

随机需求车辆路径问题及混合变邻域分散搜索算法求解

李 阳¹, 范厚明^{1†}, 张晓楠², 杨 翔¹

(1. 大连海事大学 战略管理与系统规划研究所, 大连 116026;

2. 陕西科技大学机电工程学院, 西安 710021)

摘要: 随机需求车辆路径问题(CVRPSD)是对带容量约束车辆路径问题(CVRP)的扩展, 需求不确定的特点使其较CVRP更复杂, 对求解方法要求更高. 基于先预优化后重调度思想, 提出两阶段的混合变邻域分散搜索算法(VNSS)对该问题进行求解: 预优化阶段构建随机机会约束规划模型, 对客户点随机需求作机会约束确定型等价处理, 生成最优预优化方案; 重调度阶段采用新的点重优化策略进行线路调整, 降低因失败点而产生的额外成本, 减少对人工和车辆的占用. 算例验证表明, 随机机会约束模型和两阶段变邻域分散搜索算法在求解CVRPSD时较为有效, 点重优化策略调整效果更佳.

关键词: 车辆路径问题; 随机需求; 点重优化策略; 分散搜索算法; 变邻域搜索算法

中图分类号: TP301.6 文献标识码: A

Two-phase variable neighborhood scatter search for the capacitated vehicle routing problem with stochastic demand

LI Yang¹, FAN Hou-ming^{1†}, ZHANG Xiao-nan², YANG Xiang¹

(1. Institute of Strategy Management and System Planning, Dalian Maritime University, Dalian 116026, China;

2. College of Mechanical and Electrical Engineering, Shaanxi University of Science and Technology, Xian 710021, China)

Abstract: The capacitated vehicle routing problem with stochastic demand (CVRPSD) is an extension of capacitated vehicle routing problem (CVRP) which is a well-known NP-hard problem. Due to the stochastic characteristic of customer's demand, the solution process is evidently different from deterministic CVRP and it is rather complicated to be solved. Based on the principles of pre-optimization and re-dispatch, a two-stage variable neighborhood scatter search algorithm (VNSS) is proposed. In first stage, the stochastic chance constrained optimization model is constructed and the stochastic constrain of customer's demand is transformed to a certain constrain. On base of the equivalent formation, the optimal solutions of pre-optimization scheme are generated by VNSS and it could be participated in the re-dispatch optimization. In second stage, the paper proposes a new re-dispatch policy to deal with the so-called failure point in pre-optimization scheme. The failure point and the customers afterwards are re-optimized to avoid unnecessary vehicle routings which may cause extra costs and more vehicles. Numerical results show that the two-stage VNSS and re-dispatch policy is rather effective.

Key words: vehicle routing problem; stochastic demand; re-dispatch policy; scatter search algorithm; variable neighborhood search algorithm

1 引言(Introduction)

车辆路径问题(Vehicle Routing Problem)是组合优化和运筹学领域的研究热点, 许多学者对经典的确定型VRP予以了极大的关注, 但其局限于一系列已知的客户点及相关配送信息, 与现实情况存在差异. 在实际配送环节中, 往往客户需求、位置、路况等信息是

未知的, 这些未知因素的影响引起了对不确定车辆路径问题的思索, 目前不确定VRP的研究多集中于随机需求车辆路径问题(Capacitated Vehicle Routing Problem with Stochastic Demand, CVRPSD)^[1].

在确定型VRP研究方面, 多种启发式算法已被应用于模型及算例求解, 文献[2]提出了模拟动物捕食策

收稿日期: xxxx-xx-xx; 录用日期: xxxx-xx-xx.

†通信作者. E-mail: fhm468@163.com; Tel.: +86 0411-84725868.

国家自然科学基金(61473053, 70801007), 辽宁省社会科学规划基金重点项目(L16AGL004), 辽宁省教育厅科学技术研究一般项目(L2014046), 大连市科学技术计划项目(2015D12ZC181)

Supported by National Natural Science Foundation of China(61164015, 61305132), Key Project of Social Science Planning Foundation of Liaoning Province(L16AGL004), Science and Technology Foundation of Educational Commission of Liaoning Province(L2014046), Science and Technology Planning Foundation of Dalian(2015D12ZC181).

略的捕食搜索算法;文献[3-4]设计量子进化算法,对量子位操作进行改进;文献[5]构造了结合2-OPT的混合遗传算法;文献[6]通过调整粒子惯性权值对粒子群算法做了改进;文献[7]将大邻域搜索与蚁群算法相结合,优化了解方案寻优环节;文献[8]基于局部改进原理提出了新的随机递归算法;通过构造路径生成和节约计算准则,文献[9]设计了改进节约算法.然而由于CVRPSD中客户点需求随机的影响,以往求解确定型VRP的理论和方法不再有效,目前在求解CVRPSD时多先采用机会约束规划或随机规划方法对随机变量进行处理,然后在预优化生成的路线基础上执行重调度操作^[10-12],值得注意的是与经典的确定型VRP不同,CVRPSD目标函数求解的是最小期望总成本^[13].在求解方法方面精确算法和启发式算法均有采用,但由于随机问题的复杂性,精确算法(如L-shaped和Branch-and-price算法等)只能在合理的时间内对中小规模的算例进行求解^[14-15],启发式算法则被更多应用于一系列各种规模的不确定问题求解^[16-19],取得了较好的结果.

目前国内学者对CVRPSD的研究还相对较少^[20-21],本文以CVRPSD为研究对象,构建了随机机会约束规划模型,对需求随机变量作机会约束确定型等价处理,设计两阶段启发式的混合变邻域分散搜索算法(VNSS)对问题进行求解,在预优化阶段保证生成质量较高的解方案,随后在重调度阶段执行新的点重优化策略对其进行调整,以合理降低因客户点需求随机和线路失败所增加的配送线路额外成本,有效减少人工和配送车辆占用,最后通过CVRP与CVRPSD算例集对比实验对本文模型和算法进行了测试.

2 问题描述及模型建立(Problem description and mathematical model)

2.1 CVRPSD问题描述(Description of CVRPSD)

本文研究的CVRPSD是对CVRP拓展而来,对该问题描述如下:配送网络中存在一个配送中心、多客户点和多配送车辆,在一系列因素(车辆载重、客户点需求等)影响下,合理安排车辆线路和客户点服务次序,使总体配送期望成本最小.对CVRPSD有如下假设:①客户点需求不确定,其需求量大小满足随机分布且相互独立,期望值不大于配送车辆最大装载量,且只有当配送车辆到达该点时其需求才会被确认;②客户点需求本文假设其服从泊松分布;③配送中心、客户点信息已知,配送网络中所有节点完全相互连通,配送线路起点和终点均为配送中心;④配送车辆信息已知,所有车辆均为同一型号,性能相同,预优化阶段每辆车仅有一条服务路径且路径中客户点随机需求量总和不超过配送车辆最大装载量,客户点仅接受一辆车进行服务;⑤配送中心服务能力满足客户点要求,

可同时派出多辆车(不超过可用车辆总数)进行配送服务,客户需求量充足;⑥文中研究的是向客户点交付货物问题,车辆从配送中心出发时均处于满载状态.

在CVRPSD中,假设有完备的有向图 $G = (V, E)$,其中所有点集合 $V = \{0\} \cup V_o$, 0为配送中心,客户点集合 $V_o = \{1, 2, \dots, n\}$;边集合 $E = \{(i, j) | i, j \in V\}$, c_{ij} 为点 i 和 j 之间的路径成本;可用配送车辆集合 $K = \{1, 2, \dots, \varphi\}$,车辆的最大载重量为 Q ;文中选用泊松分布对客户点需求信息进行表征^[15],客户点 $i(i \in V_o)$ 的需求 ξ_i 是随机变量,服从相互独立的离散随机概率分布,其期望 $E[\xi_i]$ 和方差 $Var[\xi_i]$ 相同均为 λ , i 点需求量为 q 的概率为 $p_{iq} = Pr(\xi_i = q)$, $q = 0, 1, 2, \dots, 0 \leq q \leq Q, i \in V_o$;决策变量 x_{ijk} 表示车辆 k 是否直接从点 i 到达点 j ,是为1,否为0; y_{ik} 表示客户 $i(i \in V_o)$ 是否由车辆 k 服务,是为1,否为0.

2.2 随机机会约束规划模型构建(Stochastic chance-constrained model)

确定型CVRP中客户点需求量已知,仅需合理安排车辆路径以使总配送成本达到最小,而CVRPSD中客户需求具有离散的随机分布特点,且只在配送车辆到达后才明确,因此CVRPSD较CVRP更复杂.在对其求解时先生成预优化车辆路径方案,但该方案可能会出现线路失败,即配送车辆容量不足不能按照预设线路中客户点顺序进行服务,相关客户点称为失败点,应对失败线路或失败点重新调整车辆路径最终完成配送服务使总期望成本最小.文中采用随机机会约束规划对客户点随机信息进行处理,将复杂的随机优化问题转化为由概率密度函数表征的等价确定型优化问题,进而可通过两阶段求解获得最优配送方案.

假设某车辆 k 按照集合 $V_k = \{s, t, u, w\}$, $V_k \subseteq V_o$ 中预优化的客户顺序进行服务,车辆从中心满载(Q)出发,由于客户点需求均服从独立的泊松分布且具有累加性,因此多个独立泊松分布的随机变量 ξ_i ($i \in V_o$)相加仍服从泊松分布^[22].当车辆服务完客户点 s, t, u 时,其实时剩余装载量为 $Q_u = Q - \xi_s - \xi_t - \xi_u$, Q_u 也是随机变量,有 $0 \leq Q_u \leq Q$.模型引入随机机会约束对客户点 w 是否可以接受车辆 k 服务进行检验,首先给定预先设置的风险偏好值 α ,此时 w 点需求量与车辆服务 u 后剩余装载量的关系 $\xi_w \leq Q_u$ 为随机约束,我们希望随机约束以预设的偏好值水平 α 成立,即基于随机概率测度有机会约束 $Pr\{\xi_w \leq Q_u\} \geq \alpha$ 成立^[23]. ξ_w 与 Q_u 的随机关系如式(1)所示:

$$\begin{aligned} Pr\{\xi_w \leq Q_u\} &= Pr\{\xi_w \leq Q - \xi_s - \xi_t - \xi_u\} \\ &= Pr\left\{\sum_{i \in V_k} \xi_i \leq Q\right\} \end{aligned} \quad (1)$$

$Pr\{\xi_w \leq Q_u\}$ 表达的是概率测度($Pr \in [0, 1]$),不妨假设客户点 w 尚未指定车辆进行配送服务,此时

车辆 k 线路中已有客户点 s, t, u , 在决策时 Pr 与 α 共同决定是否以安排车辆 k 对客户点 w 进行服务, 即车辆 k 剩余装载量能否满足客户点 w 随机需求, Pr 越大车辆能够满足其随机需求的概率越大. 在随机机会约束 $Pr\{\xi_w \leq Q_u\} \geq \alpha$ 下, α 是决策者在随机环境下对某客户点执行服务与否的意向量化, 与决策者的风险偏好程度有关, $\alpha \in [0, 1]$, 一般不接受 $\alpha = 0$. α 值越大表明决策者对路线可行性的要求越高, 反之表明决策者比较重视车辆使用情况, 要求单车尽量多执行配送. 值得注意的是, $Pr\{\xi_w \leq Q_u\} \geq \alpha$ 是一种概率, 即使满足该机会约束, 在实际配送环节中仍可能会出现线路失败(不包含 $\alpha = 1$).

由此可知在预优化阶段, 对于给定的 α 值, 若 $Pr \geq \alpha$, 则当前车辆可指派给客户点进行服务, 若 $Pr < \alpha$, 当前车辆不能满足客户点随机需求, 应指派新车进行路径优化, 所有客户点均已安排车辆进行配送后, 即生成了一个可行方案. 相应的预优化随机机会约束规划模型如下:

$$\min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{ij} x_{ijk} \quad (2)$$

$$\text{s.t. } Pr \left\{ \sum_{i \in V} \sum_{j \in V_o} x_{ijk} \cdot \xi_j \leq Q \right\} \geq \alpha, \forall k \in K \quad (3)$$

$$\sum_{i \in V} x_{ijk} = \sum_{i \in V} x_{jik} = 1, \forall j \in V_o, \forall k \in K \quad (4)$$

$$x_{ijk} = 0, \forall i = j, \forall i, j \in V, \forall k \in K \quad (5)$$

$$\sum_{k \in K} y_{ik} = 1, \forall i \in V_o \quad (6)$$

$$\sum_{j \in V_o} x_{0jk} = \sum_{j \in V_o} x_{j0k} \leq 1, \forall k \in K \quad (7)$$

$$\sum_{j \in V} x_{ijk} = y_{ik}, \forall i \in V_o, \forall k \in K \quad (8)$$

$$\sum_{i \in V} x_{ijk} = y_{jk}, \forall j \in V_o, \forall k \in K \quad (9)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1, \forall S \subseteq V_o, \forall k \in K \quad (10)$$

$$x_{ijk} \in \{0, 1\}, y_{ik} \in \{0, 1\}, \forall i, j \in V, \forall k \in K \quad (11)$$

目标函数式(2)为最小化路径成本; 式(3)为随机容量机会约束, 保证预优化阶段车辆路径中客户点随机需求量之和不超过 Q 的概率大于预设的置信度水平; 式(4)表示车辆进出平衡约束, 保证每个客户均有且仅有一辆车对其进行服务; 式(5)表示相同点之间无路径连通; 式(6)保证了每个客户必被一辆车服务; 式(7)保证当车辆被启用时仅有一条服务路径且其始末点均为配送中心; 式(8)和式(9)将决策变量 x_{ijk} 和 y_{ik} 联系起来, 保证客户点被车辆服务时一定有路径与其连接; 式(10)为标准支路消除约束, S 为车辆 k 的服务路线客户集合; 式(11)为决策变量属性.

2.3 随机机会约束确定型等价处理(Equivalent formation of stochastic chance constraint)

模型中随机机会约束(3)可以进一步转化为确定型的等价形式, 保证配送线路可行性. 对于配送车辆 k 而言, 其配送路径上客户点需求 $\xi_j (j \in V_o)$ 均是独立同分布的随机变量, 由中心极限定理可知, 该路径总需求 $\xi_k (k \in K)$ 近似的服从正态分布, 有期望 $M_k = \sum_{i,j \in V} E[\xi_j] \cdot x_{ijk}$ 和标准差

$$S_k = \left(\sum_{i,j \in V} Var[\xi_j] \cdot x_{ijk}^2 \right)^{1/2} = \sqrt{M_k}^{[24]}.$$

对于车辆 k 的客户总需求随机分布, 其转化为标准正态分布后当有常数 τ 满足 $\tau = \Phi^{-1}(\alpha)$ 时(τ 为该分布 α 分位点), 约束(3)等价于

$$Pr \left\{ \left(\sum_{i \in V} \sum_{j \in V_o} \xi_j \cdot x_{ijk} - M_k \right) / S_k \leq \tau \right\} = \alpha, \forall k \in K \quad (12)$$

进而约束(3)可由如下确定型约束代换:

$$M_k + \tau \cdot S_k \leq Q, \forall k \in K \quad (13)$$

式(13)扩展形式如下所示:

$$\sum_{i,j \in V} E[\xi_j] \cdot x_{ijk} + \tau \cdot \left(\sum_{i,j \in V} Var[\xi_j] \cdot x_{ijk}^2 \right)^{1/2} \leq Q, \quad \forall k \in K \quad (14)$$

约束(12)、(13)、(14)是对随机机会约束(3)的拓展, 对于随机变量 ξ_k , 当 M_k 、 S_k 存在且 τ 为常数独立于 x_{ijk} 时, 约束(3)可由(14)替代. 由于随机变量 $\xi_j (j \in V_o)$ 之间相互独立, 且对于所有 $j (j \in V_o)$ 而言, $\left(\sum_{i,j \in V} \xi_j \cdot x_{ijk} - M_k \right) / S_k$ 与 $(\xi_j - E[\xi_j]) / \sqrt{Var[\xi_j]}$ 同分布, $E[\xi_j]$ 与 $Var[\xi_j]$ 呈线性关系(泊松分布, $E[\xi_j] = Var[\xi_j]$), 因而存在一个常数 $\bar{Q}$ 使如下线性约束:

$$\sum_{i,j \in V} E[\xi_j] \cdot x_{ijk} \leq \bar{Q} \quad (15)$$

等价于随机机会约束(3)、(12)和确定型约束(13)、(14), $\bar{Q}$ 值可由式(16)求得^[23]:

$$\bar{Q} = (2Q + \tau^2 - \sqrt{\tau^4 + 4Q \cdot \tau^2}) / 2, \tau = \Phi^{-1}(\alpha) \quad (16)$$

由此随机机会约束(3)可由概率密度确定型等价约束替换, 进而参与模型和后续算法计算.

3 两阶段混合变邻域分散搜索算法(Two-phase variable neighborhood scatter search)

CVRPSD中客户点需求随机, 当配送车辆在对给定的路径按客户点顺序进行服务出现路径失败时, 必须对其进行重新规划. 可见CVRPSD车辆路径优化具有两阶段特点, 基于先预优化, 再重调度的思想, 本文设计了两阶段混合变邻域分散搜索算法(VNSS)对其进行求解. (1)第一阶段首先根据客户点随机需求期望

和方差求解预优化方案,其中机会约束模型和中心极限定理解决了客户点随机需求不能参与模型和算法计算的问题,VNSS保证了进入第二阶段重调度的预优化方案解具有较高的质量。(2)第二阶段对车辆路径中客户点随机需求量依次进行仿真模拟,确认失败线路及失败点,执行本文点重调度策略优化车辆重调度路径,结合预优化方案和重调度路径得到最终优化方案.两阶段启发式算法具体设计如下文所述.

3.1 算法框架(Algorithm framework)

本文VNSS是对变邻域搜索算法(VNS)和分散搜索算法(SS)的集成. SS是一种种群进化算法,系统性强,其解寻优策略具有多样性和随机性的特点,搜索范围较广,不易陷入局部最优,可对当前解不断优化,易于与其他启发式算法结合,目前多被应用于求解组

合优化和路径规划等问题^[25]. VNS是文献[26]提出的一种元启发式算法,相对于一般局部搜索算法而言VNS搜索策略并无固定的轨迹,通过对当前解的不同邻域结构展开大范围的持续性搜索以获得优化,邻域搜索策略多样,可以有效的脱离局部最优,提高寻优效率. SS灵活性较高,其一般流程分为五部分,依次为初始种群生成、种群多样性优化、参考集更新策略、子集对组合策略、种群改进策略,SS的改进或组合优化主要通过上述环节来实现. 本文设计了由变邻域搜索算法和分散搜索算法组成的混合变邻域分散搜索算法(VNSS)求解CVRPSD,求解过程分两阶段,预优化阶段求得最优方案,重调度阶段对预优化方案失败点进行调整,求得最终车辆路径规划方案,算法流程如图1.

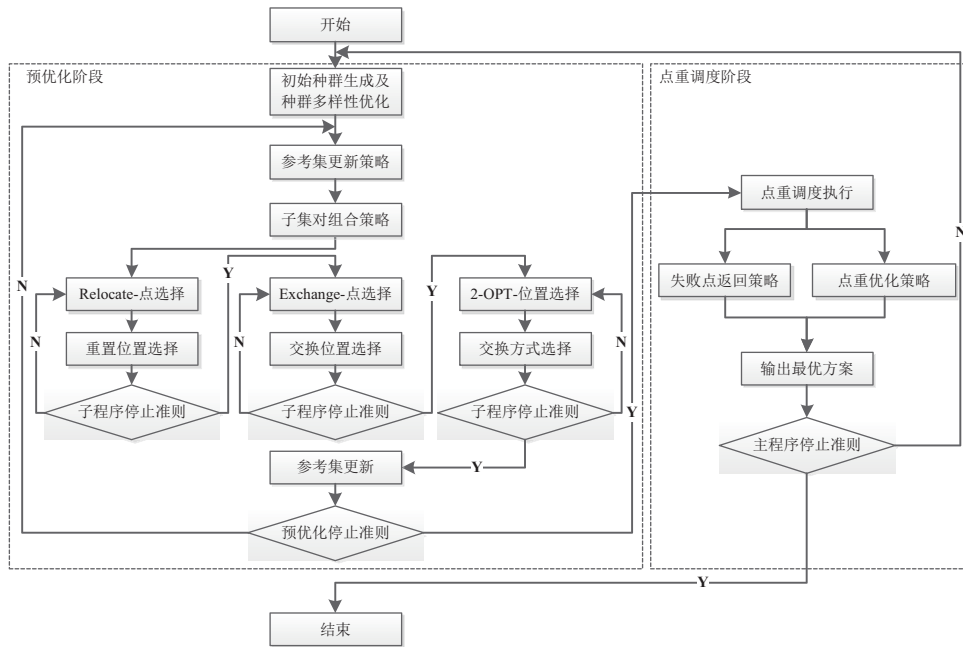


图1 VNSS流程图

Fig. 1 The basic flow of variable neighborhood scatter search

3.2 预优化启发式算法(Heuristic algorithm for pre-optimization)

3.2.1 初始种群生成及多样性优化(Generation of initial population and diversity optimization)

算法中解方案采用整数编码,配送中心为0. 预优化阶段VNSS依据改进节约算法原理选择插入法生成初始种群,客户点 l 在 i, j 点间的插入费用 $c_{ijl} = c_{il} + c_{lj} - gc_{ij} + f|c_{il} - c_{lj}|$, g, f 是随机数, $g \in [0, 3]$, $f \in [0, 1]$ ^[27-28]. 方案构建首先从 $\{0\}$ 开始插入客户点,插入过程中综合对比选取插入费用最小的点位,所生成方案应与当前种群中个体进行检验,保证初始种群中方案均不相同.在对客户点插入费用 c_{ijl} 进行计算时,随机数 g, f 的设置使插入费用不再固定,解决

了点插入方案相对单一的问题,保证了初始种群的多样性优化要求. 在整个VNSS迭代过程中新方案生成后均须进行随机机会约束验证,不满足约束的方案将增加惩罚成本 p_c .

3.2.2 参考集更新及子集对组合策略(Reference set update and subset generation)

算法在初始种群生成后对当前种群中方案进行选择组成参考集 B ,每一代迭代开始前都需更新该参考集,优化过程选择初始种群中目标函数值小(组成精英解集 $B1$)和路径多样性丰富(组成多样性解集 $B2$)的部分方案组成参考集($B = B1 \cup B2$)进入下一步算法迭代.更新时路径多样性解集从排除目标函数值小的当前种群剩余个体中选择,以多样性距离最大为参考

标准, 两方案间的距离等于两方案弧数最大值减两方案共有弧数, $B2$ 中解方案选择过程如下: (1)计算某方案与 $B1$ 中所有精英解方案的距离, 其中最小距离为该方案的多样性距离; (2)计算除(1)及 $B1$ 外所有方案的多样性距离; (3)选择多样性距离大的方案组成 $B2$.

子集对组合策略执行时随机选中参考集 B 中两方案 $X1$ 和 $X2$ 组成一个子集对, 每个子集对经过组合后

生成两新方案 $Y1$ 和 $Y2$, 组合策略反复执行最终形成新的子集进入后续迭代^[29]. $X1$ 和 $X2$ 组合时首先随机产生交叉点位 θ , $X1$ 保留 θ 前线路规划, θ 后客户点顺序与 $X2$ 直接相关, 先去除 $X2$ 中 $X1$ 前 θ 位包含的客户点, 再按照 $X1$ 中 θ 位后客户点位置依次代入处理后的 $X2$ 客户点, 最终组成新方案 $Y1$, $Y2$ 同理(方案中0位置始终不变). 具体子集对组合策略实例如图2.

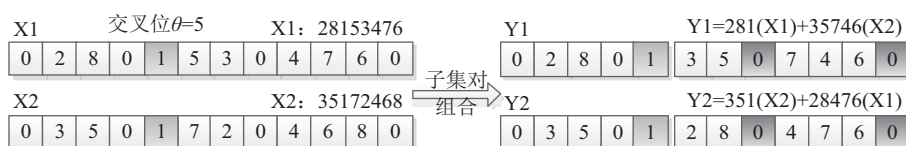


图 2 子集对组合策略示例图

Fig. 2 Examples of Subset generation method

参考集更新及子集对组合策略使VNSS在搜索时不易陷入局部最优, 令新种群与已迭代种群具有良好的分散性, 能大范围的寻找最优解, 保证对当前较优解进行深入搜索, 搜索灵活且效率较高.

3.2.3 变邻域搜索策略(Variable neighborhood search strategy)

本文SS种群改进策略选择VNS优化种群方案, 考虑方案中点及路线间的位置变更, 选取Relocate(点)、Exchange(点)、2-OPT(线)三种结构依次对方案进行变邻域搜索, 使其从邻域内局部最优逐渐向全局最优逼近. 邻域搜索在预设最大迭代次数内仍未找到其改进方案的, 直接执行后续邻域搜索策略, 在搜索时引

入邻域半径减少策略, 首先对被选中两点间的距离进行检验, 与待变更邻域点距离较远的点(大于其与所有点的距离平均值)不纳入邻域结构置换, 提高了邻域搜索效率, 避免无效邻域变更^[30]. 在进行变邻域搜索时线路内或线路间均可执行操作, 通过距离检验的点均能参与方案优化过程, 邻域搜索策略如下:

(1)重置(Relocate). 随机选中当前方案中客户点 i , 在原方案中移除该点, 假设 RV_i 为 i 点通过半径减少策略检验的可重置点 j 集合(包括配送中心), 可知 i 点可重置在 RV_i 中每个点两侧, 当重置点为配送中心时, 除两侧外 i 点还有第三个重置位, 即在两中心间重置组成新路线. 具体客户点重置策略如图3.

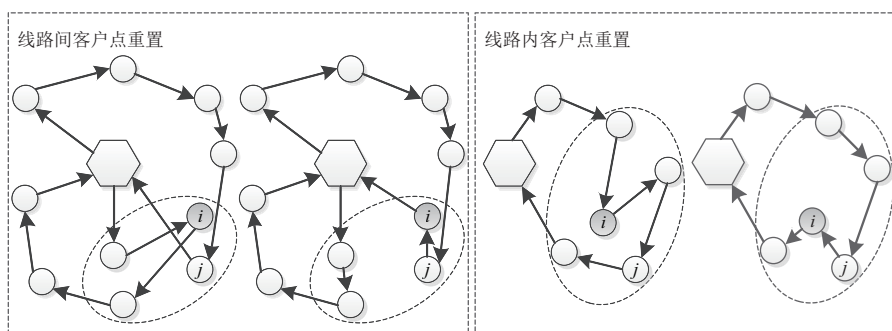


图 3 客户点重置示意图

Fig. 3 Examples of customer relocate move while Inter-Route and Intra-Route

(2)点交换(Exchange). 随机选中方案某客户点 i , 交换邻域结构下 i 点与 RV_i 中点直接交换位置, 交换点不为配送中心, 确定的两点仅有1种交换方案.

(3)边交换(2-OPT). 在当前方案中随机选中某客户点 i , 将 i 点所在线路从当前方案中单独分离记为 $R1$. 假设 j 点为某待2-OPT操作的客户点($j \in RV_i$), 先确认 i 、 j 是否在方案同一子线路中, 若 i 、 j 同路且不相邻, 则逆序两点间路径使其相连, 组成新线路 $NR1$,

反之略过; 当 i 、 j 分属不同线路时, 将 j 点所在子线路从方案中分离记为 $R2$, 不同线路间2-OPT操作通过交换 $R12$ 两线路中部分边将 i 、 j 相连组成新的子线路, 交换包括顺序交叉和逆序交叉的方式共四种方案 $NR1$ 、 $NR2$ 、 $NR3$ 、 $NR4$, 每种方案生成1、2子线路, 直接与当前方案相连构成2-OPT邻域解, 从中选取最优2-OPT邻域解继续迭代, 边交换过程实例如图4所示.

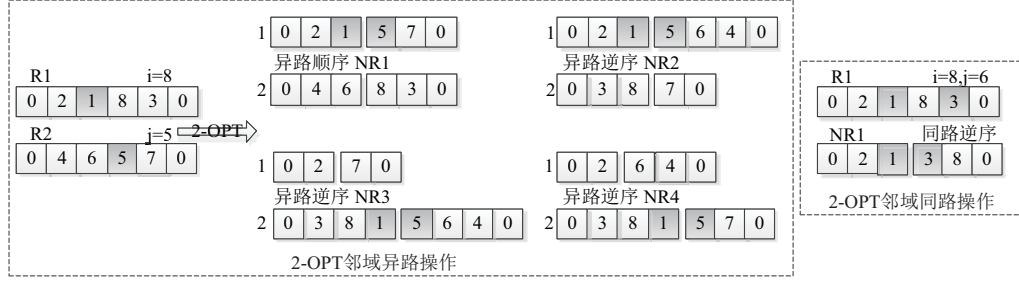


图4 2-OPT操作示意图

Fig. 4 Examples of 2-OPT move while Inter-Route and Intra-Route

3.3 失败点重调度策略(Re-dispatch strategy for failure points)

客户点需求随机使决策者在对其服务顺序进行安排时出现不确定性, 因此配送车辆能否满足线路中客户点需求是未知的, 通过对模型机会约束确定型等价计算可得车辆容量限制 $\bar{Q}$, $\bar{Q}$ 参与模型及算法对配送车辆预优化线路进行规划. 车辆容量限制 $\bar{Q}$ 约束的是一定概率下线路不失败的可能性, 通过对预优化线路中客户点随机需求进行逐点仿真, 可模拟实际车辆配送过程, 当预优化线路出现失败点时, 该子线路已不能按计划进行服务, 需要执行第二阶段重调度策略对其后续路径进行重新规划, 以满足所有随机客户的配送需求.

3.3.1 点返回策略(Point return strategy)

在以往的CVRPSD研究中重调度策略的实质多为点返回策略, 分为两种: 失败点返回策略、失败点前序点返回策略. 失败点返回策略按照预优化路径执行配送, 以车辆按计划服务完某客户 j 后的实时装载量 Q_j 为例, $Q_j > 0$ 时直接服务下一客户点, 当 $Q_j = 0$ 时返回配送中心后按预优化方案直接服务下一客户点, $Q_j < 0$ 时说明当前客户点 j 为失败点且未被完全服务, 车辆返回中心后应继续对 j 及后续客户点进行配送^[31]. 然而依据三角形三边原理, 在失败点前一点返回配送中心后继续按预优化方案服务一定比在失败点返回成本更低, 失败点前序点返回策略即基于该原理对车辆返回点进行合理选择以降低配送成本, 但由于客户点需求随机的影响, 准确的判断失败点前序点较困难, 不恰当的返回点会增加车辆线路数和配送总成本, 占用车辆和人工, 一般多通过对比各重调度方案(选择不同客户点作为返回点所得方案)总体期望成本确定返回点^[32].

3.3.2 点重优化策略(Re-optimization strategy)

鉴于失败点返回策略在重调度路径成本上有继续优化的空间, 且前序点返回策略在点选择时要求较高, 确定最合适的返回点比较困难, 本文在第二阶段点重调度时提出一种新的点重优化策略, 考虑所有线路中的失败点及后续客户点的随机需求及位置信息, 对其

线路进行重新调整优化, 以减少以往点返回策略中对配送车辆及人员的占用, 缩短路径总长度. 点重优化策略在执行时需首先确认待重调度客户点集合 V_R ($V_R = \{1, 2, \dots, r\}$, $V_R \in V_o$), V_R 由所有预优化线路中的失败点及其子线路后续客户点组成, 通过对预优化线路中客户点需求进行逐点随机模拟, 可确认线路失败点. 出现失败点后, 配送车辆应对失败点进行部分服务后直接返回配送中心, 此时失败点及其前序点需求已确认, 后续点的需求仍为随机, 未完成服务, 若 V_R 为空集, 说明配送车辆已按预优化线路完成所有客户点的服务, 无须执行第二阶段重调度策略.

确认 V_R 不为空集后, 基于节约算法原理本文设计了插入法对 V_R 执行点重优化策略, 由于失败点后续点需求随机(当 V_R 仅存在失败点时, 排除客户点随机需求影响后本文点重优化策略同样有效), 因此点重优化阶段求解的同样是CVRPSD, 与预优化阶段相同, 点重优化时线路中客户点总随机需求须满足容量机会约束 $\bar{Q}$ 限制. 值得注意的是, 点重优化线路仍可能出现线路失败, 对该情况算法执行失败点返回策略, 由于重优化时 V_R 点规模相对较小, 配送车辆载重量较大, 重优化线路中失败点出现概率极小.

3.4 时间复杂性分析(Computational complexity)

对于VNSS而言, 算例客户点规模为 n , 种群规模为 p_{size} , 算法迭代次数为 N , 变邻域搜索策略迭代次数为 M , 点重优化策略中客户点规模为 n/k , $k \in (0, 1)$, k 为常数. 依据图1的算法流程, 对各环节 i 时间复杂度 $O(T_i)$ 分析如下: (1)初始种群及其多样性优化 $O(T_1) = n^2 \cdot p_{size}$; (2)参考集更新 $O(T_2) = N \cdot n \cdot p_{size}$; (3)子集对组合策略 $O(T_3) = N \cdot n \cdot p_{size}$; (4)变邻域搜索策略包含三种邻域结构变化, $O(T_4) = 3N \cdot M \cdot n \cdot p_{size}$; (5)预优化阶段目标函数值计算 $O(T_5) = (1 + N + 3N \cdot M) \cdot n \cdot p_{size} = O(N \cdot M \cdot n \cdot p_{size})$; (6)点重优化策略 $O(T_6) = N \cdot \frac{n^2}{k} \cdot p_{size}$; (7)重调度阶段目标函数值计算 $O(T_7) = N \cdot n \cdot p_{size}$. 其中(1)至(5)为预优化阶段, (6)、(7)为重调度阶段, 综上, 算法的总计算时间复杂度可估算为

$$O(T) = (N \cdot \frac{n^2}{k} + n^2 + 4N \cdot M \cdot n + 3N \cdot n) \cdot p_{size}$$

$$= O((N \cdot n^2 + N \cdot M \cdot n) \cdot p_{size}).$$

4 算例验证及结果分析(Experimental results and comparisons)

为验证VNSS的有效性, 本文选取了确定型CVRP和CVRPSD两类算例集进行测试, 确定型CVRP算例集包括文献[3]中简单算例, 文献[33]中所有Set A、Set P测试算例以及Eilon test set中不超过60个客户的三个Set E算例, CVRPSD算例集由上述确定型CVRP算例集(不包括文献[3]中简单算例)改进而来. 改进过程如下: (1)假设客户点随机需求服从泊松分布; (2) CVRPSD算例中客户点随机需求的期望值与其确定型CVRP算例对应的客户点需求相同; (3)客户点及配送中心间距离、位置、车辆最大装载量等信息不变, 仅需求做随机改进.

文中算法编程工具采用MATLAB r2010b, 操作系统为Window7, 电脑内存4.00GB, CPU为双核酷睿i5-4460, 主频3.2GHz. 经过反复测试, 设置算法参数如下: 算法迭代次数 $N = 10 \sim 50$, 惩罚成本 $p_c = 200 \sim 1000$, 变邻域策略停止准则中最大迭代次数为 $M = 10 \sim 40$, 初始种群中个体数 $p_{size} = 30 \sim 100$. N 、 M 以及 p_{size} 与对应算例中客户点规模 n 相关, 当 $n \in [20, 40]$ 时, p_{size} 多设置为40~60, N 和 M 设置为10~20; $n \in (40, 50]$ 时, p_{size} 相应增加至50~70, N 和 M 相应增加至20左右, 不超过30; $n \in (50, 60]$ 时, p_{size} 增加至60~80, N 增加至20~40, M 设置为30左右, 客户点规模越大, 算法参数设置相应越大(针对个别算例可略增加相关参数值), 惩罚费用大小与路径总成本相关.

4.1 确定型CVRP算例验证(Experimental results of deterministic CVRP)

首先选取文献[3]中简单算例, 客户点规模为8, 配送车辆2台, 表1给出了文献[2-3, 5-6]、VNSS及点重优化策略中插入算法的计算结果, 算例计算20次, 可见本文算法较其余算法搜索效率更高, 能稳定搜索到最优解, 成功率达100%, 本文点重优化策略所采用的插入算法在面对客户点规模为8的优化问题时稳定有效, 可求得算例最优解. 此外表1中对比算法均未给出求解时间, 由于客户点较少, 本文算法能在极短时间内(5s)求得最优解.

文献[3]中算例规模较小, 另选取文献[33]中40个确定型CVRP算例组成测试集, 算例计算20次, 算例中具体数据可通过<http://www.bernabe.dorronsoro.es/vrp/>获得. 表2中列举了文献[33]中Set A部分19个确定型算例, 包括文献[7-9]和本文算法的计算结果, 其中BK表示算例已知最优解^[9], Best、Worst、Average表示算法最优值、最差值和平均值, %Dev表示与已

知最优解偏差($\%Dev = (Best - BK)/BK$), CPU为算法运行时间, 单位为秒.

表 1 文献[3]算例结果比较

Table 1 The result comparison of a simple case in Ref. [3]

算法	最优值	最差值	平均值	求得最优解比例/%
捕食搜索算法PSA ^[2]	67.5*	69	67.65	90
量子进化算法QEA ^[3]	67.5*	71	68.358	54
混合遗传算法HGA ^[5]	67.5*	69	67.875	75
改进粒子群算法IPSO ^[6]	67.5*	70.5	68.425	50
混合变邻域分散搜索算法VNSS	67.5*	67.5*	67.5*	100
点重优化插入法	67.5*	67.5*	67.5*	100

注: *表示已知最优解

由表2 Set A中算例结果对比可知, 对比文献[7]中的算法LNS-ACO性能较好, 除A-n45-k6、A-n48-k7外能求得17/19算例已知最优解, 文献[8]中SC-ESA次之(15/19), 文献[9]RRA未求得算例的已知最优解(0/19), 算法性能一般. 在算法运行时间上LNS-ACO平均耗时达1677s, 时间过长, SC-ESA平均耗时411s, RRA虽耗时较少, 但其求解精度较差. 本文所设计VNSS能求得19个算例的全部最优解, 且平均求解耗时较少为302s, 较其它算法相比性能优势明显, 可见其优化质量最高, 搜索稳定, 寻优能力较强.

在表2中A部分算例对比基础上, 表3进一步给出了文献[33]中Set E、Set P部分21个确定型算例, 包括文献[4, 7, 9]和本文算法的计算结果, 表中符号含义同表2, “—”表明相应文献未求解该算例.

表3中共有E、P算例21个, 可知文献[7]求得15/19个算例最优解, 非最优解平均偏差0.27%, 算法平均耗时1794s; 文献[9]求得7/18个算例最优解, 平均偏差0.45%, 算法耗时302s; 文献[4]求得13/18个算例最优解, 平均偏差0.1%, 文中未给出其算法时间数据; 本文算法求得15/21个算例最优解, 平均偏差0.22%, 算法平均耗时最少, 平均为257s; . 除文献[9]中算法外, LNS-ACO、NHQEA及本文VNSS算法均能求得70%以上算例最优解, 且未求得最优的算例解与最优解平均偏差均小于0.5%, 算法性能较优.

因此由表2、3中对比结果可知, 本文所设计VNSS是求解CVRP中Set A、E、P部分确定型算例的有效算法, 较对比文献中算法耗时少, 效率高, 且性能稳定, 在应用于CVRPSD算例求解时能保证求得高质量的预优化解, 进而降低第二阶段重调度路径成本.

表 2 文献[33] Set A 确定型算例集结果比较

Table 2 The results comparison of set A cases in Ref. [33]

算例	BK	LNS-ACO ^[7]			RRA ^[8]			SC-ESA ^[9]			VNSS				
		Best	%Dev	CPU	Best	%Dev	CPU	Best	%Dev	CPU	Best	Worst	Average	%Dev	CPU
A-n32-k5	784*	784*	0	856	787.1	0.40	24	784*	0	311	784*	784*	784*	0	117
A-n33-k5	661*	661*	0	900	662.1	0.17	34	661*	0	257	661*	661*	661*	0	146
A-n33-k6	742*	742*	0	948	742.6	0.08	28	742*	0	278	742*	742*	742*	0	121
A-n34-k5	778*	778*	0	909	780.9	0.37	41	778*	0	264	778*	778*	778*	0	138
A-n36-k5	799*	799*	0	1056	802.1	0.39	42	799*	0	290	799*	799*	799*	0	152
A-n37-k5	669*	669*	0	1104	672.5	0.52	27	669*	0	308	669*	669*	669*	0	146
A-n37-k6	949*	949*	0	1113	950.9	0.20	49	949*	0	271	949*	949*	949*	0	149
A-n38-k5	730*	730*	0	1139	734.2	0.58	50	730*	0	403	730*	730*	730*	0	186
A-n39-k5	822*	822*	0	1203	828.9	0.84	34	822*	0	202	822*	822*	822*	0	254
A-n39-k6	831*	831*	0	1266	835.3	0.52	46	831*	0	333	831*	831*	831*	0	290
A-n44-k6	937*	937*	0	1568	938.2	0.13	52	937*	0	390	937*	937*	937*	0	389
A-n45-k6	944*	958	1.48	1728	956	1.27	75	944*	0	611	944*	953	952.1	0	352
A-n45-k7	1146*	1146*	0	1805	1146.9	0.08	76	1146*	0	353	1146*	1146*	1146*	0	365
A-n46-k7	914*	914*	0	1979	917.7	0.40	60	914*	0	401	914*	914*	914*	0	386
A-n48-k7	1073*	1084.1	1.03	2324	1074.3	0.12	79	1084.1	1.03	370	1073*	1073*	1073*	0	410
A-n53-k7	1010*	1010*	0	2497	1020.2	1.01	79	1011	0.10	527	1010*	1017	1016.7	0	488
A-n54-k7	1167*	1167*	0	2771	1171.7	0.40	98	1168	0.09	535	1167*	1168	1167.3	0	479
A-n55-k9	1073*	1073*	0	3345	1074.9	0.18	120	1073*	0	875	1073*	1073*	1073*	0	532
A-n60-k9	1354*	1354*	0	3356	1355.9	0.14	99	1354.9	0.07	834	1354*	1360	1356.7	0	642
Average			0.13	1677		0.41	59		0.07	411				0	302

注: *表示已知最优解

表 3 文献[33] Set E、P 确定型算例集结果比较

Table 3 The results comparison of set E, P cases in Ref. [33]

算例	BK	LNS-ACO ^[7]			SC-ESA ^[9]			NHQEA ^[4]			VNSS			
		Best	%Dev	CPU	Best	%Dev	CPU	Best	%Dev	Best	Worst	Average	%Dev	CPU
E-n22-k4	375*	375*	0	447	375*	0	48	-	-	375*	375*	375*	0	33
E-n33-k4	835*	835*	0	818	839	0.48	199	-	-	835*	835*	835*	0	135
E-n51-k5	521*	-	-	-	-	-	-	-	-	521*	521*	521*	0	323
P-n16-k8	450*	450*	0	737	450*	0	2	450*	0	450*	450*	450*	0	5
P-n19-k2	212*	212*	0	364	219	3.30	2	212*	0	212*	212*	212*	0	6
P-n20-k2	216*	216*	0	353	218	0.93	6	216*	0	216*	216*	216*	0	6
P-n21-k2	211*	211*	0	400	212	0.47	6	211*	0	211*	211*	211*	0	7
P-n22-k2	216*	216*	0	413	216*	0	21	216*	0	216*	216*	216*	0	7
P-n22-k8	603*	-	-	-	-	-	-	603*	0	603*	603*	603*	0	10
P-n23-k8	529*	529*	0	616	529*	0	37	529*	0	529*	529*	529*	0	8
P-n40-k5	458*	458*	0	1228	459	0.22	109	458*	0	458*	458*	458*	0	88
P-n45-k5	510*	510*	0	1569	511	0.20	144	510*	0	510*	510*	510*	0	147
P-n50-k10	696*	696*	0	2348	697	0.14	469	697	0.14	697	697	697	0.14	481
P-n50-k7	554*	554*	0	2026	554*	0	430	554*	0	554*	554*	554*	0	523
P-n50-k8	631*	643	1.9	2142	637	0.95	452	631*	0	641	641	641	1.6	365
P-n51-k10	741*	747	0.81	2449	741*	0	494	741*	0	746	746	746	0.67	370
P-n55-k10	694*	694*	0	2877	695	0.14	653	695	0.14	698	700	698.9	0.58	589
P-n55-k15	989*	989*	0	3054	-	-	-	999	1.01	989*	999	994.5	0	564
P-n55-k7	568*	568*	0	2533	574	1.06	429	568*	0	568*	570	568.7	0	279
P-n60-k10	744*	755	1.48	3393	745	0.13	664	745	0.13	749	750	749.3	0.67	430
P-n60-k15	968*	977	0.93	3997	968*	0	916	971	0.31	977	977	977	0.93	681
Average			0.27	1794		0.45	302		0.1				0.22	257

注: *表示已知最优解

4.2 CVRPSD算例验证(Experimental results of CVRPSD)

在CVRP算例分析基础上,表4对比了多个文献的CVRPSD算例集求解结果,该集合由文献[33]改进

的CVRPSD算例组成,随机需求服从泊松分布, Best表示算法求解10次所得最优值, %Dev表示与相关最优值的偏差, α 为本文算法求得最优值时所取风险偏好值, CPU为运行时间。

表 4 文献[33] CVRPSD算例集结果比较

Table 4 The results comparison of CVRPSD cases in Ref. [33]

算例	SA ^[13]			PSO ^[17]		ALNS ₃ ^[18]		CNTGSO ^[19]		VNSS			
	Best	%Dev	CPU	Best	%Dev	Best	%Dev	Best	%Dev	Best	%Dev	α	CPU
A-n32-k5	853.6	6.71	199.8	821.65	2.72	830.67	3.84	807.61	0.96	799.93*	0	0.7	132
A-n33-k5	704.2	3.80	178.2	687.04	1.27	693.92	2.29	678.41*	0	683.09	0.69	0.7	155
A-n33-k6	793.9	4.23	141.1	769.62	1.05	771.2	1.25	763.54	0.25	761.66*	0	0.7	115
A-n34-k5	826.87	4.68	236.4	789.88*	0	813.85	3.03	797.49	0.96	790.54	0.08	0.7	151
A-n36-k5	858.71	3.33	276.1	836.05	0.61	845.57	1.75	831*	0	836.61	0.68	0.7	142
A-n37-k5	708.34	3.30	386.9	693.18	1.09	695.04	1.37	702.2	2.41	685.68*	0	0.5	149
A-n37-k6	1030.73	4.38	205.5	999.72	1.24	987.46*	0	991.4	0.40	995.75	0.84	0.8	161
A-n38-k5	775.14	3.65	313.4	756.56	1.17	753.8	0.80	747.84*	0	755.9	1.08	0.7	179
A-n39-k5	869.18	3.42	257.6	853.08	1.50	867.27	3.19	856.13	1.86	840.47*	0	0.7	236
A-n39-k6	876.6	4.27	239.9	847.92	0.86	874.65	4.04	858.32	2.09	840.71*	0	0.6	321
A-n44-k6	1025.48	6.32	281.4	978.82	1.48	990.09	2.65	979.41	1.54	964.53*	0	0.5	372
A-n45-k6	1026.73	4.03	301.4	997.41	1.06	1015.21	2.86	1001.2	1.44	986.97*	0	0.7	364
A-n45-k7	1264.99	7.62	216.0	1175.45*	0	1207.08	2.69	1211.5	3.07	1197.74	1.90	0.7	341
A-n46-k7	1002.22	2.33	314.1	984.98	0.57	1001.85	2.29	984.37	0.51	979.42*	0	0.8	389
A-n48-k7	1187.14	5.43	292.0	1132.15	0.55	1163.51	3.33	1146.6	1.83	1125.98*	0	0.7	446
A-n53-k7	1124.27	4.06	468.8	1096.6	1.50	1093.54	1.21	1100.3	1.84	1080.42*	0	0.7	538
A-n54-k7	1287.07	5.22	409.4	1223.23*	0	1281.95	4.80	1250.4	2.22	1246.4	1.89	0.8	459
A-n55-k9	1179.11	4.88	265.5	1124.3*	0	1138.7	1.28	1139.3	1.33	1132.2	0.70	0.8	547
A-n60-k9	1529.82	6.58	393.7	1454.15	1.31	1453.7	1.28	1470.5	2.45	1435.34*	0	0.7	632
E-n22-k4	411.57	9.66	104.1	390.99	4.18	398.81	6.26	375.53	0.06	375.32*	0	0.7	29
E-n33-k4	850.27	1.24	371.7	847.38	0.89	901.77	7.37	849.58	1.15	839.88*	0	0.7	139
E-n51-k5	552.26	1.78	586.0	544.86	0.42	560.05	3.22	550.85	1.52	542.6*	0	0.7	347
P-n16-k8	512.82	15.13	9.0	455.21	2.20	449.27	0.86	445.43*	0	446.24	0.18	0.7	6
P-n19-k2	224.06	7.09	234.8	213.51	2.05	212	1.32	209.23*	0	220.59	5.43	0.6	5
P-n20-k2	233.05	3.61	269.6	226.79	0.83	226.54	0.72	224.93*	0	225.16	0.10	0.7	6
P-n21-k2	218.96	2.87	332.7	218.13	2.48	220.66	3.67	218.55	2.68	212.85*	0	0.7	8
P-n22-k2	231.26	2.35	352.3	229.45	1.55	230.89	2.19	229.39	1.53	225.94*	0	0.7	7
P-n22-k8	681.06	16.02	28.6	590.72	0.63	611.96	4.25	587.03*	0	589.41	0.41	0.7	11
P-n23-k8	619.53	15.51	21.1	536.34*	0	545.58	1.72	540.55	0.78	566.01	5.53	0.8	8
P-n40-k5	472.5	1.51	367.2	471.24	1.24	488.76	5.00	471.75	1.35	465.48*	0	0.7	74
P-n45-k5	533.52	2.23	603.8	530.52	1.66	533.65	2.26	526.43	0.87	521.88*	0	0.7	157
P-n50-k10	760.94	4.43	150.6	739.51	1.49	732.48	0.52	728.67*	0	745.26	2.28	0.7	473
P-n50-k7	582.37	2.11	343.0	570.94	0.11	587.11	2.94	572.65	0.41	570.33*	0	0.8	528
P-n50-k8	669.81	2.18	225.3	659.19	0.56	673.37	2.72	663.35	1.19	655.55*	0	0.8	353
P-n51-k10	812.74	4.27	159.9	795.43	2.05	786.72	0.94	788.27	1.13	779.43*	0	0.6	381
P-n55-k10	745.7	4.73	208.9	737.87	3.63	735.36	3.28	712.03*	0	737.53	3.58	0.6	583
P-n55-k15	1068.05	9.38	117.0	1008.6	3.30	989.93	1.38	976.42*	0	1036.16	6.12	0.7	577
P-n55-k7	588.56	1.29	452.8	587.95	1.19	610.58	5.08	588.17	1.23	581.05*	0	0.5	262
P-n60-k10	804.24	4.06	296.1	772.86*	0	803.27	3.93	785.96	1.70	801.11	3.66	0.7	417
P-n60-k15	1087.41	7.34	134.8	1021.58	0.84	1013.09*	0	1042.9	2.94	1048.89	3.53	0.7	679
Average		5.18	268.7		1.23		2.59		1.09		0.97		272

注: *表示已知最优解

文献[33]中共有CVRPSD算例40个, VNSS寻优能力最强, 求得了其中22/40个算例的新最优解, 平均偏差0.97%, CNTGSO求得10/40个算例的最优解, 平均偏差1.09%, ALNS₃求得2/40个算例的最优解, 平均偏差2.59%, PSO求得6/40个算例的最优解, 平均偏差1.23%, SA未求得算例最优解, 平均偏差5.18%. 在算法求解时间方面, 对比文献[17–19] (PSO、ALNS₃、CNTGSO)未给出其CPU, SA平均求解时间为268.7s, VNSS平均运行时间为272s, VNSS与SA在求解时间上相当, 但VNSS求解精度明显高于SA.

可见本文所设计算法性能稳定且求解CVRPSD能力较强, 在两阶段求解时解方案平均偏差较小, 新的点重优化策略较点返回策略能更合理的安排车辆路径, 满足客户点的随机需求. 此外本文VNSS求得CVRPSD算例最优方案时 α 值多取0.7或0.8(α 与 $\bar{Q}$ 取值直接相关), 即 α 值取0.7~0.8时配送车辆先预留相应容量再规划路线效果较好, 由此可见确定型CVRP算例最优路径并不一定是与其对应的CVRPSD算例最好预优化路径, 预优化解方案的质量受 α 影响较大. 在算法复杂度方面, VNSS所对比的文献中[7–8, 17, 19]未给出相关分析, 文献[4, 9, 13, 18]中相应算法单次迭代的计算复杂度依次为 $O(n^2)$ 、 $O(n^2 \log n)$ 、 $O(n^2)$ 、 $O(n)$, 总体而言VNSS算法整体的计算复杂度与大部分启发式算法相当.

5 结论(Conclusions)

本文对客户点需求随机的不确定型车辆路径问题进行了研究, 构建了CVRPSD的随机机会约束模型, 对需求随机变量作机会约束确定型等价处理. 考虑到CVRPSD优化特点设计了两阶段的混合变邻域分散搜索算法(VNSS)求解该问题, 在第一阶段预优化基础上提出了新的点重优化策略, 对所有失败点及其后续客户点进行全局调整, 采用Set A、E、P(CVRP)及其随机改进型(CVRPSD)标准算例进行仿真实验. 算例验证表明, 较诸多对比算法本文所设计的两阶段VNSS性能较好, 算法效率更高, 点重优化策略整体优于点返回策略, 能求得新的随机算例最优解, 求解稳定且平均误差小, 是解决随机需求车辆路径问题的有效方法. 当求解大规模客户算例时, VNSS耗时略长, 今后将进一步改进该算法以强化其大规模求解能力.

参考文献(References):

- [1] DROR M, TRUDEAU P. Vehicle Routing with Stochastic Demands: Properties and Solution Frameworks[J]. *Transportation Science*, 1989, 23(3): 166 – 176.
- [2] JIANG Zhongzhong, WANG Dingwei. Predatory search algorithm for vehicle routing problem[J]. *Computer Integrated Manufacturing Systems*, 2006, 12(11): 1899 – 1902.
(蒋忠中, 汪定伟. 车辆路径问题的捕食搜索算法研究[J]. 计算机集

成制造系统, 2006, 12(11): 1899 – 1902.)

- [3] ZHAO Yanwei, PENG Dianjun J, ZHANG Jingling. Quantum evolutionary algorithm for capacitated vehicle routing problem[J]. *Systems Engineering - Theory & Practice*, 2009, 29(2): 159 – 166.
(赵燕伟, 彭典军, 张景玲. 有能力约束车辆路径问题的量子进化算法[J]. 系统工程理论与实践, 2009, 29(2): 159 – 166.)
- [4] CAO Gaoli, HU Rong, QIAN Bin, et al. Effective hybrid quantum evolutionary algorithm for capacitated vehicle problem[J]. *Computer Integrated Manufacturing Systems*, 2015, 21(4): 1101 – 1113.
(曹高立, 胡蓉, 钱斌, 等. 一种有效混合量子进化算法求解带容量约束的车辆路径优化问题[J]. 计算机集成制造系统, 2015, 21(4): 1101 – 1113.)
- [5] JIANG Changhua, DAI Shugui, HU Youhua. Hybrid genetic algorithm for capacitated vehicle routing problem[J]. *Computer Integrated Manufacturing Systems*, 2007, 13(10): 2047 – 2052.
(姜昌华, 戴树贵, 胡幼华. 求解车辆路径问题的混合遗传算法[J]. 计算机集成制造系统, 2007, 13(10): 2047 – 2052.)
- [6] LUO Xianguo, SHI Hongbo. Improved Particle Swarm Optimization for Vehicle Routing Problem with Non-Full Load[J]. *Journal of East China University of Science and Technology (Natural Science Edition)*, 2006, 32(7): 767 – 771.
(罗先国, 侍洪波. 非满载车辆路径问题的改进粒子群优化算法[J]. 华东理工大学学报(自然科学版), 2006, 32(7): 767 – 771.)
- [7] AKPINAR S. Hybrid large neighbourhood search algorithm for capacitated vehicle routing problem[J]. *Expert Systems with Applications*, 2016, 61(11): 28 – 38.
- [8] BU Lixin, LUO Wenyu, FENG Yuncheng. Random recursion heuristics for the VRP[J]. *Systems Engineering - Theory & Practice*, 2008, 28(11): 142 – 148.
(步立新, 罗文钰, 冯允成. 随机递归算法求解车辆路径问题[J]. 系统工程理论与实践, 2008, 28(11): 142 – 148.)
- [9] STANOJEVIC M, STANOJEVIC B, VUJOSEVIC M. Enhanced savings calculation and its applications for solving capacitated vehicle routing problem[J]. *Applied Mathematics & Computation*, 2013, 219(20): 10302 – 10312.
- [10] GENDREAU M, LAPORTE G, SEGUIN R. Stochastic vehicle routing[J]. *European Journal of Operational Research*, 1996, 88(88): 3 – 12.
- [11] TILLMAN F A. The multiple terminal delivery problem with probabilistic demands[J]. *Transportation Science*, 1969, 3(3): 192 – 204.
- [12] PSARAFTIS H N. Dynamic vehicle routing: Status and prospects[J]. *Annals of Operations Research*, 1995, 61(1): 143 – 164.
- [13] GOODSON J C, OHLMANN J W, THOMAS B W. Cyclic-order neighborhoods with application to the vehicle routing problem with stochastic demand[J]. *European Journal of Operational Research*, 2012, 217(2): 312 – 323.
- [14] LAPORTE G, LOUVEAUX F V, HAMME L V. An Integer L-Shaped Algorithm for the Capacitated Vehicle Routing Problem with Stochastic Demands[J]. *Operations Research*, 2002, 50(3): 415 – 423.
- [15] GAUVIN C, DESAULNIERS G, GENDREAU M. A branch-cut-and-price algorithm for the vehicle routing problem with stochastic demands[J]. *Computers & Operations Research*, 2014, 50(10): 141 – 153.
- [16] DUAN Zhengyu, YANG Dongyuan, WANG Shang. Improved ant colony optimization algorithm for time-dependent vehicle routing problem[J]. *Control Theory & Applications*, 2010, 27(11): 1557 – 1563.
(段征宇, 杨东援, 王上. 时间依赖型车辆路径问题的一种改进蚁群算法[J]. 控制理论与应用, 2010, 27(11): 1557 – 1563.)
- [17] MARINAKIS Y, GEORGIA-ROUMBINI I, MAGDALENE M. Particle Swarm Optimization for the vehicle routing problem with stochastic demands [J]. *Applied Soft Computing*, 2013, 13(4): 1693 – 1704.

- [18] LUO Zhixing, QIN Hu, ZHANG Dezhi, et al. Adaptive large neighborhood search heuristics for the vehicle routing problem with stochastic demands and weight-related cost[J]. *Transportation Research Part E Logistics & Transportation Review*, 2016, 85(1): 69 – 89.
- [19] MARINAKI M, MARINAKIS Y. A Glowworm Swarm Optimization algorithm for the Vehicle Routing Problem with Stochastic Demands[J]. *Expert Systems with Applications*, 2016, 46(3): 145 – 163.
- [20] LOU Shanzuo, SHI Zhongke. Vehicle routing problem with stochastic customers and demands based on the cross-entropy method[J]. *Control & Decision*, 2007, 22(1): 7 – 10.
(娄山佐, 史忠科. 基于交叉熵法解决随机用户和需求车辆路径问题[J]. 控制与决策, 2007, 22(1): 7 – 10.)
- [21] GE Xianlong, WANG Xu, DAI Ying. The Vehicle Routing Problem in the Case of Stochastic Demand Based on Hybrid Quantum Genetic Algorithm[J]. *Systems Engineering*, 2011, 29(3): 53 – 59.
(葛显龙, 王旭, 代应. 基于混合量子遗传算法的随机需求车辆调度问题[J]. 系统工程, 2011(3): 53 – 59.)
- [22] LAW A M, KELTON W D. *Simulation modeling and analysis*[M]. McGraw-Hill, Singapore, 2000.
- [23] BRUCE L. G, JAMES R. Y. A framework for probabilistic vehicle routing[J]. *IIE Transactions*, 1979, 11(2): 109 – 112.
- [24] WILLIAM R S, BRUCE L. G. Stochastic vehicle routing: A comprehensive approach[J]. *European Journal of Operational Research*, 1983, 14(4): 371 – 385.
- [25] MARTI R, LAGUNA M, GLOVER F. Principles of scatter search[J]. *European Journal of Operational Research*, 2006, 169(2): 359 – 372.
- [26] HANSEN P, MLADENOVIC N, PEREZ J A M. Variable neighborhood search[J]. *Computers & Operations Research*, 1997, 24(11): 1097 – 1100.
- [27] ZACHARIADIS E, TARANTILIS C, KIRANOUDIS C. A hybrid metaheuristic algorithm for the vehicle routing problem with simultaneous delivery and pick-up service[J]. *Expert Systems with Applications*, 2009, 36(2): 1070 – 1081.
- [28] PAESSENS H. The savings algorithm for the vehicle routing problem[J]. *European Journal of Operational Research*, 1988, 34(3): 336 – 344.
- [29] ZHANG Xiaonan, FAN Houming. Optimization and Real-Time Adjustment for Vehicle Routing Problem with Fuzzy Demand[J]. *Journal of Shanghai Jiao Tong University*, 2016, 50(1): 123 – 130.
(张晓楠, 范厚明. 模糊需求车辆路径优化及实时调整[J]. 上海交通大学学报, 2016, 50(1): 123 – 130.)
- [30] HANSEN P, MLADENOVIC N. Variable neighborhood search: Principles and applications[J]. *European Journal of Operational Research*. 2001, 130(3): 449 – 467.
- [31] GAUVIN C, DESAULNIERS G, GENDREAU M. A branch-cut-and-price algorithm for the vehicle routing problem with stochastic demands[J]. *Computers & Operations Research*, 2014, 50(10): 141 – 153.
- [32] YANG W H, MATHUR K, BALLOU R H. Stochastic Vehicle Routing Problem with Restocking[J]. *Transportation Science*, 2000, 34(1): 99 – 112.
- [33] CHRISTIANSEN C H, LYSGAARD J. A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands[J]. *Operations Research Letters*, 2007, 35(6): 773 – 781.

作者简介:

李 阳 (1990–), 男, 博士研究生, 主要研究方向为物流系统优化设计及智能算法应用研究, E-mail: liyang02030019@126.com;

范厚明 (1962–), 男, 教授, 博士生导师, 主要研究方向为交通运输系统规划与设计, 战略管理与系统规划等研究, E-mail: fhm468@163.com;

张晓楠 (1988–), 女, 讲师, 主要研究方向为物流系统优化设计, 智能优化算法, E-mail: WLxn2010@126.com;

杨 翔 (1980–), 男, 博士研究生, 主要研究方向为物流系统工程, 交通运输规划与管理, E-mail: zaiye0000@163.com.