

基于客户满意度的车辆路径问题的混合蝙蝠算法

孙 奇, 马 良

(上海理工大学 管理学院, 上海 200093)

摘要: 车辆路径问题对现实有着良好的指导意义, 自提出以来便吸引了企业界和学术界的广泛关注。然而, 传统车辆路径问题仅仅将车辆行驶里程最短作为目标, 忽视良好的客户体验对于企业的重要性。考虑客户满意度这一目标, 建立以客户满意度和车辆行驶里程最短为目标的多目标优化模型, 根据车辆路径问题的具体特征, 改变基本蝙蝠算法的编码方式。为克服基本蝙蝠算法求解精度低、易陷入局部最优的缺陷, 加入贪婪随机自适应启发式算法提高求解精度, 引入病毒进化机制以增强蝙蝠算法跳出局部最优的能力。算例分析表明: 病毒进化混合蝙蝠算法相比于基本蝙蝠算法, 在求解精度上有较大幅度提高, 是一种有效求解车辆路径问题的方法。

关键词: 车辆路径问题; 蝙蝠算法; 客户满意度; 病毒进化; 贪婪随机自适应算法

中图分类号: TP 18 **文献标志码:** A

Hybrid Bats Algorithm Based on Customer Satisfaction for Vehicle Routing Problem

SUN Qi, MA Liang

(Business School, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: The vehicle routing problem has a good guiding significance for the reality and has attracted extensive attention from the business community and academia since it was proposed. However, the traditional vehicle routing problem only takes the shortest vehicle mileage as the target, ignoring the logistics and transportation industry, and neglecting the importance of good customer experiences for enterprises. Considering the goal of customer satisfaction, a multi-objective optimization model with the goals of both the customer satisfaction and the shortest vehicle mileage was established. According to the specific characteristics of the vehicle routing problem, the coding mode of the basic bat algorithm was changed. In order to overcome the disadvantages of the basic bat algorithm, such as low accuracy and being easy to get into local optimization, the greedy randomized adaptive search procedure was added to improve the solution accuracy, and the virus evolution mechanism was introduced to enhance the ability of the bat algorithm to jump out of local optimization. The results show that compared with the basic bat algorithm, the virus evolution hybrid bat algorithm is more accurate and is an effective

收稿日期: 2018-01-26

基金项目: 国家自然科学基金资助项目 (71401106); 教育部人文社科规划基金资助项目 (16YJA630037)

第一作者: 孙 奇 (1992-), 男, 硕士研究生。研究方向: 物流工程、智能优化。E-mail: piedoggy@163.com

通信作者: 马 良 (1964-), 男, 教授。研究方向: 系统工程、智能优化。E-mail: maliang@usst.edu.cn

method to solve the vehicle routing problem.

Keywords: vehicle routing problem; bat algorithm; customer satisfaction; virus evolution; greedy randomized adaptive search procedure

车辆路径问题(vehicle routing problem, VRP)作为运筹学中一个典型的组合优化问题,属于NP难题。该问题起源于车辆运输领域,最早由Dantzig等^[1]于1959年首次提出并研究。由于车辆路径问题有着良好的实际应用价值,该问题自提出以来,便引起了管理学、运筹学、图论、组合数学、计算机应用等领域学者的关注,研究者们对其进行了大量的研究,并取得了很多成果。

带时间窗的车辆路径问题^[2](vehicle routing problem with time windows, VRPTW)最早由Savelsbergh提出,是在VRP的基础上考虑了客户对接受服务时间的需求。相比于传统的VRP,VRPTW更贴近实际情况。VRPTW也属于NP难题,当问题规模不断增大,其计算时间呈现爆炸式增长,精确算法难以甚至无法求出最优解。智能优化算法逐渐被大量使用,如遗传算法^[3-4]、蚁群算法^[5-8]、模拟退火算法^[9]、粒子群算法^[10-12]等,并取得了一定的成果。

蝙蝠算法(bat algorithm, BA)是由剑桥大学Yang^[13]于2010年最早提出。随着对该算法的进一步研究,国内外学者提出了众多的改进蝙蝠算法^[14-17]用于解决各领域优化难题。同大多数智能优化算法一样,蝙蝠算法也具有易陷入局部最优、后期收敛速度慢、局部搜索能力弱等缺点。针对蝙蝠算法的这些劣势,本文利用GRASP(greedy randomized adaptive search procedure)启发式算法为蝙蝠算法构建初始解,以提高蝙蝠算法的寻优精度。同时,在蝙蝠算法的主体中引入病毒进化(virus evolution, VE)机制,使病毒群体和蝙蝠群体以一种协同进化的方式进行演化,以增强蝙蝠算法跳出局部最优的能力。最后,通过实例验证了该算法的有效性。

1 基于客户满意度的车辆路径问题

1.1 问题描述

基于客户满意度的车辆路径问题是多目标VRPTW,该问题不仅要满足配送车辆的行驶里程

最短,还需要考虑客户满意度,使得总客户满意度最大。为此,该问题需要满足以下假设条件: a. 所有车辆从配送中心出发,当前路径配送完毕后,所有客户必须返回配送中心; b. 各条路径上的客户需求总量不能超过车辆的最大装载量; c. 所有客户点都需要被服务,一辆车可以服务多个客户点,但一个客户只能由一辆车提供服务; d. 车辆到达客户的时间可以提前,车辆提前到达客户点需要等待至可接受服务的时间节点,但不能迟于客户可接受的最晚服务时间。

1.2 模型建立

1.2.1 引入客户满意度

现有客户*i*可接受服务的时间窗为 $[ET_i, LT_i]$,现假设车辆到达客户*i*的时间为 t_i ,在硬时间窗车辆路径问题的要求下,超出时间窗以外的客户将不会接受服务。而在实际物流配送过程中,虽然货物不在时间窗内到达的情况会引起客户不满,但客户对于这种情况都有一定的容忍度,只不过如果在时间窗外,客户的满意度会降低。本文在原有时间窗的基础上构建模糊时间窗,设客户能接受服务的最早时间为 ET'_i ,能接受服务的最晚时间为 LT'_i 。货物如果在 $[ET_i, LT_i]$ 内到达,客户的满意度最高为100,在时间窗之外,但在 $[ET'_i, LT'_i]$ 内到达,虽然客户满意度呈现下降趋势,但客户仍然会接受服务。构建的模糊时间窗的客户满意度如图1所示,对应的客户满意度函数如式(1)所示。

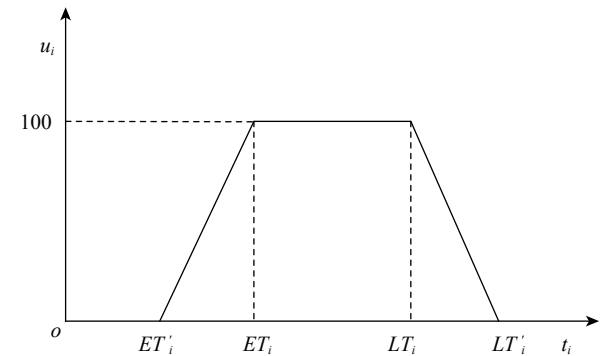


图1 客户满意度
Fig.1 Customer satisfaction

客户满意度函数:

$$u_i(t_i)=\begin{cases} \frac{t_i-ET_i'}{ET_i-ET_i'}\times 100, & t_i\in [ET_i',ET_i] \\ 100, & t_i\in [ET_i,LT_i] \\ \frac{LT_i'-t_i}{LT_i'-LT_i}\times 100, & t_i\in [LT_i,LT_i'] \\ 0, & t_i\notin [ET_i',LT_i'] \end{cases} \quad (1)$$

1.2.2 数学模型

目标函数:

$$\max P_1=\sum_{i\in N} u_i \quad (2)$$

$$\min P_2=\sum_{i\in M}\sum_{j\in M}\sum_{k\in K} c_{ij}x_{ijk} \quad (3)$$

约束条件:

$$\sum_{i\in N} y_{ik}d_i\leq Q, \quad \forall k\in K \quad (4)$$

$$\sum_{k\in K} y_{ik}=1, \quad \forall i\in N \quad (5)$$

$$\sum_{i\in M} x_{ijk}=y_{jk}, \quad \forall j\in N, \forall k\in K \quad (6)$$

$$\sum_{j\in M} x_{ijk}=y_{ik}, \quad \forall i\in N, \forall k\in K \quad (7)$$

$$\sum_{i,j\in M} x_{ijk}\leq |S-1|, \quad \forall k\in K \quad (8)$$

$$ET_i'\leq t_i+w_i\leq LT_i', \quad \forall i\in N \quad (9)$$

式中: $M=\{0,1,2,\cdots,n\}$ 表示所有节点集合, 其中 0 表示配送中心; $N=\{1,2,3,\cdots,n\}$ 为客户点; $K=\{1,2,3,\cdots,m\}$ 为配送车辆集合; Q 表示车辆的最大载重量; d_i 表示客户点 i 的需求量; t_i 为车辆到达客户 i 的时间; c_{ij} 为客户 i 到客户 j 的距离; w_i 为车辆在客户 i 的等待服务时间; S 为某辆车的配送客户集, $|S|$ 为集合 S 中所包含顶点的个数; x_{ijk} 和 y_{ik} 为决策变量。

$$x_{ijk}=\begin{cases} 1, & \text{车辆}k\text{由}i\text{驶向}j \\ 0, & \text{否则} \end{cases}$$
$$y_{ik}=\begin{cases} 1, & \text{客户}i\text{的需求由车辆}k\text{完成} \\ 0, & \text{否则} \end{cases}$$

式(2)和式(3)分别表示要求的最大化客户满意度和最小化总车辆行驶里程的目标函数; 式(4)为车辆载重限制; 式(5)、式(6)和式(7)保证每个客户点被且仅被服务一次, 每个客户的需求只由一辆车来完成; 式(8)为消除子回路; 式(9)为时间窗约束。本文采用线性加权法将两个目标转化为单目标优化, 即

$$\min Z=\omega P_1-(1-\omega)P_2 \quad (10)$$

式中, ω 表示目标权重, $\omega\in(0,1)$ 。

2 求解基于客户满意度的车辆路径问题的混合蝙蝠算法设计

2.1 基本蝙蝠算法

蝙蝠算法是受到自然界蝙蝠回声定位的启发演变而来的新型智能优化算法, 蝙蝠通过改变声波的频率、声音的响度、脉冲发生率, 从而锁定目标进行捕食。

a. 蝙蝠速度位置更新。

基本蝙蝠算法是通过频率 f 实现对蝙蝠移动步长和范围的控制, 从而达到更新蝙蝠的速度和位置。设 m 只蝙蝠于时刻 t 散布在 D 维搜索空间内, 则第 i 只蝙蝠在 $t+1$ 时刻位置和速度更新公式如下:

$$f_i=f_{\min}+(f_{\max}-f_{\min})\beta \quad (11)$$

$$v_i^{t+1}=v_i^t+(X_i^t-X_*)f_i \quad (12)$$

$$X_i^{t+1}=X_i^t+v_i^{t+1} \quad (13)$$

式中: f_i 表示第 i 只蝙蝠发出的脉冲频率; $f_{\max}$ 和 $f_{\min}$ 分别表示蝙蝠发出脉冲频率的最大值和最小值; β 是一个随机变量, 其取值范围为 $\beta\in[0,1]$; v 表示蝙蝠速度; X_* 表示当前蝙蝠种群中的最优位置。

同时基本蝙蝠算法内部可设定局部搜索行为, 产生一个随机数 $rand1$, 通过比较 $rand1$ 和脉冲发射率 r_i , 若 $rand1>r_i$, 根据式(14)在当前最优解集中选取一个最优蝙蝠, 进行随机扰动, 从而产生一个新位置。

$$X_{\text{new}}=X_{\text{old}}+\varepsilon A^t \quad (14)$$

式中: X_{new} 表示更新后的蝙蝠位置; X_{old} 表示更新前的蝙蝠位置; $\varepsilon\in[1,1]$ 为一个随机数; A^t 表示当前时刻下所有蝙蝠的平均响度。

b. 蝙蝠响度和脉冲发生率更新。

蝙蝠在找到猎物时, 会降低自身的声波响度并且增加脉冲发生率。响度和脉冲发生率按照以下公式进行更新。

$$A_i^{t+1}=\alpha A_i^t \quad (15)$$

$$r_i^{t+1}=r_i^t[1-\exp(-\gamma t)] \quad (16)$$

式中: r_i^t 表示第 i 只蝙蝠在 t 时刻的脉冲发生率; $\alpha\in(0,1)$ 表示响度衰减系数, $\gamma>0$ 表示脉冲发生率增加系数。 α 与 γ 均为常量, 通过 α 与 γ 的调节, 从

而达到平衡整个算法的全局搜索和局部搜索。

2.2 混合蝙蝠算法

2.2.1 基本蝙蝠算法速度的重新定义与编码

基本蝙蝠算法最初被用来解决的是连续函数优化问题, 在面对离散的组合优化问题时, 基本蝙蝠算法很难直接应用, 需要根据实际问题对蝙蝠位置的编码方式以及速度进行重新定义。本文借鉴文献[18], 利用汉明距离(HD)来对速度进行定义, 并采取二维编码方式。现以一个有 7 个客户点的车辆路径问题为例, 蝙蝠位置编码如下:

$$X_1 = [1\ 3\ 5\ 2\ 4\ 7\ 6]$$
$$X_2 = [1\ 1\ 2\ 2\ 2\ 3\ 3]$$

上述编码表示使用 3 辆车, 其中, 车辆 1 路径: 0—1—3—0; 车辆 2 路径: 0—5—2—4—0; 车辆 3 路径: 0—7—6—0。

蝙蝠速度定义位置更新操作如式(17)和(18)所示:

$$V_i^{t+1} = HD(X_i^t, X_*) \quad (17)$$

$$X_i^{t+1} = (X_i^t, V_i^{t+1}, 2\text{-opt}) \quad (18)$$

汉明距离表示蝙蝠两个位置之间的差距, 举例如下:

$$X_i = [1\ 3\ 5\ 2\ 4\ 7\ 6]$$
$$X_* = [1\ 2\ 5\ 3\ 7\ 4\ 6]$$

则在此例中, 蝙蝠当前位置和最优位置之间的汉明距离为 4。在本文中具体操作为: a. $HD > \frac{n}{2}$, 针对蝙蝠当前位置随机选取两对对位点进行 2-opt 操作; b. $HD \leq \frac{n}{2}$, 则在蝙蝠当前位置中随机选取一对对位点进行 2-opt 操作。

2.2.2 GRASP 启发式算法构造初始解

GRASP 启发式算法^[19]已经被证明是一种求解组合优化问题的有效方法, 它由两个阶段构成, 分别是构造阶段和局部搜索阶段。在构造阶段中通过带有随机特性的贪心算法进行初始解的选择, 该算法具有贪心和随机两种特性。而在局部搜索阶段中, 通过局部搜索策略对构造阶段所得到的解进行优化, 以提高解的质量。本文局部搜索阶段所采取的局部搜索策略为 2-opt 方法, 产生数量为 $0.8N$ (N 为种群总数量) 的初始解, 剩余的种群初始解则由随机策略产生。

2.2.3 引入病毒进化机制

病毒是自然界一种神奇的生物, 它具有特殊的感染能力和进化能力。在病毒进化算法中, 病毒一方面通过感染能力, 将自身的遗传信息传递

给其他个体; 另一方面通过自身的进化能力, 从宿主个体中获取部分遗传信息来改变自身的遗传信息, 从而达到进化的目的。病毒进化算法的具体设计如下所示。

a. 病毒个体的编码方式: 病毒个体(virus)是从宿主个体(host)中抽取一定片段, 长度与宿主个体保持一致均为 n , 在病毒个体中加入通配符“*”。



图 2 病毒个体编码
Fig.2 Virus individual coding

b. 病毒的感染操作: 病毒个体以 P_{infect} (感染概率) 感染宿主个体, 使得病毒个体上的基因片段替换宿主个体基因上的片段。根据车辆路径的实际情况, 为了避免路径重复的情况, 将感染操作定义为病毒个体基因片段替换宿主个体中对应位置, 并将所替换的宿主个体基因与宿主个体中与病毒用来替换的基因重复的片段进行交换, 从而产生一个新的宿主个体, 具体操作如图 3 所示。

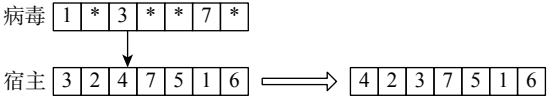


图 3 病毒感染操作
Fig.3 Virus infect operation

c. 病毒个体适应度: 一个病毒个体是否优劣是由它的适应度函数值来表现, 适应度值(fit)用病毒在感染宿主个体后, 宿主个体适应度值的变化来表达。假设宿主个体集合为 N , 则第 i 个病毒的适应度计算为

$$fitV_i = \sum_{j \in N} (fitH_j^* - fitH_j) \quad (19)$$

式中, $fitH_j^*$, $fitH_j$ 分别表示宿主个体被感染后和被感染前的适应度值。

d. 病毒的复制操作: 病毒通过复制操作将宿主个体的基因以概率 P_{copy} 复制到自身基因中, 以产生拥有更多有效字符串的病毒个体。在初始化病毒种群的时候随机选取宿主个体经复制操作产生病毒个体的初始种群, 具体操作如图 4 所示。

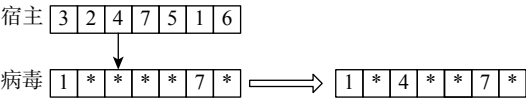


图 4 病毒复制操作
Fig.4 Virus copy operation

e. 病毒的剪切操作：病毒以 P_{cut} 的概率进行剪切操作，即用通配符“*”来替代病毒基因中的有效片段，剪切操作的意义在于通过减少有效基因的个数来降低病毒个体继续进化的能力。具体操作如图5所示。

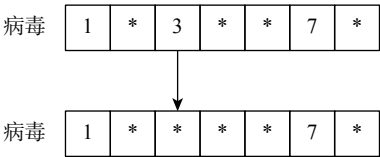


图5 病毒剪切操作
Fig.5 Virus cut operation

f. 病毒的生命力：每一个病毒都有自己的生命力(Life), 当生命力小于0时, 意味着病毒已经死亡, 此时需要产生新的病毒来维持病毒种群, 病毒生命力定义为

$$L_i(t+1)=\theta L_i(t)+fitV_i$$

(20)

式(20)表示第*i*个病毒在*t*+1时刻的生命力, 式中, $\theta \in (0,1)$ 为病毒的生命力衰减系数。

2.3 混合蝙蝠算法设计

本文针对基本蝙蝠算法的局限性以及车辆路径问题的具体情况, 引入 GRASP 启发式算法和病毒进化机制, 对基本蝙蝠算法中速度以及位置更新方式进行重新定义, 提出了 GRASP-VREBA 算法, 该算法具体步骤如下:

Step 1 初始化 GRASP 各参数, 定义贪婪函数, 通过构造阶段和局部搜索两个阶段得到数量为 $0.8N$ 的初始解, 剩余初始解则采取随机生成的方法产生;

Step 2 初始化蝙蝠算法和病毒进化相应的参数, 包括蝙蝠速度 v_i 、响度 A_i 、脉冲频率 r_i 、响度衰减系数 α 、脉冲频率增加系数 γ , 以及病毒感染、复制和剪切概率(P_{infect} , P_{copy} , P_{cut})、病毒生命力衰减系数 θ 等, 并通过病毒复制操作产生病毒的初始种群;

Step 3 根据式(10)计算各蝙蝠的适应度值, 排列蝙蝠, 确定最优蝙蝠的位置;

Step 4 进行蝙蝠主种群的进化操作, 根据重新定义的速度和位置更新公式, 进行蝙蝠位置和速度的更新, 并产生新解;

Step 5 将每一个病毒对所有蝙蝠进行感染操作, 并计算蝙蝠的适应度值和生命力, 若生命力小于0, 则随机选择一个宿主个体进行病毒复制操作, 否则对病毒个体进行剪切操作, 当病毒生命

力小于0时, 通过病毒复制产生新的病毒;
Step 6 判断是否达到最大迭代次数或者已经达到最优解, 否则返回 Step 3 继续运行。

3 仿真实验与分析

3.1 算例描述与数据

为了验证算法的有效性, 本文对文献[8]算例的时间窗进行扩展, 扩展方式为客户可接受服务的最早时间扩展为在原时间窗的基础上提前 20%, 客户可接受服务的最晚时间扩展为在原时间窗的基础上延长 10%。

算例中, 各参数如下: 车辆最大载重量为 8 t, 为各客户点提供货物配送服务, 各客户信息如表1所示; 车辆行驶速度为 40 km/h, 如果车辆提前到达必须等待至可接受服务时间节点后才可进行服务。在所有客户被服务完毕后, 所选路径需要达到的目标为: 车辆总行驶里程最少, 客户满意度最大。本文权重取 $\omega=0.2$ 。

表 1 配送中心与客户信息
Tab.1 Distribution center and customer data

客户 编号	横坐 标/km	纵坐 标/km	需求 量/t	时间窗/h				服务 时间/h
				ET_i'	ET_i	LT_i	LT_i'	
0	70	70	0	0	0	0	0	0
1	107	77	3.4	1.2	1.5	3.5	3.85	0.2
2	109	139	0.8	2.4	3.0	5.50	6.05	0.2
3	120	22	3.9	1.6	2.0	5.0	5.50	0.2
4	48	47	1.9	1.2	1.5	5.0	5.50	0.4
5	116	22	3.2	1.6	2.0	5.5	6.05	0.2
6	12	138	1.4	3.2	4.0	8.0	8.80	0.5
7	86	40	2.2	1.2	1.5	3.0	3.30	0.4
8	121	124	2.1	2.0	2.5	5.0	5.50	0.2
9	61	57	3.5	3.6	4.5	8.0	8.80	0.2
10	40	113	2.3	1.6	2.0	3.5	3.85	0.2
11	129	24	1.8	2.0	2.5	5.5	6.05	0.4
12	12	84	1.6	1.6	2.0	4.5	4.95	0.4
13	44	116	2.7	2.4	3.0	6.0	6.60	0.2
14	102	52	1.5	2.8	3.5	5.5	6.05	0.2
15	41	36	1.3	3.2	4.0	7.0	7.70	0.2
16	132	138	2.4	2.4	3.0	6.0	6.60	0.4
17	104	139	2.9	2.4	3.0	5.0	5.50	0.2
18	104	54	1.3	0.8	1.0	3.5	3.85	0.4
19	22	104	1.1	1.6	2.0	4.5	4.95	0.4
20	46	133	0.7	1.6	2.0	4.5	4.95	0.4

注: 编号 0 表示配送中心; 1~20 表示客户

3.2 结果分析

本文应用 Matlab R2014a 对基本蝙蝠算法、GRASP-BA 算法、GRASP-VEBA 算法进行编程实现,这 3 种算法均在 Intel Core i7-7700 3.6 GHz (8.00 GB RAM)、操作系统为 Win10 的环境下运行。3 种算法的参数设置如下: $\alpha=0.95$, $\gamma=0.95$, 初始响度 $A_0 \in [0, 1]$; 初始脉冲频率 $r_0 \in [0, 1]$; 蝙蝠种群数 $n=100$, 最大迭代次数为 600。在基本蝙蝠算法中,蝙蝠种群为 300。GRASP-VEBA 算法中,病毒种群数目 $virus_num=0.3n$; $P_{copy}=0.2$; $P_{infect}=0.2$, $P_{cut}=0.1$ 。GRASP-BA 和 GRASP-VEBA 算法中,阈值 $\alpha=0.3$ 。

3 种算法均随机运行 15 次,优化结果如表 2 所示。基本蝙蝠算法求出的最优路径中,车辆总行驶里程为 1 207.2 km,最大客户满意度为 1 524.3; GRASP-BA 算法求出的最优路径中,车辆总行驶里程为 1 148.8 km,最大客户满意度为 1 524.3; GRASP-VEBA 算法求出的最优路径中,车辆总行驶里程为 1 064.8 km,最大客户满意度为 1 524.3。可见改进的蝙蝠算法均优于基本蝙蝠算法。尤其在 GRASP-VEBA 算法中,相比于 BA 算法和 GRASP-BA 算法平均长度均减少超过 10%,分别为 16.9% 和 12.2%,平均客户满意度提高分别为 3.1% 和 1.9%,验证了算法的有效性和稳定性。

表 2 3 种算法计算结果比较
Tab.2 Comparison of the calculation results of three algorithms

算法	最优路径长度/km	最优路径客户满意度	最差路径长度/km	最差路径客户满意度	平均长度/km	平均客户满意度
BA	1 207.2	1 524.3	1 314.8	1 424.3	1 358.9	1 475.6
GRASP-BA	1 148.8	1 524.3	1 343.2	1 471.4	1 286.0	1 492.6
GRASP-VEBA	1 064.8	1 524.3	1 230.7	1 524.3	1 129.0	1 521.9

最优路径图如图 6 所示,最优路径各车配送路线为

- 车辆 1: 0—18—5—11—14—0
- 车辆 2: 0—15—4—9—0
- 车辆 3: 0—12—19—13—6—0
- 车辆 4: 0—3—7—0
- 车辆 5: 0—20—10—17—2—0
- 车辆 6: 0—1—16—8—0

贴合生活实际。建立基于客户满意度的车辆路径问题模型,针对该问题,在基本蝙蝠算法的基础上引入 GRASP 启发式算法构造初始解和病毒进化机制进行协同进化,设计出求解多目标车辆路径问题的混合蝙蝠算法,并用其求解测试算例。求解结果表明算法的有效性,能够较好地弥补基本蝙蝠算法后期收敛速度慢、缺乏有效局部搜索机制等的不足。

参考文献:

[1] DANTZIG G B, RAMSER J H. The truck dispatching problem[J]. *Management Science*, 1959, 6(1): 80–91.
[2] SAVELSBERGH M W P. Local search in routing problems with time windows[J]. *Annals of Operations Research*, 1985, 4(1): 285–305.
[3] URSANI Z, ESSAM D, CORNFORTH D, et al. Localized genetic algorithm for vehicle routing problem with time windows[J]. *Applied Soft Computing*, 2011, 11(8): 5375–5390.
[4] GHOSEIRI K, GHANNADPOUR S F. Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm[J]. *Applied Soft Computing*, 2010, 10(4): 1096–1107.
[5] DING Q L, HU X P, SUN L J, et al. An improved ant colony optimization and its application to vehicle routing

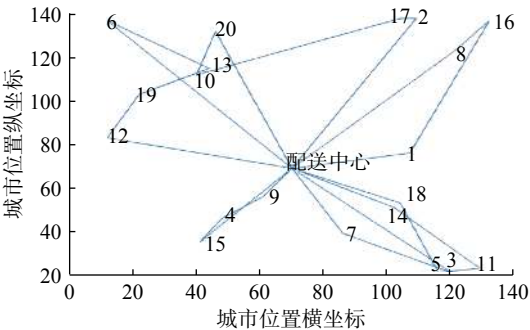


图 6 最优方案路径图
Fig.6 Optimal scheme routes

4 结 论

本文在传统单目标车辆路径问题的基础上增加客户满意度最大化的目标,使车辆路径问题更

- problem with time windows[J]. *Neurocomputing*, 2012, 98: 101–107.
- [6] 张勇. 基于改进蚁群算法物流配送路径优化的研究[J]. *控制工程*, 2015, 22(2): 252–256.
- [7] 何小锋, 马良. 带时间窗车辆路径问题的量子蚁群算法[J]. *系统工程理论与实践*, 2013, 33(5): 1255–1261.
- [8] 刘云, 张惠珍. 多目标带时间窗的车辆路径问题的单亲遗传混合蚁群算法[J]. *公路交通科技*, 2016, 33(6): 95–100, 106.
- [9] 王超, 穆东. 基于模拟退火算法求解 VRSPDTW 问题[J]. *系统仿真学报*, 2014, 26(11): 2618–2623.
- [10] MARINAKIS Y, MARINAKI M, DOUNIAS G. A hybrid particle swarm optimization algorithm for the vehicle routing problem[J]. *Engineering Applications of Artificial Intelligence*, 2010, 23(4): 463–472.
- [11] 徐杰, 黄德先. 基于混合粒子群算法的多目标车辆路径研究[J]. *计算机集成制造系统*, 2007, 13(3): 573–579, 584.
- [12] 温惠英, 孙博. 基于离散粒子群算法的协同车辆路径问题[J]. *公路交通科技*, 2011, 28(1): 149–153.
- [13] YANG X S. A new metaheuristic bat-inspired algorithm[M]//GONZÁLEZ J R, PELTA D A, CRUZ C, et al. *Nature Inspired Cooperative Strategies for Optimization* (NICSO 2010). Berlin Heidelberg: Springer, 2010: 65–74.
- [14] LIN J H, CHOU C W, YANG C H, et al. A chaotic levy flight bat algorithm for parameter estimation in nonlinear dynamic biological systems[J]. *Computer and Information Technology*, 2012, 2(2): 56–63.
- [15] NAKAMURA R Y M, PEREIRA L A M, COSTA K A, et al. BBA: a binary bat algorithm for feature selection[C]//*Proceedings of the 2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*. Ouro Preto, Brazil: IEEE, 2012: 291–297.
- [16] 刘长平, 叶春明. 具有混沌搜索策略的蝙蝠优化算法及性能仿真[J]. *系统仿真学报*, 2013, 25(6): 1183–1188, 1195.
- [17] 马祥丽, 张惠珍, 马良. 蝙蝠算法在物流配送车辆路径优化问题中的应用[J]. *数学的实践与认识*, 2015, 45(24): 80–86.
- [18] 殷亚, 张惠珍. 易腐生鲜货品车辆路径问题的改进混合蝙蝠算法[J]. *计算机应用*, 2017, 37(12): 3602–3607.
- [19] FEO T A, RESENDE M G C. Greedy randomized adaptive search procedures[J]. *Journal of Global Optimization*, 1995, 6(2): 109–133.

(编辑: 丁红艺)



(上接第 159 页)

参考文献:

- [1] 仪洪勋, 杨重骏. 亚纯函数唯一性理论[M]. 北京: 科学出版社, 1995.
- [2] HAYMAN W K. Meromorphic functions[M]. Oxford: Clarendon Press, 1964.
- [3] 杨乐. 值分布论及其新研究[M]. 北京: 科学出版社, 1982.
- [4] 方明亮. 涉及微分多项式的亚纯函数的唯一性[J]. *数学进展*, 1995, 24(3): 244–249.
- [5] RUBEL L A, YANG C C. Values shared by an entire function and its derivative[M]//BUCKHOLTZ J D, SUFFRIDGE T J. *Complex Analysis*. Berlin: Springer Verlag, 1977: 101–103.
- [6] BRÜCK R. On entire functions which share one value CM with their first derivative[J]. *Results in Mathematics*, 1996, 30(1/2): 21–24.
- [7] GUNDERSEN G G, YANG L Z. Entire functions that share one value with one or two of their derivative[J]. *Journal of Mathematical Analysis and Applications*, 1998, 223(1): 88–95.
- [8] YU K W. On entire and meromorphic functions that share small functions with their derivatives[J]. *Journal of Inequalities in Pure and Applied Mathematics*, 2003, 4(1): 21.
- [9] 刘礼培, 袁建军. 亚纯函数与其微分多项式分担小函数的唯一性[J]. *西南师范大学学报 (自然科学版)*, 2006, 31(6): 11–14.

(编辑: 石 瑛)