

考虑异质物资合车运输的灾后救援选址 - 路径 - 配给优化

郭鹏辉¹, 朱建军¹, 王翥华²

(1. 南京航空航天大学 经济与管理学院, 南京 211106; 2. 金陵科技学院 商学院, 南京 211169)

摘 要 研究灾后应急救援中的双层选址 - 路径 - 配给问题. 针对灾害发生之后第一时间内各类型救援物资供给受限的情况, 以救援及时性、综合满意度和物资供给公平性为优化目标, 建立多工厂节点、多品种物资的考虑异质物资合车运输的多目标双层选址 - 路径 - 配给优化模型. 采用融合差分进化和约束优化的方法, 将多目标优化问题分解为三个单目标子迭代过程和一个多目标迭代过程, 同时优化车辆行驶路线和需求节点物资分配方案. 实验表明, 采用合车运输的策略能够有效减少派出车辆的数量和车辆行驶时间.

关键词 应急救援; 双层选址 - 路径 - 配给问题; 多目标优化; 协同进化算法

Location-routing-allocation problem with consolidated shipping of heterogeneous relief supplies in post-disaster rescue

GUO Penghui¹, ZHU Jianjun¹, WANG Hehua²

(1. College of Economics and Management, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China;

2. School of Business, Jinling Institute of Technology, Nanjing 211169, China)

Abstract The paper studies on the two-echelon location-routing-allocation problem arising in post-disaster rescue. In view of the limited supply of various types of relief supplies in the first time after a disaster, a multi-objective two-echelon location-routing-allocation optimization model considering multi-depot and heterogeneous materials is proposed. The arriving time, satisfaction degree of demand nodes and the fairness of materials supplies are the optimization goals. The multi-objective optimization problem is solved by three single-objective sub-iterative processes and a multi-objective iterative process based on the method using the differential evolution strategy and the constrained optimization strategy. The vehicle routes and distribution of materials are optimized at the same time. Experiments show that transport different kinds of materials using the same vehicle can effectively reduce the number of vehicles used and the driving time of vehicles.

Keywords emergency rescue; two-echelon location-routing-allocation problem; multi-objective optimization; coevolutionary algorithm

1 引言

根据国家减灾委员会发布的数据, 2017 年各类自然灾害共造成全国 1.4 亿人次受灾, 直接经济损失 3018.7 亿元. 尽管面对大规模自然灾害往往举全国之力进行救援, 但灾害发生第一时间内调配的各种物资仍无法完

收稿日期: 2018-01-16

作者简介: 郭鹏辉 (1994-), 男, 汉, 河南济源人, 博士研究生, 研究方向: 应急管理, 优化算法等, E-mail: guopenghui@nuaa.edu.cn; 朱建军 (1976-), 男, 汉, 江苏丹阳人, 教授, 博士生导师, 研究方向: 决策理论与方法, 复杂系统建模等, E-mail: zhujianjun@nuaa.edu.cn.

基金项目: 国家自然科学基金 (71502073); 南京航空航天大学研究生创新基地 (实验室) 开放基金 (KFJJ20170908)

Foundation item: National Natural Science Foundation of China (71502073); Foundation of Graduate Innovation Center in NUAA (KFJJ20170908)

中文引用格式: 郭鹏辉, 朱建军, 王翥华. 考虑异质物资合车运输的灾后救援选址 - 路径 - 配给优化 [J]. 系统工程理论与实践, 2019, 39(9): 2345-2360.

英文引用格式: Guo P H, Zhu J J, Wang H H. Location-routing-allocation problem with consolidated shipping of heterogeneous relief supplies in post-disaster rescue[J]. Systems Engineering — Theory & Practice, 2019, 39(9): 2345-2360.

全满足灾区的全部需求,关于灾区各类物资紧缺的报道仍常见诸报端。救援物资在灾害发生后成为一种稀缺资源,导致救援物资稀缺性产生的主要原因有以下几方面,一方面灾害发生的时间和规模难以预测,而维持大量的应急救援物资储备是不经济也是不现实的,因此储备物资往往难以满足灾区的全部需求;另一方面,尽管灾后可以通过各种社会渠道调集物资,但灾害的突发性和调集物资的实效性是难以调和的冲突,灾害突发导致大量且集中的食品、药品、帐篷等物资需求,各种渠道现有的储备可能难以完全覆盖灾区需求。大规模自然灾害发生之后,往往需要在第一时间内调集包括救援设备、帐篷、被褥、食物、药品等在内的多种类型的大量救援物资,灾区对物资的需求具有异质性的特征。根据对灾后救援现实需求的分析,如何在灾难发生后将有限的异质物资快速合理地向灾区配送是一个值得研究的问题。

由于灾害往往导致原有道路受损,无法通行大型车辆,因此往往需要在受灾区域外围通过大型运输车辆将所需物资从工厂节点运往受灾区域附近的中转节点,再在受灾区域内部通过小型车辆或直升机运输至需求节点,中转节点的选址及车辆路径的设计直接决定救援的效果。文献研究来看,选址-路径问题(location-routing problem, LRP)整合了经典的设施选址问题(facility location problem)和车辆路径问题(vehicle routing problem, VRP),“设施选址”属于长期决策而“车辆路径设计”属于短期决策,但在灾后应急救援过程中若割裂两者可能造成所选方案与全局最优相差甚远^[1]。LRP的研究至今约有几十年的历史^[2],已被广泛应用于城市物流系统设计、应急救援物资调度等领域。双层或多层LRP(multi-echelon location routing problem)是对经典LRP的扩展,多层LRP假设顾客并非直接被中心仓库节点服务,而是将货物通过一个包含 $N+1$ 类物流设施的 N 层物流网络逐层进行运输,直至货物到达顾客^[3]。Nagy等^[4]在2007年的综述文章中仅提到了3个关于多层LRP的研究,近10年来多层LRP的研究增多并被认为是LRP众多变种中最重要的一个,且多数研究关注双层的LRP(two-echelon location routing problem, LRP-2E)^[3]。已有的研究多以成本作为优化目标,且较少关注多品种商品的情况。Rath等^[5]研究了供给能力受限条件下的灾害救援LRP,第一阶段从货物供应方到达中转节点使用大型车辆一次性运输,第二阶段从中转节点到需求节点使用小型车辆运输,但在上述第一阶段的运输过程中并未考虑车辆的行驶路线,仅关注从各个工厂运往各个中转站的物资数量,且仅考虑了存在单品种物资的情况。王绍仁等^[6]研究了地震灾害应急响应阶段的选址-路径问题,以应急物流系统耗费时间最少为优化目标,研究如何确定物资集散点和应急物资运输路径以满足需求点的应急资源需求。詹沙磊等^[7]将应急决策过程划分为车辆选址决策、车辆路径选择和物资配送决策,研究了多出救点、多受灾点、多物资、多车型的选址-路径问题。王海军等^[8]以平均车辆运输时间最小化和系统总成本最小化为目标,建立多车型的双目标开放式选址-路径问题混合整数规划模型,采用基于非支配排序的遗传算法求解。王旭坪等^[9]研究了运力受限情况下的救援车辆路径选择和应急物资分配模型,将应急救援网络分为中央储备库、大型枢纽站、物资集散地和救援点四级,优化车辆路径、物资分配和运力配置,设计分层求解算法进行求解。Lin等^[10]将物流网络划分为工厂、集散中心、大客户、小客户四个层级,优化运输网络第一层车辆在工厂、集散中心和大客户之间的路线,以及第二层的车辆在集散中心和小客户之间的运输路线,建立了以固定成本和可变成本最小为优化目标的混合整数规划模型。Zhou等^[11]研究了“最后一公里”配送中的多仓库节点双层选址-路径问题,考虑了具有多个一级仓库节点的情况,但未考虑仓库节点提供异质物资,仅研究了不同仓库节点提供的物资具有可替代性的情况。Boccia等^[12]研究了多商品的LRP,建立基于“流拦截(flow intercepting)”的选址-路径模型,采用分支-切割法(branch-and-cut algorithm)进行求解。黄凯明等^[13]研究了多层级的设施选址-路径规划问题,以选址费用、车辆配送固定费用和运输费用之和最小为优化目标,采用量子进化算法与遗传算法协同的集成求解方案对问题进行求解。

现有的LRP-2E研究主要集中于常规条件下单品种商品的问题,多以成本作为优化目标,而灾后应急救援条件下的LRP-2E具有需求紧急性、资源稀缺性、商品异质性等特点,相关研究较少。本文研究自然灾害发生后资源受限条件下的多品种物资调度策略,将应急救援物资配送网络划分为两层,上层由工厂节点和中转节点构成,下层由中转节点和需求节点构成,以“救援及时性”、“综合满意度”和“公平性”作为优化目标,采用异质物资合车运输的策略发挥车辆运力并提高运输效率,提出异质物资合车运输的双层选址-路径-配给问题(two-echelon location-routing-allocation problem, LRAP-2E)模型,并研究基于救援目标要求下的应

急救援配送网络优化求解算法. 本文的主要创新和贡献为, 提出资源稀缺条件下考虑异质物资合车运输的多目标 LRAP-2E 模型, 并针对这一问题提出一种基于目标函数分解的进化多目标优化算法.

2 问题定义与描述

灾后救援物资的配送过程为首先使用大型车辆将各种不同类型的物资运至灾区外围的中转节点, 由于灾区道路受损, 在灾区内部使用运量较小但更加灵活的小型车辆进行运输, 从各中转节点运输各种救援物资至需求节点. 假设第一阶段工厂节点 - 中转节点的运输过程, 各中转节点仅能被运输某类型物资的车辆访问一次, 第二阶段中转节点 - 需求节点的运输过程, 考虑不同类型物资的合车运输, 且各需求节点仅能被运输物资的车辆访问一次. 根据 Boccia 等^[14] 和 Laporte 等^[15] 对 LRP-2E 分类的表示方法, 本文研究问题可以表示为 $3/T/\bar{T}$, 即将运输网络分为三种类型的节点, 存在两个阶段的车辆路径问题, 且两个阶段均考虑车辆的具体行驶路径, 仅考虑中转节点的选址问题.

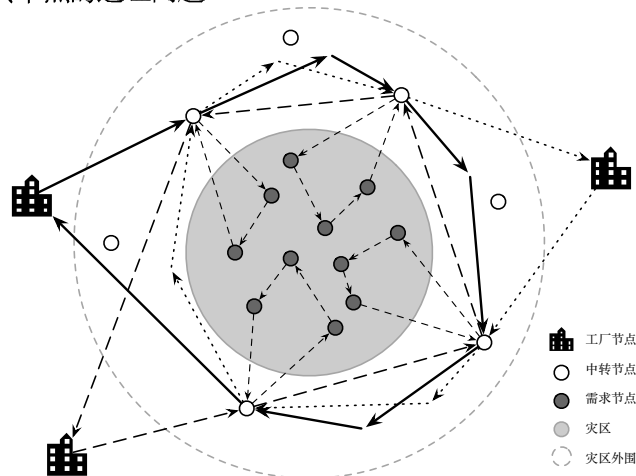


图1 物流网络结构

将应急救援物资配送物流网络划分为如图1所示的三部分, 第一部分为若干生产异质商品的工厂, 第二部分为若干可供使用的中转节点, 第三部分为若干需求节点. 救援物资从工厂节点出发, 经过中转节点, 最终到达需求点. 由于本文研究确定性的选址 - 路径 - 配给问题, 假设开始运输物资之前工厂节点的数量、位置和各节点可提供物资数量等信息已经给定. 灾害发生后首先从受灾区域附近的应急储备仓库、工厂、血站等地点安排准备救援物资, 本文将这些节点统称为工厂节点. 然后通过大型运输车辆将各种物资从距离受灾区域较远的工厂节点运往受灾区域外围的中转节点, 最后将各类物资从中转节点合车运往需求节点. 本文假设各工厂节点仅供应单一品种的物质, 且工厂节点至中转节点的运输过程不考虑物资的合车运输, 这一考虑基于以下原因. 中转节点至需求节点运输过程中各类物资数量相对于车辆容量较小, 因此合车运输能够充分发挥车辆的运力, 并一次性满足某个需求节点的多种物资需求, 从而提高运输效率; 而工厂节点至中转节点的运输过程货物运输量大, 往往需要使用大型车辆运输, 现实生活中货物大批量运输时采用合车运输方式带来的效率提升是十分有限的, 并且会带来额外的调度成本, 同时增大问题的求解难度. 本文提出的双层选址 - 路径 - 配给优化模型存在几个有待解决的问题: 1) 如何选择合适的中转节点; 2) 如何分配各需求节点接收物资的数量; 3) 如何设计从工厂节点至中转节点以及从中转节点至需求节点的车辆路径. 事实上, 上述问题可以视为两层的车辆路径问题, 存在两个阶段的决策问题, 第一阶段确定开设哪些中转节点以及从各个工厂通过何种路线运往各个中转节点多少物资, 第二阶段确定从各个中转节点通过何种路线运往各个需求节点多少物资. 和 VRP 相比的困难之处在于中转节点的信息不完备性, 即无法事先知道哪些中转节点被选择, 以及各个被选中的中转节点中转的各类型物资数量.

由于不同种物资合车运输能够充分发挥车辆的运输能力, 使得运输同样数量物资所需的车辆数量更少, 或同样数量的车辆能够运输更多的救援物资, 因此本文考虑异质物资合车运输的情况. 不失一般性, 以图2所示的单工厂节点的多商品 VRP 为例, 假设工厂节点的供给量无限, 车辆容量为 100, 需求节点两类物资的需求量在大括号中给出, 对比合车运输和分车运输的差异.

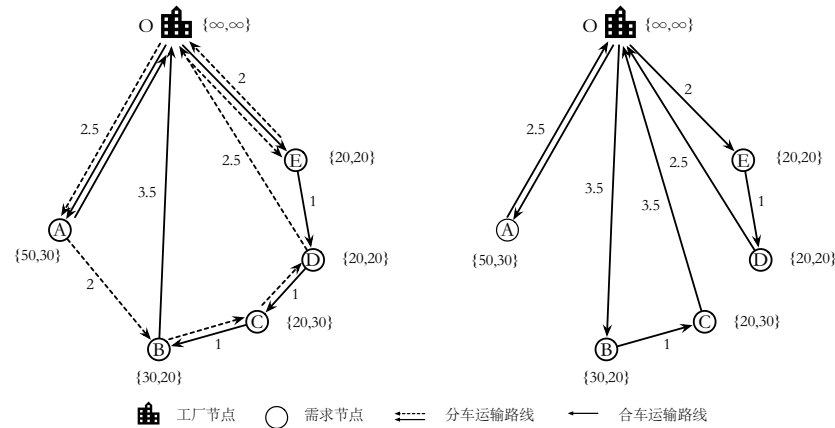


图 2 分车运输和合车运输对比

分车运输时, 左图中实线为运输第一类物资的车辆路径, 分别派出 2 辆载货量为 50 和 90 的车辆, 其中 1 号车辆的路线为 O-A-O, 里程为 5, 2 号车辆的路线为 O-E-D-C-B-O, 里程为 8.5; 左图中虚线为运输第二类物资的车辆路径, 分别派出 2 辆载货量为 100 和 20 的车辆, 3 号车辆的路线为 O-A-B-C-D-O, 里程为 9, 4 号车辆的路线为 O-E-O, 里程为 4. 综上, 合车运输时车辆数为 4, 总行驶里程为 26.5, 载货率 (载货率 = 总运货量 / (车辆容量 × 车辆数量)) 为 65%. 合车运输时, 分别派出 3 辆车, 其中 1 号车辆装载第一类物资 50, 第二类物资 30, 路线为 O-A-O, 里程为 5; 2 号车辆装载第一类物资 50, 第二类物资 50, 路线为 O-B-C-O, 里程为 8; 3 号车辆装载第一类物资 40, 第二类物资 40, 路线为 O-E-D-O, 里程为 5.5. 综上, 合车运输时车辆数为 3, 总行驶里程为 18.5, 载货率为 87%. 因此, 合车运输相比分车运输, 在同等条件下可以减少所需车辆数量和车辆行驶里程, 并充分发挥车辆运力. 异质物资分车运输的问题可以视为若干个同质物资问题的叠加, 且各子问题间不存在交互关系. 若考虑异质物资合车运输, 则存在车辆容量有限的条件下如何分配不同物资的装载量的问题. 综上, 本文提出采用异质物资合车运输策略的 LRAP-2E 优化模型.

本文模型和算法与已有研究的区别主要有以下几个方面: 考虑有多种类型物资合车运输的情况: 自然灾害发生后受灾区域的物资需求往往包含多种类型, 例如救援设备、帐篷、被褥、衣物、食品、药品、血液等, 由于存在体积差异的异质物资合车运输更能充分发挥救援车辆的运输能力, 本文考虑中转节点 - 需求节点的运输过程中不同品种的物资经由同一车辆运输的情况. 考虑供给能力受限的情况: 由于在灾害发生的第一时间内总的供给能力有限, 无法满足灾区的全部需求, 相比于常规条件下选址 - 路径问题考虑的资金成本, 更需要考虑如何快速满足各受灾地点的需求且避免某些群体被不公平对待, 即本文考虑的救援及时性、综合满意度率和公平性三个优化目标. 基于目标函数分解的协同进化多目标优化算法: 当模型中优化目标数量较多时, 传统的多目标优化方法会面临复杂度较高或非支配解在种群中占比较大导致算法效率较低的问题, 使得算法无法有效地得到 Pareto 前沿, 对此, 本文采用基于目标分解的协同进化算法对提出的多目标双层选址 - 路径 - 配给问题进行求解.

3 模型设置与分析

本文相关符号描述如下, O 、 S 、 D 、 P 分别表示工厂节点、中转节点、需求节点和物资类型的集合, $T_{ij}, \forall i, j \in O \cup S \cup D$ 表示从 i 节点到 j 节点所需时间; U 和 V 表示从工厂节点至中转节点的车辆集合以及从中转节点至需求节点的车辆集合, 两种不同类型的车辆容量分别为 C_u 和 C_v ; 由于不考虑物资在同一层节点内部的中转, 即从工厂节点 i 运到工厂节点 j 或从中转节点 i 运到中转节点 j , 因此设置节点通路集合 $A_1 = \{(i, j) | i \in O, j \in S \cup i, j \in S \cup i \in S, j \in O\}$ 和 $A_2 = \{(i, j) | i \in S, j \in D \cup i, j \in D \cup i \in D, j \in S\}$; 路径变量 $x_{ij}^u(p) \in \{0, 1\}, \forall (i, j) \in A_1, u \in U, p \in P$ 和 $x_{ij}^v(p) \in \{0, 1\}, \forall (i, j) \in A_2, v \in V, p \in P$ 表示运输 p 物资的 u 或 v 车辆是否在通路 (i, j) 上经过, 流量变量 $f_{ij}^u(p) \in \mathcal{R}^+, \forall (i, j) \in A_1, u \in U, p \in P$ 和 $f_{ij}^v(p) \in \mathcal{R}^+, \forall (i, j) \in A_2, v \in V, p \in P$ 表示 u 或 v 车辆运输的 p 物资在通路 (i, j) 上的流量, 选址变量 $z_j \in \{0, 1\}, \forall j \in S$ 表示节点 j 是否被选作中转节点, $z_j = 1$ 当且仅当 $\sum_{u \in U} \sum_{(i, j) \in A_1} \sum_{p \in P} x_{ij}^u(p) \geq 1, \forall j \in S$; $M_o(p)$ 表示工厂节点 $o \in O$ 能够供应的物资 $p \in P$ 的最大数量, $M_j(p)$ 表示需求节点 $j \in D$ 对物资 $p \in P$

的需求.

目标函数 F_1 定义为车辆从工厂节点将货物运往中转节点与车辆将货物从中转节点运往需求节点所花费的所有时间, 这个目标函数在模型求解时等价于所有车辆路径的总长度; 将每个需求节点的满意度定义为 $\sum_{p \in P} m_j(p)/M_j(p), \forall j \in D$, 其中 $m_j(p) = \sum_{v \in V} (\sum_{i \in S \cup D} f_{ij}^v(p) - \sum_{h \in S \cup D} f_{jh}^v(p))$ 表示需求节点 j 接收的 p 物资数量, 则所有需求节点的综合满意度为各需求节点的满意度之和, 但是当需求节点数量或物资数量不同时上述指标在不同的问题上缺乏可比性, 因此, 将所有需求节点的综合满意度 F_2 定义为 (2) 式; 公平性目标函数 F_3 定义为各需求节点满意度的方差. 模型中的约束主要有车辆数量和容量约束、工厂节点供给能力约束. 设置虚拟变量 $m_i^*(p)$ 表示中转节点 i 中转的 p 物资数量, 则工厂节点至中转节点与中转节点至需求节点的建模过程类似.

$$\min \quad F_1 = \sum_{u \in U} \sum_{(i,j) \in A_1} \sum_{p \in P} T_{ij} x_{ij}^u(p) + \sum_{v \in V} \sum_{(i,j) \in A_2} \sum_{p \in P} T_{ij} x_{ij}^v(p) \quad (1)$$

$$\max \quad F_2 = \left(\sum_{j \in D} \sum_{p \in P} \frac{m_j(p)}{M_j(p)} \right) / (|D| \cdot |P|) \quad (2)$$

$$\min \quad F_3 = \sum_{j \in D} \left(\sum_{p \in P} \frac{m_j(p)}{M_j(p)} - \left(\sum_{j \in D} \sum_{p \in P} \frac{m_j(p)}{M_j(p)} \right) / |D| \right)^2 / |D| \quad (3)$$

$$\text{s.t.} \quad f_{oj}^u(p) \leq C_u, \forall o \in O, p \in P, u \in U, j \in S, \quad (4)$$

$$\sum_{u \in U} \sum_{j \in S} f_{oj}^u(p) \leq M_o(p), \forall o \in O, p \in P, \quad (5)$$

$$\sum_{u \in U} \sum_{j \in S} \sum_{o \in O} \sum_{p \in P} x_{oj}^u(p) \leq |U|, \quad (6)$$

$$\sum_{u \in U} \sum_{h \in S \cup \{o\}} x_{ih}^u(p) = \sum_{u \in U} \sum_{h \in S \cup \{o\}} x_{hi}^u(p) \leq 1, \forall i \in S, o \in O, p \in P, \quad (7)$$

$$\sum_{u \in U} \sum_{h \in S \cup \{o\}} f_{hi}^u(p) - \sum_{u \in U} \sum_{h \in S \cup \{o\}} f_{ih}^u(p) = m_i^*(p), \forall i \in S, o \in O, p \in P, \quad (8)$$

$$m_j^*(p) x_{ij}^u(p) \leq f_{ij}^u(p) \leq (C_u - m_i^*(p)) x_{ij}^u(p), \forall (i,j) \in A_1, p \in P, \quad (9)$$

$$\sum_{p \in P} f_{ij}^v(p) \leq C_v, \forall i \in S, j \in D, v \in V, \quad (10)$$

$$\sum_{v \in V} \sum_{j \in D} f_{ij}^v(p) = m_i^*(p), \forall i \in S, p \in P, \quad (11)$$

$$\sum_{i \in S} \sum_{j \in D} \sum_{v \in V} \sum_{p \in P} x_{ij}^v(p) \leq |V|, \quad (12)$$

$$\sum_{p \in P} \sum_{v \in V} \sum_{h \in S \cup D} x_{ih}^v(p) = \sum_{p \in P} \sum_{v \in V} \sum_{h \in S \cup D} x_{hi}^v(p) = 1, \forall i \in D, \quad (13)$$

$$\sum_{v \in V} \sum_{h \in S \cup D} f_{hi}^v(p) - \sum_{u \in U} \sum_{h \in S \cup D} f_{ih}^v(p) \leq M_i(p), \forall i \in D, p \in P, \quad (14)$$

$$\sum_{p \in P} f_{ij}^v(p) \leq \left(C_v - \sum_{p \in P} M_i(p) \right) x_{ij}^v(p), \forall (i,j) \in A_2, p \in P, v \in V, \quad (15)$$

$$f_{ij}^v(p) \geq M_j(p) x_{ij}^v(p), \forall (i,j) \in A_2, p \in P, v \in V, \quad (16)$$

$$\sum_{p \in P} x_{ij}^v(p) \in \{0, |P|\}, \forall (i,j) \in A_2, v \in V, \quad (17)$$

$$x_{ij}^u(p), x_{ij}^v(p) \in \{0, 1\}, \forall (i,j) \in A_1 \cup A_2, u \in U, v \in V, p \in P, \quad (18)$$

$$f_{ij}^u(p), f_{ij}^v(p) \in \mathcal{R}^+, \forall (i, j) \in A_1 \cup A_2, u \in U, v \in V, p \in P, \quad (19)$$

$$z_j \in \{0, 1\}, \forall j \in S. \quad (20)$$

(1) 式表示车辆将所需物资从工厂节点运往中转节点以及从中转节点运往需求节点所需的时间之和; (2) 式表示所有需求节点的综合满意度; (3) 式表示用方差衡量的各需求节点需求被满足程度的差异, 其中 $\sum_{p \in P} (m_j(p)/M_j(p))$ 为各需求节点的满意度, (3) 式为各需求节点满意度的方差; (4)、(5)、(6) 式分别为单个车辆的容量约束、工厂节点供给量约束、车辆总数约束, (7) 式表示中转节点被运送某类型物资的所有车辆至多访问一次, (8) 式设置虚拟变量 $m_i^*(p)$ 表示运往 i 节点的 p 物资数量, (9) 式的含义为, 当 $x_{ij}^u(p) = 1$ 时 $m_j^*(p) \leq f_{ij}^u(p) \leq C_u - m_i^*(p)$ 成立, 式中 $m_i^*(p)$ 和 $m_j^*(p)$ 分别为节点 i 和节点 j 的 p 物资中转量, 因此不等式的含义为当车辆离开 i 节点前往 j 节点时, 车上的 p 类型物资量 $f_{ij}^u(p)$ 大于 j 节点的 p 类型物资需求量 $m_j^*(p)$, 且在 i 节点卸货前车上的物资数量 $m_i^*(p) + f_{ij}^u(p)$ 不超过车辆容量 C_u ; 当 $x_{ij}^u(p) = 0$ 时 $f_{ij}^u(p) = 0$, 表明当运输 p 物资的 u 车辆不从路径 ij 经过时, 路径 ij 上不产生物资流量, (8)、(9) 两式共同定义了物资的运输过程; (10)~(17) 式为物流网络第二层的约束, 与物流网络第一层的约束 (4)~(9) 对应, 其中 (13) 式表示每个需求节点仅被车辆访问一次, (14) 式表示各需求节点的各种类型需求均被部分满足. 与 (9) 式类似, (15) 式表示当车辆到达 i 节点时, 车上各类物资总量不超过车辆容量; (16) 式表示当车辆离开 i 节点前往 j 节点时, 车上各类物资量大于等于 j 节点需求. (17) 式表示每个需求节点的各种物资需求被同一车辆服务. (18)、(19)、(20) 表示路径变量和选址变量为 0-1 变量, 流量变量为实数变量.

应急救援中的 LRAP-2E 首先需要考虑的并非开设救援设施的成本或车辆行驶成本等经济指标, 而是“救援及时性”这一效率指标 (F_1). 救援及时性主要受车辆的行驶时间影响, 包括将物资从工厂运往中转设施的时间以及从中转设施运往终端需求的时间两部分, 若选择的中转节点距离工厂较近而距离需求节点较远则第一部分时间花费较小第二部分时间花费较大, 反之则第一部分时间花费较大第二部分时间花费较小, 因此需要选择合适的中转节点使得总时间花费较小. 此外, 由于灾害发生的第一时间内所能调集的救援资源往往十分有限, 难以马上满足所有受灾区域的所有需求, 此时便需要考虑各种物资在各个需求点的配给方案, 尽最大可能满足各需求点的需求, 即优化模型中第二个目标函数——“综合满意度” (F_2). 若仅考虑上述的 F_1 和 F_2 , 优化结果会倾向于优先满足距离中转节点近的、需求量小的需求节点, 而忽视距离中转节点远的、需求量大的需求节点. 救援物资作为一种稀缺资源在上述条件下出现了分配不公平的现象, 因此需要考虑第三个指标——“公平性” (F_3), 使得各个节点需求被满足程度的差异尽可能小. 若完全按照需求量和供给量的比例关系分配各种物资, 虽然可以实现绝对的公平, 但是满意度指标无法同时达到最优, 因为同样数量的某种物资供给给需求量小的节点对“综合满意度”的贡献比供给给需求量较大的节点时大. 而物资的分配方式又必然会对车辆路径产生影响, 进而影响目标函数 F_1 的值. 上述 LRAP-2E 模型是对已有的如 LRP、LRP-2E 等 \mathcal{NP} -hard 问题的推广, 因此也是 \mathcal{NP} -hard 问题.

4 求解算法

4.1 算法整体框架

本文模型为约束多目标优化, 各个目标函数存在冲突无法同时达到最优, 存在权衡各个目标函数值的问题. 求解多目标优化问题的方法主要可以分为线性加权法、目标规划法、 ϵ -约束法等方法^[16], 但以上方法存在需要先验知识或计算复杂等问题. 进化多目标优化 (multi-objective evolutionary algorithms, MOEA) 同时对一组解进行操作, 是求解多目标优化问题的有效算法, 但当目标函数数量较多时, 传统的基于 Pareto 排序的进化多目标优化算法会面临非支配解在所有解中所占比列过高使得算法效率低下的问题^[17]. 本文采用基于目标函数分解的协同进化算法 (cooperative CMOEA based on objective functions decomposition), 根据目标函数数量设置种群数量, 各个种群仅优化一个目标函数, 避免了各个目标函数之间的冲突. 但是, 存在最终的解倾向于取得各个目标函数的极端值而偏离 Pareto 前沿的问题. 因此, 设置额外的种群, 采用一定的信息共享策略控制各种群的进化方向, 并通过比较各种群迭代得到的最优解得到最终的 Pareto 非支配解集, 即多目标优化中的最优解集.

本文模型还是一个约束优化问题, 对于约束优化, 存在权衡可行解和不可行解进入子代种群可能性的问题. [18] 表明通过控制算法对不可行域的搜索能增强算法的性能, 因此本文的搜索空间包括可行域和不可行域. 模型存在的约束主要包括供给量约束、车辆容量约束和车辆数量约束等, 实验发现若允许各类型的不可行解进行交叉、变异和选择操作会造成种群中不可行解占多数的情况, 因此在进化算法的编码过程中确保各个解均不违反车辆容量约束. 对不满足供给量约束和车辆数量约束的不可行个体的选择步骤在 4.2.3 详细介绍.

本文算法的框架如图 3. 针对本文模型具有多个目标函数的特点, 采用目标分解的方法设立 3 个单目标种群, 每个种群优化单一目标函数并与非支配解存档进行信息交互; 针对本文模型既有离散变量又有连续变量的特点, 采用遗传算法中的基因编码对表示车辆路径结构的 0-1 变量进行编码, 采用差分进化策略对表示物资配给量的连续变量进行优化; 针对本文模型具有多个约束的特点, 采用基于贪婪策略的约束支配 (constraint-dominate)^[25] 关系选择目标函数较优的可行解进入新种群.

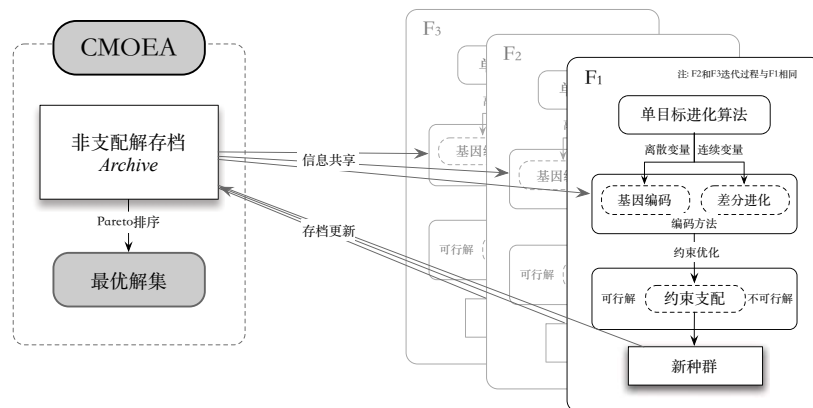


图 3 算法框架

本文算法中 3 个单目标种群输出较优解进行 Pareto 排序, 每次迭代得到的非支配解被存入非支配解存档, 并与之后单目标种群输出的较优解进行 Pareto 排序. 在 Pareto 占优的概念下无需事先设定目标函数权重, 并最终得到最优解集. 因此, 本文算法避免了现实决策环境中需要预先确定决策者偏好信息的难题. 此外, 针对决策者对不同目标函数存在偏好的情况, 当得到最优解集后可引入偏好信息以突出目标函数重要程度的差异, 并据此对非支配解进行进一步优选.

4.2 算法模块设计

4.2.1 编码和初始种群

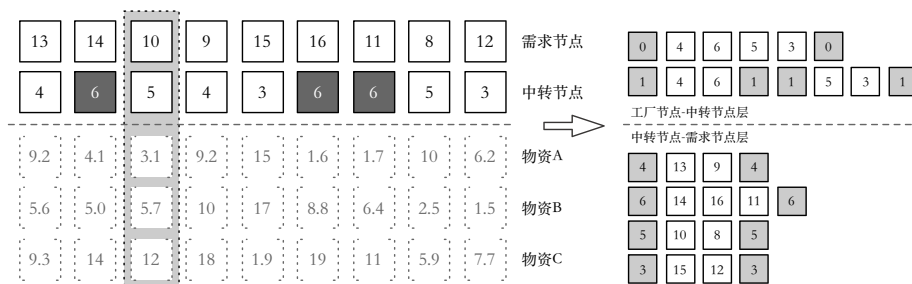


图 4 染色体编码

本文编码方式受 [11] 启发, 用 3 条染色体分别表示节点的分配关系、工厂节点-中转节点车辆路径和中转节点-需求节点车辆路径, 构成问题的一个解. 与其主要差异在于, 由于本文模型同时优化运输路线结构和运输物资的数量, 不仅存在 0-1 变量且存在实数变量, 编码方式需要做出相应的调整, 后续的交叉、变异操作与之截然不同. 如图所示的染色体第一层为需求节点编号, 第二层为中转节点编号, 第三层及其之后为经由中转节点运往某需求节点的某类型物资数量. 图 4 中长方形内的一段染色体表示, 经由 5 号中转节点运往 10 号需求节点 A 类型物资 3.1 单位容量, B 类型物资 5.7 单位容量, C 类型物资 12 单位容量.

车辆路径根据如图 4 所示的分配关系自动生成, 且保证满足车辆容量约束. 工厂节点-中转节点车辆路

径和中转节点 - 需求节点车辆路径的生成方式的核心思想为, 工厂节点按照染色体第二层中转节点出现的先后顺序向中转节点运输货物, 每个中转节点仅被生产某产品的工厂节点访问一次; 中转节点按照分配给自己的需求节点出现的先后顺序构造从中转节点向需求节点运送物资的车辆路径, 上述过程均保证运货量满足车辆容量满足约束. 以中转节点 6 为例说明中转节点 - 需求节点的路径生成方式: 首先, 验证 14 和 16 的各物资总需求量是否超出车辆的总容量限制, 若超出则生成路线 6-14-6, 若未超出则进一步验证 14、16、11 的各物资总需求量是否超出车辆的容量限制, 若超出则生成路线 6-14-16-6, 若未超出则生成路线 6-14-16-11-6. 根据染色体的第二行中转节点出现的先后顺序确定中转节点序列 [4, 6, 5, 3], 采用和中转节点 - 需求节点路径生成方式类似的方法生成从各个工厂节点到达中转节点的车辆路径, 各中转节点的物资需求根据其所服务的需求节点的物资需求量确定. 图 4 右边为根据分配关系生成车辆路径的过程, 其中工厂节点 1 派出了 2 辆车进行运输, 其他工厂节点和中转节点均派出 1 辆车进行运输. 中转节点 2 和 7 根据图 4 的染色体编码未被选为使用的中转节点. 在现实问题中可能存在某些物资必须使用专车运输, 上述编码方式同样可被用于存在必须专车运输物资的情况, 只需对车辆路径生成算法进行修改, 使得每辆车均运输不超过其容量的单品种物资, 且各需求节点被运输每类物资的车辆均访问 1 次即可.

初始种群生成方式为, 首先对需求节点随机排序生成染色体的第一行, 然后对各个需求节点随机分配中转节点生成染色体的第二行, 最后依 $x \sim \mathcal{N}(M_o(p_i)/|D|, M_o(p_i)/(3 \cdot |D|))$ 随机生成运往各个需求节点的各种类型物资数量构成染色体的其他部分, 再由上述分配关系生成车辆路径. 需求节点和中转节点的分配关系、工厂节点 - 中转节点车辆路径、中转节点 - 需求节点车辆路径共同构成一个个体. 重复上述步骤 μ 次生成初始种群.

4.2.2 交叉和变异

交叉过程应尽可能地保留个体中已有的模式, 使得较优的模式在进化过程中能够得以保留. 本文采用的交叉方式为多点交叉. 首先从一个个体上随机选择一段染色体片段, 由于染色体第一行的需求节点编号在每一个个体中均出现且仅出现一次, 因此可将其视为染色体片段的索引. 以图 5 为例, 发生交叉的点位为 11, 8, 16, 12, 两个个体的对应片段交换生成 2 个新的个体.

13	15	11	8	16	12	9	10	14	13	11	8	9	15	12	14	10	16
7	6	3	5	3	5	4	5	5	4	6	5	4	3	6	6	5	3
9.2	4.1	3.1	9.2	15	1.6	1.7	10	6.2	7.6	4.4	8.8	5.8	2.3	5.8	3.8	10	5.8
5.6	5.0	5.7	10	17	8.8	6.4	2.5	1.5	5.5	6.9	4.6	10	16	14	7.9	7.6	11
9.3	14	12	18	1.9	19	11	5.9	7.7	13	14	12	19	3.6	9.8	12	5.0	8.9

图 5 交叉操作

变异过程的困难之处在于, 简单的单点或多点变异方法虽然能够使得中转节点和需求节点的分配关系发生变化, 避免解陷入局部最优, 但是由于实数变量即染色体的最后几行的存在, 变异过程应该保证实数变量的搜索过程尽可能覆盖可行域的所有范围. 因此, 这里对染色体的前两行和最后几行采用不同的变异操作. 需求节点和中转节点的分配关系, 即染色体前两行的变异方式为对染色体的第二行进行逆序排列. 对需求节点接收物资数量, 即染色体的后几行的变异引入差分进化策略^[19]:

$$V_{i,G}^m = X_{i,G}^m + f_1 \cdot (X_{best,G}^m - X_{i,G}^m) + f_2 \cdot (X_{i1,G}^m - X_{i2,G}^m) + f_3 \cdot (A_{r,G} - X_{i,G}^m). \quad (21)$$

式 (21) 中 $X_{i,G}^m$ 为原始的变量向量 (向量维数由物资种类数量和需求节点数量确定, 如图 5 对应的 $X_{i,G}^m$ 维数为 3×9), $X_{best,G}^m$ 为 F_m 目标函数子种群中的最优解, $X_{i1,G}^m$ 和 $X_{i2,G}^m$ 为从种群中随机选择的两个个体, f_i 为变异因子, $A_{r,G}$ 为从非支配解存档中随机选择的解, 起到了调节种群进化方向的作用. 式 (21) 中, $X_{i,G}^m$ 保存了个体自身的搜索信息, 有助于保证算法的收敛性; $(X_{best,G}^m - X_{i,G}^m)$ 增强算法在目标 F_m 方向上的搜索能力; $(X_{i1,G}^m - X_{i2,G}^m)$ 加入随机扰动维持种群的多样性; $(A_{r,G} - X_{i,G}^m)$ 引入了其他种群的搜索信息, 增强算法在 Pareto 最优解集方向上的搜索能力, 是式 (21) 与经典的单目标优化差分进化算法的关键区别. 变异过程先执行需求节点接收物资数量的变异, 再执行中转节点和需求节点分配关系的变异.

4.2.3 个体评价和子代生成

对每个单目标优化子种群, 本文参考经典的差分进化子代生成策略, 与其不同之处在于由于在搜索过程考虑了不可行解, 因此个体的选择不能仅考虑目标函数值的大小, 还要考虑对约束违反程度的大小. 参考经典的差分进化算法^[20]的子代生成过程, 本文从种群中所有的 μ 个个体中随机选择 $\lambda/2$ 对个体执行以下操作:

$$u_{i,j,G} = \begin{cases} v_{i,j,G}, & \text{if rand}(0, 1) < mtt_{\text{prob}}, \\ x_{i,j,G}, & \text{otherwise.} \end{cases} \quad (22)$$

式 (22) 中 $x_{i,j,G}$ 为一对原始的变量向量, $v_{i,j,G}$ 为根据 “4.2.2 交叉和变异” 的变异算法生成的变量向量, mtt_{prob} 为变异率, $u_{i,j,G}$ 称为实验向量. 对实验向量根据 4.2.2 执行交叉操作生成两个新的个体, 通过比较新生成的个体和原始个体的目标函数值以及对约束的违反程度确定进入子代种群的个体. 为保证随算法的迭代, 种群中可行解比例不断提高, 故采用如式 (23) 所示的贪婪选择策略, 优先选择目标函数值较优的可行解进入子代.

$$X_{G+1} = \begin{cases} U_{i,G}, & \text{if } \begin{cases} \phi(X_{i,G}) > 0 \text{ and } \phi(U_{i,G}) = 0, \\ \phi(X_{i,G}) = 0 \text{ and } \phi(U_{i,G}) = 0, & F(U_{i,G}) \succ F(X_{i,G}), \\ \phi(X_{i,G}) > 0 \text{ and } \phi(U_{i,G}) > 0, & \phi(U_{i,G}) < \phi(X_{i,G}), \end{cases} \\ X_{i,G}, & \text{if } \begin{cases} \phi(U_{i,G}) > 0 \text{ and } \phi(X_{i,G}) = 0, \\ \phi(U_{i,G}) = 0 \text{ and } \phi(X_{i,G}) = 0, & F(X_{i,G}) \succ F(U_{i,G}), \\ \phi(U_{i,G}) > 0 \text{ and } \phi(X_{i,G}) > 0, & \phi(X_{i,G}) < \phi(U_{i,G}). \end{cases} \end{cases} \quad (23)$$

式 (23) 中 X_{G+1} 为进入新种群的个体, $U_{i,G}$ 为交叉操作后生成的个体, $X_{i,G}$ 为原始个体, $F(X_{i,G})$ 为原始个体的目标函数值, $\phi(X_{i,G})$ 为原始个体对约束的违反程度 ($F(U_{i,G})$ 和 $\phi(U_{i,G})$ 含义与之对应). (23) 式的含义为, 以下 3 种情况时选择新生成的子代个体进入新种群: 1) 原始个体为不可行解, 子代个体为可行解; 2) 原始个体和子代个体均为可行解, 但子代个体目标函数值较优; 3) 原始个体和子代个体均为不可行解, 但子代个体对约束的违反程度较小. 选择原始个体进入新种群的情况与之类似. $\phi(X_{i,G}) \in [0, 1]$ ($\phi(U_{i,G}) \in [0, 1]$) 的定义为:

$$\phi(X_{i,G}) = (\varphi_1(X_{i,G}) + \varphi_2(X_{i,G}) + \varphi_3(X_{i,G}))/3. \quad (24)$$

式 (24) 中 $\varphi_1(X_{i,G})$ 为标准化后的 $X_{i,G}$ 对工厂节点总供给量约束的违反值, $\varphi_2(X_{i,G})$ 为差分算法迭代过程中可能生成的 $X_{i,G}$ 供应量为负数的部分标准化后的值, $\varphi_3(X_{i,G})$ 为标准化后的 $X_{i,G}$ 对车辆数量约束的违反值. 标准化方法为 $\varphi_i(X_{i,G}) = (\varphi_i(X_{i,G}) - \varphi_i \min(pop))/(\varphi_i \max(pop) - \varphi_i \min(pop))$.

4.2.4 多目标优化流程

对每个单目标优化子种群按照 4.2.2 的交叉变异算法和 4.2.3 的个体评价和子代生成策略并行地执行单目标优化算法, 每次迭代用各个单目标优化子种群最优的前 k 个可行个体 $pop_{m.best_k}$ 更新多目标优化种群, 对多目标优化种群进行 Pareto 排序得到 Pareto 最优解集. 详细过程见算法 1.

5 实验和算法分析

5.1 实验数据和参数标定

Breunig 等^[21]对已有的双层车辆路径问题的测试数据进行了收集和整理^[22], 将已有研究所使用的基准测试数据分为 5 个数据集共 200 余个测试用例, 中转节点数量为 2~10, 需求节点数量为 20~200. 但现有数据均无法直接用于本文提出的模型和算法, 因为: 1) 与现有的大部分双层车辆路径问题研究相对应, 上述数据中工厂节点的数量均为 1, 而本文模型研究有多个工厂节点时的情况; 2) 上述数据中需求节点的需求为单品种商品的需求, 而本文研究异质商品需求下的问题, 即商品的品种数量为 2 或 2 以上; 3) 上述数据无工厂节点的供应能力信息, 假设工厂节点可以满足所有需求点的需求, 而本文研究供应能力受限下的问题, 需要明确各工厂节点的供应能力上限. 本文基于 Set4^[23] ($|O| = 1$, $|S| = [2, 3, 5]$, $|D| = 50$), 增加工厂节点数量 (令 $|O| = [2, 3]$), 随机从 50 个需求节点中选择部分节点 (令 $|D| = [30, 50]$), 并按照 $x \sim U(0.5 \cdot l_i, 0.5 \cdot u_i)$ 及 $y \sim U(0.3 \cdot l_i, 0.3 \cdot u_i)$ (其中 l_i 和 u_i 为原始数据中结点需求量的下界和上界) 随机生成各需求节点其他两种

物资的需求量, 各供给节点的供给能力按照其所对应物资总需求量的 80% 给出, 生成 4 类 ($|O|, |S|, |D| = [2, 3, 30], [2, 3, 50], [3, 5, 30], [3, 5, 50]$) 共 72 个的“供应能力受限的多工厂节点多商品双层选址 - 路径 - 配给问题”测试数据^[24].

本文算法的参数可以分为 2 类: 1) 种群容量 (μ)、子代数量 (λ)、最优子代数量 (k)、非支配解集容量 ($|Archive|$) 和变异率 (mtt_{prob}) 等遗传算法的基本参数; 2) 差分进化算法中指导连续变量进化方向的参数 f_i . 对于种群容量等遗传算法的基本参数, 虽然较大的种群容量往往能够保证解的效果较好, 但维护种群中的大量个体, 以及对个体的交叉变异操作会产生大量的计算开销, 因此需要权衡解的质量和算法效率确定第一类参数取值. f_i 参数标定问题可以视为一个确定参数为何值时目标函数最优的优化问题. 因此本文从工厂节点数为 3, 中转节点数为 5 的测试用例中随机选取 5 个, 尝试 $f_1 = \{0.3, 0.4, 0.5, 0.6, 0.7\}$, $f_2 = \{0.3, 0.4, 0.5, 0.6, 0.7\}$, $f_3 = \{0.3, 0.4, 0.5, 0.6, 0.7\}$ 中所有的组合, 确定参数为何值时测试用例取得最理想的解. 不同参数下得到的 Pareto 最优解集优劣的比较方式为, 本解集支配其他解集解的数量, 该值越大越好. 各参数最终取值为 $\mu, \lambda, k, |Archive|, mtt_{prob} = \{500, 300, 300, 400, 0.3\}$, $f_1, f_2, f_3 = \{0.4, 0.5, 0.5\}$.

算法 1 多目标选址-路径-配标进化路标

```

input:  $\mu, \lambda, k, |Archive|, mtt_{prob}, f_i$ 
output:  $Archive$ 
1: /* framework */
2: function FRAMEWORK( $\mu$ )
3:    $Archive = \emptyset; best_{k_s} = \emptyset$ 
4:   for each single objective population  $m = 1 \rightarrow 3$  do
5:      $pop_m = \emptyset$ 
6:     for  $i = 1 \rightarrow \mu$  do
7:        $ind_{rand} \leftarrow$  individual generated according to 4.2.1
8:        $pop_m \leftarrow pop_m \cup \{ind_{rand}\}$ 
9:     end for
10:     $pop_{m.best_k}, offspring_m \leftarrow$  SOE( $pop_m, pop_m$ )
11:     $best_{k_s} \leftarrow best_{k_s} \cup \{pop_{m.best_k}\}$ 
12:  end for
13:   $Archive \leftarrow$  MOE( $Archive, best_{k_s}$ )
14:  while stop criterion is not met do
15:    for each single objective population  $m = 1 \rightarrow 3$  do
16:       $pop_{m.best_k}, offspring_m \leftarrow$  SOE( $pop_m, Archive$ )
17:       $best_{k_s} \leftarrow best_{k_s} \cup \{pop_{m.best_k}\}$ 
18:    end for
19:     $Archive \leftarrow$  MOE( $Archive, best_{k_s}$ )
20:  end while
21:  return  $Archive$ 
22: end function
23:

```

算法 2 路目标配进化(SOE)

```

input:  $pop_m, |Archive|$ 
output:  $pop_{m.best_k}, pop_m$ 
1: /* single objective evolution function */
2: function SOE( $pop_m, |Archive|$ )
3:    $P \leftarrow$  randomly chose  $\lambda/2$  pairs of individuals from  $pop_m$ 
4:   for each  $ind \in P$  do
5:      $pop_m \leftarrow pop_m \setminus \{ind\}$ 
6:   end for
7:   for each pair of individuals  $(ind_1, ind_2) \in P$  do
8:     if random()  $< mtt_{prob}$  then
9:        $ind_1, ind_2 \leftarrow ind_1, ind_2$  mutation according to 4.2.2
10:    end if
11:     $ind_{1son}, ind_{2son} \leftarrow ind_1, ind_2$  crossover according to 4.2.2
12:     $ind_{1new} \leftarrow$  chose one of  $ind_{1son}$  and  $ind_1$  according to Eqs(23)
13:     $ind_{2new} \leftarrow$  chose one of  $ind_{2son}$  and  $ind_2$  according to Eqs(23)
14:     $pop_m \leftarrow pop_m \cup \{ind_{1new}, ind_{2new}\}$ 
15:  end for
16:   $pop_{m.best_k} \leftarrow$  the best  $k$  feasible solutions in  $pop_m$ 
17:  return  $pop_{m.best_k}, pop_m$ 
18: end function

```

算法 3 Pareto解配维护(MOE)

```

input:  $|Archive|, best_{k_s}$ 
output:  $Archive$ 
1: /* multi objective evolution function */
2: function MOE( $|Archive|, best_{k_s}$ )
3:    $Archive = \emptyset; P = Archive \cup best_{k_s}$ 
4:   for each individual  $ind = P_1 \rightarrow P_{|P|}$  do
5:     if  $ind$  is not dominated by any other individual  $\in P$  then
6:        $Archive \leftarrow Archive \cup \{ind\}$ 
7:     end if
8:   end for
9:   return  $Archive$ 
10: end function

```

5.2 算例分析

本部分以 5·12 四川汶川特大地震为例, 分析本文算法的运行效果. 需求节点为图 6 中所列出的 15 个受灾严重的城市, 中转节点为人为指定的 5 个位于高速公路或国道附近且避开灾区的中转站 (其中节点 D 为了验证算法的选址效果, 由于 D 既远离灾区又远离工厂节点, 因此最优解集中 D 未被选作中转节点), 工厂节点为成都市及双流机场. 灾害发生后从全国通过铁路运输或航空运输将帐篷和食品等物资运往成都市和双流机场. 帐篷通过成都市经中转节点运往受灾城市, 食品通过双流机场经中转节点运往受灾城市. 在地震发生后的第一时间所调集的物资仅能满足灾区所有需求的 80%. 各需求节点、中转节点、工厂节点的经纬度、需求量和供给量见表 1. 表中各需求节点的需求量为帐篷和食品的总量 $p_1 + p_2$, 令 $p_1 : p_2 = [5 : 5], [6 : 4], [7 : 3], [8 : 2], [9 : 1]$, 设计 5 个不同的实验, 以模拟所需的救援物资体积或质量差异悬殊的情况, 从而验证“不同物资体积或质量差异较大时合车运输能够减少所需车辆数量和车辆行驶里程并充分发挥车辆的运力”的假设.

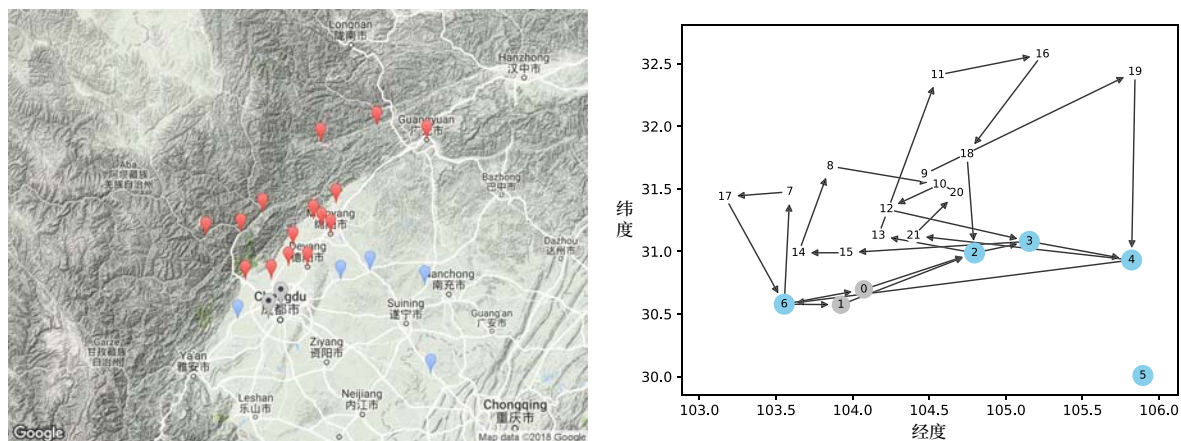
图 6 节点分布和合车运输 $p_1 : p_2 = [5 : 5]$ 的 1 个最优解

表 1 节点位置和物资需求量

$ O , S , D = 2, 5, 15 \quad U = 4 \quad C_u = 6000 \quad V = 10 \quad C_v = 2000$									
序号	节点	经度	纬度	需求 (供给)	序号	节点	经度	纬度	需求 (供给)
0	成都	104.073370	30.696811	$80\% \sum M_j(p_1)$	11	平武县	104.555583	32.409675	882
1	双流县	103.923651	30.574474	$80\% \sum M_j(p_2)$	12	绵竹市	104.220750	31.338077	164
2	A	104.795866	30.987896	\	13	什邡市	104.167501	31.126780	361
3	B	105.153533	31.080407	\	14	都江堰市	103.646912	30.988434	220
4	C	105.820597	30.931548	\	15	彭州	103.958014	30.990108	607
5	D	105.894298	30.008766	\	16	青川县	105.238842	32.575484	879
6	E	103.553531	30.578562	\	17	理县	103.166852	31.436473	560
7	汶川县	103.590386	31.476822	237	18	江油市	104.745823	31.778022	583
8	茂县	103.853522	31.681154	682	19	广元市	105.843357	32.435435	767
9	北川县	104.467970	31.617202	967	20	绵阳市	104.679004	31.467460	488
10	安县	104.567187	31.534886	921	21	德阳市	104.397894	31.126855	907

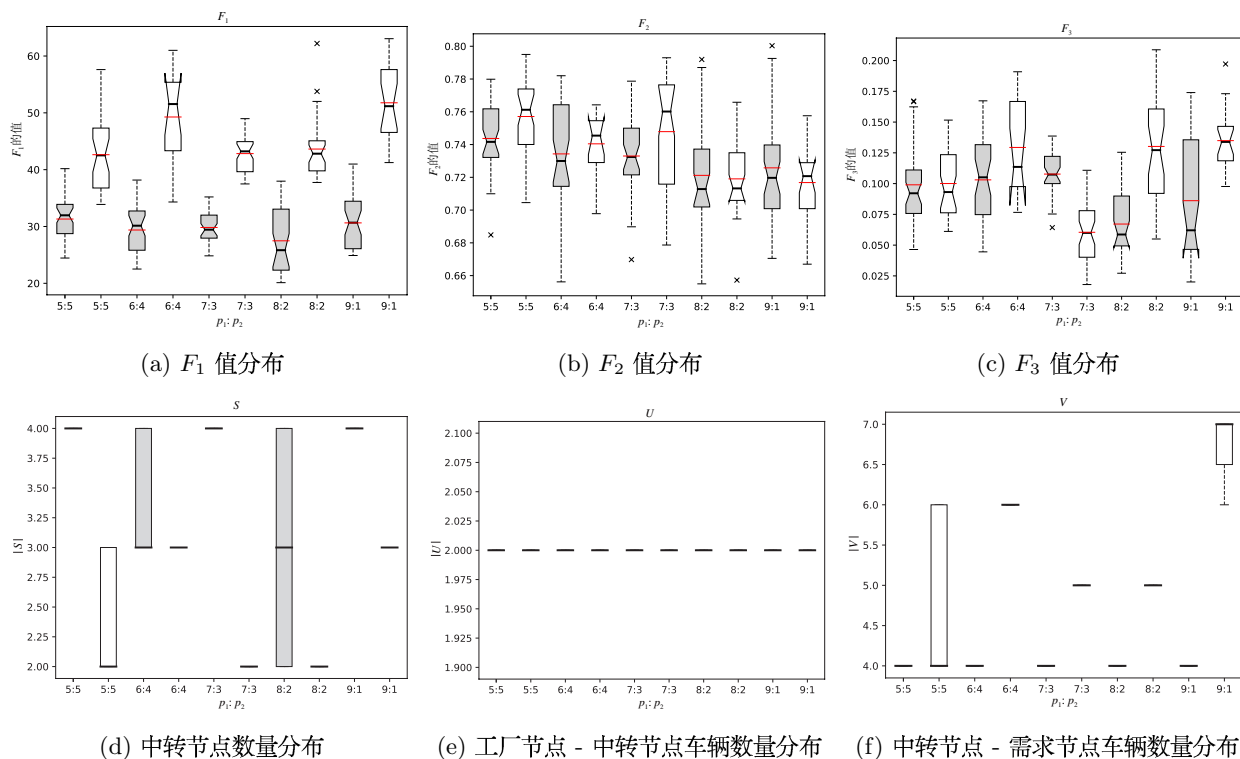


图 7 合车运输和分车运输对比

除对比不同的 $p_1 : p_2$ 外, 本文还对比了各类物资合车运输和分车运输的差异. 分车运输的实验假设每个需求节点的不同物资需求仅能被唯一中转节点满足. 算法实现仅需对中从中转节点到需求节点的路径生成算法做出调整, 算法其他部分及参数与合车运输时保持一致. 合车运输和分车运输时的目标函数值及使用的中转节点、各类车辆数量如图 7.

图 7(a)~(c) 为根据第 4 节的多目标优化算法得到的 Pareto 最优解的分布情况. 图中横坐标为两种物资的体积比 (5 个实验用例中每个需求节点两种物资需求总量在表 1 中给出), 各横坐标对应一个实验用例, 三个图对应三个目标函数. 箱线图描述某个测试用例的 Pareto 最优解集中某个目标函数值的分布情况, 灰色和白色的图形分别表示合车运输和分车运输时的情况. 在救援及时性这一目标函数上合车运输策略显著优于分车运输; 在综合满意度这一目标函数上, 当两类物资数量差异不大时分车运输略优于合车运输, 当两类物资量差异较大 ($p_1 : p_2 = [8 : 2], [9 : 1]$) 时两种策略差别较小; 在公平性这一目标函数上, 除 $p_1 : p_2 = [7 : 3]$ 外, 合车运输均优于分车运输.

图 7(d)~(f) 为 Pareto 最优解集中每个可行解实际使用的中转节点数量和两类车辆数量. 合车运输时中转节点的使用数量在各种条件下均高于分车运输, 这在一定程度上解释了合车运输在救援及时性指标上的优异表现; 各最优解的工厂节点 - 中转节点车辆数量均为 2, 即从 2 个工厂节点各派出 1 辆车; 合车运输时各最优解的中转节点 - 需求节点车辆数量均为 4, 而分车运输时仅有极少数解的车辆使用数量为 4, 且当两类物资数量差异较大 ($p_1 : p_2 = [9 : 1]$) 时甚至出现了使用 7 辆车的解. 虽然合车运输使用的中转节点略多于分车运输, 但合车运输在使用车辆数较少的条件下目标函数 F_1 显著优于分车运输时的目标函数值, 这印证了本节开头的假设.

5.3 测试数据实验

5.2 的算例分析表明同等条件下合车运输相比分车运输使用车辆数量更少, 车辆行驶里程更少, 本部分主要目的在于验证本文提出算法面对更大规模问题时的运行速度和优化结果. 此外, 还对比了本文提出的算法和基于 Pareto 排序的 NSGA-III 算法^[17,25] 在算法效率和目标函数寻优上的差异. NSGA-III 种群容量、子代数量和变异率均与本文提出算法的各单目标种群保持一致, 即 $\mu', \lambda', mtt'_{\text{prob}} = \{500, 300, 0.3\}$ (未令 $\mu' = 3\mu$ 的原因为, 此时算法进行 Pareto 排序会十分缓慢, 与本文提出的算法效率相差巨大), 种群中存在可行解和不可行解时, 依照 [25] 提出的约束支配关系选择精英解. 篇幅所限, 仅展示从 5.1 中构造的测试数据中选取的共 4 组 ($[|O|, |S|, |D|] = [2, 3, 30], [2, 3, 50], [3, 5, 30], [3, 5, 50]$) 每组 5 个测试用例的求解结果. 算法实现环境为 Intel Xeon E5-2682 v4 (1core) CPU, 2 GB RAM, CentOS7.3, Python3.6.3. 实验结果如表 2~5, 表头中 $|O|, |S|, |D|$ 分别为工厂节点、中转节点、需求节点数量, $|U|, |V|$ 分别为工厂节点 - 中转节点层和中转节点 - 需求节点层车辆数量, C_u, C_v 为两种车辆的容量. 表中算例名称右侧依次为 2 种不同算法得到的合车运输和分车运输时 Pareto 最优解的数量 ($n_{\text{opt}}, n'_{\text{opt}}$)、算法运行时间 (t, t') 及 Pareto 最优解集中救援及时性 ($F_{1\text{median}}, F'_{1\text{median}}$)、综合满意度 ($F_{2\text{median}}, F'_{1\text{median}}$)、公平性 ($F_{3\text{median}}, F'_{3\text{median}}$) 三个目标函数的中位数.

为了对比本文提出的算法和 NSGA-III 算法生成解集的差异, 采用 Hypervolume 指标^[26,27] $I_{\text{hv}}(A)$, Schott's spacing metric 指标^[28] $I_{\text{ss}}(A)$ 和 Coverage 指标^[27] $I_{\text{c}}(A, B)$ 来衡量算法的性能. 通常从收敛性和多样性两方面评价多目标优化算法得到解集的优劣, 收敛性方面解集应尽可能接近真实的 Pareto 前沿, 多样性方面解集中各解的差异应尽可能大. $I_{\text{hv}}(A)$ 指标为解集与参考点形成的超体积, 可同时衡量解集的收敛性和多样性. 该指标越大越好, 因为当解分布集中或离真实的 Pareto 前沿较远时该值不会太大. $I_{\text{ss}}(A)$ 指标测度解集的一致性, 该指标越小越好. 其定义为 $I_{\text{ss}}(A) = [\sum_{z \in A} (\bar{d} - d(z))^2 / (|A| - 1)]^{1/2}$, 其中 $d(z) = \min_{z' \in A} \{\sum_{m=1}^M |z_m - z'_m|\}$, \bar{d} 为 $d(z)$ 的均值, z_m 为目标函数向量 z 的 m 分量. $I_{\text{c}}(A, B)$ 指标为 B 中被 A 支配或与 A 相等的解的数量在 B 中所占比例, 该指标越接近 1 表明 A 相比 B 越好. 其定义为 $I_{\text{c}}(A, B) = |z^2 \in B | \exists z^1 \in A : z^1 \succeq z^2| / |B|$, 其中 $z^1 \succeq z^2$ 表示 z^1 支配 z^2 或与 z^2 相等. 计算 $I_{\text{hv}}(A)$ 和 $I_{\text{ss}}(A)$ 需要先将解集标准化. 标准化方法为, 首先将原始模型转化为最小化问题, 即令解集中 z_{2i} 均取对应的相反数; 然后使用 $z'_{mi} = (z_{mi} - \min_m^{(\cdot)}) / (\max_m^{(\cdot)} - \min_m^{(\cdot)})$ 对解集中所有解的 $|M| = 3$ 个目标函数值分别标准化处理, 其中 $\max_m^{(\cdot)}$ 为解集 (\cdot) 中所有解向量 m 分量的最大值, 对于 I_{ss} 指标, $\max_m^{(\cdot)} = \max_m^{(A)}$, 对于 I_{hv} 指标,

表 2 $|O|, |S|, |D| = 2, 3, 30$ 合车运输和分车运输对比

算例	算法	n_{opt}	n'_{opt}	$t(s)$	$t'(s)$	$ O , S , D = 2, 3, 30 \quad U = 6 \quad C_u = 12500 \quad V = 12 \quad C_v = 500$									
						$F_{1\text{median}}$	$F'_{1\text{median}}$	$F_{2\text{median}}$	$F'_{2\text{median}}$	$F_{3\text{median}}$	$F'_{3\text{median}}$	I_{hv}	I'_{hv}	I_{ss}	I'_{ss}
Set4a_20_2_3_30	本文算法	90	100	93.05	90.98	249171.76	387141.29	0.800638	0.797558	0.011619	0.015643	0.8655	0.6925	0.1324	0.1833
	NSGA-III	84	60	275.52	280.85	278656.96	449039.93	0.723578	0.722736	0.108430	0.113914	0.0374	0.0140	0.3875	0.4412
Set4a_24_2_3_30	本文算法	87	73	92.36	90.34	244535.03	33825.19	0.820302	0.826861	0.037961	0.040853	0.5564	0.7772	0.1490	0.2114
	NSGA-III	55	72	285.18	287.94	278117.34	467840.77	0.746921	0.748843	0.182657	0.191024	0.0122	0.0121	0.2854	0.4280
Set4a_28_2_3_30	本文算法	74	74	93.83	90.24	229349.89	390991.24	0.797452	0.799234	0.027106	0.019681	0.8965	0.9779	0.1689	0.1947
	NSGA-III	67	64	285.51	258.84	267916.39	426368.76	0.729741	0.727968	0.163092	0.167780	0.0151	0.0128	0.4271	0.4161
Set4a_32_2_3_30	本文算法	78	80	94.02	89.71	241835.59	379919.43	0.813713	0.816966	0.031842	0.034046	0.4867	0.6327	0.1471	0.1686
	NSGA-III	57	63	284.56	238.32	268572.77	431896.84	0.727887	0.726262	0.162762	0.161528	0.0119	0.0040	0.5203	0.3830
Set4a_36_2_3_30	本文算法	63	67	95.75	89.96	232593.29	391464.62	0.799482	0.789610	0.016619	0.013168	0.8573	0.2113	0.2066	0.1819
	NSGA-III	62	66	298.12	209.05	276205.13	438267.07	0.728373	0.727801	0.131728	0.133766	0.0070	0.0138	0.3702	0.3372
均值	本文算法	78.4	78.8	93.80	90.25	239497.11	388668.35	0.806317	0.806046	0.025029	0.024678				
	NSGA-III	65	65	285.78	255.00	273893.72	442682.67	0.731300	0.730722	0.149734	0.153602				

表 3 $|O|, |S|, |D| = 2, 3, 50$ 合车运输和分车运输对比

算例	算法	n_{opt}	n'_{opt}	$t(s)$	$t'(s)$	$ O , S , D = 2, 3, 50 \quad U = 6 \quad C_u = 12500 \quad V = 12 \quad C_v = 500$									
						$F_{1\text{median}}$	$F'_{1\text{median}}$	$F_{2\text{median}}$	$F'_{2\text{median}}$	$F_{3\text{median}}$	$F'_{3\text{median}}$	I_{hv}	I'_{hv}	I_{ss}	I'_{ss}
Set4a_20_2_3_50	本文算法	65	42	116.69	106.44	386708.10	617377.23	0.757092	0.741470	0.046617	0.047898	0.4868	0.3624	0.2003	0.2890
	NSGA-III	15	14	208.98	180.11	414685.18	694291.11	0.724269	0.726947	0.130435	0.124538	0.0152	0.0143	0.5415	0.5800
Set4a_24_2_3_50	本文算法	71	27	121.08	99.89	391431.74	651201.79	0.799388	0.756809	0.062107	0.151295	0.5759	0.6212	0.2398	0.1786
	NSGA-III	36	26	180.76	201.18	423086.94	690498.99	0.740942	0.735294	0.180494	0.185265	0.0060	0.0060	0.5114	0.4735
Set4a_28_2_3_50	本文算法	65	38	120.31	97.57	366997.62	573563.59	0.792892	0.744836	0.047427	0.087478	0.2066	0.3568	0.2144	0.2357
	NSGA-III	25	14	219.66	210.87	382408.62	615013.36	0.732716	0.735580	0.149325	0.149608	0.0066	0.0220	0.4763	0.4760
Set4a_32_2_3_50	本文算法	40	35	117.74	104.76	375303.82	590984.75	0.779306	0.761307	0.052205	0.080036	0.3087	0.4748	0.2703	0.3421
	NSGA-III	11	15	228.24	194.67	407341.57	660342.89	0.723121	0.728187	0.133636	0.145931	0.0097	0.0086	0.5305	0.5101
Set4a_36_2_3_50	本文算法	49	64	119.13	106.59	367053.66	615744.35	0.791543	0.740686	0.059921	0.078705	0.4395	0.2085	0.3021	0.1553
	NSGA-III	20	2	203.48	214.76	404878.61	668941.47	0.724794	0.714839	0.149503	0.168147	0.0114	0.0069	0.4591	0.7071
均值	本文算法	58	41.2	118.99	103.05	377498.99	609774.34	0.784044	0.749022	0.053655	0.089082				
	NSGA-III	21.4	14.2	208.22	200.32	406480.18	665817.56	0.729168	0.728169	0.148679	0.154698				

表 4 $|O|, |S|, |D| = 3, 5, 30$ 合车运输和分车运输对比

算例	算法	n_{opt}	n'_{opt}	$t(s)$	$t'(s)$	$ O , S , D = 3, 5, 30 \quad U = 9 \quad C_u = 12500 \quad V = 18 \quad C_v = 500$											
						$F_{1\text{median}}$	$F'_{1\text{median}}$	$F_{2\text{median}}$	$F'_{2\text{median}}$	$F_{3\text{median}}$	$F'_{3\text{median}}$	I_{hv}	I'_{hv}	I_{ss}	I'_{ss}	I_c	I'_c
Set4a_40_3_5_30	本文算法	72	24	103.85	93.59	305030.06	703527.76	0.786606	0.765560	0.024586	0.166921	0.8769	0.1745	0.2045	0.2179	1	1
	NSGA-III	37	2	294.83	292.47	355058.14	603718.76	0.731075	0.740710	0.229273	0.241625	0.0121	0.0094	0.5004	0.7071	0	0
Set4a_44_3_5_30	本文算法	61	39	104.23	96.36	302362.29	621375.92	0.801078	0.767592	0.014811	0.086595	0.9005	0.4418	0.2496	0.2861	1	1
	NSGA-III	23	29	300.86	306.56	351190.37	730696.96	0.739355	0.737241	0.168398	0.194362	0.0238	0.0080	0.6125	0.5146	0	0
Set4a_48_3_5_30	本文算法	61	44	102.47	93.70	292761.05	554150.79	0.789580	0.738682	0.015359	0.120308	0.8105	0.5663	0.2683	0.2265	1	0.9615
	NSGA-III	22	26	298.78	298.70	332457.16	705990.03	0.728933	0.732624	0.208921	0.207993	0.0173	0.2635	0.4858	0.4088	0	0
Set4a_52_3_5_30	本文算法	65	44	103.63	97.03	302028.06	592007.65	0.799344	0.759173	0.033579	0.090655	0.7707	0.5267	0.2344	0.2491	1	1
	NSGA-III	31	28	288.24	291.16	357574.62	717942.91	0.721994	0.728388	0.260442	0.252063	0.0128	0.0226	0.4784	0.4190	0	0
Set4a_54_3_5_30	本文算法	73	41	102.98	99.10	307456.44	591438.38	0.780623	0.754779	0.025939	0.137272	0.5653	0.3194	0.1862	0.2863	1	1
	NSGA-III	41	2	279.40	276.63	366133.67	646653.40	0.723037	0.702123	0.239630	0.252061	0.0208	0.0035	0.4501	0.7071	0	0
均值	本文算法	66.4	38.4	103.43	95.96	301927.58	612500.10	0.791446	0.757157	0.022855	0.120350						
	NSGA-III	30.8	17.4	292.42	293.10	352482.79	681000.41	0.728879	0.728217	0.221333	0.229621						

表 5 $|O|, |S|, |D| = 3, 5, 50$ 合车运输和分车运输对比

算例	算法	n_{opt}	n'_{opt}	$t(s)$	$t'(s)$	$ O , S , D = 3, 5, 50 \quad U = 9 \quad C_u = 12500 \quad V = 18 \quad C_v = 500$											
						$F_{1\text{median}}$	$F'_{1\text{median}}$	$F_{2\text{median}}$	$F'_{2\text{median}}$	$F_{3\text{median}}$	$F'_{3\text{median}}$	I_{hv}	I'_{hv}	I_{ss}	I'_{ss}	I_c	I'_c
Set4a_40_3_5_50	本文算法	74	43	139.27	128.77	456914.09	1013726.90	0.789411	0.758149	0.063522	0.166065	0.1091	0.4426	0.1548	0.2265	1	1
	NSGA-III	21	27	236.22	248.54	516392.42	1095198.42	0.733231	0.733455	0.248819	0.241071	0.0031	0.0321	0.5916	0.5607	0	0
Set4a_44_3_5_50	本文算法	78	18	138.34	128.43	411273.61	951784.27	0.780673	0.751196	0.050642	0.109915	0.4968	0.4344	0.2170	0.2948	1	1
	NSGA-III	29	18	210.73	228.35	466276.46	972598.25	0.732678	0.729039	0.196781	0.196863	0.0153	0.0032	0.4774	0.5332	0	0
Set4a_48_3_5_50	本文算法	77	32	136.59	130.07	426057.03	905624.13	0.775476	0.748957	0.038896	0.112860	0.6372	0.3139	0.2314	0.2356	1	1
	NSGA-III	22	19	187.85	212.65	473415.83	990840.13	0.735367	0.730625	0.197124	0.202763	0.0031	0.0073	0.6489	0.4383	0	0
Set4a_52_3_5_50	本文算法	85	40	137.32	126.07	447671.44	955365.75	0.776822	0.755524	0.058524	0.143713	0.3030	0.5274	0.2465	0.2479	1	1
	NSGA-III	26	28	216.93	251.70	506109.84	1069592.29	0.730264	0.731640	0.237121	0.236366	0.0354	0.0114	0.4525	0.5314	0	0
Set4a_54_3_5_50	本文算法	103	41	138.35	125.37	443916.04	980235.09	0.772218	0.761268	0.065694	0.127671	0.5134	0.4577	0.1923	0.1499	1	1
	NSGA-III	34	26	213.94	226.11	508304.51	1062634.19	0.728971	0.732887	0.240308	0.230225	0.0119	0.0120	0.4100	0.4355	0	0
均值	本文算法	83.4	34.8	137.97	127.74	437166.44	961347.23	0.778920	0.755019	0.055456	0.132045						
	NSGA-III	26.4	23.6	213.13	233.47	494099.81	1038172.66	0.732102	0.731529	0.224031	0.221458						

$\max_m^{(\cdot)} = \max_m^{(A \cup B)}$, 其中 A 为待评价算法得到的解集, B 为与待评价算法对比的算法得到的解集, $\min_m^{(\cdot)}$ 与之类似. 计算 I_{hv} 指标的参考点取 $[1.1, 1.1, 1.1]$.

首先对比本文提出的算法求解合车运输和分车运输问题的结果, 以比较两种不同策略的差异, 表中加粗字体表示合车运输和分车运输的优劣关系. 当 $|O|, |S|, |D|$ 数量增加时, Pareto 最优解的数量变化较小, 合车运输 $|O|, |S|, |D| = 3, 5, 50$ 相比 $|O|, |S|, |D| = 2, 3, 30$ 算法运行时间增长约 47%. $|O|, |S|, |D| = 2, 3, 30$ 时本文算法存在部分测试用例分车运输时在综合满意度和公平性指标上优于合车运输, 当 $|O|, |S|, |D|$ 为其他值时本文算法采用合车运输策略在救援及时性和公平性目标函数上的表现均优于分车运输, 且随着问题规模扩大优势更加明显. 各组测试用例及两种不同运输策略在综合满意度目标函数上的表现差异较小, 大致呈现随问题规模扩大综合满意度略微减小的趋势. 然后对比本文提出的算法和 NSGA-III 得到结果的差异, 表 2~5 中加下划线的表示同等条件下 NSGA-III 算法的表现优于本文算法的表现. 除 Set4a_40_3_5_30 的 $F'_{1median}$ 外, 无论是算法运行效率还是“救援及时性”、“综合满意度”和“公平性”目标函数, 本文算法均明显优于 NSGA-III. 通过 I_{hv} 和 I_{ss} 指标可以看出本文算法在收敛性和解的多样性上均优于 NSGA-III 算法, 通过 I_c 指标可以看出针对各种规模问题采用各种运输策略时本文算法得到的解集均优于 NSGA-III 得到的解集, 对于测试用例 Set4a_48_3_5_30, 采用分车运输策略时, NSGA-III 解集中存在不被本文算法解集支配的解. 综上, 本文提出的算法对上述的不同规模问题均可在可接受的时间内给出非支配解集, 本文提出的异质物资合车运输策略明显优于分车运输的策略, 且本文基于目标函数分解的多目标优化算法在解的质量上优于基于 Pareto 排序的 NSGA-III 算法.

6 结论

针对灾害发生之后第一时间各类型救援物资供给受限的情况, 本文研究多工厂节点、多品种物资的异质物资合车运输多目标双层选址 - 路径 - 配给优化问题. 文章建立了包含救援及时性、综合满意度、公平性三个目标函数的多目标混合整数规划模型, 采用基于目标分解的协同进化算法对优化模型进行求解. 通过算例分析和基准测试数据实验验证了算法的在不同规模问题下的优化结果和运行速度. 本文将应急救援物流网络分为工厂节点、中转节点、需求节点构成的双层网络, 且考虑了多品种物资的情况. 各类物资首先从工厂节点运往中转节点, 然后不同品种物资合车运输从中转节点运往需求节点. 由于灾害发生第一时间救援物资具有稀缺性, 本文还考虑了总供给无法满足总需求的情况, 并由此引出了多目标优化模型中的“综合满意度”和“公平性”优化目标. 当多目标优化模型中有三个或三个以上目标函数时, 基于 Pareto 排序的多目标优化算法由于非支配解的比例较高, 会出现算法效率较低的问题. 本文采用基于目标分解的多目标协同进化算法对提出的混合整数规划模型进行求解, 同时优化车辆路径结构和物资配给方案. 进化算法的搜索空间包括可行域和不可行域, 通过一定的控制策略指导算法的搜索方向, 最终得到一组 Pareto 最优解. 本文提出了资源稀缺条件下的异质物资合车运输双层选址 - 路径 - 配给优化模型, 设计的算法可以对各种规模问题在可接受的时间内给出 Pareto 最优解集. 需求紧急性、资源稀缺性、商品异质性是应急救援物流优化问题的重要特征, 应急救援物流优化问题另外的重要特征是需求的不确定性及路网可靠性的不确定性, 如何解决上述不确定性灾后应急救援物流优化问题是十分有现实意义的研究课题. 此外, 本文模型未考虑物资在同一层节点间的中转, 在需求不确定的双层车辆路径问题中, 考虑物资在同一层节点间的中转也是一个很重要的问题.

参考文献

- [1] Salhi S, Nagy G. A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling[J]. Journal of the Operational Research Society, 1999, 50(10): 1034-1042.
- [2] Schneider M, Drexel M. A survey of the standard location-routing problem[J]. Annals of Operations Research, 2017, 259(1-2): 389-414.
- [3] Drexel M, Schneider M. A survey of variants and extensions of the location-routing problem[J]. European Journal of Operational Research, 2015, 241(2): 283-308.
- [4] Nagy G, Salhi S. Location-routing: Issues, models and methods[J]. European Journal of Operational Research, 2007, 177(2): 649-672.

- [5] Rath S, Gutjahr W J. A math-heuristic for the warehouse location-routing problem in disaster relief[J]. *Computers & Operations Research*, 2014, 42: 25–39.
- [6] 王绍仁, 马祖军. 震害紧急响应阶段应急物流系统中的 LRP[J]. *系统工程理论与实践*, 2011, 31(8): 1497–1507.
Wang S R, Ma Z J. Location-routing problem in emergency logistics system for post-earthquake emergency relief response[J]. *Systems Engineering — Theory & Practice*, 2011, 31(8): 1497–1507.
- [7] 詹沙磊, 刘南. 基于灾情信息更新的应急物资配送多目标随机规划模型 [J]. *系统工程理论与实践*, 2013, 33(1): 159–166.
Zhan S L, Liu N. Multi-objective stochastic programming model for relief allocation based on disaster scenario information updates[J]. *Systems Engineering — Theory & Practice*, 2013, 33(1): 159–166.
- [8] 王海军, 杜丽敬, 马士华. 震后应急物流系统中双目标开放式选址: 路径问题模型与算法研究 [J]. *管理工程学报*, 2016, 30(2): 108–115.
Wang H J, Du L J, Ma S H. Model and algorithms for integrated open location and routing problem in emergency logistics under earthquake[J]. *Journal of Industrial Engineering and Engineering Management*, 2016, 30(2): 108–115.
- [9] 王旭坪, 马超, 阮俊虎. 运力受限的应急物资动态调度模型及算法 [J]. *系统工程理论与实践*, 2013, 33(6): 1492–1500.
Wang X P, Ma C, Ruan J H. Model and algorithm of relief materials dynamic scheduling without sufficient vehicle quantity[J]. *Systems Engineering — Theory & Practice*, 2013, 33(6): 1492–1500.
- [10] Lin J R, Lei H C. Distribution systems design with two-level routing considerations[J]. *Annals of Operations Research*, 2009, 172(1): 329–347.
- [11] Zhou L, Baldacci R, Vigo D, et al. A multi-depot two-echelon vehicle routing problem with delivery options arising in the last mile distribution[J]. *European Journal of Operational Research*, 2018, 265(2): 765–778.
- [12] Boccia M, Crainic T G, Sforza A, et al. Multi-commodity location-routing: Flow intercepting formulation and branch-and-cut algorithm[J]. *Computers & Operations Research*, 2018, 89: 94–112.
- [13] 黄凯明, 卢才武, 连民杰. 多层次设施选址 - 路径规划问题建模及算法 [J]. *控制与决策*, 2017, 32(10): 1803–1809.
Huang K M, Lu C W, Lian M J. Research on modeling and algorithm for multi-echelon location-routing problem[J]. *Control and Decision*, 2017, 32(10): 1803–1809.
- [14] Boccia M, Crainic T G, Sforza A, et al. Location-routing models for designing a two-echelon freight distribution system[R]. Montréal: Université de Montréal, 2011.
- [15] Laporte G. Location-routing problems[M]// Golden B L, Assad A A. *Vehicle Routing: Methods and Studies*. Amsterdam: North-Holland Publishing Company, 1988.
- [16] Cho J H, Wang Y, Chen I R, et al. A survey on modeling and optimizing multi-objective systems[J]. *IEEE Communications Surveys & Tutorials*, 2017, 19(3): 1867–1901.
- [17] Deb K, Jain H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints[J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(4): 577–601.
- [18] Glover F, Hao J K. The case for strategic oscillation[J]. *Annals of Operations Research*, 2011, 183(1): 163–173.
- [19] Wang J, Zhang W, Zhang J. Cooperative differential evolution with multiple populations for multiobjective optimization[J]. *IEEE Transactions on Cybernetics*, 2016, 46(12): 2848–2861.
- [20] Das S, Suganthan P N. Differential evolution: A survey of the state-of-the-art[J]. *IEEE Transactions on Evolutionary Computation*, 2011, 15(1): 4–31.
- [21] Breunig U, Schmid V, Hartl R F, et al. A large neighbourhood based heuristic for two-echelon routing problems[J]. *Computers & Operations Research*, 2016, 76: 208–225.
- [22] Breunig U. Two-echelon vehicle routing problem — Collection of benchmark instances[EB/OL]. [2018-01-05]. <http://prolog.univie.ac.at/research/TwoEVRP/>.
- [23] Beasley J E. OR-library[EB/OL]. [2018-01-05]. [http://people.brunel.ac.uk/\\$mastjjb/jeb/orlib/files/vrp2e.rar](http://people.brunel.ac.uk/$mastjjb/jeb/orlib/files/vrp2e.rar).
- [24] Guo P H. LRP2E-instances[EB/OL]. [2018-01-05]. <https://github.com/phguo/LRP2E-instances/>.
- [25] Jain H, Deb K. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach[J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(4): 602–622.
- [26] Zitzler E, Thiele L. Multiobjective optimization using evolutionary algorithms — A comparative case study[C]// *International Conference on Parallel Problem Solving from Nature*, 1998: 292–301.
- [27] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach[J]. *IEEE Transactions on Evolutionary Computation*, 1999, 3(4): 257–271.
- [28] Schott J R. *Fault tolerant design using single and multicriteria genetic algorithm optimization*[D]. Boston: Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1995.