

# 包装废弃物回收车辆路径问题的改进遗传算法

张异

(重庆工商职业学院, 重庆 401520)

**摘要:** **目的** 采用优化传统遗传算法 (GA) 研究包装废弃物回收车辆路径问题 (VRP) 的性能。**方法** 提出改进遗传算法 (IGA)。首先, 设计基于贪婪算法的初始种群生成算子, 提高初始种群质量; 其次, 设计根据适应度值大小、进化代数等自适应调整的交叉和变异概率; 然后, 设计最大保留交叉算子, 保证种群的多样性; 最后, 对企业实例和标准算例进行仿真测试。**结果** 采用 IGA 算法、蚁群算法 (ACO) 能求得算例最优解, 且 IGA 算法运行速度快于 ACO 算法, 分支界定算法 (BBM)、传统 GA 算法无法求得算例最优解。**结论** 与 BBM 算法、传统 GA 算法和 ACO 算法相比, IGA 算法求解包装废弃物回收 VRP 问题的整体性能更优。

**关键词:** 包装废弃物; 回收; 车辆路径问题; 遗传算法

**中图分类号:** TP301 **文献标识码:** A **文章编号:** 1001-3563(2018)17-0147-06

**DOI:** 10.19554/j.cnki.1001-3563.2018.17.024

## Improved Genetic Algorithm for Vehicle Routing Problem in Packaging Waste Recycling

ZHANG Yi

(Chongqing Technology and Business Institute, Chongqing 401520, China)

**ABSTRACT:** The work aims to optimize the performance of traditional genetic algorithm (GA) used to solve the vehicle routing problem (VRP) in packaging waste recycling. An improved genetic algorithm (IGA) was put forward. Firstly, in order to improve the quality of initial population, the initial population generation operator based on greedy algorithm was designed; secondly, the crossover and mutation probabilities adaptively adjusted based on fitness values and evolutionary algebras were designed; then, the maximum preserved crossover operator was designed to ensure population diversity. Finally, simulation tests were carried out on an enterprise instance and standard examples. IGA and ant colony algorithm (ACO) were used to get the optimal solution of the example, and IGA ran faster than ACO. The branch and bound algorithm (BBM) and traditional GA could not find the optimal solution of the example. Compared with the BBM, traditional GA and ACO, IGA has better overall performance in solving the VRP problem of packaging waste recycling.

**KEY WORDS:** packaging waste; recycling; vehicle routing problem; genetic algorithm

近年来, 随着电子商务高速发展, 电商包裹需求量呈井喷式增长<sup>[1-2]</sup>。这在推动包装行业高速发展的同时, 也加剧了我国资源紧缺的严峻现状, 并对环境造成严重污染, 加强包装废弃物的循环利用已经迫在眉睫。对此, 国家先后出台了《包装废弃物的处理与利用通则》《农药包装废弃物回收处理管理办法》等法律法规, 对包装废弃物回收的重大战略意义、回收处理体系建设等做出明确规定。在此背景下, 包装废

物回收问题成为了理论界和实践界关注的热点问题, 其中 VRP 问题及求解算法更成为了重点研究对象。目前, 包装废弃物回收 VRP 问题的求解算法非常多, 大致可以分为精确算法和智能算法 2 类<sup>[3]</sup>。精确算法通常指可以寻找到全局最优解的算法, 但其计算量会随着问题规模的增大而呈指数增加。智能算法是学者们研究的重点, 国内外学者先后提出了 GA 算法、粒子群算法、ACO 算法<sup>[4]</sup>等求解大规模 VRP 问题。

收稿日期: 2018-05-24

基金项目: 重庆市教委人文社科项目 (16SKGH209); 重庆工商职业学院重点项目 (ZD2014-03)

作者简介: 张异 (1973—), 女, 硕士, 重庆工商职业学院副教授, 主要研究方向为供应链管理、物流管理。

传统 GA 算法是由 John. H. Holland 教授于 1975 年首次提出的一种智能进化算法, 具备局部搜索能力强、并发性能较好等优点<sup>[5-7]</sup>。经过几十年的发展, GA 算法被广泛应用到组合优化、机器学习、信号处理、人工智能等领域。例如, 文献[8]借鉴 GA 算法的精英保留策略和交叉机制, 提出了一种基于精英交叉的二进制蝙蝠算法, 为 0—1 背包问题研究提供了新方向。文献[9]和文献[10]则分别在 GA 算法基础上设计了噪音分割算法和医学图像保护算法。文献[11]采用 GA 算法对 BP 神经网络训练的初始权值阈值进行优化, 有效提高了 BP 神经网络的整体性能。然而, 分析相关文献可知, 传统 GA 算法在具体应用中, 仍然存在运行速度较慢、容易陷入局部最优解等问题, 算法性能有待提升。基于上述分析, 文中提出了求解包装废弃物回收 VRP 问题的 IGA 算法。

### 1 包装废弃物回收的车辆路径问题

包装废弃物回收的 VRP 问题可以描述为<sup>[12]</sup>: 回收中心派遣一定数量的车辆去回收客户包装废弃物。回收中心位置、客户位置及需要回收的包装废弃物质量已知, 且车辆载重量给定, 要求在满足如下几个条件的情况下合理规划车辆行驶线路, 使整体行驶距离最短: 回收中心约束, 所有车辆均从同一回收中心出发, 完成任务后返回回收中心; 访问唯一性约束, 所有客户均被访问且只能被访问 1 次; 车辆载重约束, 所有车辆均不能超载。

用  $V$  表示回收网络节点集合,  $V=\{0,1,\dots,n\}$ , 其中 0 表示回收中心, 其余节点表示客户;  $K$  表示车辆集合,  $K=\{1,2,\dots,m\}$ ,  $m$  为完成回收任务所需车辆数;  $Q$  为车辆最大载质量;  $d_{ij}$  为客户  $i$  到  $j$  的距离;  $q_i$  为客户  $i$  需要回收的包装废弃物质量。

设定 2 个决策变量:

$$x_{ijk} = \begin{cases} 1, & \text{车辆 } k \text{ 完成客户 } i \text{ 的回收任务后前往客户 } j \\ 0, & \text{否则} \end{cases}$$

$$y_{ik} = \begin{cases} 1, & \text{车辆 } k \text{ 执行客户 } i \text{ 的回收任务} \\ 0, & \text{否则} \end{cases}$$

则包装废弃物回收 VRP 问题的数学模型为:

$$\min Z = \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} d_{ij} x_{ijk} \quad (1)$$

约束条件为:

$$\sum_{i \in V \setminus \{0\}} q_i \sum_{j \in V} x_{ijk} \leq Q, \forall k \in K \quad (2)$$

$$\sum_{k \in K} \sum_{i \in V} x_{ijk} = 1, \forall j \in V \setminus \{0\} \quad (3)$$

$$\sum_{k \in K} \sum_{j \in V} x_{ijk} = 1, \forall i \in V \setminus \{0\} \quad (4)$$

$$\sum_{i \in V \setminus \{0\}} x_{i0k} = \sum_{j \in V \setminus \{0\}} x_{0jk} = 1, \forall k \in K \quad (5)$$

$$\sum_{k \in K} \sum_{i \in V \setminus \{0\}} x_{i0k} = \sum_{k \in K} \sum_{j \in V \setminus \{0\}} x_{0jk} = m \quad (6)$$

$$\sum_{k \in K} \sum_{i \in S} \sum_{j \in \bar{S}} x_{ijk} \geq 1, \forall S \subseteq V \setminus \{0\}, |S| \geq 2 \quad (7)$$

$$\sum_{j \in V} x_{ijk} = \sum_{j \in V} x_{jik}, \forall k \in K, \forall i \in V \setminus \{0\} \quad (8)$$

$$x_{ijk} (x_{ijk} - 1) = 0, \forall i, j \in V, \forall k \in K \quad (9)$$

$$y_{ik} (y_{ik} - 1) = 0, \forall i \in V, \forall k \in K \quad (10)$$

其中, 式 (1) 表示行驶路径最短的目标函数; 式 (2) 表示车辆载重约束; 式 (3) 和式 (4) 表示客户访问唯一性约束; 式 (5) 和式 (6) 回收中心约束; 式 (7) 为子回路约束; 式 (8) 表示客户车辆流守恒约束; 式 (9) 和式 (10) 为 0~1 变量约束。

### 2 改进混合遗传算法

文中主要对传统 GA 算法的初始种群生成方式、交叉变异概率和操作进行改进, 具体步骤如下所述。

#### 2.1 编码设计

采用自然数编码结构, 将每条行驶路线编译成长度为  $n+m+1$  的个体, 如  $(0, i_1, i_2, \dots, i_e, 0, i_o, \dots, i_k, 0, \dots, 0, i_p, \dots, i_n, 0)$ , 其中 0 表示回收中心,  $i_j$  为对应子路径上第  $j$  个客户。

#### 2.2 种群初始化

现有 GA 算法大多是随机生成初始种群, 导致初始种群中含有大量适应度值较差甚至不可行个体, 对算法求解质量和效率造成了严重影响<sup>[13]</sup>。针对该问题, 文中提出了基于贪婪算法的初始种群生成算子。

1) 随机生成客户全排列。随机选取一个客户作为当前服务对象  $C_{\text{current}}$ , 并将其纳入客户全排列集合  $C_{\text{route}}$ ; 对未纳入  $C_{\text{route}}$  的客户进行搜索, 选出距离  $C_{\text{current}}$  距离最近的客户  $C_{\text{next}}$ , 将其纳入  $C_{\text{route}}$ , 并作为  $C_{\text{current}}$  继续搜索距离最近的客户, 直到所有客户都纳入  $C_{\text{route}}$  中,  $C_{\text{route}}$  即为客户全排列。

2) 生成含 0 的行驶路线。首先, 访问  $C_{\text{route}}$  中的基因, 若  $\sum_{i=1}^{s-1} q_i \leq Q$  且  $\sum_{i=1}^s q_i > Q$ , 则将 0 插入第  $s$  位, 把原来第  $s$  位及以后的基因逐一往后移动一位。其次, 继续按约束条件在  $C_{\text{route}}$  中插入 0 基因, 直到访问到字符串末端。最后, 在  $C_{\text{route}}$  首部尾部分别插入 0, 则可生成一条完整路径。

3) 个体数量是否达到初始种群规模? 如果是, 初始种群生成完毕; 否则进入步骤 1)。

#### 2.3 适应度评价

在式 (1) 基础上, 引入惩罚系数计算个体目标函数值。如式 (11) 所示, 其中惩罚系数  $M$  取非常大的正实数。

$$f = \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} d_{ij} x_{ijk} + M \sum_{k \in K} \max(\sum_{i \in V} q_i y_{ik} - Q, 0) \quad (11)$$

2.4 选择操作

按照适应度值对所有个体进行从小到大排序。假设算法种群规模为  $N$ ，适应度值最小的个体编号为 0，则适应度值最大的个体编号为  $N-1$ 。随机生成一个符合标准正态分布的数值  $r$ ，可知  $r$  会有 99.9% 的概率落在  $[-3,3]$  内。取  $r^* = \lfloor \frac{r}{3} \rfloor$ （若  $r > 3$ ，则重新生成），

则  $r^*$  必然在  $[0,1]$  中。令  $a = r^*(N-1)$ ，选取第  $a$  号个体进入新种群。根据正态分布的特点可知，选取个体越靠近 0，其概率值越大，则选取较优个体的可能性就越高。

2.5 交叉概率和变异概率自适应调整

交叉概率和变异概率是交叉操作和变异操作的重要参数，直接影响着 GA 算法的全局寻优能力和收敛速度。如式 (12) 和 (13) 所示，文中设计了根据适应度值大小、进化代数等自适应调整的交叉概率  $p_c$  和变异概率  $p_m$ 。

$$p_c = \begin{cases} p_{c1} - \frac{(p_{c1} - p_{c2})(f_{avg} - f')}{(f_{avg} - f_{best})[1 + \exp(-t \cdot R/T \cdot N)]}, & f' < f_{avg} \\ p_{c1}^{[1 + \exp(-t \cdot R/T \cdot N)]}, & f' \geq f_{avg} \end{cases} \quad (12)$$

$$p_m = \begin{cases} p_{m1} - \frac{(p_{m1} - p_{m2})(f - f_{best})}{(f_{avg} - f_{best})[1 + \exp(t \cdot R/T \cdot N)]}, & f < f_{avg} \\ p_{m1}^{[1 + \exp(t \cdot R/T \cdot N)]}, & f \geq f_{avg} \end{cases} \quad (13)$$

式中： $f_{best}$  为当前种群最优适应度值； $f_{avg}$  为当前种群平均适应度值； $f'$  为待交叉个体中的较优适应度值； $f$  为待变异个体适应度值； $t$  为当前进化代数； $T$  为最大进化代数； $R$  为进化过程中最优解未发生改变的个体数； $N$  为种群规模； $p_{c1}, p_{c2}$  为最大和最小交叉概率； $p_{m1}, p_{m2}$  为最大和最小变异概率，且  $p_{c1} > p_{c2}, p_{m1} > p_{m2}$ ，取值范围为  $(0,1)$ ，可在进化过程中调整。

分析式 (12) 和 (13) 可知，在算法进化初期，由于种群个体差异较大， $f_{avg} - f_{best}$  的值也较大，再加之进化代数  $t$  和最优解未发生改变的个体数  $R$  较小，所以交叉概率  $p_c$  较大，可加快算法收敛速度。在算法进化后期，虽然种群个体趋于相同， $f_{avg} - f_{best}$  变小，进化代数  $t$  和最优解未发生改变的个体数  $R$  逐渐变大，导致交叉概率  $p_c$  逐渐变小，但变异概率  $p_m$  却变大，提高了种群多样性，有效避免了算法陷入局部最优解。

2.6 交叉算子操作

传统交叉算子容易对优秀子路径造成破坏，且在父个体相同时，无法产生新的子个体，降低了种群多样性<sup>[14-15]</sup>。对此，文中提出了最大保留交叉算子。如图 1 所示，首先将父个体中车辆实载率最高的子路径复制到另一个父个体对应的临时字符串首部。然后，删除该父个体中与临时字符串相同的基因。最后，按照 2.2 节所述方法对其余基因进行重新优化排列并合并到临时字符串中，即可得到交叉后的子个体。

2.7 变异操作

随机选取 2 条子路径中的非零点作为变异基点，将其尾部进行交换，完成变异操作。

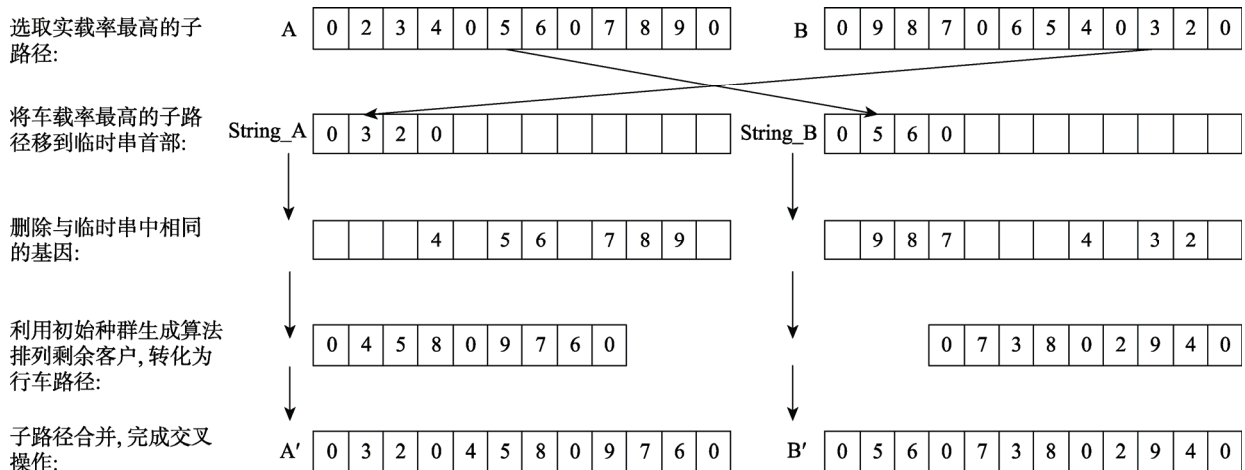


图 1 最大保留交叉算子

Fig.1 The maximum reservation crossover operator

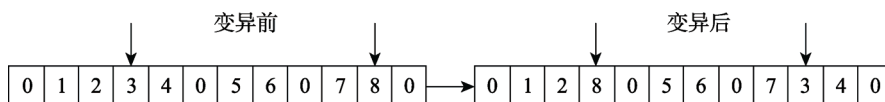


图 2 变异算子

Fig.2 The mutation operator

至此一次进化完毕,判断算法是否达到最大进化代数?达到,算法停止,输出最优解即为最优配送方案;否则,重复上述步骤。

### 3 仿真分析

为验证 IGA 算法的有效性,文中分别采用包装废弃物回收实例和 VRP 问题的标准测试算例进行仿真实验。

#### 3.1 回收实例

如表 1 所示,某企业要对 30 个客户的包装废弃物进行回收,客户位置和包装废弃物质量已知,回收中心坐标为 (50 km, 50 km),车辆最大载质量为 8 t。

在“Intel i5-8250U, 8GB RAM, Win XP, MATLAB 2012”环境下,利用 IGA 算法对该实例进行求解,最优配送方案见表 2。算法参数设置为:种群规模  $N=100$ 、最大进化次数  $T=60$ 、最大交叉概率  $p_{c1}=0.6$ 、最小交叉概率  $p_{c1}=0.3$ 、最大变异概率  $p_{m1}=0.6$ 、最小变异概率  $p_{m2}=0.002$ 、惩罚系数  $M=1\ 000\ 000$ 。

表 1 客户信息  
Tab.1 Customer information

序号	位置/km	回收量/t	序号	位置/km	回收量/t
1	[45,83]	2.13	16	[23,58]	0.67
2	[17,63]	2.64	17	[56,28]	1.19
3	[31,52]	0.37	18	[27,75]	0.23
4	[81,69]	1.68	19	[77,31]	2.89
5	[17,35]	2.10	20	[45,59]	3.55
6	[48,36]	1.02	21	[11,25]	2.04
7	[25,10]	0.53	22	[48,39]	1.37
8	[19,40]	1.27	23	[37,75]	1.80
9	[78,88]	0.99	24	[52,4]	0.51
10	[20,87]	1.01	25	[41,51]	2.44
11	[34,51]	2.00	26	[26,37]	0.38
12	[45,98]	4.25	27	[16,52]	0.78
13	[25,47]	1.98	28	[66,44]	2.80
14	[32,53]	0.62	29	[36,57]	3.70
15	[74,61]	1.73	30	[71,39]	1.50

表 2 最优配送方案  
Tab.2 Optimal distribution scheme

序号	行驶路径	行驶距离/km	序号	行驶路径	行驶距离/km
1	0→6→24→7→21→5→8→26→0	146.53	5	0→1→12→10→18→0	123.55
2	0→29→14→3→11→0	41.92	6	0→20→0	20.59
3	0→25→0	18.11	7	0→23→2→16→27→13→0	104.01
4	0→28→15→4→9→0	112.94	8	0→30→19→17→22→0	79.7

由表 2 可知,完成此次回收任务所需车辆数为 8 辆,最优行驶距离为 647.35 km。为进一步验证 IGA 算法的求解性能,分别采用传统 GA 算法<sup>[7]</sup>、ACO 算法<sup>[4]</sup>和分支界定法 (BBM)<sup>[16]</sup>对上述实例计算 40 次,统计最优值、最劣值、平均计算时间等指标,见表 3。

表 3 各算法仿真结果对比分析  
Tab.3 Comparison and analysis of simulation results of each algorithm

算法	最优值 /km	最劣值 /km	平均值 /km	计算时间/s	搜索成功率/%
ACO 算法	647.35	863.76	755.14	19.32	15
GA 算法	660.27	958.90	850.40	11.56	0
BBM 算法	960.33	960.33	960.33	3.52	0
IGA 算法	647.35	658.13	650.68	9.30	95

通过表 3 看出,在全局寻优能力上,IGA 算法和 ACO 算法均能求得最优解,GA 算法和 BBM 算法均陷入局部最优解,IGA 算法最优解搜索成功率、平均值和最劣值最优分别为 95%、650.68 km、658.13 km,

ACO 算法次之,GA 算法和 BBM 算法最差,说明 IGA 算法全局优化能力优于 ACO 算法、GA 算法和 BBM 算法。在计算时间上,BBM 计算时间最短,IGA 算法次之,GA 算法和 ACO 算法最长,说明 IGA 算法在提高全局寻优能力的同时并没有降低运行速度。

#### 3.2 标准算例

从文献[17]中下载关于 VRP 问题的标准算例。分别采用上述算法进行求解,并对每个算例运行 40 次,统计最优值、平均值、平均计算时间等指标,见表 4。

根据表 4,IGA 算法能求得 C1, C2, C3 和 C4 最优解,各算例对应平均值最优,ACO 算法仅能求得 C1 最优解,各算例对应平均值次之,GA 算法和 BBM 算法均陷入了局部最优解,对应平均值最大,表明 IGA 算法全局寻优能力最强,ACO 算法、GA 算法和 BBM 算法依次降低。在计算时间上,BBM 算法计算各算例所耗时间最短,IGA 算法次之,GA 算法和 ACO 算法最长,说明 BBM 算法、IGA 算法、GA 算法和 ACO 算法运行速度依次降低。综上分析,IGA 算法全局优化能力优于 ACO 算法、GA 算法和 BBM 算法,

表 4 各算法仿真结果对比分析  
Tab.4 Comparison and analysis of simulation results of each algorithm

实例	客户数量	已知最优解/km	ACO 算法			GA 算法			IGA 算法			BBM 算法		
			最优值/km	平均值/km	计算时间/s	最优值/km	平均值/km	计算时间/s	最优值/km	平均值/km	计算时间/s	最优值/km	平均值/km	计算时间/s
C1	50	524.6	524.6	599.3	19.9	531.1	636.5	15.4	524.6	560.3	10.1	637.8	637.8	6.3
C2	75	835.3	840.5	903.6	22.5	854.9	998.8	23.3	835.3	870.4	19.7	1005.4	1005.4	14.2
C3	100	826.1	851.3	911.5	29.8	855.3	1000.2	25.7	826.1	893.6	20.1	1023.7	1023.7	13.5
C4	150	1028.4	1143.7	1300.4	34.7	1238.7	1367.4	30.2	1028.4	1263.1	21.3	1531.4	1531.4	15.6
C5	199	1291.3	1399.8	1552.4	40.2	1417.4	1712.1	34.1	1310.3	1422.7	25.5	1800.3	1800.3	19.2

虽然其运行速度比 BBM 算法稍慢，但以此为代价来换取解的质量的大幅度提高也是值得的，因此，IGA 算法整体求解性能优于 ACO 算法、GA 算法和 BBM 算法，能够适应包装废弃物回收等大规模 VRP 问题的求解需求。

#### 4 结语

主要对包装废弃物回收 VRP 问题的求解算法进行了研究。针对传统 GA 算法容易陷入局部最优解、收敛速度较慢等不足，采用贪婪算法对初始种群生成方式进行改进，根据适应度值大小和进化代数自适应调整交叉和变异概率，设计了能够提高种群多样性的最大保留交叉算子，并在此基础上提出了 IGA 算法。仿真测试表明，IGA 算法整体求解性能优于 ACO 算法、GA 算法和 BBM 算法，为 VRP 问题的研究提供了一种新的思路。

#### 参考文献：

[1] 樊贵香. 混合模拟植物生长算法在包装件配送中的应用[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.

[2] 詹文瑶. B2C 电子商务网购包装设计研究[J]. 包装工程, 2017, 38(20): 255—258.  
ZHAN Wen-yao. Packaging Design of B2C Electronic Commerce Online Shopping[J]. Packaging Engineering, 2017, 38(20): 255—258.

[3] GAUR D R, MUDGAL A, SINGH R R. Approximation Algorithms for Cumulative VRP with Stochastic Demands[M]. Discrete Applied Mathematics, 2018.

[4] GIANLUCA. Ant Colony Optimization for Capacitated

Vehicle Routing Problem[J]. Journal of Computer Science, 2012, 8(6): 846—852.

[5] FRIEDRICH T, KOTZING T, KREJCA M S, et al. The Compact Genetic Algorithm is Efficient under Extreme Gaussian Noise[J]. IEEE Transactions on Evolutionary Computation, 2017, 21(3): 477—490.

[6] PAPPU S M J, GUMMADI S N. Artificial Neural Network and Regression Coupled Genetic Algorithm to Optimize Parameters for Enhanced Xylitol Production by Debaryomyces Nepalensis, in Bioreactor[J]. Biochemical Engineering Journal, 2017, 120(7): 136—145.

[7] 邢文训, 谢金星. 现代优化计算方法[M]. 北京: 清华大学出版社, 2005.  
XING Wen-xun, XIE Jin-xing. The Modern Optimization Calculation Method[M]. Beijing: Tsinghua University Press, 2005.

[8] 吴聪聪, 贺毅朝, 陈焱瑛, 等. 精英交叉二进制蝙蝠算法求解 0-1 背包问题[J]. 计算机应用研究, 2017, 34(6): 1677—1681.  
WU Cong-cong, HE Yi-chao, CHEN Yi-ying, et al. Elite Cross Binary Bat Algorithm for Solving 0-1 Knapsack Problem[J]. Application Research of Computers, 2017, 34(6): 1677—1681.

[9] PWINT M, SATTAR F. A Segmentation Method for Noisy Speech Using Genetic Algorithm[J]. Expert Systems, 2016, 33(1): 60—69.

[10] PAREEK N K, PATIDAR V. Medical Image Protection Using Genetic Algorithm Operations[J]. Soft Computing, 2016, 20(2): 763—772.

[11] 袁圃, 毛剑琳, 向凤红, 等. 改进的基于遗传优化 BP 神经网络的电网故障诊断[J]. 电力系统及其自动化学报, 2017, 29(1): 118—122.  
YUAN Pu, MAO Jian-lin, XIANG Feng-hong, et al.

- Grid Fault Diagnosis Based on Improved Genetic Optimization BP Neural Network[J]. Proceedings of the CSU-EPISA, 2017, 29(1): 118—122.
- [12] 张异. 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.
- [13] MOHAMMED M A, GANI M K A, HAMED R I, et al. Solving Vehicle Routing Problem by Using Improved Genetic Algorithm for Optimal Solution[J]. Journal of Computational Science, 2017, 21(5): 255—262.
- [14] HIASSAT A, DIABAT A, RAHWAN I. A Genetic Algorithm Approach for Location-inventory-routing Problem with Perishable Products[J]. Journal of Manufacturing Systems, 2017, 42(1): 93—103.
- [15] COSTA PRDOD, MAUCERI S, CARROLL P, et al. A Genetic Algorithm for a Green Vehicle Routing Problem[J]. Electronic Notes in Discrete Mathematics, 2018, 64(6): 65—74.
- [16] 曹平方, 李灵, 李诗珍. 基于分枝界定的 VRP 模型精确算法研究及应用[J]. 包装工程, 2014, 35(17): 97—101.
- CAO Ping-fang, LI Ling, LI Shi-zhen. Research and Application of the Accurate Algorithm of VRP Model Based on Branch and Bound Method[J]. Packaging Engineering, 2014, 35(17): 97—101.
- [17] LAST U. Networking and Emerging Optimization[EB/OL]. [2013-07-07]. <http://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-instances/>.