

可持续性包装

混合蝙蝠算法在包装废弃物逆向物流中的应用

彭维, 朱云波

(重庆城市管理职业学院, 重庆 401331)

摘要: **目的** 为了提高蝙蝠算法(BA)求解包装废弃物逆向物流问题的性能。**方法** 在标准BA算法的基础上提出混合蝙蝠算法(HBA)。首先,构建新型蝙蝠表达式,使BA算法适用于包装废弃物逆向物流问题的求解。其次,引入自适应惯性权重,改造蝙蝠速度更新公式;然后,引入粒子群算法(PSO),对每次迭代中任一随机蝙蝠进行粒子群操作;最后,利用HBA算法对企业实例和标准算例进行仿真测试。**结果** 企业最优回收距离为776.63 km。与遗传算法(GA)、蚁群算法(ACO)和禁忌搜索算法(TS)相比,HBA算法能够求得已知最优解的标准算例个数最多为6个,求得最好解与已知最优解的平均误差最小为8.58%,平均运行时间最短为4.39 s。**结论** HBA算法的全局寻优能力、稳定性和运行速度均优于GA算法、ACO算法和TS算法。

关键词: 包装废弃物; 逆向物流; 蝙蝠算法; 粒子群算法

中图分类号: X79 **文献标识码:** A **文章编号:** 1001-3563(2019)01-0253-06

DOI: 10.19554/j.cnki.1001-3563.2019.01.037

Application of Hybrid Bat Algorithm in Packaging Waste Reverse Logistics

PENG Wei, ZHU Yun-bo

(Chongqing City Management College, Chongqing 401331, China)

ABSTRACT: The work aims to improve the performance of the bat algorithm (BA) used in solving the problem of packaging waste reverse logistics. Based on standard BA, a hybrid bat algorithm (HBA) was proposed. Firstly, a new bat expression was constructed to make BA suitable for the solution to the problem of packaging waste reverse logistics. Secondly, the speed update formula of bat was reconstructed by introducing the adaptive inertia weight. Then, particle swarm optimization algorithm (PSO) was introduced, which carried out particle swarm operation on any random bat in each iteration. Finally, simulation tests were carried out on enterprise examples and standard examples by HBA. The optimal recovery distance of enterprise was 776.63 km. HBA could get better results compared with genetic algorithm (GA), ant colony optimization algorithm (ACO) and tabu search algorithm (TS); such as, the number of standard examples of the known optimal solution was 6, the average error between the best solution and known optimal solution was 8.58%, and the minimum average running time was 4.39 s. HBA algorithm is superior to GA, ACO and TS in global optimization ability, stability and running speed.

KEY WORDS: packaging waste; reverse logistics; bat algorithm; particle swarm algorithm

包装废弃物逆向物流关系着资源环境的保护,历来是理论界和实践界关注的热点领域,其中的车辆路

收稿日期: 2018-08-30

基金项目: 重庆市教育委员会 2017 年度科学技术研究项目 (1609155488)

作者简介: 彭维 (1984—), 女, 硕士, 重庆城市管理职业学院讲师, 主要研究方向为物流与供应链管理。

通信作者: 朱云波 (1985—), 男, 硕士, 重庆城市管理职业学院实验师, 主要研究方向为逆向物流。

径优化问题 (Vehicle Routing Problem, VRP) 及求解算法更成为了重点研究对象^[1]。经过多年研究,学者们提出了大量求解 VRP 问题的启发式算法。例如, AZAD A S 等^[2]设计了基于 VRP 问题的随机模因算法。WANG K 等^[3]和周生伟等^[4]对传统 GA 算法种群生成方式和交叉操作进行改进,提高了算法全局寻优能力。WANG X 等^[5]则简化了蚂蚁构建 VRP 问题可行解的过程,提出了一种新型蚁群算法。启发式算法已经逐渐成为 VRP 问题的主要求解算法。

蝙蝠算法是模拟蝙蝠觅食过程而设计的一种新型启发式算法,由 Yang Xinshe 教授于 2010 年首次提出。自提出以后,蝙蝠算法被广泛运用到函数优化、生产调度、模式识别和神经网络等领域。例如, LEMMA T A 等^[6]采用 BA 算法对燃气轮机发电系统模型进行优化,验证了 BA 算法和局部线性模型树算法结合的有效性。YANG X S 等^[7]将 BA 算法应用到拓扑优化领域,解决了其中的弹簧问题和减速器问题。JOSE J T 等^[8]利用 BA 算法求解了风力发电经济负荷分配问题。国内对 BA 算法的研究尚处于初级阶段,应用领域主要集中在连续域函数方面,很少将其用于 VRP 等离散型问题的求解。

1 包装废弃物逆向物流的数学描述与模型

1.1 定义参数

用 $G=(F,E)$ 表示包装废弃物逆向物流网络;其中, $F=(0,1,\dots,n)$ 表示节点集, 0 为回收中心, 其余节点为客户; $E=\{(i,j)|i,j\in F,i\neq j\}$ 表示弧集, d_{ij} 为客户 i 到 j 之间的距离; $K=\{1,2,\dots,m\}$ 表示车辆集合, m 为完成回收任务所需车辆总数; Q 为车辆最大载重; q_i 为客户 i 待回收包装废弃物质量; $x_{ijk}=\{0,1\}$ 表示车辆 k 是否由客户 i 前往客户 j 服务; $y_{ik}=\{0,1\}$ 表示车辆 k 是否服务客户 i 。

1.2 建立模型

包装废弃物逆向物流问题可以描述为^[9-10]:客户有一定重量的包装废弃物需要回收,回收中心位置、客户位置及包装废弃物重量已知,且车辆最大承载能力给定。要求在如下限制条件下设计行驶距离最短的回收路径:所有客户都有且只有一辆车服务;所有客户包装废弃物重量都不大于车辆最大承载能力;车辆不能超载。

包装废弃物逆向物流问题的数学模型为:

$$\min Z = \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m d_{ij} x_{ijk} \quad (1)$$

$$\text{条件: } \sum_{i=1}^n q_i y_{ik} \leq Q, k=1, 2, \dots, m \quad (2)$$

$$\sum_{k=1}^m y_{ik} = 1, i=1, 2, \dots, n \quad (3)$$

$$\sum_{i=1}^n x_{ijk} = y_{jk}, j=1, 2, \dots, n, k=1, 2, \dots, m \quad (4)$$

$$\sum_{j=1}^n x_{ijk} = y_{ik}, i=1, 2, \dots, n, k=1, 2, \dots, m \quad (5)$$

$$x_{ijk}(x_{ijk}-1)=0 \quad (6)$$

$$y_{ik}(y_{ik}-1)=0 \quad (7)$$

其中,式(1)表示行驶距离最短的目标函数;式(2)表示车辆不能超载;式(3)表示客户只能由1辆车服务;式(4—5)表示变量 x_{ijk} 和 y_{ik} 之间的关系;式(6—7)表示 0~1 变量约束。

2 混合蝙蝠算法在包装废弃物逆向物流中的应用

2.1 基本蝙蝠算法

BA 算法主要通过飞行速度、脉冲频率和脉冲响度的更新来搜索最优解,算法核心步骤如下所述^[11]。

1)根据式(8—10)更新蝙蝠 i 在时刻 t 的速度 V_i^t 和位置 X_i^t 。

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \quad (8)$$

$$V_i^t = V_i^{t-1} + (X_i^{t-1} - X_s)f_i \quad (9)$$

$$X_i^t = X_i^{t-1} + V_i^t \quad (10)$$

式中: $\beta \in [0,1]$ 为服从均匀分布的随机向量; X_s 为当前最优蝙蝠。

2)根据式(11—12)对每只蝙蝠进行局部搜索。

$$A^t = \langle A_i^t \rangle \quad (11)$$

$$X_{\text{new}} = X_{\text{old}} + \varepsilon A^t \quad (12)$$

式中: ε 为随机数, $\varepsilon \in [-1,1]$; A_i^t 为时刻 t 蝙蝠 i 的脉冲响度; A^t 为时刻 t 所有蝙蝠平均响度。

3)根据式(13—14)更新脉冲响度 A 和脉冲发射频率 r 。

$$A_i^{t+1} = \alpha A_i^t \quad (13)$$

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)] \quad (14)$$

式中: α, γ 为常量,通常取值为 0.9。随着迭代次数的增加,可得到:

$$A_i^t \rightarrow 0, r_i^t \rightarrow r_i^0, t \rightarrow \infty \quad (15)$$

2.2 改进蝙蝠算法在包装废弃物逆向物流中的应用

2.2.1 构造个体表达式

根据包装废弃物逆向物流的特点,文中设计了一种离散型个体表达式,即用向量 $X=(X_v, X_r)$ 表达蝙蝠位置向量,其中 n 维向量 X_v 表示服务客户的车辆,

n 维向量 X_i 表示车辆在对对应路径中服务客户的次序。已知某包装废弃物逆向物流网络有 9 个客户, 需要 3 辆车才能完成回收任务, 某蝙蝠对应位置向量为:

客户点: 1 2 3 4 5 6 7 8 9

X_v : 1 3 2 3 3 1 2 1 2

X_r : 2 3 2 2 1 1 3 3 1

该蝙蝠表示的回收路径为:

车辆 1: 0 6 1 8 0

车辆 2: 0 9 3 7 0

车辆 3: 0 5 4 2 0

相应地, 蝙蝠速度向量用 $V=(V_v, V_r)$ 表示。

2.2.2 改进蝙蝠算法在包装废弃物逆向物流中的实现步骤

1) 随机生成 n 维 $[1, m]$ 的整数和 $[1, n]$ 的实数作为蝙蝠 i 位置向量 $X=(X_a, X_b)$, n 维 $[-(m-1), (m-1)]$ 的整数和 $[-(n-1), (n-1)]$ 的实数作为蝙蝠 i 速度向量 $V=(V_a, V_b)$, 直到达到初始种群规模。

2) 根据 Eval 函数计算每只蝙蝠 i 的目标函数值 $Z(i)$, 找出种群最优蝙蝠 X_* 。

3) 判断算法是否达到最大迭代次数。若达到, 输出最优蝙蝠, 算法停止; 反之, 进入步骤 4)。

4) 根据式 (8—10) 更新蝙蝠 i 的位置向量和速度向量, 得到 $X(i)$ 和 $V(i)$ 。其中, $V(i)$ 向上取整, 且当 $X(i)$, $V(i)$ 超出边界时按边界取值。

5) 生成随机数 rand1, 若 rand1 $> r(i)$, 则根据式 (11—12) 对蝙蝠 i 进行局部搜索, 得到待定最新解 X_{new} , 转入步骤 6)。反之, 转入步骤 7)。

6) 生成随机数 rand2, 若 rand2 $> A(i)$ 且 $Z[X_{\text{new}}] < Z[X_{\text{old}}]$, 接受最新解 X_{new} 。反之, 拒绝 X_{new} 。

7) 根据式 (13) 和式 (14) 更新脉冲响应度 $A(i)$ 和脉冲发射频率 $r(i)$, 返回步骤 3)。

2.2.3 Eval 函数

根据蝙蝠表达式特征, 文中提出求解目标函数值的 Eval 函数, 具体计算过程为如下所述。

1) 整数化处理。将蝙蝠 X_b 向量升幂排序, 并进行整数化处理。假设某只蝙蝠位置向量为:

客户点: 1 2 3 4 5 6 7 8 9

X_a : 1 2 3 3 3 1 3 1 2

X_b : 2.1 3.4 3.6 1.2 1.9 4.3 2.7 4.8 5.1

可知, 客户 3, 4, 5, 7 均由车辆 3 服务, 对应的 X_b 值分别为 (3.6 1.2 1.9 2.7)。其中, 客户 4 的 X_b 值最小为 1.2, 可以将其处理为 1; 客户 3 的 X_b 值最大为 3.6, 可处理为 4。由此可得, 这些客户对应的 X_b 值整数化处理后为 (4 1 2 3)。同理, 对其他 X_b 值进行整数化处理可得该蝙蝠 X_b 向量为 (1 4 1 2 2 3 3 2)。

2) 计算目标函数值。在式 (1) 中加入惩罚系数

C (非常大的正实数), 可得到蝙蝠目标函数值计算公式为:

$$\min Z = \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m d_{ij} x_{ijk} + C \sum_{k=1}^m \max(\sum_{i=0}^n q_i y_{ik} - Q, 0) \quad (16)$$

2.3 混合蝙蝠算法在包装废弃物逆向物流中的应用

与其他启发式算法一样, 改进 BA 算法存在“早熟”、收敛速度过慢等缺陷。对此, 文中在改进 BA 算法基础上提出了 HBA 算法。

2.3.1 自适应惯性权重的引入

如式 (17) 所示, 为加快算法收敛速度, 提高全局寻优能力, 文中引入惯性权重因子 ω 对蝙蝠速度更新公式进行改造:

$$V_i^t = \omega V_i^{t-1} + (X_i^{t-1} - X_*) f_i \quad (17)$$

其中, ω 呈现递减规律。

$$\omega(t) = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \cdot \exp[-\eta(\frac{t}{T})^2] \quad (18)$$

式中: $\omega_{\max}$ 和 $\omega_{\min}$ 为 $\omega(t)$ 的最大值和最小值; η 为 (1, T) 之间的随机数; t 为当前迭代次数; T 为算法最大迭代次数。分析可知, 在算法进化前期, ω 取值较大, 有助于扩大算法搜索空间, 提高算法全局寻优能力。在算法进化后期, ω 逐渐变小, 有助于加强算法局部搜索能力, 加快算法收敛速度。

2.3.2 粒子群算法的引入

PSO 算法是根据鸟群觅食过程而设计的一种智能启发式算法^[12-13]。算法中, 粒子通过利用自身最优解和种群最优解不断更新自己的方式, 来寻找问题最优解。用 $X_i(x_{i1}, x_{i2} \cdots x_{iD})$ 和 $V_i(v_{i1}, v_{i2} \cdots v_{iD})$ 分别表示 D 维空间中粒子 i 的位置和速度, 假设粒子 i 最优解为 $P_i = (p_{i1}, p_{i2} \cdots p_{iD})$, 种群最优解为 $P_g = (p_{g1}, p_{g2} \cdots p_{gD})$, 则粒子的位置和速度更新公式为^[8]:

$$v_{id}^{t+1} = w v_{id}^t + c_1 r_1 (p_{id}^t - x_{id}^t) + c_2 r_2 (p_{gd}^t - x_{id}^t) \quad (19)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (20)$$

式中: w 为粒子权重; c_1 和 c_2 为学习因子, 通常取值为 2; r_1 和 r_2 为 (0, 1) 之间的随机数; $c_1 r_1 (p_{id}^t - x_{id}^t)$ 被称为“认知部分”; $c_2 r_2 (p_{gd}^t - x_{id}^t)$ 为社会部分。

PSO 算法具有随机全局搜索能力强、局部深度搜索能力强、调控参数少等优点, 被广泛应用于神经网络、电力系统、控制工程、物流配送等领域。文中通过算法融合的形式将 PSO 算法嫁接到 BA 算法中, 利用 PSO 算法优点克服 BA 算法进化后期种群多样性差、局部搜索能力弱等问题, 进一步提高 BA 算法寻优能力和收敛速度。具体而言, 在 BA 算法每次迭代中, 首先根据第 2.2.1 节的方式构造粒子

个体,然后利用 PSO 算法随机对一个蝙蝠进行局部优化,并将优化结果与式(12)产生个体进行比较,选择较优个体作为局部搜索备选蝙蝠。为减少算法时间复杂度,每次迭代仅对一个蝙蝠进行一次粒子群操作。

2.3.3 混合蝙蝠算法在包装废弃物逆向物流中的应用

综上所述, HBA 算法求解包装废弃物逆向物流问题的步骤如下所述。

1) 算法参数设置。包括:算法最大迭代次数 T , 蝙蝠种群规模 N , 蝙蝠 i 的初始脉冲响度 A_i^0 , 脉冲发射频率 r_i^0 , 脉冲频率最小值 $f_{\min}$ 和最大值 $f_{\max}$, 脉冲响度衰减系数 α 和脉冲发射频率增加系数 γ , 粒子群操作进化代数 M , 粒子权重 w , 学习因子 c_1 和 c_2 。

2) 蝙蝠种群初始化。根据第 2.2.2 节方式生成 N 个蝙蝠作为初始种群,按照第 2.2.3 节 Eval 函数计算每只蝙蝠 i 的目标函数值 $Z(i)$, 找出种群最优蝙蝠 X_* 。

3) 判断算法是否达到最大迭代次数 T 。若达到, 输出最优蝙蝠, 算法停止; 反之, 进入步骤 4)。

4) 更新蝙蝠位置和速度向量。根据式(8)(10), (17)和(18)更新蝙蝠 i , 转入步骤 5)。其中, $V(i)$ 向上取整, 且当 $X(i)$, $V(i)$ 超出边界时按边界取值。

5) 局部搜索。生成随机数 rand1 , 若 $\text{rand1} > r(i)$, 则根据式(11)和(12)对蝙蝠 i 进行局部搜索, 得到 $\bar{X}_{\text{new}}$; 将当前蝙蝠种群作为 PSO 算法初始种群, 根据式(19)和式(20)进行粒子群操作, 找出最优粒子 $\underline{X}_{\text{new}}$, 比较 $Z[\bar{X}_{\text{new}}]$ 与 $Z[\underline{X}_{\text{new}}]$ 大小, 选择较小个体作为蝙蝠 i 的待定最新解 X_{newo} 。

每次迭代只随机对一个蝙蝠进行粒子群操作。未进行粒子群操作的蝙蝠, 根据式(11)和式(12)得出蝙蝠对应待定最新解 X_{new} 即可。

6) 生成随机数 rand2 , 若 $\text{rand2} > A(i)$ 且 $Z[X_{\text{new}}] < Z[X_{\text{old}}]$, 接受最新解 X_{newo} 。反之, 拒绝 X_{newo} 。根据式(13)和式(14)更新 $A(i)$, $r(i)$, 返回步骤 3)。

3 仿真分析

为验证算法的有效性和求解性, 分别对企业实例和标准算例^[14]进行仿真测试。实例中, 已知回收中心坐标为(30,30), 需要向 30 个客户回收包装废弃物, 各客户待回收量及位置信息见表 1, 车辆最大载质量为 8 t。

为合理安排线路, 需要预先估计完成回收任务所需车辆数。文中采用式(21)确定参与回收任务的车辆数:

$$m = \left\lceil \sum_{i=1}^n q_i / \partial Q \right\rceil + 1 \quad (21)$$

表 1 客户信息
Tab.1 Customer information

序号	$[X,Y]$	待回收量	序号	$[X,Y]$	待回收量
1	[45,83]	3.01	16	[66,58]	2.34
2	[73,63]	0.43	17	[60,82]	1.19
3	[26,52]	1.03	18	[59,75]	0.23
4	[81,69]	1.57	19	[23,31]	0.23
5	[37,36]	2.00	20	[45,59]	3.14
6	[44,42]	1.02	21	[11,43]	2.99
7	[25,10]	0.78	22	[17,37]	1.37
8	[40,40]	1.14	23	[40,75]	1.90
9	[23,88]	3.04	24	[52,4]	0.45
10	[18,87]	0.38	25	[89,51]	2.22
11	[64,45]	1.00	26	[37,37]	0.37
12	[36,98]	4.02	27	[34,45]	0.96
13	[69,55]	2.14	28	[20,23]	2.50
14	[36,53]	0.36	29	[44,57]	4.03
15	[37,61]	1.79	30	[71,19]	2.54

式中: $[\]$ 为取不大于括号内数值的最大整数; $0 < \partial < 1$ 为车容量装载系数, 文中设置为 0.95, 可得所需车辆数 $m=7$ 。

将 HBA 算法的参数设置为: 最大迭代次数 $T=80$; 蝙蝠种群规模 $N=50$, 脉冲响度衰减系数 $\alpha=0.9$, 脉冲发射频率增加系数 $\gamma=0.9$; 粒子群操作进化代数 $M=40$, 粒子权重 $w=0.729$, 学习因子 $c_1=c_2=2$ 。在“Intel i5-8250U, 8GB RAM, Win XP, Matlab 2012”环境下, 利用 HBA 算法对该实例进行求解, 求得最优行驶距离为 776.63 km, 具体回收方案见表 2。

表 2 最优回收方案
Tab.2 Optimal recycling scheme

序号	回收路径	回收距离/km
1	0→19→22→21→28→0	58.18
2	0→11→13→30→24→7→0	156.88
3	0→3→23→1→17→18→0	134.36
4	0→25→4→2→16→6→26→0	146.63
5	0→20→29→0	65.30
6	0→12→9→10→0	148.01
7	0→5→8→15→14→27→0	67.27

由表 2 可知, HBA 算法对包装废弃物逆向物流问题的求解是可行的, 且求解效果较佳。为进一步验证算法求解性能, 采用 HBA 算法、GA 算法^[4]、TS 算法^[15]和 ACO 算法^[16]对各标准算例分别进行 80 次求解, 统计求得的最好解、最好解与已知最优解的误差 (e_1)、平均值、平均值与已知最优解的误差 (e_2) 和平均运行时间 (t) 等仿真结果, 见表 3。表 3 中,

表 3 GA 算法、TS 算法、ACO 算法和 HBA 算法仿真结果对比
Tab.3 Simulation results comparison of GA, TS, ACO and HBA

算例	客户数	GA算法			TS算法			ACO算法			HBA算法		
		$e_1/\%$	$e_2/\%$	t/s	$e_1/\%$	$e_2/\%$	t/s	$e_1/\%$	$e_2/\%$	t/s	$e_1/\%$	$e_2/\%$	t/s
A-n32-k5	32	0.00	2.23	4.33	0.98	2.87	2.13	0.00	1.19	5.10	0.00	1.78	2.05
A-n38-k5	38	0.00	4.89	4.87	1.74	5.91	3.54	0.78	5.46	6.32	0.00	4.31	3.58
A-n39-k5	39	0.00	5.05	3.76	3.65	7.06	2.07	0.00	6.77	5.66	0.00	4.28	2.02
A-n48-k7	46	2.25	7.44	6.15	2.11	7.98	3.97	3.56	7.13	6.37	0.00	6.48	3.45
A-n55-k9	55	2.19	9.93	7.14	5.86	12.43	4.51	4.01	11.42	8.29	0.00	9.75	4.42
A-n60-k9	60	3.08	12.34	7.19	6.78	11.14	4.29	6.87	13.09	7.73	0.00	10.48	4.18
A-n65-k9	65	4.40	11.69	8.48	6.94	14.22	5.11	6.26	12.56	9.65	1.28	11.36	4.96
A-n69-k9	69	4.62	13.31	9.33	8.13	16.13	7.03	8.77	15.23	11.28	0.42	12.99	7.05
A-n80-k10	80	6.78	17.09	11.20	9.67	22.83	7.92	9.13	20.03	14.52	2.30	15.79	7.83
平均值		2.59	9.33	6.94	5.61	11.17	4.51	4.38	10.32	8.32	0.44	8.58	4.39

e_1 =(最好解-已知最优解)/已知最优解 $\times 100\%$,单位为%; e_2 =(平均值-已知最优解)/已知最优解 $\times 100\%$,单位为%; t 表示算法 80 次求解的平均运行时间,单位为 s。

根据表 3,在全局寻优能力上,HBA 算法能够求得最优解(e_1 值为 0)的算例个数为 6 个,GA 算法为 3 个,ACO 算法为 2 个,TS 算法不能求得任一算例的最优解,且各算法对应的 e_1 平均值依次为 0.44%,2.59%,4.38%和 5.61%,说明 HBA 算法的全局寻优能力最强,GA 算法次之,而 ACO 算法和 TS 算法的全局寻优能力最差。在算法稳定性上,HBA 算法对应的 e_2 平均值最小为 8.58%,所以 HBA 算法的稳定性最强;GA 算法对应的 e_2 平均值次之为 9.33%,GA 算法的稳定性次之;ACO 算法和 TS 算法对应的 e_2 平均值最大,分别为 10.32%和 11.17%,故 ACO 算法和 TS 算法的稳定性最差。在算法运行速度上,HBA 算法、GA 算法、ACO 算法和 TS 算法对应的 t 平均值分别为 4.39,6.94,8.32,4.51 s,说明 HBA 算法运行速度最快,TS 算法次之,而 GA 算法和 ACO 算法最慢。

由此可见,HBA 算法是有效的,且在全局寻优能力、稳定性和运行速度上均优于 GA,ACO 和 TS 等智能算法。这主要得益于自适应惯性权重和 PSO 算法的引入,使算法前期搜索空间得以扩大,算法后期种群多样性得以提高,进而提高了算法寻优能力。同时,由于算法进行局部搜索时仅对一个蝙蝠进行一次粒子群操作,因此在运行时间上也没有造成过多的浪费。

4 结语

文中提出了求解包装废弃物逆向物流问题的 HBA 算法。该算法构造了适应 VRP 问题的蝙蝠个体

表达式,并针对算法“早熟”、收敛速度较慢等缺陷,引入了自适应惯性权重和 PSO 算法。仿真实验表明,HBA 算法的求解性能优于 GA,ACO 和 TS 等智能算法,能够很好地适应包装废弃物逆向物流问题的求解。今后将对 HBA 算法在其他领域的应用进行研究。

参考文献:

[1] 张昇. 包装物回收物流中的车辆路径优化问题[J]. 包装工程, 2017, 38(17): 233—238.
ZHANG Yi. Vehicle Routing Optimization Problem in Package Recycling Logistics[J]. Packaging Engineering, 2017, 38(17): 233—238.

[2] AZAD A S, ISLAM M, CHAKRABORTY S. A Heuristic Initialized Stochastic Memetic Algorithm for MDPVRP With Interdependent Depot Operations[J]. IEEE Transactions on Cybernetics, 2017, 47(12): 4302—4315.

[3] WANG K, LAN S, ZHAO Y. A Genetic-algorithm-based Approach to the Two-echelon Capacitated Vehicle Routing Problem with Stochastic Demands in Logistics Service[J]. Journal of the Operational Research Society, 2017: 1—13.

[4] 周生伟, 蒋同海, 张荣辉. 改进遗传算法求解 VRP 问题[J]. 计算机仿真, 2013, 30(12): 140—143.
ZHOU Sheng-wei, JIANG Tong-hai, ZHANG Rong-hui. Improved Genetic Algorithm for VRP[J]. Computer Simulation, 2013, 30(12): 140—143.

[5] WANG X, CHOI T M, LIU H, et al. Novel Ant Colony Optimization Methods for Simplifying Solution Construction in Vehicle Routing Problems[J]. IEEE Transactions on Intelligent Transportation Systems, 2016, 17(11): 3132—3141.

[6] LEMMA T A, HASHIM F B M. Use of Fuzzy Systems and Bat Algorithm for Exergy Modeling in a Gas Tur-

- bine Generator[C]// Humanities, Science and Engineering, IEEE, 2012: 305—310.
- [7] YANG X S, KARAMANOGLU M, Fong S. Bat Algorithm for Topology Optimization in Microelectronic Applications[C]// International Conference on Future Generation Communication Technology, IEEE, 2013: 150—155.
- [8] JOSE J T. Economic Load Dispatch Including Wind Power Using Bat Algorithm[C]// International Conference on Advances in Electrical Engineering IEEE, 2014: 1—4.
- [9] 樊贵香. 混合模拟植物生长算法在包装废弃物配送中的应用[J]. 包装工程, 2016, 37(13): 43—49.
FAN Gui-xiang. Application of Hybrid Plant Growth Simulation Algorithm in Packaged Products Distribution[J]. Packaging Engineering, 2016, 37(13): 43—49.
- [10] 裴小兵, 贾定芳. 基于模拟退火算法的城市物流多目标配送车辆路径优化研究[J]. 数学的实践与认识, 2016(2): 105—113.
PEI Xiao-bing, JIA Ding-fang. Optimizing Multi-objective Vehicle Routing Problem in City Logistics Based on Simulated Annealing Algorithm[J]. Mathematics in Practice and Theory, 2016(2): 105—113.
- [11] GANDOMI A H, YANG X S, ALAVI A H, et al. Bat Algorithm for Constrained Optimization Tasks[J]. Neural Computing and Applications, 2013, 22(6): 1239—1255.
- [12] AI T J, KACHITVICHYANUKUL V. Particle Swarm Optimization and Two Solution Representations for Solving the Capacitated Vehicle Routing Problem[J]. Computers & Industrial Engineering, 2009, 56(1): 380—387.
- [13] GOKSAL F P, ALTIPARMAK F, KARAOGLAN I. A Hybrid Particle Swarm Optimization for Vehicle Routing Problem with Simultaneous Pickup and Delivery[J]. Computers & Industrial Engineering, 2013, 65(1): 39—53.
- [14] SALMEN A, AHMAD I, AI M B. Particle Swarm Optimization for Task Assignment Problem[J]. Microprocessors and Microsystems, 2002, 26: 363—371.
- [15] REN C Y. Tabu Search Algorithm for Capacitated Vehicle Routing Problem[J]. Advanced Materials Research, 2013, 753—755: 3060—3063.
- [16] GIANLUCA. Ant Colony Optimization for Capacitated Vehicle Routing Problem[J]. Journal of Computer Science, 2012, 8(6): 846—852.