

汽车装配线电动车配送路径及换电站选址优化^{*}

周炳海[†], 谭芬

(同济大学 机械与能源工程学院, 上海 201804)

摘要:考虑将电动小车用来进行基于厂内循环配送策略的汽车装配线的物料配送, 提出了汽车装配线电动车配送路径及换电站选址问题, 以最小化系统总成本为优化目标建立了数学规划模型. 针对这一复杂的混合优化问题, 对该问题的性质进行了分析, 提出了两阶段动态规划算法获取小规模问题的最优解; 对于中、大规模问题, 通过种群分割技术并在 Lévy 飞行中融入深度邻域搜索算子构建了改进型离散布谷鸟算法. 最后, 进行了仿真实验, 分别对比了两阶段动态规划算法, 实数遗传算法及改进人工蜂群算法在解决该问题方面的性能, 结果表明改进型离散布谷鸟算法的有效性以及在算法稳定性、搜索深度以及收敛性三个方面的较大优势.

关键词:厂内物料配送; 超市; 电动小车; 换电站; Lévy 飞行; 深度邻域搜索

中图分类号: TP391

文献标志码: A

Electric Vehicle Delivery Routing and Battery Swap Station Location Optimization for Automotive Assembly Lines

ZHOU Binghai[†], TAN Fen

(School of Mechanical Engineering, Tongji University, Shanghai 201804, China)

Abstract: Considering employing the electric vehicles to deliver parts to stations for automotive assembly lines based on in-plant milk-run delivery strategy, an electric vehicle delivery routing and battery swap station location problem was presented, and a mathematical programming model with an objective function of minimizing total cost of the system was set up. To tackle this complicated problem, the property was analyzed, and a two-phase dynamic programming method was adopted to obtain the global optimum for small scale problems. For medium and large scale problems, both the population decomposition strategy and the depth neighborhood search operator based on Lévy flight were applied to develop an improved discrete cuckoo search algorithm. Finally, through the comparison of the two-phase dynamic programming method, real genetic algorithm and modified artificial bee colony algorithm, the simulation experiments were carried out to illustrate the effectiveness and great advantages in stability, deep searching ability and convergence of the algorithm.

Key words: in-plant material delivery; supermarket; electric vehicles; battery swap station; Lévy flight; depth neighborhood search

^{*} 收稿日期: 2017-04-17

基金项目: 国家自然科学基金资助项目(71471135), National Natural Science Foundation of China(71471135)

作者简介: 周炳海(1965-), 男, 浙江浦江人, 同济大学教授, 博士生导师

[†] 通讯联系人, E-mail: bhzhou@tongji.edu.cn

随着经济的不断发展,能源危机和环境污染的问题日益凸显,发展绿色生产物流成为制造业发展的必然趋势^[1].电动小车(Electric Vehicle, EV)由于其清洁、节能等优点逐步被用来取代传统燃料小车进行厂外“最后一公里”以及厂内的物料配送^[2].如何采用EV进行汽车混流装配线的物料配送对制造企业的节能减排、降成本增效益具有重要意义.

近年来,许多国内外学者对汽车混流装配系统中的物料准时化配送调度问题进行了较深入的研究. Emde等^[3]对基于物料超市的循环配送问题进行了建模分析,分别研究了路径和配送次数问题,并在多项式时间内求解;Fathi等^[4]由此构建了循环配送问题的多目标配送调度模型,并提出了融合优先级规则算法以优化配送次数和线边库存. Golz等^[5]以德国某汽车制造商为例,以最小化小车司机为目标研究了其路径、调度和装载问题.上述文献都是考虑传统小车的物料配送,而EV配送的不同点在于一是节能减排,二是由于其电量有限、可行里程较传统燃料车较短,需要考虑到电量的补给工作.

而关于EV配送的电动车辆路径问题(Electric Vehicle Routing Problem, EVRP)多是对厂外物料进行配送. Artmeier等^[6]第一次引进了EV并考虑小车的能耗成本,通过改进经典的最短路径法来研究“最优能耗”路径;Schneider等^[2]提出了带时间窗的可充电车辆路径问题,小车采用在客户点充电方式补给电量. Yang等^[7]采用更换电池方式补给电量,随之而来的便是换电站(Battery Swap Station, BSS)选址问题.目前有两种充电方式:直接充电和换电池.从效率和成本方面考虑,本文选择换电池的方式来补给电量.上述文献中每个客户仅被访问一次,不同于装配车间需要进行多次物料配送.

为此,本文在上述文献的基础上,提出了汽车装配线电动车配送路径优化及换电站选址优化问题,通过决策EV的配送路径及BSS位置来构建优化模型,提出了两阶段动态规划算法和改进型离散布谷鸟算法以最小化配送系统的总成本.

1 数学建模

1.1 问题描述

对于汽车混流装配车间的工位段 S ,物料超市负责暂存和分拣 S 所需的物料.考虑到车间突发情

况多,为了使得小车的柔性更高,EV由人员驾驶操作,根据配送列表将各工位所需的物料在需要的时间送达,实现对工位段 S 多频次、小批量的物料配送.装配线不允许缺货,EV的电量不允许降为0,即需在BSS对小车补给电量.在超市附近有一系列换电站BSSs的备选点,需要决策的就是在备选BSSs中选择合适的BSS进行换电池操作以及EV的配送路径使得系统的总成本最小.

为有效描述要研究的优化问题,作如下基本假设:1)系统选用节拍时间作为基本的时间单位;2)计划期内的需求已知;3)不考虑EV负载对电池耗电量的影响;4)EV采用循环配送策略进行物料配送;5)各EV的配送路线不允许交叉(no overlapping);6)EV可在各个备选换电站(Battery Swap Station, BSS)进行换电池操作,且在所负责工位段进行配送时不允许中断,即换电池操作只允许在到达第一个工位之前或离开最后一个工位之后进行;7)每辆EV最多进行一次换电池操作.

1.2 数学模型

为方便形式化描述,现定义符号,见表1~表4.

表1 集合符号及定义
Tab. 1 Notations and definitions of sets

集合符号	定义
S	待装配工位集合,下标为 s
$\{0\}$	物料超市
F	BSS备选点集合
K	EV的集合,下标为 k
S_k	小车 k 负责的配送工位集合
R_k	小车 k 访问的节点集合, $R_k = S_k \cup F \cup \{0\}$
R	所有节点集合, $R = S \cup F \cup \{0\}$

表2 决策变量符号及定义
Tab. 2 Notations and definitions of decision variables

决策变量符号	定义
x_k	小车 k 负责配送区域的最后一个工位,则 $S_k = (x_{k-1} + 1, \dots, x_k)$
y_{ibk}	二进制变量,若小车 k 的BSS位于节点 i 且在第 b 次配送时进行换电池则为1;否则,为0
z	二进制变量,当 $y_{ibk} = 1$ 时,若在离开超市后换电池则为1,在到达超市前则为0
n	电动小车的实际使用数量

表 3 问题参数符号及定义

Tab. 3 Notations and definitions of parameters

参数符号	定义
A	小车的装载容量
T	计划期(生产节拍总数)
D_s	工位 s 在计划期内的物料需求
d_{gh}	节点 g 和节点 h 之间的距离, $g, h \in R$
rt	超市中的物料补充时间
st	换电池操作时间
v	小车的速度
λ	单位距离成本
g	EV 的使用成本
f	BBS 的建设成本
Q	EV 的电池容量
q	电池的消耗率
P_k	小车 k 首次从超市出发时的电量

表 4 辅助变量符号及定义

Tab. 4 Notations and definitions of auxiliary variables

辅助变量符号	定义
u_{ghbk}	二进制变量,若小车 k 在第 b 次配送时从节点 g 直接行驶到节点 h 则为 1;否则,为 0
TD_k	不换电池时小车 k 负责配送区域的全程距离
DT_k	小车 k 的总配送时间
B_k	小车 k 所在路线的配送次数
ΔSD_k	小车 k 负责路线上换电池产生的距离增量
AB_{gbk}	小车 k 第 b 次配送到达节点 g 时的电量
LB_{gbk}	小车 k 第 b 次配送离开节点 g 时的电量

据上述问题描述、模型假设及符号定义,对汽车装配线电动车配送路径及换电站选址问题建模如下:

目标函数:

$$\text{Minimize } ng + f \cdot \sum_{k=1}^n \sum_{b=1}^{B_k} \sum_{j \in F} y_{jkb} + \lambda \cdot \sum_{k=1}^n (TD_k \cdot B_k + \Delta SD_k) \quad (1)$$

约束如下:

$$x_0 = 1 \quad (2)$$

$$x_{k-1} + 1 \leq x_k, \quad \forall k = 1, \dots, n \quad (3)$$

$$x_n = |S| \quad (4)$$

$$\left(\sum_{s \in S_k} D_s \right) / B_k \leq A, \quad \forall k = 1, \dots, n \quad (5)$$

$$\sum_{b=1}^{B_k} \sum_{j \in F} y_{jkb} \leq 1, \quad \forall k = 1, \dots, n \quad (6)$$

$$AB_{hbk} = LB_{gbk} - q \cdot d_{gh} \cdot u_{ghbk}, \quad \forall g, h \in R_k, k = 1, \dots, n, b = 1, \dots, B_k \quad (7)$$

$$LB_{gbk} = AB_{gbk} (1 - y_{gkb}) + Qy_{gkb}, \quad \forall g \in R_k \setminus \{0\}, k = 1, \dots, n, b = 1, \dots, B_k \quad (8)$$

$$LB_{01k} = P_k, \quad \forall k = 1, \dots, n \quad (9)$$

$$LB_{0bk} = AB_{0(b-1)k}, \quad \forall k = 1, \dots, n, b = 1, \dots, B_k \quad (10)$$

$$y_{jkb}, u_{ghbk} \in \{0, 1\} \quad \forall j, g, h \in R_k, k = 1, \dots, n, b = 1, \dots, B_k \quad (11)$$

$$n \leq |K| \quad (12)$$

其中:

$$TD_k = d_{0, x_{k-1}+1} + \sum_{g=x_{k-1}+1}^{x_k-1} d_{g, g+1} + d_{x_k, 0} \quad \forall k = 1, \dots, n \quad (13)$$

$$\Delta SD_k = z(d_{0j} + d_{j, x_{k-1}+1} - d_{0, x_{k-1}+1}) + (1-z)(d_{j0} + d_{x_k, j} - d_{x_k, 0}) \quad (14)$$

$$DT_k = T - \left(\sum_{b=1}^{B_k} \sum_{j \in F} y_{jkb} \right) (st + \Delta SD_k / v) \quad \forall k = 1, \dots, n \quad (15)$$

$$B_k = \lceil DT_k / \lceil TD_k / v + rt \rceil \rceil, \quad \forall k = 1, \dots, n \quad (16)$$

目标函数(1)表示最小化包括 BSS 建设成本、EV 使用成本及运输成本的系统总成本;约束(2)~(4)表示每辆 EV 至少负责配送 1 个工位并且配送区域不重复;约束(5)表示每次配送不能超过 EV 的容量;约束(6)表示每辆 EV 最多进行一次换电池操作;约束(7)表示 EV 到达某节点的电量;约束(8)表示 EV 离开某节点的电量,其中,当 EV 离开 BSS 换完电池后变为满电量;约束(9)表示 EV 首次从超市出发的电量为 P_k ;约束(10)表示 EV 从超市离开的电量等于前一次配送到达超市的电量;约束(11)~(12)表示变量的取值范围。

1.3 问题性质

为了进一步深入分析问题,针对上述构建的优化模型,给出了相关的引理、定理。

定义 1 由于电池可用里程数的限制,电动小车在配送时可能会出现“断点”(Breaking Point, BP). 若 $\gamma_k = (g, b)$ 满足下列条件,则称 γ_k 为小车 k 配送的 BP.

$$\{\gamma_k = (g, b) \mid AB_{gbk} < 0, g \in S_k \cup \{0\}, b \in B_k\}$$

它表示小车在第 b 次配送到达节点 g 时电池电量已耗尽。

推论 1 若 b^* 表示小车 k 负责配送区域的第一个 BP 对应的配送次数,则当 $b^* = 0$ 时,表明小车电池不需要进行换电池操作;当 $b^* > 0$ 时,表明需要换电池。

证 根据定义 1,确定小车 k 负责配送区域的第一个 BP,表示为 $\gamma_k^* = (g^*, b^*)$:

$$b^* = \{b \mid AB_{gb^*k} < 0, AB_{gb^*k} \geq 0, \forall b' < b^*, g \in S_k \cup \{0\}, b', b^* \in B_k\}$$

$$g^* = \{g \mid AB_{g^* b^* k} < 0, AB_{g^* b^* k} \geq 0,\$$

$$\forall g' > g^*, g', g^* \in S_k \cup \{0\}\}$$

若 $b^* = 0$, 即 $\forall b \in B_k, AB_{gbk} \geq 0$, 不需要换电池; 若 $b^* > 0$, 即 $\exists b \in B_k$, 使得 $AB_{gbk} < 0$, 电池电量耗尽, 需要换电池。

证毕

性质 1 当小车的配送区域和 BSS 确定时, 若小车 k 在第 \bar{b} 次配送进行换电池操作, 则当 $b_{\min} \leq b^*$ 时, \bar{b} 在范围 $[b_{\min}, b^*]$ 内均可行; 当 $b_{\min} > b^*$ 时, 该解为非可行解, b_{\min} 和 b^* 值的求解见证明。

证 如图 1 所示, 一方面, 小车必须在电池电量耗尽前换电池, 根据推论 1 对于 b^* 的定义, 求得 b^* 。



图 1 性质 1 示意图

Fig. 1 Illustration of property 1

另一方面, 由于假设(7), 每辆小车最多进行一次换电池操作, 因此, 即使提前进行, 也需保证后续电量足够使用, 由此确定 b_{\min} 的值. 若 Δ 表示小车在第 \bar{b} 次配送换完电池后剩余行驶的距离, 小车 k 的 BSS 在节点 \tilde{g} , 则由 $\Delta + q(B_k - \bar{b})TD_k \leq Q$ 可得:

$$\bar{b} \geq B_k - (Q/q - \Delta)/TD_k \quad \text{即} \quad b_{\min} = \lceil B_k - (Q/q - \Delta)/TD_k \rceil, \quad \text{其中,}$$

$$\Delta = \begin{cases} d_{g, xk-1} + \sum_{g=x_{k-1}+1}^{x_k-1} d_{g, g+1} + d_{x_k, 0} & \text{当 } z=1 \text{ 时} \\ d_{g0} & \text{当 } z=0 \text{ 时} \end{cases}$$

而当 $b_{\min} > b^*$ 时, 说明小车换一次电池并不能满足电量需求, 违反了假设(7), 因此为非可行解。

证毕

性质 2 若 t_{0k} 表示小车 k 最大单次配送时间, 如果 $\forall k=1, \dots, n, (TD_k/v + rt) \leq t_{0k}$, 则所有小车满足容量约束, 在其他约束都满足的情况下该解为可行解; 如果 $\exists k=1, \dots, n, (TD_k/v + rt) > t_{0k}$, 则小车 k 违反容量约束, 该解为非可行解。

证 根据约束(5)可得, 若要满足小车的容量约束, 小车 k 最大单次配送时间 t_{0k} 由下式计算:

$$t_{0k} = A / \left(\sum_{s=x_{k-1}+1}^{x_k} D_s / DT_k \right)$$

证毕

2 两阶段动态规划算法

由于各小车的配送路线不交叉, 配送距离和需求仅与当前小车负责的配送区域有关, 因此, 提出两阶段动态规划法 (Two-stage Dynamic Programming, TSDP) 来解决该问题: 第一步确定最优工位划分方案, 即解决 EV 配送路径的问题 (Dynamic Programming Routing, DPR); 第二步解决相应小车的 BSS 站点问题 (Dynamic Programming BSS, DPB)。

2.1 DPR 问题

在确定最优工位划分方案时不考虑在 BSS 站点换电池的问题, 故将决策过程划分成 $|S| + 1 (0, \dots, |S|)$ 个阶段, 其中 0 为虚拟的开始阶段. 每个阶段由状态 i 表示, 表示小车负责配送区域的最后一个工位. 从状态 i 到状态 j 即表示小车负责的工位为 $i+1, \dots, j$, 其状态转移方程为:

$$f(i, j) = g + TD_{i+1, j} \cdot \lceil T / (TD_{i+1, j} / v + rt) \rceil \lambda \quad (17)$$

它表示状态转移即增加一辆小车时对应的目标函数值的变化, 其中 $TD_{i+1, j} = (d_{0, j+1} + \sum_{k=i+1}^{j-1} d_{k, k+1} + d_{j0})$ 。

对于由状态 i 转移至状态 j 的过程, 既要考虑小车要满足容量约束, 又要保证换电池后的电量足够使用, 因此, 根据性质 1、性质 2 分别可得出 $TD_{i+1, j}$ 应满足式(18)和式(19):

$$TD_{i+1, j} \cdot \lceil T / (TD_{i+1, j} / v + rt) \rceil \leq (Q + P_k) / q \quad (18)$$

$$TD_k / v + rt = A / \left(\left(\sum_{s \in S_k} D_s \right) / T \right) \quad (19)$$

$G(j)$ 表示工位 0 到工位 j 对应的最优目标函数值并且 $G(0) = 0$, 则:

$$G(j) = \min_{0 \leq i \leq j-1} \{G(i) + f(i, j)\} \quad (20)$$

结合式(17)至式(20)可得 DPR 问题的最优解 $G(|S|)$ 及其对应的小车数 n 。

2.2 DPB 问题

根据上一步求得的 EV 数, 将决策过程划分为 $n+1$ 个阶段, 每个阶段 k 表示每个小车, 其中阶段 0 表示虚拟的开始阶段. 每个阶段由状态 bs_x^k 表示, 表示阶段 k 所在的 BSS 站点为 x , 其中 $bs_x^k = 0$ 表明该阶段电量充足不需要进行换电池操作. 因此在每个

阶段开始之前根据推论 1 计算 b^* , 若 $b^* = 0$, 则 $bs_x^k = 0$. 同样, 需要保证换电池后的电量足够使用, 根据性质 1 则有 $(T - st - \Delta/v)/(TD_k/v + rt)TD_k + \Delta_x \leq (P_k + Q)/q$ 可得到阶段 k 的所有可能状态的集合, 记为 $\Theta(bs_x^k)$

$$\Theta(bs_x^k) \in \{x | \Delta_x \leq (P_k + Q)(TD/(rt \cdot v) + 1)/q - TD(T - st)/rt, x \in F\} \quad (21)$$

从状态 bs_x^k 转移到状态 bs_y^{k+1} 时相应的目标函数值的变化为:

$$\omega(bs_x^k, bs_y^{k+1}) = (T - st - \Delta/v)/(TD_k/v + rt) \cdot TD_k + f \cdot \xi + g + \Delta_y, bs_y^{k+1} \in \Theta(bs_x^k) \quad (22)$$

其中, 当 y 在前 k 阶段对应的 BSS 站点中出现过, 则 $\xi = 0$; 否则, $\xi = 1$. 如果 $H(bs_y^k)$ 表示直到阶段 k 产生的最小总成本且 $H(bs_0^0) = 0$, 则有:

$$H(bs_y^{k+1}) = \min_{bs_y^{k+1} \in \Theta(bs_x^k)} \{H(bs_x^k) + \omega(bs_x^k, bs_y^{k+1})\} \quad (23)$$

完成 n 个阶段的决策之后可获得该问题的最优目标函数值 $H(bs_y^n)$. 对于 TSDP 算法的时间复杂度, 在第一步解决 DPR 问题时, 共有 $|S| + 1$ 个节点, 每个节点相连的节点最多 $|S|$ 个, 则每次最多有 $|S|^2$ 相连的弧线需要计算 $|S|$ 次, 因此, 第一步的复杂度为 $O(|S|^3)$; 在第二步解决 DPB 问题时, 共有 $n + 1$ 个阶段, 即需要决策 n 次, 每次最多有 F 个 BSS 站点可选, 由于第一阶段只是确定了最优的小车数, 则有 $C_{|S|-1}^n$ 种方案, 因此, 第二步的复杂度为 $O(C_{|S|-1}^n F^n)$, 所以, TSDP 算法的时间复杂度为 $O(|S|^3 + C_{|S|-1}^n F^n)$, 随着问题规模 $|S|$ 的扩大, TSDP 算法所需的计算时间急剧增加, 算法求解效率降低, 故需构建启发式算法来求解中、大规模的问题.

3 改进型离散布谷鸟算法

布谷鸟搜索算法 (Cuckoo Search, CS) 是由 Yang 等^[8] 在 2009 年提出的一种新兴的启发式群体智能优化算法. 该算法基于布谷鸟寄生育雏的习性, 并结合一些鸟类和果蝇 Lévy 飞行的行为. CS 算法结构较简单、控制参数较少, 且 Lévy 飞行的高随机性增强了算法探索解空间的性能. 研究表明 CS 算法比粒子群算法 (PSO)、遗传算法 (GA) 以及人工蜂群算法 (ABC) 等群体智能算法求解效率更高, 目前已成功应用于多种工程优化问题的研究. 因此, 本

文针对研究问题的特性提出改进型离散型布谷鸟算法 (Improved Discrete Cuckoo Search, IDCS).

本文实行种群分割技术, 根据各鸟巢适应度的不同将布谷鸟分为三大种群, 不同子群体中的鸟巢采用不同的搜索方法, 从而保持种群搜索的多样性. 对前 P_d 部分的优秀布谷鸟 NP_{best} 进行精英保留策略^[9]; 对前 P_c 部分的较优布谷鸟 NP_{better} 在 Lévy 飞行中融入深度邻域搜索算子来获取新解, 提高 CS 算法局部进行精细搜索的能力, 提高收敛速度^[10]; 另外, 对于种群中 P_a 部分的较差个体 NP_{worse} 采取重建技术, 利用其随机扰动性引导算法跳出局部最优.

3.1 鸟巢编码方式

鸟巢 $\bar{\omega}_1, \dots, \bar{\omega}_{N_{pop}}$ 采用变长三层混合编码方式, N_{pop} 为鸟巢的数量, 也即种群规模的大小, 编码长度表示该优化方案所采用的 EV 总数. 编码的第一层为分区层, 表示各 EV 负责配送区域的最后一个工位; 编码的第二、三层为 BSS 层, 第二层表示对应 EV 换电池的 BSS 所在备选点, 其中 0 表示不需要换电池; 第三层表示 EV 何时进行换电池操作, 1 表示 EV 在离开超市后换电池, 0 表示 EV 在到达超市前进行换电池. 假设有 $|S| = 12$ 个工位、 $|F| = 5$ 个 BSS 备选点, 如图 2 所示, 给出了两鸟巢 $\bar{\omega}_1, \bar{\omega}_2$ 的编码示意图.

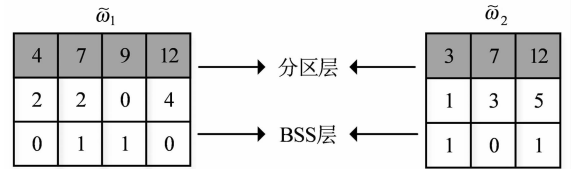


图 2 鸟巢编码示意图

Fig. 2 The schematic diagram of encoded mode for nest

3.2 鸟巢种群初始化

$j\bar{\omega}_i^f$ 表示鸟巢 $\bar{\omega}_i$ 第 f 层的第 j 列, ξ_i 表示鸟巢 $\bar{\omega}_i$ 的编码长度. 给定种群规模 N_{pop} , 对于编码的分区层, 结合约束 (2) ~ (4), 在满足条件 (12) 的情况下随机生成即 $j\bar{\omega}_i^1 \geq j^{-1}\bar{\omega}_i^1 + 1, {}^0\bar{\omega}_i^1 = 1, \xi_i\bar{\omega}_i^1 = |S|, \forall j = 1, \dots, \gamma_i$; 对于编码的 BSS 层, 根据推论 1, 确定 b^* 的值, 对于 $\forall j = 1, \dots, \gamma_i$, 若 $b_k^* = 0$, 则 $j\bar{\omega}_i^2 = 0, j\bar{\omega}_i^3 = 0$; 若 $b_k^* > 0$, 则 $j\bar{\omega}_i^2$ 和 $j\bar{\omega}_i^3$ 均随机产生.

3.3 适应度值计算

由于上述的编码过程没有考虑到小车的装载容量约束, 且可能存在违反约束 (6) 的情况, 考虑到非

可行解的存在,因此,向目标函数引入惩罚值. 对于一个鸟巢 $\bar{\omega}$, 其适应度值根据下式计算:

$$Z_{\text{penalized}} = Z + M \cdot \sum_{k=1}^n \{ \max \{ 0, TD_k - t_{0k} \} + \max \{ 0, b_k^* - b_{k\min} \} \} \quad (24)$$

其中, M 是一个非常大的正实数.

3.4 基于 Lévy 飞行的深度邻域搜索

CS 算法运用 Lévy 飞行的高随机性来进行全局搜索. 标准的 CS 算法是用来解决连续优化问题, 通过 Lévy 分布产生随机数来确定搜索的距离. 而在组合优化问题中, 解空间比较复杂, 解之间的距离也不能用数学形式很好地去刻画, 因此, 离散 CS 算法需要根据离散问题的特性来定义搜索的距离. 所以, 根据 Lévy 分布产生的随机数来确定搜索的步长或深度^[11]. 在本文提出的 IDCS 中, 在鸟巢之间增加信息交换即交叉操作以加快算法的收敛速度, 此外, 在 Lévy 飞行部分嵌入了结合问题特征的深度邻域搜索算子, 提高算法的精细搜索能力, 具体包括: 交叉算子(Crossover)、BSS 层交换算子(BSS-swap)、BSS 层插入算子(BSS-insert)、BSS 层反转算子(BSS-inverse).

3.4.1 Crossover 算子

Step1: 对于 NP_{better} 中的鸟巢 $\bar{\omega}_1$, 随机选取 NP_{best} 中的鸟巢 $\bar{\omega}_2$, 记 $Len1 = |\bar{\omega}_1|$, $Len2 = |\bar{\omega}_2|$, 最后交叉所得的鸟巢为 new ;

Step2: 若 $Len1 = Len2$, 则不执行交叉操作, 即 $new = \bar{\omega}_1$; 否则, 随机选取 $\bar{\omega}_1$ 、 $\bar{\omega}_2$ 的交叉点 A 和 B , 则交叉所得的新鸟巢 $new1$ 、 $new2$ 的长度分别为 $len1 = A + Len2 - B$, $len2 = B + Len1 - A$;

Step3: 若 $A^{+1}\bar{\omega}_1^1 > B\bar{\omega}_2^1$ & $B^{+1}\bar{\omega}_2^1 > A\bar{\omega}_1^1$ & $len1 \leq K$ & $len2 \leq K$, 则转 Step4; 否则, 转 Step5;

Step4: 对 $\bar{\omega}_1$ 、 $\bar{\omega}_2$ 分别在交叉点 A 和 B 处交叉, 得到新鸟巢 $new1$ 、 $new2$, 转 Step6;

Step5: 重新生成交叉点 B , 并转 Step3;

Step6: 比较 $new1$ 、 $new2$ 的适应度值 $Z_{\text{penalized}}$, 适应度值较小的鸟巢即鸟巢 new .

如图 3 所示, 对 $|S| = 30$ 的两个鸟巢进行交叉操作.

3.4.2 BSS 层算子

1) BSS-swap: 随机选择当前鸟巢中的两个位置, 交换其所在 BSS;

2) BSS-insert: 随机选择当前鸟巢中的一个位置, 重新安排其 BSS 所在位置, 其余 BSS 依次后移;

3) BSS-inverse: 随机选择当前鸟巢中的两个位

置, 将两个位置之间的 BSS 颠倒顺序.

邻域搜索的步长如下:

$$d = \min(\lceil 3 \times rk \rceil, 4) \quad (25)$$

其中, rk 为服从 Lévy 分布产生的随机数, d 决定了邻域搜索的操作方式, 若 $d = 1$, 则仅执行 Crossover 算子; 若 $d = 2$, 则执行 Crossover 算子 + BSS-swap 算子; 若 $d = 3$, 则执行 Crossover 算子 + BSS-insert 算子; 否则, 执行 Crossover 算子 + BSS-inverse 算子.

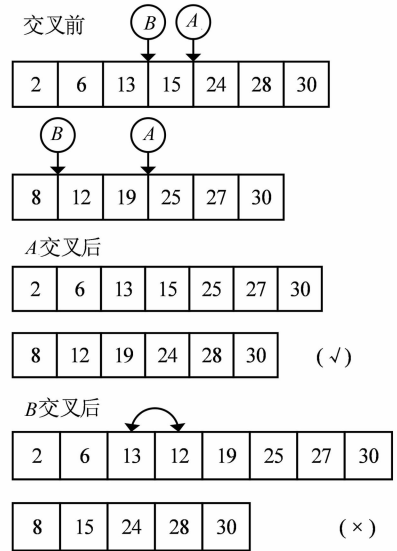


图 3 Crossover 算子的示意图
Fig. 3 Illustration of Crossover operator

3.5 重建鸟巢

在标准的 CS 算法中, n 个巢中有 P_a 部分将会被新的鸟巢取代, 这些鸟巢由于不够好, 所以很容易被宿主鸟发现从而无法保留至下一代. 因此, 需要重建鸟巢, 重建技术包括变异、合并、拆分算子, 使得鸟巢的长度分别为保持不变, 减 1, 加 1, 如图 4 所示.

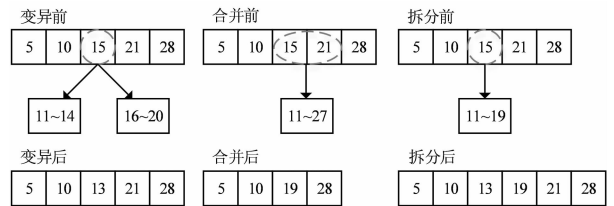


图 4 变异、合并、拆分算子示意图
Fig. 4 Illustration of mutation, combination and separation operator

3.6 IDCS 算法流程

Step 1: 初始化算法参数和问题参数;

Step 2: 构建初始种群;

Step 3: 根据各鸟巢适应度的不同将布谷鸟分为三大种群,实行种群分割策略;

Step 4: 针对优秀布谷鸟 NP_{best} 进行精英保留策略;

Step 5: 针对较优布谷鸟 NP_{better} , 结合基于 Lévy 飞行的深度邻域搜索操作进行精细搜索;

Step 6: 针对较差布谷鸟 NP_{worse} , 借助变异算子引导算法跳出局部最优;

Step 7: 若算法达到最大迭代次数 Ite_{max} 或连续 P_{max} 步没有更新最优解,则终止优化流程并输出当前最优解;否则,转 Step 3.

4 实验与分析

4.1 算例构建

参考文献[3]和[7]中的相关参数,考虑计划期 $T = 400$ 、EV 装载容量 $A = 10$ 的物料配送问题,各工位之间的距离为 6,超市中的物料补充时间 $rt = 1$,换电池操作时间 $st = 4$,小车速度为 25,,单位距离成本 $\lambda = 0.05$,EV 的使用成本和 BSS 的建设成本分别为 $\lceil \alpha \cdot d_{avra} \rceil$ 、 $\lceil \beta \cdot d_{avra} \rceil$,其中 α 和 β 为成本系数, d_{avra} 为 EV 负责配送区域的平均距离,电池容量 $Q = 100$,电池消耗率 $q = Q/(0.9d_{avra})$,EV 首次从超市出发时的电量 $P_k = \text{rand}(0.15, 0.35) \cdot Q$. 对于问题规模,根据文献[3]可知,每个超市至多负责 60 个工位.

本文的所有算法在主频为 2.50 GHz、内存 4 GB、Intel(R) Core(TM) i5-4200M CPU 的 PC 机上进行仿真实验,采用 Matlab(2014a)环境编程实现,实验结果分析如下.

4.2 算法参数设置分析

本文提出的 IDCS 算法包含 3 个重要参数,分别是种群规模 N_{pop} 、执行基于 Lévy 飞行的深度邻域搜索操作的种群比例 P_c 以及执行重建操作的种群比例 P_a . 为了使得 IDCS 算法的搜索性能最优,对中、规模问题进行正交试验,选用中等规模 $|S| = 30$ 的测试算例来说明试验过程. 每个因素包含四个水平,具体的试验参数配置如表 5 所示.

表 5 试验参数配置

Tab. 5 Combination of parameter values

算法参数	因素水平			
	1	2	3	4
N_{pop}	40	80	120	160
P_c	0.5	0.6	0.7	0.8
P_a	0.15	0.2	0.25	0.3

由于试验参数为三因素四水平,因此选用 $L_{16}(4^3)$ 的正交表,如表 6 所示. 对于每种试验方案, IDCS 算法运行 20 次,选用其平均值 \bar{C} 作为评价指标,试验结果的统计分析见表 7.

表 6 正交表和 \bar{C} 值统计

Tab. 6 Orthogonal table and \bar{C} values

试验号	因素			\bar{C}
	N_{pop}	P_c	P_a	
1	1	1	1	5 961.2
2	1	2	2	5 973.2
3	1	3	3	5 963.2
4	1	4	4	6 009.7
5	2	1	2	5 994.9
6	2	2	1	5 957.3
7	2	3	4	5 986.6
8	2	4	3	5 963.9
9	3	1	3	5 938.0
10	3	2	4	5 937.1
11	3	3	1	5 964.3
12	3	4	2	5 948.2
13	4	1	4	5 942.6
14	4	2	3	5 929.9
15	4	3	2	5 966.7
16	4	4	1	5 954.1

表 7 试验结果统计分析

Tab. 7 Statistical analysis of experimental results

因素水平	N_{pop}	P_c	P_a
1	5 976.825	5 959.175	5 959.225
2	5 975.675	5 949.375	5 970.75
3	5 946.9	5 970.2	5 948.75
3	5 948.325	5 968.975	5 969
极差 R	29.925	20.825	22
因素主次顺序	1	3	2
最优水平	3	2	3
最优组合	120	0.6	0.25

由表 7 可知,种群规模 N_{pop} 对 IDCS 算法性能影响最大, N_{pop} 值偏大将会导致较多的计算时间, N_{pop} 值偏小又会使得搜索不充分; P_c 和 P_a 值很好地平衡了算法搜索的深度和广度.

综上,当问题规模 $|S| = 30$ 时 IDCS 算法的参数设置如下:种群规模 $N_{pop} = 120$,执行基于 Lévy 飞行的深度邻域搜索操作的种群比例 $P_c = 0.6$,执行变异操作的种群比例 $P_a = 0.25$.

4.3 算法有效性验证

由于 TSDP 算法具有指数级别的时间复杂度,随着问题规模 $|S|$ 的扩大,所需的计算时间急剧增加,故用其来解决小规模问题. 因此,为验证 IDCS 算法的有效性,首先将其与 TSDP 算法在小规模算

例下进行比较,将计算时间设置为 1 200 s,是可接受的 TSDP 算法运行的时间. 针对每个算例, IDCS 算法运行 20 次,问题规模参数和测试结果如表 8 所示(符号“—”表示 TSDP 算法无法在规定时间内获取最优解或无法比较). 其中, C^* 、 $CPU1$ 为 TSDP 算法所得的最优目标函数值和相应的运行时间, \bar{C} 、 $CPU2$ 为 IDCS 算法运行 20 次的平均值和平均运行时间.

表 8 小规模算例的参数和测试结果

Tab. 8 Parameters and results of small scale instances

算例	S	C^*	$CPU1(S)$	\bar{C}	$CPU2(S)$	Δ	$PD/\%$
1	5	2 058	0.02	2 058	1.58	0	0
2	10	2 102.9	0.21	2 103.25	2.23	0.35	0.02
3	15	3 320.7	11.34	3 322.36	3.21	1.66	0.05
4	20	3 371.6	136.93	3 375.34	3.76	3.74	0.11
5	25	4 642.8	724.45	4 653.37	5.71	10.57	0.23
6	30	—	—	5 933.29	7.65	—	—

由表 8 可知, IDCS 算法可以有效地解决小规模问题. 各算例求得结果的平均值与 TSDP 算法求得的最优值的绝对偏差 Δ 值虽然随着问题规模的扩大而增大,但各算例优化结果的百分比偏差 (percentage deviation, PD)^[12] 在区间 $[0, 0.23]$ 内.

为了进一步验证本文构建的 IDCS 算法的搜索性能,利用 IDCS 算法求解中、大规模算例,并将其与文献[13]、[14]中的实数遗传算法(RGA)、改进人工蜂群算法(MABC)进行比较分析,包括不同问题规模下优化结果的百分比偏差 PD 值以及标准差 σ 值分析,其中,百分比偏差 PD 值可以反映算法的搜索深度,标准差 σ 值反映出了算法的稳定性, PD 值越大,表明本文构建的 IDCS 算法的搜索深度较其他算法更优. σ 值越小,表明算法的稳定性更好. 三种算法的优化结果如表 9 所示. 可以很明显的看出 IDCS 算法的搜索深度和稳定性均优于其它两种算法.

4.4 算法收敛性能分析

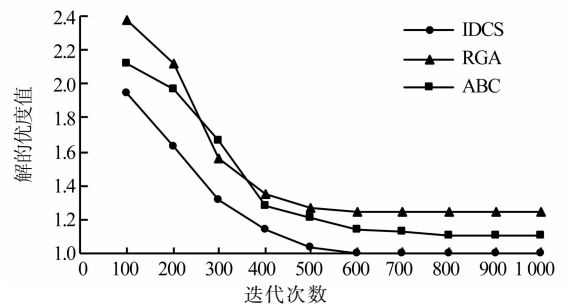
此外,本文根据文献[15]中提出的解的优度(performance ratio, PR)作为算法收敛性能的评价指标. $PR = \frac{V(H, T, Y)}{\text{Best}(Y)}$,其中 $V(H, T, Y)$ 表示算法 H 在迭代 T 次后对算例 Y 的求解结果, $\text{Best}(Y)$ 表示对算例 Y 求解获得的参照最优值. 由于上文已验证的本文构建的 IDCS 算法的有效性,因此,下文将 IDCS 算法对算例 Y 经过 1 000 次迭代后的结果作为 $\text{Best}(Y)$. PR 值能很好地反应出算法的收敛性

能,其值越小,表明算法的收敛性能越好. 图 5 给出了问题规模 $|S| = 50$ 时迭代次数对三种算法的 PR 值的影响.

表 9 中大规模算例的参数及测试结果

Tab. 9 Parameters and results of medium and large scale instances

算例	S	算法	C	$PD/\%$	σ
1	30	IDCS	5 933.29	—	4.26
		RGA	5 940.63	0.12	8.65
		MABC	5 940.04	0.11	5.74
2	40	IDCS	8 567.97	—	11.55
		RGA	8 731.36	1.90	25.49
		MABC	8 797.17	2.68	13.08
3	50	IDCS	12 678.35	—	39.35
		RGA	14 351.35	13.20	79.70
		MABC	13 814.95	8.96	55.34
4	60	IDCS	20 006.8	—	82.21
		RGA	24 271.7	21.32	131.76
		MABC	22 855.1	14.23	92.36

图 5 迭代次数对不同算法的 PR 值的影响Fig. 5 Number of iterations versus the values of PR for different algorithms

5 结论

1) 提出了汽车装配线电动车配送路径及换电站选址问题,对企业的节能减排、降成本增效益具有重要意义;

2) 将该问题划分为两个子问题,并构建了 TSDP 算法以获得小规模问题的最优解,并证明该算法具有指数级别的复杂度;

3) 构建 IDCS 算法求解中、大规模问题,通过在 Lévy 飞行中融入深度邻域搜索算子提高算法精细搜索的能力;

4) 通过将 IDCS 算法与 TSDP 算法在小规模问题下进行比较,并与其它算法进行对比,验证其有效性.

参考文献

- [1] 聂凯, 谢丹凤, 李巍. 新能源汽车城市物流碳排放模型的构建与分析[J]. 湖南大学学报(自然科学版), 2015, 42(9): 134-140. NIE K, XIE D F, LI W. Modeling and analysis of the carbon emission of new energy vehicle in urban logistics industry[J]. Journal of Hunan University(Natural Sciences), 2015, 42(9): 134-140. (In Chinese)
- [2] SCHNEIDER M, STENGER A, GOEKE D. The electric vehicle-routing problem with time windows and recharging stations[J]. Transportation Science, 2014, 48(4): 500-520.
- [3] EMDE S, BOYSEN N. Optimally routing and scheduling tow trains for JIT-supply of mixed-model assembly lines[J]. European Journal of Operational Research, 2012, 217(2): 287-299.
- [4] FATHI M, RODRIGUEZ V, ALVAREZ M J. A novel memetic ant colony optimization-based heuristic algorithm for solving the assembly line part feeding problem[J]. The International Journal of Advanced Manufacturing Technology, 2014, 75(1/4): 629-643.
- [5] GOLZ J, GUJJULA R, GÜNTHER H O, *et al.* Part feeding at high-variant mixed-model assembly lines[J]. Flexible Services and Manufacturing Journal, 2012, 24(2): 119-141.
- [6] ARTMEIER A, HASELMAYR J, LEUCKER M, *et al.* The shortest path problem revisited: Optimal routing for electric vehicles[C]//Annual Conference on Artificial Intelligence. Berlin Heidelberg: Springer, 2010: 309-316.
- [7] YANG J, SUN H. Battery swap station location-routing problem with capacitated electric vehicles[J]. Computers & Operations Research, 2015, 55: 217-232.
- [8] YANG X S, DEB S. Cuckoo search via Lévy flights[C]//World Congress on Nature & Biologically Inspired Computing. IEEE, 2009: 210-214.
- [9] 张泉, 杜亚星, 张林峰, 等. 基于遗传算法的单U地源热泵钻孔内热阻研究[J]. 湖南大学学报(自然科学版), 2014, 41(9): 100-105.
- [10] ZHANG Q, DU Y X, ZHANG L F, *et al.* A study on the heat resistance of single U pipe in the ground source heat pump system based on genetic algorithm[J]. Journal of Hunan University(Natural Sciences), 2014, 41(9): 100-105. (In Chinese)
- [10] 何禹清, 彭建春, 文明, 等. 基于改进遗传算法的配电网动态无功优化[J]. 湖南大学学报(自然科学版), 2010, 37(3): 38-43.
- HE Y Q, PENG J C, WEN M, *et al.* Dynamic reactive power optimization method for distribution network based on improved genetic algorithm[J]. Journal of Hunan University (Natural Sciences), 2010, 37(3): 38-43. (In Chinese)
- [11] WANG J, WANG L, SHEN J. A hybrid discrete cuckoo search for distributed permutation flowshop scheduling problem[C]//2016 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2016: 2240-2246.
- [12] SINGH M R, MAHAPATRA S S. A swarm optimization approach for flexible flow shop scheduling with multiprocessor tasks[J]. International Journal of Advanced Manufacturing Technology, 2011, 62(1/4): 267-278.
- [13] ALNAHHAL M, NOCHE B. A genetic algorithm for super-market location problem[J]. Assembly Automation, 2015, 35(1): 122-127.
- [14] 周炳海, 刘子龙. 考虑衰退的流水车间生产与预防性维护集成调度方法[J]. 计算机集成制造系统, 2016, 22(5): 1273-1279.
- ZHOU B H, LIU Z L. Integrated scheduling method of production and preventive maintenance in flow shops with degradations[J]. Computer Integrated Manufacturing Systems, 2016, 22(5): 1273-1279. (In Chinese)
- [15] QU P, MASON S J. Metaheuristic scheduling of 300-mm lots containing multiple orders[J]. IEEE Transactions on Semiconductor Manufacturing, 2005, 18(4): 633-643.