

求解自行车共享系统静态再平衡问题的单亲遗传算法

冯炳超¹ 吴璟莉^{1,2,3}

1 广西师范大学计算机科学与信息工程学院 广西 桂林 541004

2 广西师范大学广西多源信息挖掘与安全重点实验室 广西 桂林 541004

3 广西区域多源信息集成与智能处理协同创新中心 广西 桂林 541004

(1620438678@qq.com)

摘要 自行车共享系统具有改善城市交通出行结构,减少交通污染等优点。各站点自行车数量相对平衡对于提高共享系统的利用率非常重要,自行车共享系统再平衡问题应运而生。该问题属于NP难问题。2017年,Fábio等提出求解单车多访问静态再平衡问题的ILS算法,获得了较好的结果,但是该算法结构较为复杂,修复算子耗费大量时间,且修复后得到劣质解的概率较大,影响了优化结果。针对该问题,提出基于单亲遗传算法的求解方法P-SMSBR,设计了较为简练的优化过程,运用十进制编码表示运载车路径方案,引入7种变异算子参与演化,并采用精英策略增强算法的搜索能力。利用大量模拟数据和真实数据对算法性能进行测试,实验结果表明,P-SMSBR算法具有较好的优化效果,能够在较短的时间内获得较ILS算法更短的运载车路径方案,且随着站点数的增多,P-SMSBR算法优势更加显著,是一种求解自行车共享系统静态再平衡问题的有效方法。

关键词: 自行车共享系统;NP难;单亲遗传算法;静态再平衡问题;启发式

中图法分类号 TP301

Partheno-genetic Algorithm for Solving Static Rebalance Problem of Bicycle Sharing System

FENG Bing-chao¹ and WU Jing-li^{1,2,3}

1 College of Computer Science and Information Technology, Guangxi Normal University, Guilin, Guangxi 541004, China

2 Guangxi Key Lab of Multi-source Information Mining & Security, Guangxi Normal University, Guilin, Guangxi 541004, China

3 Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing, Guilin, Guangxi 541004, China

Abstract The bicycle sharing system has the advantages of improving urban traffic travel structure and reducing traffic pollution. The relative balance of the number of bicycles at each site is very important for improving the utilization of the sharing system, and the problem of rebalancing bicycle sharing system is proposed. Since the problem is NP-hard one, Fábio et al. proposed the ILS algorithm for solving the single-vehicle and multiple-visit static bicycle rebalancing case in 2017, and obtained good results. However, the ILS algorithm has very complicated structure, and the repair operator, which consumes a lot of time, has a good chance of generating inferior solution. To solve this problem, in this paper, a partheno-genetic algorithm based method P-SMSBR is presented. A more concise optimization process is designed, and decimal code is used to represent a vehicle path solution. Seven mutation operators are introduced, and elite strategy is adopted to enhance the search ability of the algorithm. A large number of simulation and real data were adopted to test the performance of the algorithm. The experimental results indicate that the P-SMSBR algorithm proposed in this paper has better optimization effect, which can obtain a shorter vehicle path than that obtained by the ILS algorithm in a shorter time. In addition, the P-SMSBR algorithm shows more significant advantages with the increase of the number of sites. It is an effective method for solving the single-vehicle and multiple-visit static bicycle rebalancing problem.

Keywords Bicycle sharing system, NP hard, Partheno-genetic algorithm, Static bike rebalancing problem, Heuristic

1 引言

城市的发展使环境污染和交通拥堵问题日益加重,自行车共享系统能够显著减少交通污染,为市民短距离出行带来便利。截止2019年4月,大约14850000辆自助式公共自行

车和电动自行车在世界各个城市使用^[1]。然而,各停车点自行车数量是否平衡大大影响着共享系统的利用率,运营商需要定期使用运载车为各站点调度自行车,以恢复各站点存量的目标值,这类问题被称为自行车共享系统再平衡问题或再定位问题(Bike Rebalancing Problem, BRP)^[2]。

基金项目:国家自然科学基金项目(61762015,61502111,61662007,61763003);广西自然科学基金项目(2016GXNSFAA380192);广西研究生教育创新计划项目(XYCSZ2018078);“八桂学者”工程专项;广西科技基地和人才专项(AD16380008)

This work was supported by the National Natural Science Foundation of China(61762015,61502111,61662007,61763003),Guangxi Natural Science Foundation (2016GXNSFAA380192),Innovation Project of Guangxi Graduate Education(XYCSZ2018078) and “Bagui Scholar” Project Special Funds and Guangxi Science Base and Talent Special Support(AD16380008).

通信作者:吴璟莉(wjlhappy@mailbox.gxnu.edu.cn)

自行车共享系统再平衡问题分为静态再平衡问题(Static Bike Rebalancing Problem, SBRP)^[3]和动态再平衡问题(Dynamic Bike Rebalancing Problem, DBRP)^[4]两大类。SBRP 假设在运载车为各站点调度自行车期间,各站点的自行车不可使用,即各站点自行车数目只能由调度所改变^[5];DBRP 则没有该限制,允许用户在调度过程中使用自行车,即各站点的自行车数据是动态变化的^[4]。由于站点自行车数量的变化会影响调度方案,因此动态再平衡问题通常将整体调度时间分为较小的时间片,然后在某个时间片内假设站点自行车数目是固定的,将其视以静态再平衡问题进行求解^[6]。因此,SBRP 是研究 DBRP 的基础。静态再平衡问题根据运载车数目和站点访问次数又可划分为单车单访问^[7-12]、单车多访问^[2,3,5-6]、多车单访问^[13-17]和多车多访问^[18-22] 4 类子问题,本文主要研究单车多访问的静态再平衡问题,即使用单辆运载车进行调度,且调度过程中允许对站点进行多次访问。

单车多访问 SBRP 问题近年来得到了研究者的关注。2011年,Benchimol等^[3]证明其是一个 NP 难问题,并提出近似求解算法;2013年,Chemla等^[6]提出求解该问题的禁忌搜索算法;2015年,Erdoĉan等^[5]提出基于分支切割算法的精确方法,但该方法受到问题规模的限制,仅在小规模问题上获得了较好结果;2017年,Fábıo等^[2]提出求解该问题的迭代局部搜索算法 ILS,获得了较禁忌搜索算法^[6]和分支切割算法^[5]更好的优化效果,但 ILS 算法结构较为复杂,且耗费大量时间的修复算子得到劣质解的概率较大,影响了优化结果。针对该问题,本文提出基于单亲遗传算法的求解方法 P-SMSBR,设计了较为简练的优化过程,运用十进制编码表示运载车路径方案,引入 7 种变异算子参与演化,并采用精英策略增强算法的搜索能力。大量实验结果表明,P-SMSBR 算法具有较 ILS 算法更好的优化效果,能够在较短的时间内获得较 ILS 算法更短的运载车路径方案,且随着站点数的增多,算法优势更加显著。

2 问题模型

假设 $V = \{0, 1, \dots, n\}$, 其中 0 表示运载车库,其余取值表示 n 个站点,由此构造带权完全有向图 $G = (V, A, C)$, 其中 V 表示顶点集, $A = \{\langle i, j \rangle \mid i, j \in V\}$ 表示边集, $C = \{c_{ij} \mid c_{ij} > 0, i, j \in V\}$, c_{ij} 记录顶点 i 到顶点 j 的代价(如距离、运行时间等),且满足式(1)的三角不等式。

$$c_{ij} + c_{jk} \geq c_{ik}, \forall i, j, k \in V \quad (1)$$

对于 $\forall i \in V$, 令 p_i 为站点 i 初始的自行车数量, p_i' 为站点 i 的目标自行车数量,其需求记为 d_i , 即 $d_i = p_i' - p_i$ ($p_i, p_i' \in N$)。 $d_i > 0$ 表示应从站点 i 补充自行车; $d_i < 0$ 表示应为站点 i 移走自行车; $d_i = 0$ 表示站点 i 为平衡站点,即该站点不强制被访问。令站点 $i \in V$ 的容量为 q_i ($q_i \in N$), 且 $q_0 = p_0 = p_0' = 0$, 运载车容量为 Q ($Q \in N$)。

假设 S 为不平衡站点的集合, 即 $S \subseteq V \setminus \{0\}$, $\delta^+(S) = \{\langle i, j \rangle \mid i \in S, j \notin S\}$ 表示从集合 S 中站点指向其补集中站点的有向弧集。若集合 S 中至少包含一个初始不平衡站点, 定义 $\mu(S) = 1$, 否则 $\mu(S) = 0$ 。令 x_{ij} 表示弧 $\langle i, j \rangle$ 被运载车行驶的次數, y_{ij} 表示在弧 $\langle i, j \rangle$ 上运输的自行车数目。基于以上定义, 下面给出单车多访问 SBRP 问题的数学模型^[6]。

$$\min \sum_{\langle i, j \rangle \in A} c_{ij} x_{ij} \quad (2)$$

$$\text{s. t. } \sum_{i \in V \setminus \{0\}} x_{0i} = 1 \quad (3)$$

$$\sum_{j \in V} x_{ij} = \sum_{j \in V} x_{ji}, \forall i \in V \quad (4)$$

$$\sum_{i \in V \setminus \{0\}} y_{0i} = 0 \quad (5)$$

$$p_i + \sum_{j \in V \setminus \{i\}} y_{ij} = p_i' + \sum_{j \in V \setminus \{i\}} y_{ji} \quad (6)$$

$$\sum_{\langle i, j \rangle \in \delta^+(S)} x_{ij} \geq \mu(S), S \subseteq V \setminus \{0\} \quad (7)$$

$$0 \leq y_{ij} \leq Q x_{ij} \quad (8)$$

$$0 \leq p_i + \sum_{j \in V \setminus \{i\}} y_{ji} \leq q_i + \sum_{j \in V \setminus \{i\}} y_{ij} \quad (9)$$

式(2)为目标函数,表示要找到一条成本最低的运载车调度方案;约束(3)表示运载车从车库出发;约束(4)保证调度路径不能结束于站点,运载车最后应返回车库;约束(5)表示运载车出库时运载量为 0;约束(6)确保站点 i 经调度后,自行车数量达到其目标值;约束(7)限定了各站点访问次数的范围,由于包括初始平衡站点在内的任何站点在调度过程中均可作为临时装卸点,因此非平衡站点的访问次数大于或等于 1,而平衡站点访问次数大于或等于 0;约束(8)表示运输量不超过运载车容量;约束(9)表示调度期间,站点 i 的自行车数目不超过该站点的最大容量。

3 P-SMSBR 算法

本节提出求解单车多访问静态再平衡问题的单亲遗传算法 P-SMSBR (Partheno-genetic-algorithm for Single-vehicle and Multiple-visit Static Bicycle Rebalancing)。算法输入包括站点需求情况、图 G 、运载车容量 Q 、参数 α , 算法输出为调度路径。下面首先介绍算法的关键要素,最后给出算法流程。

3.1 染色体编码及初始种群

染色体采用十进制编码方式,是由车库和站点编号构成的长度为 L 的序列, 即 $Z = \langle z_1, z_2, \dots, z_L \rangle$ ($z_1 = z_L = 0, z_i \in \{1, 2, \dots, n\}, i = 1, 2, \dots, n$)。初始染色体的生成方法如下:利用贪婪算法^[2]随机生成 M 条的染色体,每一条染色体代表一个初始解,以此生成 M 条初始染色体构成初始种群。

3.2 适应度函数

每条染色体表示一个可行的调度方案,需对该方案的优劣进行评价。给定染色体 G , 适应度函数 $Fitness(G)$ 的定义如式(10)所示, 其中 $\sum_{i=1}^{L-1} c_{g_i, g_{i+1}}$ 表示该调度方案对应的路径长度。

$$Fitness(G) = \frac{1}{\sum_{i=1}^{L-1} c_{g_i, g_{i+1}}} \quad (10)$$

3.3 选择算子

采用轮盘赌选择和精英策略来产生新一代种群。将适应度最高的 $\lceil M/10 \rceil$ 个个体直接从父代遗传到子代, 如果 $M/10$ 不是整数, 则四舍五入取偶, 然后运用轮盘赌选择算子来生成其余 $M - \lceil M/10 \rceil$ 个个体。

3.4 变异算子

算法 P-SMSBR 引入 7 种变异算子^[6,23] 对给定染色体执行变异操作。

3.4.1 算子 Addbuff

由模型定义可知,任何站点在调度过程中均可作为临时装卸点,该算子试图在调度路径中增加对某个站点的访问次数。给定染色体 $Z = \langle z_1, z_2, \dots, z_L \rangle$, 随机生成站点序号 $2 \leq i \leq L-1$ 。若站点 z_i 已在调度路径中, 将 z_i 随机插入调度路

径 Z 的某个位置;否则将 z_i 随机插入调度路径 Z 中两个不相邻的位置,这表示初始平衡站点在调度中也可充当临时装卸点被访问。

3.4.2 算子 Reinsertion

给定染色体 $Z = \langle z_1, z_2, \dots, z_L \rangle$, 随机生成 $2 \leq i \leq L-1$, $2 \leq j \leq L (i \neq j)$, 将 z_i 从原位置删除, 并插入 z_j 前一个位置, 得到新染色体。

3.4.3 算子 Or-opt2

给定染色体 $Z = \langle z_1, z_2, \dots, z_L \rangle$, 随机生成 $2 \leq i \leq L-2$, $2 \leq j \leq L (i \neq j)$, 将 z_i 和 z_{i+1} 从原位置删除, 并插入 z_j 前一个位置, 得到新染色体。

3.4.4 算子 Or-opt3

给定染色体 $Z = \langle z_1, z_2, \dots, z_L \rangle$, 随机生成 $2 \leq i \leq L-3$, $2 \leq j \leq L (i \neq j)$, 将 z_i, z_{i+1} 和 z_{i+2} 从原位置删除, 并插入 z_j 前一个位置, 得到新染色体。

3.4.5 算子 2-opt

给定染色体 $Z = \langle z_1, z_2, \dots, z_i, z_{i+1}, \dots, z_{j-1}, z_j, \dots, z_L \rangle$, 随机生成 $2 \leq i < j \leq L-1$, 将两个位置之间的序列反转, 得到新染色体 $Z = \langle z_1, z_2, z_j, z_{j-1}, \dots, z_{i+1}, z_i, \dots, z_L \rangle$ 。

3.4.6 算子 Swap

给定染色体 $Z = \langle z_1, z_2, \dots, z_L \rangle$, 随机生成 $2 \leq i < j \leq L-1$, 交换 z_i 和 z_j 的位置, 得到新染色体 $Z = \langle z_1, z_2, \dots, z_j, \dots, z_i, \dots, z_L \rangle$ 。

3.4.7 算子 Suppression

给定染色体 $Z = \langle z_1, z_2, \dots, z_L \rangle$, 创建站点 $z_i (p_{z_i} = p'_{z_i} \text{ 或 } p_{z_i} \neq p'_{z_i}, \text{ 且 } z_i \text{ 被访问次数大于 } 1, 2 \leq i \leq L-1)$ 构成的集合 Z_s , 随机在其中选择一个站点从染色体移除, 确保得到一个适应值更大的新染色体, 否则保持原染色体不变。

3.5 算法流程

根据上述算法要素, 算法 1 给出算法流程。

算法 1 P-SMSBR

输入: $p_i, p'_i, d_i, q_i (i \in V), Q, G, \maxgen, \maxt, M, P_m$

输出: 调度路径 Z

1. 随机生成初始种群 pop_0 , 将 pop_0 中最优个体保存到变量 best 中;
2. $\text{gen} = 0, t = 0, i = 0$;
3. while $\text{gen} < \maxgen$ or $t < \maxt$ do
4. 将 pop_{gen} 中适应值最高的 $\lceil M/10 \rceil$ 个个体放入 $\text{pop}_{\text{gen}+1}$;
5. 运用轮盘赌选择算子选出其余 $M - \lceil M/10 \rceil$ 个个体放入 $\text{pop}_{\text{gen}+1}$;
6. 对 $\text{pop}_{\text{gen}+1}$ 中个体按适应值从大到小排序, 得到有序染色体 $\{Z_1, Z_2, \dots, Z_N\}$;
7. for ($i = 1; i < M; i++$)
8. if ($i < \lceil M/5 \rceil$) or ($\text{rand}(0,1) < p_m$) then
9. 随机选择某个变异算子对 Z_i 变异, 得到新个体 Z_i' ;
10. if Z_i' 可行 & $\text{Fitness}(Z_i') > \text{Fitness}(Z_i)$ then
11. $Z_i \leftarrow Z_i'$;
12. $\text{gen} \leftarrow \text{gen} + 1$;
13. if $\text{pop}_{\text{gen}+1}$ 最优个体优于 best 个体 then
14. $\text{best} \leftarrow \text{pop}_{\text{gen}+1}$ 最优个体;
15. else
16. $t = t + 1$;
17. 由 best 个体得到路径 Z , 输出 Z 。

4 实验结果

利用模拟和真实数据集对算法 P-SMSBR 和 ILS 进行实

验对比分析。实验在一台联想工作站 (Intel(R) Core(TM) i5-6500 3.20GHz CPU, 内存为 12GB) 上进行, 操作系统为 Windows 7, 编译环境为 myeclipse 2016。P-SMSBR 算法的参数设置如下: $M = 200, \maxgen = 1000, p_m = 0.01, \maxt = 200$ 。ILS 算法的参数设置与文献[2]一致: 外循环 $IR = 10$, 内循环 $I_{ILS} = \max\{160, 4 \times n\}$, 其中 n 为站点数。

4.1 实验数据和评价指标

模拟数据源于文献[24]的 980 个数据样例, 其中站点数 n 取值 20, 30, 40, 50 和 60, 运载车容量 $Q \in \{10, 15, 20, 25, 30, 35, 40, 45, 1000\}$ 。当 $n = 100$ 时, $Q \in \{10, 30, 45, 1000\}$ 。对于 n 和 Q 的任意一个组合, 如 $n = 20, Q = 10$, 有以 n20Q10A 到 n20Q10J 命名的 10 个原始实例, 其中包括顶点 $i \in V (i \neq 0)$ 的欧几里德坐标及其整数需求值 $d_i \in [-10, 10]$ 。对于站点 $i \in V (i \neq 0)$, 以如下方式生成其他参数取值: 随机设置 $q_i (q_i \in \{10, 20, 40\})$, 根据 d_i 的取值设置 p'_i 和 p_i 。若 $d_i > 0$, 则 $p'_i \in [d_i, q_i]$ 且 $p'_i \in \mathbb{N}$; 若 $d_i < 0$, 则 $p'_i \in [0, q_i + d_i]$ 且 $p'_i \in \mathbb{N}$; 最后通过计算得到 $p_i = p'_i - d_i$ 。

真实数据选取 Bari, Denver, Minneapolis 3 个城市的数据样例^[14], 包括顶点 $i \in V$ 的欧几里德坐标及其整数需求值 d_i 。由于所有站点的需求总体并不平衡, 即 $\sum_{i \in V \setminus \{0\}} d_i \neq 0$, 因此设置 $d_o = -\sum_{i \in V \setminus \{0\}} d_i$ 以平衡总体需求^[5]。各站点容量 $q_i = \max_{i \in V} |d_i| (i \in V, i \neq 0)$, p'_i 和 p_i 的设置方法与模拟数据一致。

对每个实例, 运行算法 10 次, 令 b_sol 记录最短的路径长度, a_sol 记录 10 次的平均路径长度, $time$ 表示 10 次的平均运行时间。

4.2 模拟数据结果

针对参数 $\alpha = 1$ 和 $\alpha = 3$ 设置两组实验数据, 则 $p_i = \alpha \times p'_i, p'_i = \alpha \times p'_i, q_i = \alpha \times q_i$ 。为便于表述, 某个 n 和 Q 组合下的 10 个实例的结果, 取均值记为 n 和 Q 组合下的结果。如 n20Q10A 到 n20Q10J 实例的结果均值记为 n20Q10。

图 1 给出了不同 n 设置下的调度路径长度。从图中可以看出, 当 $n = 20$ 时, 两种算法得到的最佳调度路径长度是相等的, 但随着站点数的不断增加, P-SMSBR 算法寻径的优势越来越明显, 能够获得较 ILS 算法更短的调度路径。此外, 当参数 α 增大时, 调度路径长度变短, 这是由于参数 α 的增大使站点容量 q_i 变大, 放宽了站点约束, 因此解质量得以提高。

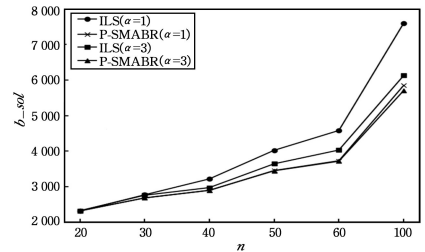


图 1 不同 n 下的路径长度

Fig. 1 Path length with different n

图 2 针对不同 n 设置下的算法运行时间进行比较。从图中可以看出, 随着站点数的不断增多, 两种算法运行时间的差距不断增大。以参数 $\alpha = 1$ 为例, 当 n 从 20 增加到 100 时, P-SMSBR 算法的运行时间从 1.37 s 增加到 25.79 s, 而 ILS 算法的运行时间从 1.89 s 增加到 580.83 s, 其增量是 ILS 算法的 22.52 倍。其主要原因在于 ILS 算法迭代次数多, 外循环

$IR=10$,内循环 $I_{ILS} = \max\{160, 4 \times n\}$,其中 n 为站点数,站点数越多,迭代次数越多。P-SMSBR 算法的最大演化代数次数为 1000,最优值保持恒定阈值 max_t 为 200。此外,ILS 算法结构较为复杂,修复算子耗费大量时间,且修复后得到劣质解的概率较大,影响了优化结果。

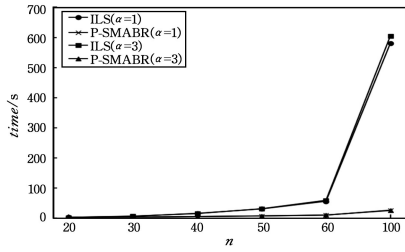


图 2 不同 n 下的运行时间

Fig. 2 Running time with different n

图 3 和图 4 给出了不同 Q 取值下的路径长度。在图 3 中,站点数 $n=40$;在图 4 中,站点数 $n=60$ 。

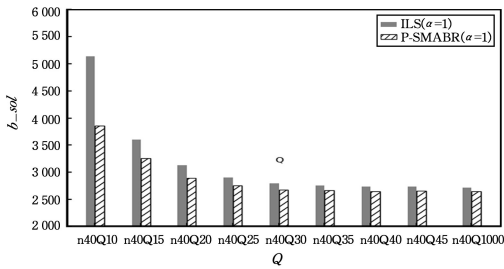


图 3 不同 Q 下的路径长度($\alpha=1, n=40$)

Fig. 3 Path length with different $Q(\alpha=1, n=40)$

图中数据表明,P-SMSBR 算法在不同运载车容量下均能获得较 ILS 算法更短的调度路径。此外,随着运载车容量的增大,由于站点访问次数会有所减少,因此两种算法的调度路径长度均得以缩短,且差距逐渐减小。由此可见,当运载车容量较小时,P-SMSBR 算法的优势较为明显。

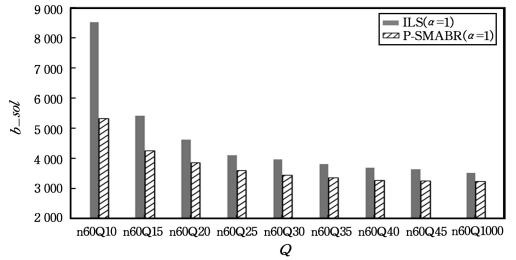


图 4 不同 Q 下的路径长度($\alpha=1, n=60$)

Fig. 4 Path length with different $Q(\alpha=1, n=60)$

4.3 真实数据结果

与模拟数据类似,针对参数 $\alpha=1$ 和 $\alpha=3$ 设置了两组真实数据实验,实验结果如表 1 和表 2 所列。实验结果显示,当站点数目较少时,算法 ILS 和 P-SMSBR 均可求得最优解,且运行时间差别不大;但随着站点数的增多,算法 P-SMSBR 的优势愈加显著,能够在更短的时间内获得较算法 ILS 更短的调度路径。在同一城市站点数不变的情况下,随着运载车容量 Q 的增大,两种算法获得的调度路径长度差距逐渐减小。参数 α 的增加,使得调度路径长度整体缩短,但运行时间整体有所增加。由此可见,真实数据实验结果与模拟数据实验结果具有相同的变化规律。

表 1 不同城市的结果比较($\alpha=1$)

Table 1 Comparison of results with different cities($\alpha=1$)

城市	V	Q	ILS			P-SMSBR		
			b_sol	a_sol	time/s	b_sol	a_sol	time/s
Bari	13	30	14 600	14 640	0.61	14 600	14 620	0.41
Bari	13	20	15 700	15 760	0.53	15 700	15 920	0.34
Bari	13	10	20 600	21 170	0.85	20 600	21 510	0.41
Denver	51	30	59 226	61 626	29.52	52 654	55 942	7.53
Denver	51	20	57 894	65 952	34.69	55 736	58 300	7.42
Denver	51	10	84 989	88 469	49.30	75 434	79 228	7.97
Minneapolis	116	30	189 542	198 770	1170.98	17 6443	189 587	35.33
Minneapolis	116	20	221 955	237 259	1 216.86	218 422	234 034	36.26
Minneapolis	116	10	358 662	380 863	1 529.49	329 860	348 202	43.30

表 2 不同城市的结果比较($\alpha=3$)

Table 2 Comparison of results with different cities($\alpha=3$)

城市	V	Q	ILS			P-SMSBR		
			b_sol	a_sol	time/s	b_sol	a_sol	time/s
Bari	13	30	14 600	14 600	0.59	14 600	14 640	0.39
Bari	13	20	15 700	15 700	0.54	15 700	16 000	0.48
Bari	13	10	20 600	21 200	0.83	20 600	21 760	0.42
Denver	51	30	56 148	60 608	30.17	51 583	54 649	7.41
Denver	51	20	60 043	64 590	33.73	55 673	58 759	7.41
Denver	51	10	82 402	88 798	49.13	74 914	78 947	8.35
Minneapolis	116	30	184 600	195 878	1 241.05	17 5865	190 663	43.94
Minneapolis	116	20	224 337	237 663	1 454.32	218 405	233 544	52.85
Minneapolis	116	10	352 081	377 011	1 542.78	321 225	348 601	75.99

结束语 单车多访问静态再平衡问题是自行车共享系统中重要的研究问题之一。本文提出一种基于单亲遗传算法的优化方法 P-SMSBR。该方法优化过程简练,采用十进制编码表示运载车路径方案,引入 7 种变异算子参与演化,并采用精

英策略增强算法的搜索能力。大量实验结果表明,本文提出的 P-SMSBR 算法具有较好的优化效果,能够在较短的时间内获得较 ILS 算法更短的运载车路径方案,且随着站点数的增多,算法优势更加显著。因此,P-SMSBR 算法是一种求解

自行车共享系统静态再平衡问题的有效方法。

参 考 文 献

- [1] GOOGLE. Interactive bike sharing world map (V10.17.1) [OL]. <http://bike-sharing-map.com>.
- [2] CRUZ F, SUBRAMANIAN A, BRUCK B P, et al. A heuristic algorithm for a single vehicle static bike sharing rebalancing problem [J]. *Computers & Operations Research*, 2017, 79: 19-33.
- [3] BENCHIMOL M, BENCHIMOL P, CHAPPERT B, et al. Balancing the stations of a self service "bike hire" system [J]. *RAIRO Oper Res*, 2011, 45(1): 33-61.
- [4] CAGGIANI L, OTTOMANELLI M. A dynamic simulation based model for optimal fleet repositioning in bike sharing systems [J]. *Procedia Social and Behavioral Sciences*, 2013, 87: 203-210.
- [5] ERDOĞAN G, BATTARRA M, CALVOR W, et al. An exact algorithm for the static rebalancing problem arising in bicycle sharing systems [J]. *Eur J Oper Res*, 2015, 245(3): 667-679.
- [6] CHEMLA D, MEUNIER F, CALVOR W, et al. Bike sharing systems: solving the static rebalancing problem [J]. *Discrete Optimization*, 2013, 10(2): 120-146.
- [7] LI Y F, SZETO W Y, LONG J C, et al. A multipletype bike repositioning problem [J]. *Transportation Research*, 2016, 90: 263-278.
- [8] ERDOĞAN G, LAPORTE G, CALVOR W, et al. The static bicycle relocation problem with demand intervals [J]. *European Journal of Operational Research*, 2014, 238(2): 451-457.
- [9] HOS C, SZETO W Y. Solving a static repositioning problem in bike-sharing systems using iterated tabu search [J]. *Transportation Research Part E*, 2014, 69: 180-198.
- [10] CAGGIANI L, OTTOMANELLI M. A modular soft computing based method for vehicles repositioning in bike-sharing systems [J]. *Proc Social Behav*, 2012, 54: 675-684.
- [11] TING C K, LIAO X L. The selective pickup and delivery problem: Formulation and a memetic algorithm [J]. *Int J Prod Econ*, 2013, 141(1): 199-211.
- [12] SALAZAR J J, SANTOS B. The split-demand one-commodity pickup-and-delivery travelling salesman problem [J]. *Transp Res Part B: Methodol*, 2015, 75: 58-73.
- [13] DELL'AMICO M, IORI M, NOVELLANI S, et al. A destroy and repair algorithm for the bike sharing rebalancing problem [J]. *Comput Oper Res*, 2016, 71: 149-162.
- [14] DELL'AMICO M, HADJICOSTANTINO E, IORI M, et al. The bike sharing rebalancing problem: mathematical formulations and benchmark instances [J]. *Omega*, 2014, 45: 7-19.
- [15] FORMAI A, RAVIV T, TZUR M, et al. A3-step math heuristic for the static repositioning problem in bike-sharing systems [J]. *Transp Res Part B: Methodol*, 2015, 71: 230-247.
- [16] RAVIV T, TZUR M, FORMAI A, et al. Static repositioning in a bike-sharing system: models and solution approaches [J]. *EURO J Transp Logist*, 2013, 2: 187-229.
- [17] LIN J H, CHOU T C. A geo-aware and VRP-based public bicycle redistribution system [J]. *International Journal of Vehicular Technology*, 2012, 10: 1-14.
- [18] ALVAREZ R, BELENGUER J M, BENAVENTE, et al. Optimizing the level of service quality of a bike-sharing system [J]. *Omega*, 2016, 62: 163-175.
- [19] GASPEROLD, RENDLA, URLIT, et al. A Hybrid ACO+CP for Balancing Bicycle Sharing Systems [J]. *Hybrid Metaheuristics*, 2013, 7919: 198-212.
- [20] BULHÕES T, SUBRAMANIAN A, ERDOČA N G, et al. The static bike relocation problem with multiple vehicles and visits [J]. *European Journal of Operational Research*, 2018, 264: 508-523.
- [21] RAINER M, PAPAZEK P, RAIDL G, et al. Raidl, PILOT, GRASP, and VNS approaches for the static balancing of bicycle sharing systems [J]. *J Glob Optim*, 2015, 63: 597-629.
- [22] ESPEGREN H M, KRISTIANS LUND J, ANDERSSON H, et al. The static bicycle repositioning problem [J]. *Literature Survey and Newformulation*, 2016, 9855: 337-351.
- [23] MLADENOVIĆ N, HANSEN P. Variable neighborhood search [J]. *Comput Oper Res*, 1997, 24(11): 1097-1100.
- [24] HERNÁNDEZ H, SALAZAR J J. A branch-and-cut algorithm for a traveling salesman problem with pickup and delivery [J]. *Discret Appl Math*, 2004, 145: 126-139.



FENG Bing-chao, born in 1993, post-graduate. His research interest is intelligent optimization algorithm.



WU Jing-li, born in 1978. Her current research interests include bioinformatics, computational biology, algorithms and complexity.