ij}) \quad (16)$$

式中: $G_{L,j}$ 为全局领导者在第 j 维上的位置。

(4) 全局领导者学习阶段: 寻找群体中的最优解, 被选中的蜘蛛猴代表群体的全局领导者。检查全局领导者的位置, 若位置更新, 则将全局领导者的全局限制计数设置为0, 否则增加1。

(5) 局部领导者学习阶段: 依次对每组成员进行贪婪选择, 确定每组局部领导者的位置。检查局部领导者的位置, 若位置更新, 则将局部领导者的局部限制计数设置为0, 否则增加1。

(6) 局部领导者决策阶段: 若局部限制计数达到了局部领导限制, 即在初始迭代次数内局部领导者的位置未更新, 则该组成员采用随机初始化方式或式(16)重新进行位置更新。

$$E_{sm,newij} = E_{sm,ij} + U(0,1) \times (G_{L,j} - E_{sm,ij}) + U(0,1) \times (E_{sm,rj} - L_{L,kj}) \quad (17)$$

(7) 全局领导者决策阶段: 若全局限制计数达到了全局领导限制, 即在初始迭代次数内全局领导者的位置未更新, 则对群体重新进行分组。

群体中的蜘蛛猴通过以上步骤不断更新位置, 直至达到其最优位置和最优函数值时终止算法。SMO 算法的主要参数及其作用如表1所示。

表1 蜘蛛猴优化算法的主要参数与作用

参数	作用
最大探索次数	决定局部和全局领导者决策阶段的探索次数
迭代次数	决定迭代的次数
蜘蛛猴数量	决定蜘蛛猴的数量
局部搜索次数	决定全局领导者阶段的局部搜索次数
LLDP_PR	决定局部领导者决策阶段的更新概率
LLP_PR	决定局部领导者阶段的更新概率
最大组数	决定最大的现有组数, 控制分裂与融合过程

3 实验结果与分析

3.1 实验环境

本文所有实验均在配备 Intel (R) Core (TM) i7-9750H 2.60 GHz 处理器和 16 GB RAM 的笔记本计算机上进行。该 PC 的操作系统为 64 位 Windows 10。代码均以 Java 语言编写。

3.2 SMO 算法参数分析

由于最大探索次数、迭代次数、蜘蛛猴数量和局部

搜索次数这4个参数通常来说是越大越好的,故初步暂定它们的值分别为50、10、200和200。接下来,本文对剩余的LLDP_PR、LLP_PR和最大组数这3个参数进行实验分析。

首先固定LLP_PR为0.8,最大组数为10,对LLDP_PR进行最佳取值搜索,结果如图4所示。由图可知,LLDP_PR的最佳取值搜索值为0.5。

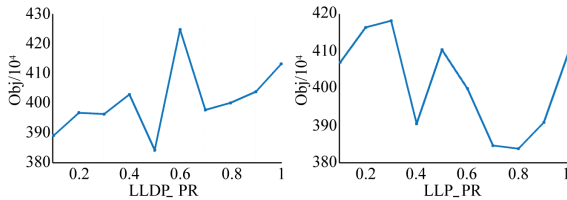


图4 LLDP_PR的取值结果

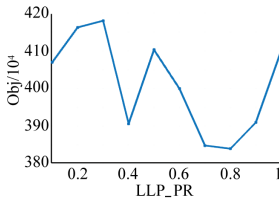


图5 LLP_PR的取值结果

因此,固定LLDP_PR为0.5,最大组数为10,然后对LLP_PR进行最佳取值搜索,结果如图5所示。由图可知,LLP_PR的最佳取值搜索值为0.8。

同理,对最大组数进行最佳取值搜索,结果如图6所示。由图可知,当最大组数为2时,目标函数值较小,又由于最大组数对目标值的影响和迭代次数紧密相连,最终决定设置最大组数为迭代次数的1/5,即最大组数为2。

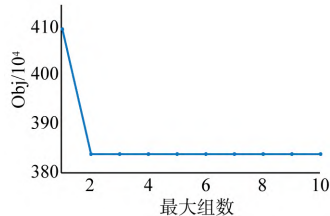


图6 最大组数的取值结果

3.3 结果与分析

本文测试数据来源于2022年数字汽车大赛创新组赛题一,详细数据可在大赛官网下载。本文对求解PD模型的不同方法分别进行了测试,并得到了不同求解时间下,不同算法得到的结果,结果如表2所示。由于大部分方法属于随机算法,具有随机性,本文取10次实验结果的平均值作为表3的数据来源。为验证SMO算法的有效性,将其与粒子群优化算法(Particle Swarm Optimization, PSO)算法进行比较,并采用较大规模的算例进行测试。其中,PSO算法主要参数设定为:惯性权重 $\omega=0.6$ 、个体学习因子 $c_1=0.5$ 、社会学习因子 $c_2=0.4$ 。由表2可得,SMO算法的求解结果都比PSO算法的求解结果好,说明SMO算法在求解该问题时具有更好的性能。

表2 比赛测试数据结果

		$10^{-5} \text{m} \cdot \text{kW} \cdot \text{h}$	
卡时/s	PD模型	卡时/s	PD模型
	PSO SMO		PSO SMO
10	4 548 336.84 4 519 646.24	1 800	2 851 012.78 2 057 965.03
60	3 686 572.49 3 551 079.19	3 600	2 708 294.12 2 000 211.31
600	3 192 741.84 2 418 090.06	10 800	2 562 773.61 1 838 596.39
900	2 875 488.35 2 409 755.93		

不同卡时不同算法求解结果如图7所示。由图可知,当求解时间从60 s变为600 s时,求解结果的下降最为明显,而后即使将时间增加至10 800 s,下降也没有60~600 s这一阶段明显,说明群体智能算法在600 s左右时即可求得较优解。

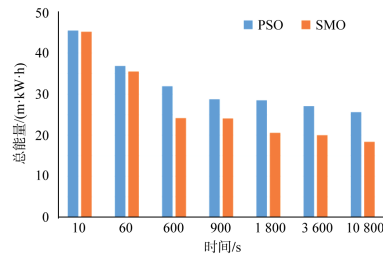


图7 不同卡时不同算法求解测试数据结果对比柱状图

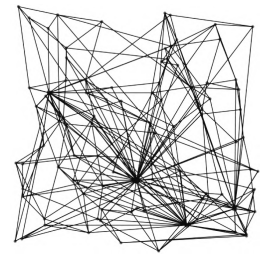


图8 车辆路径可视化

当算法运行时间为10 800 s时,SMO算法求解PD模型的结果如表3所示,车辆路径可视化结果如图8所示。

3.4 随机测试数据结果与分析

为了进一步测试和验证算法的可移植性,构造了17组不同站点数、不同目的地数的随机测试案例,每个随机案例的特征取值范围依据3.3节比赛测试数据中对应特征的最大和最小值所确定。表4所示为比赛测试数据的描述性信息,包括每个特征的平均值Avg、最小值Min、最大值Max、标准差Std。

为了更简洁地表达随机案例的属性,将随机案例命名为PX-Y,其中X为随机案例中的站点数(包含目的地和车辆统一停靠区域),Y为目的地数。例如P60-2表示一个拥有60个站点和2个目的地的随机案例。针对17组随机构造的测试案例,进行了卡时300 s的测试,表5所示为17组随机案例的测试结果。由表可知,300 s内求解17组随机测试案例的最佳结果有14组来自SMO算法,说明SMO算法在大部分案例中都可以在较短时间内求得较好解;也有3组最佳结果来自PSO算法,3组案例都是较大规模和较多目的地的随机案例,说明PSO算法在一定时间内对大规模的多目的地多车型车辆路径规划问题可能有更好的求解效果。图9所示为卡时300 s不同算法求解不同随机案例的结果对比柱状图。

4 结束语

本文研究了多目的地和多车型的车辆路径规划问题,根据通勤车辆接送员工上下班实际运营情况,构建相应的数学模型对此过程进行完整刻画。设计了一种基于S-N链的解表示方法以及对应的解码过程和评价准则,采用蜘蛛猴优化算法对问题进行求解。通过对多组算例进行测试,验证了所提数学模型和蜘蛛猴优化算法能够有效解决多目的地和多车型的车辆路径问题。

本文提出的PD模型和SMO算法在求解多目的地/小规模问题时有更好的效果,在未来研究中,会进一步考

表3 蜘蛛猴优化算法求解PD模型结果示例

序号	路径	选用 车型	车单位 车容量	车单位	路程 长度/m	能耗 /
				能耗 / (kW·h ⁻¹)		(10 ⁻³ ·m·kW·h)
1	[5, 27, 0, 65, 10, 78, 80]	3	30	28	573.01	16 044.52
2	[5, 43, 81, 17, 64, 45, 44, 30]	8	55	38	582.10	22 123.46
3	[5, 99, 41, 51, 94, 84, 80]	6	45	34	461.82	15 701.94
4	[5, 99, 29, 51, 78, 10]	4	35	30	457.69	13 730.77
5	[5, 99, 74, 39, 6, 30]	5	40	32	738.75	23 640.12
6	[5, 43, 49, 75, 87, 0, 10]	6	45	34	682.70	23 211.95
7	[5, 29, 18, 66, 97, 58, 80]	4	35	30	707.80	21 234.03
8	[5, 65, 0, 27, 56, 59, 30]	7	50	36	844.38	30 398.02
9	[5, 49, 69, 55, 11, 26, 10, 78, 80]	7	50	36	1 021.87	36 788.06
10	[5, 52, 67, 4, 22, 79, 30]	3	30	28	661.20	18 513.75
11	[5, 49, 69, 95, 16, 1, 89, 30]	5	40	32	657.52	21 040.68
12	[5, 43, 17, 81, 71, 68, 80]	7	50	36	995.30	35 831.13
13	[5, 3, 61, 56, 52, 32, 41, 10]	9	60	40	1 051.84	42 073.99
14	[5, 32, 38, 72, 30]	2	25	26	709.08	18 436.17
15	[5, 91, 36, 82, 37, 71, 80, 10]	6	45	34	1 189.37	40 438.85
16	[5, 11, 75, 73, 25, 7, 46, 30]	9	60	40	1 022.37	40 894.96
17	[5, 74, 16, 95, 11, 47, 10]	5	40	32	1 095.26	35 048.54
18	[5, 94, 84, 66, 12, 28, 96, 80, 10]	4	35	30	884.02	26 520.60
19	[5, 29, 35, 34, 77, 30]	5	40	32	1 099.11	35 171.76
20	[5, 32, 77, 13, 2, 19, 85, 76, 80, 10]	8	55	38	1 663.15	63 199.76
21	[5, 41, 78, 51, 94, 61, 30]	5	40	32	1 164.63	37 268.17
22	[5, 17, 25, 93, 9, 73, 10]	7	50	36	1 420.67	51 144.12
23	[5, 56, 39, 64, 50, 7, 10, 80]	6	45	34	1 591.83	54 122.43
24	[5, 34, 54, 57, 53, 24, 31, 35, 80, 10]	8	55	38	1 525.64	57 974.49
25	[5, 11, 75, 87, 60, 10, 80]	2	25	26	835.68	21 727.68
26	[5, 45, 6, 44, 79, 22, 59, 61, 10, 80]	6	45	34	1 681.09	57 157.30
27	[5, 84, 71, 48, 37, 76, 54, 30]	9	60	40	1 678.29	67 131.71
28	[5, 8, 14, 70, 83, 30, 33, 93, 47, 10, 80]	4	35	30	2 727.88	81 836.55
29	[5, 51, 48, 68, 97, 58, 80, 10, 55, 50, 30]	3	30	28	1 834.44	51 364.32
30	[5, 88, 85, 62, 19, 53, 57, 30]	6	45	34	1 654.95	56 268.63
31	[5, 38, 77, 42, 32, 41, 10, 80]	0	15	22	1 129.67	24 852.92
32	[5, 45, 89, 15, 14, 20, 4, 59, 10, 80]	6	45	34	2 106.69	71 627.71
33	[5, 11, 9, 33, 21, 25, 30, 1, 95, 10, 80]	2	25	26	2 252.91	58 575.75
34	[5, 90, 23, 98, 62, 35, 92, 80, 78, 10]	3	30	28	2 092.20	58 581.62
35	[5, 18, 68, 96, 28, 12, 66, 10, 60, 30]	3	30	28	1 854.29	51 920.34
36	[5, 2, 24, 63, 40, 86, 70, 15, 30, 46, 16, 18, 80, 84, 10]	1	20	24	3 209.23	77 021.58
37	[5, 81, 3, 91, 88, 36, 37, 92, 68, 80, 10, 27, 30]	0	15	22	2 133.86	46 944.94
38	[5, 55, 16, 74, 67, 72, 52, 34, 35, 48, 80]	2	25	26	1 702.51	44 265.48
39	[5, 42, 13, 82, 29, 65, 87, 47, 73, 93, 21, 25, 50, 30, 83, 86, 72, 67, 10]	1	20	24	3 964.86	95 156.72
40	[5, 76, 98, 23, 90, 54, 31, 34, 77, 79, 15, 6, 30]	0	15	22	2 108.29	46 382.47
41	[5, 17, 0, 60, 10]	0	15	22	868.94	19 116.88
42	[5, 11, 49, 56, 20, 30]	0	15	22	970.69	21 355.39
43	[5, 39, 89, 7, 21, 10, 80]	0	15	22	1 613.01	35 486.26
44	[5, 26, 97, 58, 18, 30]	0	15	22	1 673.22	36 810.94
45	[5, 40, 80]	0	15	22	1 566.30	34 458.73

表4 比赛测试数据的描述性统计结果

项目	X_1	X_2	School1	School2	School3
Avg	500.82	531.98	7.11	6.53	6.76
Min	18	10	0	0	0
Max	996	985	14	14	14
Std	298.08	271.69	4.29	4.19	4.24

表5 17组随机案例测试数据结果

Instance	PD模型		Instance	PD模型	
	PSO	SMO		PSO	SMO
P20-1	259 827.35	248 085.53	P80-1	671 064.35	572 540.90
P20-2	269 123.02	254 651.03	P80-2	1 362 515.64	1 124 291.17
P20-3	431 727.37	357 973.20	P80-3	2 133 427.98	1 855 532.31
P40-1	432 086.30	399 598.02	P80-4	2 935 273.25	2 955 471.61
P40-2	663 122.82	540 909.47	P100-1	892 836.31	742 308.15
P40-3	1 129 973.70	852 870.03	P100-2	1 865 986.87	1 693 335.60
P60-1	514 923.99	414 751.53	P100-3	2 941 018.24	2 952 447.867
P60-2	1 032 282.80	946 283.90	P100-4	4 002 706.72	4 341 190.057
P60-3	1 668 767.7	1 342 909.34			

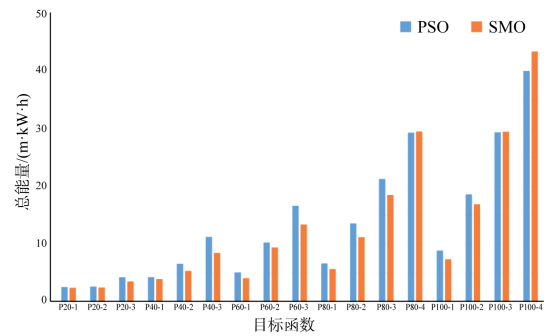


图9 卡时 300 s 不同算法求解不同随机案例的结果对比

虑求解少目的地/大规模问题。其中,在问题模型上,对PD模型进行简化,减少决策变量和提升求解速度;在求解算法上,探索启发式-精确两阶段算法此问题进行求解,从而提高算法的精确度。

参考文献:

- [1] 王科峰,叶春明,李永林.同时取送货车辆路径问题算法研究综述[J].计算机应用研究,2013,30(2):334-340.
- [2] 徐宁,胡义秋,姚康.同时取送货配送路径优化的两阶段混合策略搜索算法[J].物流科技,2023,46(1):1-6.
- [3] 裴颂文,沈天马,宁钟,等.KMG:考虑逆向物流的无人机路径规划策略研究[J].系统工程理论与实践,2019,39(12):3111-3119.
- [4] 吴腾宇,陈嘉俊,寒洁,等.O2O模式下的配送车辆实时取送货路径选择问题[J].系统工程理论与实践,2018,38(11):2885-2891.
- [5] 陈雪,胡蓉,王辉,等.学习型蚁群算法求解一类复杂两级车辆路径问题[J].系统仿真学报.
- [6] 包胜男,靳文舟.基于改进遗传算法多车辆同城配送问题[J].广西大学学报(自然科学版),2022,47(3):813-820.
- [7] 李博威,户佐安,贾叶子,等.带软时间窗的同时取送货车辆路径问题研究[J].工业工程,2020,23(5):75-81.

- [7] 关丽霞. 带软时间窗和同时取送货的车辆路径问题研究[D]. 长沙:中南大学,2010.
- [8] 李志滔,方雄发,吴雪菲,等. 基于二次清运的回收车辆路径研究[J]. 机电工程技术,2021,50(4):91-93.
- [9] BELGIN O, KARAOGLAN I, ALTIPARMAK F. Two-echelon vehicle routing problem with simultaneous pickup and delivery: mathematical model and heuristic approach[J]. Computers & Industrial Engineering, 2017,115(1): 1-16.
- [10] MAJIDI S, HOSSEINI-MOTLAGH S M, IGNATIUS J. Adaptive large neighborhood search heuristic for pollution-routing problem with simultaneous pickup and delivery[J]. Soft Computing, 2018,22(9):2851-2865.
- [11] CAROLINA L, GUILLERMO G, ENRIQUE C, et al. An improved particle swarm optimization algorithm for the VRP with simultaneous pickup and delivery and time windows[J]. IEEE Latin America Transactions, 2018,16(6): 1732-1740.
- [12] CHENTLI H, OUAFI R, CHERIF-KHETTAF W R. A selective adaptive large neighborhood search heuristic for the profitable tour problem with simultaneous pickup and delivery services[J]. RAIRO-Operations Research, 2018,52(4/5): 1295-1328.
- [13] QIU Y, NI M, WANG L, et al. Production routing problems with reverse logistics and remanufacturing[J]. Transportation Research Part E Logistics & Transportation Review, 2018,111:87-100.
- [14] ZHAO Y, LENG L, ZHANG C. A novel framework of hyper-heuristic approach and its application in location-routing problem with simultaneous pickup and delivery[J]. Operational Research, 2021(9):8-14.
- [8] 兰凤崇,郑文杰,李志杰,等. 车用动力电池的挤压载荷变形响应及内部短路失效分析[J]. 华南理工大学学报(自然科学版), 2018,46(6):65-72.
- [9] KUKREJA J, NGUYEN T, SIEGMUND T, et al. Crash analysis of a conceptual electric vehicle with a damage tolerant battery pack [J]. Extreme Mechanics Letters, 2016:371-378.
- [10] 董晴雯,王丽娟,陈宗渝,等. 电池包侧面碰撞仿真响应特性和安全性分析[J]. 南昌大学学报(工科版), 2020,42(4):386-391.
- [11] HE Y, DAVIM J P, MA Y E. 2D macro-mechanical FE simulations for machining unidirectional FRP composite: the influence of damage models[J]. Science & Engineering of Composite Materials, 2015,23.
- [12] AMODEO C M, ALI M Y, PAN J. Computational models for simulations of lithium-ion battery modules under quasi-static and dynamic constrained compression tests[J]. International Journal of Crashworthiness, 2017, 22(1/2):1-14.
- [13] 李志杰,陈吉清,兰凤崇,等. 机械外力下动力电池包的系统安全性分析与评价[J]. 机械工程学报, 2019,55(12):137-148.
- [14] SAHRAEI E, MEIER J, WIERZBICKI T. Characterizing and modeling mechanical properties and onset of short circuit for three types of lithium-ion pouch cells[J]. Journal of Power Sources, 2014,247:503-516.
- [15] WIERZBICKI T, SAHRAEI E. Homogenized mechanical properties for the jellyroll of cylindrical Lithium-ion cells[J]. Journal of Power Sources, 2013,241:467-476.
- [16] BELHAJ L A, AUBINEAU A, CANNIEUX A, et al. Simulation of Electric Vehicle Battery Behaviour for Frequency Regulation Use: Profitability Versus Mobility Constraints and Grid Needs [C]//Proceedings of First International Conference on Green Communications, Computing and Technologies, 2016.
- [17] KUCHENBUCH K, CRULL S, WEBER J, et al. Battery arrangement for use in vehicle, particularly electric vehicle, has traction battery, which is arranged below vehicle floor and is protected against side crash with lateral deformation elements: DE102011120960A1[P]. 2013.
- [18] XU J, WANG L B, LIU B H, et al. Review for mechanical integrity of lithium-ion battery[J]. Journal of Automotive Safety and Energy, 2017.
- [19] ZHANG X, SAHRAEI E, WANG K. Li-ion Battery Separators, Mechanical Integrity and Failure Mechanisms Leading to Soft and Hard Internal Shorts[J]. Scientific Reports, 2016,6:32578.
- 第一作者简介: 刘 婷 (1998—), 女, 硕士, 研究领域为车辆路径优化。
- ※通讯作者简介: 魏丽军 (1982—), 男, 博士, 教授, 研究领域为车辆路径优化。
- 作者简介:
闫鹏飞 (1991—), 男, 硕士, 工程师, 研究领域为基于虚拟测评的电池包结构安全分析, 已发表论文6篇。
马天翼 (1990—), 男, 博士, 高级工程师, 研究领域为电池性能安全测试分析评价, 已发表论文30篇。