

# 基于改进自适应遗传算法的物流配送路径优化研究

吴 聪, 陈侃松, 姚 静

(湖北大学 计算机与信息工程学院物联网工程研究所, 武汉 430062)

**摘要:** 针对物流运输中带软时间窗车辆路径优化问题, 提出一种改进的自适应遗传算法; 为消除遗传算法初始种群随机性强, 个体分散的缺陷, 采用精英保留选择方法, 加快算法的收敛速度, 同时提出了交叉概率和变异概率自适应调整的交叉和变异方法, 进化过程中交叉概率和变异概率根据适应度、进化代数 and 进化过程中个体未改变数目个数来自适应变化, 提高算法的局部搜索能力, 有效避免了算法出现未成熟收敛的情况; 将新的自适应遗传算法 (new improved adaptive genetic algorithm, NIAGA) 应用于该路径优化问题的求解, 实验结果表明改进后的自适应遗传算法在求解物流配送路径优化问题上有明显优势。

**关键词:** 物流配送; 路径优化问题; 遗传算法

## Study on Optimization of Logistics Distribution Route Based on Improved Adaptive Genetic Algorithm

Wu Cong, Chen Kansong, Yao Jing

(Institute of Internet of Things, School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China;)

**Abstract:** A new improved adaptive genetic algorithm is proposed for solving vehicle routing problem with soft time windows (VRPSTW). The initial population of genetic algorithm (GA) is highly stochastic and individual distribution is dispersed, so elitism strategy is adopted to improve operation speed. GA is prone to "premature" and in local optima trap when searching to near the local optima solution. To avoid algorithm immature convergence the paper proposes an adaptive adjustment method of crossover probability and mutation probability. The new improved adaptive genetic algorithm is applied to vehicle routing problem. The results clearly demonstrate that the proposed method has better solution.

**Keywords:** logistics distribution; routing optimization; genetic algorithm

### 0 引言

车辆的路径问题是物流配送中的核心问题, 合理地规划物流运输过程中的车辆路径能为企业降低生产成本, 提高企业市场竞争力。带软时间窗的车辆路径问题 (vehicle routing problem with soft time windows, VRPSTW), 是指在满足车辆容量不超过核载容量和规定时间范围等约束条件的前提下, 合理规划车辆运输路径, 使物流配送过程中产生的费用最低。

遗传算法是一种智能进化算法, 它具有全局搜索能力强, 算法灵活, 计算速度快等优势<sup>[1]</sup>, 被广泛应用于求解物流配送路径优化问题<sup>[2]</sup>, 但其存在初始种群的随机性强, 个体分布比较分散, 同时局部搜索能力差, 搜索到全局最优解附近时收敛

速度变慢, 容易出现未成熟收敛, 陷入局部最优<sup>[3]</sup>等缺点, 为克服这些缺陷 sriniva<sup>[4]</sup>等人提出了自适应遗传算法, 搜索过程中交叉概率和变异概率根据进化的情况进行自适应调整<sup>[5]</sup>。任子武<sup>[6]</sup>等人对传统的自适应遗传算法进行改进, 遗传参数随适应度的大小自适应调整, 算法的收敛速度和收敛精度得到了一定的提高, 但该方法对适应值小于平均适应值的劣质个体的处理方式单一, 算法进化到晚期还是存在种群中个体多样性降低, 收敛速度变慢的缺点, 针对这些问题本文提出新的改进自适应遗传算法, 对遗传参数的自适应调整方案进行改进, 并利用该方法来求解 VRPSTW 问题。

### 1 数学模型的建立

VRPSTW 问题的优化目标是在满足车辆容量不超过核载容量和规定时间范围等约束条件的前提下, 优化车辆运输路径, 使物流配送过程中产生的费用最低, 费用函数可表示如下:

$$Z = P + P(t) + P(q) \tag{1}$$

其中:  $P$  为车辆配送过程中产生的费用,  $P(t)$  表示配送车辆超出时间窗范围产生的惩罚费用,  $P(q)$  表示超过车辆的核载容量产生的惩罚费用。车辆在运输过程中必须满足以下约束条件:

收稿日期: 2017-07-10; 修回日期: 2017-08-07。

基金项目: 国家科技支撑计划项目(2015BAK03B02)。

作者简介: 吴 聪(1991-), 女, 湖北随州人, 硕士研究生, 主要从事智能优化算法的应用方向的研究。

陈侃松(1972-), 男, 湖北荆州人, 教授, 博士生导师, 主要从事无线传感网络及应用方向的研究。

通讯作者: 姚 静(1976-), 女, 湖北天门人, 博士, 讲师, 主要从事智能优化算法的应用方向的研究。

$$x_{ijk} = \begin{cases} 1, & \text{车辆 } k \text{ 由客户 } i \text{ 到客户 } j \\ 0, & \text{车辆 } k \text{ 不由客户 } i \text{ 到客户 } j \end{cases} \quad (2)$$

$$y_{ik} = \begin{cases} 1, & \text{客户 } i \text{ 由车辆 } k \text{ 服务} \\ 0, & \text{客户 } i \text{ 不由车辆 } k \text{ 服务} \end{cases} \quad (3)$$

$$\sum_{k=1}^K \sum_{j=1}^{N+1} x_{ijk} = K \quad (4)$$

$$\sum_{k=1}^K y_{ki} = 1, \quad \forall i \in \{2, 3, \dots, N+1\} \quad (5)$$

$$\sum_{i=1}^n x_{ijk} \leq y_{jk}, \quad i = 1, 2, 3, \dots, n, \quad k = 1, 2, 3, \dots, k \quad (6)$$

$$\sum_{j=1}^n x_{ijk} \leq y_{ik}, \quad j = 1, 2, 3, \dots, n, \quad k = 1, 2, 3, \dots, k \quad (7)$$

$$\sum_{i=2}^{N+1} \sum_{j=1}^{N+1} q_i x_{ijk} \leq Q_k, \quad \forall k = 1, 2, 3, \dots, k \quad (8)$$

$$\sum_{i=2}^{N+1} \sum_{j=1}^{N+1} x_{ijk} (t_{ij} + s_i + w_i) \leq I_0, \quad \forall k = 1, 2, 3, \dots, k \quad (9)$$

$$t_0 \geq e_0 \quad (10)$$

$$e_i \leq t_i \leq l_i, \quad \forall i \in \{2, 3, \dots, N+1\} \quad (11)$$

其中:  $N$  为客户编号,  $K$  为车辆数,  $Q_k$  为每辆车的限制容量,  $e_0$  为每辆车开始工作的时间,  $I_0$  为每辆车最晚回到配送中心的时间,  $t_{ij}$  为从客户  $i$  到客户  $j$  的时间,  $s_i$  表示在客户  $i$  的服务时间,  $w_i$  表示在客户  $i$  的等待时间。

式 (2) ~ 式 (3) 表示车辆服务的规则; 式 (4) ~ 式 (11) 是 VRPSTW 的约束条件。式 (4) 表示  $K$  辆车生成  $K$  条配送路线, 并且是从配送中心出发, 最终回到配送中心, 式 (5) 表示每辆车只给一个客户  $i$  配送一次货物, 式 (6) ~ (7) 表示的是每辆车只能从一个边进一个边出, 式 (8) 表示每辆车每条路径上送货的总重量不超过最大容量, 式 (9) 表示每辆车的行驶的总时间不超过最终回配送中心的限制时间, 式 (10) 表示车辆要在开始工作时间之后出发晚, 式 (11) 表示每个客户规定的服务时间窗。

为了保证遗传操作生成的染色体是有效的, 需要将相应的约束条件设计成惩罚函数<sup>[7]</sup>, 如果配送车辆超出时间窗范围或者超过车辆的核载容量, 就会进行相应的惩罚<sup>[8]</sup>。

1) 时间惩罚函数:

$$P(t) = \begin{cases} c_1 \max(0, (e_i - t_i)), & t_i \leq e_i \\ 0, & t_i \in (e_i, l_i) \\ c_2 \max(0, (t_i - l_i)), & t_i \geq l_i \end{cases} \quad (12)$$

$c_1, c_2$  为早到和晚到的时间惩罚系数,  $e_i$  为客户要求的最早服务时间,  $t_i$  为车辆到达客户  $i$  的时间,  $l_i$  为客户  $i$  要求的最晚服务时间。

2) 容量惩罚函数:

$$P(q) = c_0 \max(0, (q_i y_{ik} - Q)) \quad (13)$$

$c_0$  为超过核载容量的惩罚系数,  $q_i$  表示客户  $i$  的货物需求量  $Q$  表示车辆的限制容量。

由此, 得到物流路径优化问题的目标函数如下:

$$\min Z = \sum_{i=0}^{N+1} \sum_{j=0}^{N+1} \sum_{k=1}^K (p_0 + p d_{ij} x_{ijk}) + c_0 \sum_{k=1}^K \max \left[ \sum_{j=2}^{N+1} q_i y_{ik} - Q, 0 \right] + \sum_{i=2}^{N+1} \{c_1 \max[0, (e_i - t_i)] + c_2 \max[0, (t_i - l_i)]\} \quad (14)$$

其中:  $p_0$  为车辆发车所需要的费用,  $p$  为单位里程所产生的费用,  $d_{ij}$  为车辆从客户  $i$  到客户  $j$  的路程。

配送路径方案的优劣是由适应度来评价的<sup>[9]</sup>, 适应度值越大方案越优, 目标函数值越小, 这里选择目标函数的倒数作为适应度函数, 如下:

$$f = 1/Z \quad (15)$$

## 2 自适应遗传算法的改进

### 2.1 基本自适应遗传算法

自适应遗传算法在进行搜索的过程中交叉概率和变异概率会根据进化的情况进行自适应调整<sup>[10]</sup>。传统的自适应遗传算法 (Adaptive Genetic Algorithm, AGA) 是由 sriniva 等人提出的。其中交叉概率和变异概率的自适应调整公式如下:

$$p_c = \begin{cases} \frac{k_1 (f_{\max} - f')}{f_{\max} - f_{\text{avg}}, f'} \geq f_{\text{avg}} \\ k_2, f' < f_{\text{avg}} \end{cases} \quad (16)$$

$$p_m = \begin{cases} \frac{k_3 (f_{\max} - f)}{f_{\max} - f_{\text{avg}}, f} \geq f_{\text{avg}} \\ k_4, f < f_{\text{avg}} \end{cases} \quad (17)$$

其中:  $f_{\max}$  为所有个体适应度值中最大的适应值,  $f_{\text{avg}}$  为所有个体的平均适应度  $f$  为要变异个体的适应值  $f'$  为准备交叉的两个个体中较大的适应值, 系数  $k_1, k_2, k_3, k_4$  取 (0, 1) 区间的值。

由式中可以看出,  $f_{\max} - f_{\text{avg}}$  越大, 即种群中个体适应度比较分散时, 则  $p_c, p_m$  比较小; 如果  $f_{\max} - f_{\text{avg}}$  越小, 即种群中的个体适应度比较集中时, 则  $p_c, p_m$  越大,  $p_c$  和  $p_m$  会根据个体的适应度值自适应调整。

但是 AGA 也存在缺点: 一方面由于传统的自适应交叉和变异概率主要取决于取值为 (0, 1) 的随机系数  $k_1, k_2, k_3, k_4$ , 随机性比较大, 影响了遗传算法的种群个体质量, 可能会导致遗传算法陷入局部最优; 另一方面,  $f' = f_{\max}$  或者  $f = f_{\max}$  时, 即交叉和变异的个体适应度达到最高适应度时, 交叉概率和变异概率为 0, 种群个体会处于停滞不前的状态。

针对 AGA 的缺点, 任子武等人提出了改进自适应遗传算法, 公式如下:

$$p_c = \begin{cases} p_{c1} - \frac{(p_{c1} - p_{c2})(f' - f_{\text{avg}})}{f_{\max} - f_{\text{avg}}}, & f' \geq f_{\text{avg}} \\ p_{c1}, & f' < f_{\text{avg}} \end{cases} \quad (18)$$

$$p_m = \begin{cases} p_{m1} - \frac{(p_{m1} - p_{m2})(f_{\max} - f)}{f_{\max} - f_{\text{avg}}}, & f \geq f_{\text{avg}} \\ p_{m1}, & f < f_{\text{avg}} \end{cases} \quad (19)$$

式中,  $f_{\max}$  为所有个体适应度值中最大的适应值,  $f_{\text{avg}}$  为所有个体的平均适应度,  $f$  为要变异个体的适应值,  $f'$  为要交叉的个体中比较大的适应度,  $p_{c1} > p_{c2}, p_{m1} > p_{m2}$ , 取 (0, 1) 区间的值, 可在优化过程中调整。

以上改进后  $p_c$  和  $p_m$  不仅可以随适应度的大小自适应调整, 而且, 若  $f' = f_{\max}$ , 则  $p_c = p_{c2}$ , 若  $f = f_{\max}$ , 则  $p_m = p_{m2}$ , 不存在最大适应度个体的  $p_c$  和  $p_m$  为 0 的情况。

但该算法还有如下缺点:

1) 该算法没有对劣质个体进行处理, 当个体的适应度小于平均适应度时,  $p_c$  和  $p_m$  为设定的固定值。

2) 随着算法进化到晚期, 虽然个体的适应度都比较高, 但是个体之间趋于相似, 种群中个体多样性降低, 收敛速度慢, 进行交叉操作已经没有意义。

### 2.2 自适应遗传算法的改进

下文针对传统自适应遗传算法和经典的改进自适应遗传算法的缺点提出了新的自适应遗传算法 (new improved adaptive genetic algorithm, NIAGA) 如下:

#### 2.2.1 选择算子的改进

首先, 将种群中所有个体的适应度按从大到小的顺序进行排列, 新种群中有 1/2 的个体直接复制初始种群中适应度最高的个体而产生, 剩下的个体采用轮盘赌的方式进行选择; 然后, 将选择的个体组合成新的种群, 并将新种群中所有个体的适应度按从大到小的顺序排列, 选择出适应度最高的个体保存下来; 最后完成遗传算法的所有操作, 生成新的种群, 如果新种群中适应度最高的个体适应度值比旧种群中保存的适应度值低, 那么就旧种群中适应度值最高的个体替代新种群中适应度最低的个体。

按以上改进方法进行选择操作, 不仅在种群个体的总数目没有改变的前提下增加了适应高的个体比率, 保持了种群中个体的多样性; 还在一定程度上解决了轮盘赌选择法种群单一随机化和容易陷入局部最优的缺陷。

#### 2.2.2 交叉算子和变异算子的改进

由于交叉概率和变异概率是进行交叉和变异操作的重要参数, 对遗传算法的搜索能力有很大影响, 而基本遗传算法的交叉概率和变异概率是设定的固定值, 缺乏理论依据。本文提出了新的自适应遗传算法, 它的交叉概率和变异概率会根据适应度, 进化代数和进化过程中最优解没有改变的个体数目的变化而自适应调整。不仅可以增加算法的收敛速度, 还突出个体之间的差异以增加个体之间的竞争力。

根据传统和改进自适应遗传算法存在的问题, 本文提出了交叉概率和变异概率根据适应度、进化代数和进化过程中个体未改变数目自适应调整的改进方法, 交叉概率和变异概率变化公式如下:

$$p_c = \begin{cases} p_{c1} - \frac{(p_{c1} - p_{c2})(f' - f_{\text{avg}})}{(f_{\max} - f_{\text{avg}})[1 + \exp(-t * S/T * M)]}, & f' \geq f_{\text{avg}} \\ p_{c1}^{[1 + \exp(-t * S/T * M)]}, & f' < f_{\text{avg}} \end{cases} \quad (20)$$

$$p_m = \begin{cases} p_{m1} - \frac{(p_{m1} - p_{m2})(f_{\max} - f)}{(f_{\max} - f_{\text{avg}}) * [1 + \exp(t * S/T * M)]}, & f \geq f_{\text{avg}} \\ p_{m1}^{[1 + \exp(t * S/T * M)]}, & f < f_{\text{avg}} \end{cases} \quad (21)$$

其中:  $f_{\max}$  为每代种群最大的适应值,  $f_{\text{avg}}$  为每代种群的平均适应值,  $f'$  为要交叉的个体中比较大的适应度,  $f$  为要变异个体的适应值,  $t$  为当前进化代数,  $S$  为进化过程中最优解

没改变的个体数目,  $T$  为最大终止代数,  $M$  为种群大小, 其中  $p_{c1} > p_{c2}$ ,  $p_{m1} > p_{m2}$ , 取 (0, 1) 区间的值, 可以在优化过程中调整。

进化初期, 种群中的个体之间差异比较大,  $f_{\max} - f_{\text{avg}}$  的值也比较大, 此时进化代数  $t$  和进化过程中个体未改变数目  $S$  比较小, 所以交叉概率  $p_c$  比较大, 种群收敛速度较快。进化后期, 由于种群中个体之间的差异变小, 个体之间相似度变高, 收敛速度变慢, 有陷入局部最优的趋势,  $f_{\max} - f_{\text{avg}}$  的值变小, 进化代数  $t$  和进化过程中个体未改变数目  $S$  变大, 为了提高收敛效率, 交叉概率  $p_c$  变小, 变异概率  $p_m$  增加, 提高了种群个体的多样性, 充分显示了变异算子的局部搜索能力。经过改进的遗传算法在进化的过程中根据适应度大小、进化代数  $t$  和进化过程中个体未改变数目  $S$  自适应调整交叉和变异概率, 不仅能保证交叉操作的全局搜索能力, 并且能充分发挥变异操作的局部搜索能力。

## 3 改进自适应遗传算法在 VRPSTW 中的应用

改进遗传算法求解 VRPSTW 问题主要包括编码, 种群初始化, 求解适应度, 选择, 交叉, 变异, 终止条件判定等步骤。

### 3.1 参数设定

首先对实验参数设定如下:

种群规模:  $M = 50$

交叉概率:  $p_c = 0.6$ ,  $p_{c1} = 0.6$ ,  $p_{c2} = 0.3$

变异概率:  $p_m = 0.01$ ,  $p_{m1} = 0.01$ ,  $p_{m2} = 0.002$

终止代数:  $T = 200$

每辆车最大容量:  $Q = 6000$

发车成本:  $p_0 = 100$

单位运输成本:  $p = 10$

车辆行驶速度:  $v = 5$

超载惩罚系数:  $c_0 = [0 \ 10]$

早到惩罚系数:  $c_1 = [0 \ 5.0 \ 6.0 \ 14.0 \ 5.0 \ 4.0 \ 4.0 \ 8.0 \ 5.0 \ 3.0 \ 6.0 \ 15.0 \ 8.0 \ 7.0 \ 15.0 \ 5.0 \ 4.0 \ 6.0]$

迟到惩罚系数:  $c_2 = [0 \ 150 \ 300 \ 210 \ 160 \ 180 \ 320 \ 210 \ 2700 \ 220 \ 180 \ 210 \ 2500 \ 310 \ 210 \ 100 \ 1500 \ 180]$

### 3.2 遗传操作

#### 3.2.1 选择

采用精英保留策略和轮盘赌方法分别选择个体 (路径) 组成新的种群, 该种群一共包含 50 条城市路径 (即 50 条染色体), 首先对所有个体的适应度按从大到小的顺序进行排列, 选择部分适应度最高的个体 (所用费用最少的路径), 剩下的个体通过轮盘赌选择法选择, 然后组合成的规模为 50 的新种群, 将这 50 个个体按照适应度从大到小的顺序进行排列, 将适应度最高的个体保存, 然后进行交叉和变异操作, 生成新的种群, 如果新的种群中最高适应度小于旧种群中最高适应度, 则用旧种群中适应度最高的个体代替新种群中适应度最低的个体。

#### 3.2.2 交叉

本文采用两点交叉的方式进行交叉操作, 首先随机选取两

个交叉位点, 对交叉点中间的基因模块进行交叉路径, 比如:

交叉前

父代 1: 1 5 6 | 4 1 3 1 2 | 8 7 9 1

父代 2: 1 5 4 | 1 8 6 2 1 | 3 9 7 1

交叉后

子代 1: 1 5 6 | 1 8 6 2 1 | 8 7 9 1

子代 2: 1 5 4 | 1 8 6 2 1 | 3 9 7 1

### 3.2.3 变异

本文在用遗传算法求解 VRPSTW 问题的过程中, 采用的是随机选择基因位点(城市编号)进行变异操作的, 首先根据变异概率选中进行变异的个体(路径), 然后在该个体上随机选择两个基因位, 最后将这两个基因位上的基因相互交换, 形成新的个体。

变异前

父代: 1 5 6 | 4 1 3 1 2 | 8 7 9 1

变异后

子代: 1 5 6 | 8 1 3 1 2 | 4 7 9 1

## 4 仿真实验及结果分析

应用提出的算法解决如下物流路径问题: 6 辆车从 1 个配送中心出发, 为 17 个客户提供配送服务。配送中心(城市编号为 1)和 17 个客户所在城市坐标, 每个客户的服务时间窗和货物需求量信息如表 1 所示。

表 1 VRPSTW 问题初始城市坐标和客户需求信息

城市编号	X	Y	服务时间窗	货物需求量
1	18.70	15.29	[0, 24]	0
2	16.47	8.45	[7, 10]	1 600
3	20.07	10.14	[7, 11]	700
4	19.39	13.37	[7, 13]	1 000
5	25.27	14.24	[7, 9]	1 900
6	22.00	10.04	[7, 10]	800
7	25.47	17.02	[7, 12]	1 500
8	15.79	15.10	[7, 14]	1 700
9	16.60	12.38	[7, 8]	100
10	14.05	18.12	[7, 9]	300
11	17.53	17.38	[7, 10]	1 600
12	23.52	13.45	[7, 13]	1 200
13	19.41	18.13	[7, 14]	1 100
14	22.11	12.51	[7, 8]	2 500
15	11.25	11.04	[7, 11]	1 000
16	14.17	9.76	[7, 10]	900
17	24.00	19.89	[7, 9]	800
18	12.21	14.50	[7, 12]	1 700

根据配送过程中产生的费用和车辆行驶的路径来衡量 NIAGA 求解 VRPSTW 问题的优化效果。对比的算法是任子武等人提出了改进自适应遗传算法(improved adaptive genetic algorithm, 简称 IAGA)。设定进化代数为 500, 一共进行了 5 次实验, 呈现的结果是求取 5 次实验结果的平均值。将 IAGA 和 NIAGA 求得的 VRPSTW 问题的实验结果进行对比, 如表

2 所示。

表 2 IAGA 和 NIAGA 求解 VRPSTW 问题实验结果对比

	AGA		NIAGA	
	最优成本	行驶里程	最优成本	行驶里程
1	5 918.8 684	129.1537	3 228.9 212	109.3 619
2	5 358.8 551	132.8 608	3 814.7 192	123.4 226
3	6 419.2 541	119.5 565	3 996.453	123.7 368
4	5 729.0 411	130.9 948	3 323.9 388	116.4 567
5	6 757.8 177	123.5 251	3 797.9 083	123.5 856
平均值	6 036.7 673	127.2 182	3 632.3 881	119.3 127

从表中可以看出, NIAGA 在求解 VRPSTW 问题的过程中所产生的费用比 IAGA 低很多, 最短路径也小很多, 更进一步验证了本文提出的改进遗传算法在求解 VRPSTW 问题中有明显优势。

为了更直观地对 IAGA 和 NIAGA 求解 VRPSTW 问题进行观察, 取其中一次的 IAGA 和 NIAGA 求解 VRPSTW 问题的路径仿真图, 并对其详细参数进行对比。如下:

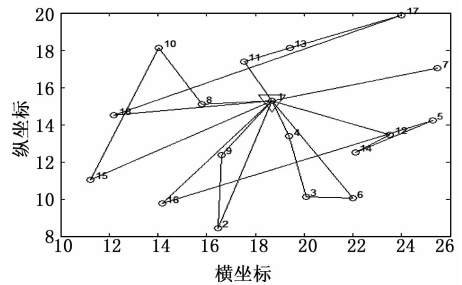
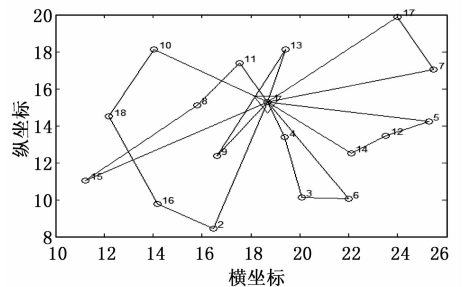


图 1 IAGA 在 VRPTW 中应用的路径图



注: 三角形表示的是配送中心, 数字表示的是城市编号, 小圆圈表示客户所在城市。

图 2 NIAGA 在 VRPTW 问题中应用的路径图

从图中可以看出, NIAGA 的路径图比 IAGA 的路径图简单一些。其具体参数如表 3 所示。

表 3 IAGA 和 NIAGA 求解 VRPSTW 问题的最优解对比

求解算法	最优成本	行驶里程
IAGA	5 689.7 345	125.3 745
NIAGA	3 689.823	106.8 263

从表中可以看出, NIAGA 在求解 VRPSTW 问题与 IAGA 相比, 求得的最优成本和行驶里程都比较小。其中, NIAGA 求解最优路径中, 每辆车的行驶路径及货物容量如表 4 所示。

表 4 每辆车的行驶路径及货物容量

车辆号	路径	路径长度	货物容量
车辆 1	1->13->9->1	12.8 625	1200
车辆 2	1->11->8->15->1	24.117	4300
车辆 3	1->10->18->16->2->1	24.4 747	4500
车辆 4	1->4->3->6->1	13.4 746	2 500
车辆 5	1->14->12->5->1	14.6 676	5 600
车辆 6	1->7->17->1	17.2 299	2 300

每辆车的核载总容量是 6 000，由上表可以看出，本次 NIAGA 求解最优路径中，每辆车的载货容量都没有超过核载总容量。

由每个城市的坐标可以得出每两个城市之间的距离，根据车辆的行驶速度，可以得出每辆车在每两个城市之间的行驶时间，从而计算出车辆到达目的地的时间。与表 4.1 给出的时间窗进行对比，判断车辆是否满足约束条件。每辆车的具体路径长度及到达目的地的时间如表 5 ~ 表 10 所示。

表 5 车辆 1 的具体路径长度及到达目的地的时间

车辆 1 的路径	路径长度	行驶时间	到达目的地的时间
1->13	2.8 739	0.5 748	7.5 748
13->9	6.4	1.28	8.8 548
9->1	3.5 886	0.7 177	9.5 725

表 6 车辆 2 的具体路径长度及到达目的地的时间

车辆 2 的路径	路径长度	行驶时间	到达目的地的时间
1->11	2.3 952	0.479	7.479
11->8	6.9 384	1.3 877	8.667
8->15	6.0 906	1.2 181	9.8 851
15->1	8.6 928	1.7 386	11.6 237

表 7 车辆 3 的具体路径长度及到达目的地的时间

车辆 3 的路径	路径长度	行驶时间	到达目的地的时间
1->10	5.4435	1.0887	8.0887
10->18	4.0608	0.8122	8.0009
18->16	5.1292	1.0258	9.0267
16->2	2.6469	0.5294	9.5561
2->1	7.1943	1.4389	10.995

表 8 车辆 4 的具体路径长度及到达目的地的时间

车辆 4 的路径	行驶路径	行驶时间	到达目的地的时间
1->4	2.0402	0.408	7.408
4->3	3.3008	0.66	8.068
3->6	1.9326	0.3865	8.4545
6->1	6.201	1.2402	9.6947

表 9 车辆 5 的具体路径长度及到达目的地的时间

车辆 5 的路径	行驶路径	行驶时间	到达目的地的时间
14	4.3996	0.88	7.88
14->12	1.6946	0.3382	8.2189
12->5	1.92	0.384	8.6029
5->1	6.6534	1.3307	9.9336

表 10 车辆 6 的具体路径长度及到达目的地的时间

车辆 6 的路径	行驶路径	行驶时间	到达目的地的时间
1->7	6.9875	1.3975	8.3975
7->17	3.2246	0.6449	9.0424
17->1	7.0178	0.1269	9.1693

从表 5 ~ 表 10 可以得出车辆 1 到达 9 号城市的时间是 8.8548，但是 9 号城市的规定的时间窗是 [7, 8]；车辆 6 到达 17 号城市的时间是 9.0424，而 17 号城市规定的时间窗是 [7, 9]。所以车辆 1 和车辆 6 到达时间都比规定时间晚，应该受到相应的惩罚。

从以上数据可以得出，NIAGA 求解 VRPSTW 问题过程中，能够求出最优解，并且产生的费用和行驶路径距离都比 IAGA 小很多；同时车辆载重没有超过车辆容量的限制，但是还是有两辆车超过出时间窗的范围，产生了相应的惩罚费用，不过超出时间窗的值比较小，对求解最优解影响不是很大，可见 NIAGA 在求解 VRPSTW 问题上有很大的优势。

### 5 结语

本文提出一种改进的自适应遗传算法，对遗传参数的自适应调整方案进行改进，进化过程中交叉概率和变异概率根据适应度、进化代数 and 进化过程中个体未改变数目个数来自适应变化。将该算法应用于求解带软时间窗的物流运输车辆路径优化问题，结果与先前的改进自适应遗传算法进行比较，分析表明该算法能有效抑制遗传算法的“早熟”，优化精度更高，得到的优化结果更靠近全局最优。

#### 参考文献：

- [1] Holland J H. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence [M]. MIT Press, 1992.
- [2] Vehicle routing: problems, methods and applications [M]. Society for Industrial and Applied Mathematics, 2014.
- [3] 葛继科, 邱玉辉, 吴春明, 等. 遗传算法研究综述 [J]. 计算机应用研究, 2008, 25 (10): 2911 - 2916.
- [4] Holland J H. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence [M]. MIT Press, 1992.
- [5] 朱整鑫. 遗传算法的适应度函数研究 [J]. 系统工程与电子技术, 1998, 20 (11): 58 - 62.
- [6] 任子武, 伞 冶. 自适应遗传算法的改进及在系统辨识中应用研究 [J]. 系统仿真学报, 2006, 18 (1): 41 - 43, 66.
- [7] 李 军, 郭耀煌. 物流配送车辆优化调度理论与方法 [M]. 中国物资出版社, 2001.
- [8] Solomon M M. Algorithms for the vehicle routing and scheduling problems with time window constraints [J]. Operations Research, 1987, 35 (2): 254 - 265.
- [9] Holland J L. Making vocational choices: A theory of vocational personalities and work environments [M]. Psychological Assessment Resources, 1997.