

基于改进蝙蝠算法的带模糊需求的车辆路径问题

朱 颢

(湖州职业技术学院, 浙江 湖州 313000)

摘要: 蝙蝠算法作为一种新的元启发式算法, 尚未被应用到模糊车辆路径问题中; 针对带模糊需求的车辆路径问题, 以极小化总运输距离为目标, 建立基于可信性理论的模糊规划模型, 提出一种改进的蝙蝠算法; 算法采用基于客户编号的编码方式, 利用随机模拟算法计算额外行驶距离; 在蝙蝠位置更新时, 引入基于非线性调整的惯性权重和基于子路径的局部搜索; 为提高全局搜索能力, 避免算法早熟, 对处于较差位置的蝙蝠进行交叉操作; 最后, 利用随机实验数据进行仿真, 分析了决策者主观偏好值对目标值的影响, 并与其它算法的寻优结果进行对比分析, 结果表明, 算法具有一定的可行性和有效性。

关键词: 蝙蝠算法; 模糊需求; 车辆路径问题

Vehicle Routing Problem with Fuzzy Demands Based on An Improved Bat Algorithm

Zhu Hao

(Huzhou Vocational Technical College, Huzhou 313000, China)

Abstract: As a new meta-heuristic, bat algorithm has not yet been applied to solve fuzzy vehicle routing problem until now. In this paper, the vehicle routing problem with fuzzy demands is considered at first, in which the final objective is to minimize the total distance, and then a fuzzy programming model based on fuzzy credibility theory is presented, in order to solve this problem, an improved bat algorithm with the coding method of customer number is introduced. In this algorithm, a stochastic simulation is proposed to calculate the additional distance, moreover, a nonlinear adjustment strategy for the inertia weight and a local search strategy on sub-route are designed at the stage of location updating of each bat, on the other hand, to improve the global search ability of this algorithm and avoid premature convergence, crossover operation on the worst bats is applied. To illustrate the effectiveness and good performance of the proposed algorithm, an example is carried out by using the random experimental data, and the influence of the decision-maker's preference on the objective of this problem is discussed, moreover, the improved bat algorithm is compared with other algorithms.

Keywords: bat algorithm; fuzzy demand; vehicle routing problem

0 引言

车辆路径问题 (VRP 问题) 作为物流领域内的一类经典的组合优化问题, 由 Dantzig 和 Ramser^[1]于 1959 年首次提出, 该问题描述为: 若干车辆从车场出发, 为一系列客户提供取货服务 (送货服务类似), 取货完毕后返回车场, 每个客户一次只能由一辆车提供取货服务, 在取货过程中需要满足车辆装载能力约束、车辆行驶里程约束和客户时间窗口约束等条件, 为此, 为每辆车选择相应的客户, 并安排访问的顺序, 使得某些目标达到最优 (如总路程最短、成本最低等)。目前, 关于车辆路径问题的文献很多, 大多数都是关于确定性的 VRP 问题, 但是在实际应用中, 由于受客观世界不确定和人类认识事物的模糊性的影响, 某些信息可能是模糊的、不确定的, 例如客户需求量大概为 30, 或者在 30 到 50 之间, 此时, 需要运用模糊变量来处理这类情况, 模糊车辆路径问题就是一类用来反映某些信息为不确定的、模糊的车辆路径问题。

针对客户需求量为模糊变量的车辆路径问题, Teodorovic^[2]等通过引入决策者偏好, 以模糊推理算法为基础, 运用扫描算法和蚁群算法进行求解; 国内学者祝崇隽等^[3]较早地运用模糊模拟的方法计算车辆的额外行驶里程, 并提出了基于可能性分布和基于需求上界的 2-OPT 算法; 张建勇^[4-6]等建立了基于可能性理论的规划模型, 先后运用遗传算法和 sweeping 算法进行求解, 并分析了决策者的偏好对优化目标的影响; 曹二保^[7-8]等建立了基于模糊可能性的规划模型, 运用差分进化算法进行求解, 在算法中为避免产生的解不可行, 设计了基于整数序规范的辅助算子; 彭北青^[9]、戎丽霞^[10]针对类似问题, 先后运用遗传算法进行了求解; 吴天羿^[11]等运用混合遗传算法求解了该问题, 在算法中运用种群扫描法初始化种群, 为了降低车辆调用数量, 提高车辆利用率和装载率, 设计了专门的混合交叉算子和差分扫描变异算子; Lian xue^[12]利用改进的差分进化算法进行求解, 在交叉环节, 交叉概率随迭代次数线性增加; R. J. Kuo^[13]等提出了基于粒子群算法和遗传算法的混合算法, 让每个粒子同其自身经历过的最优位置及全局最优位置进行多次交叉, 选取较优的子代粒子作为下一代粒子; Cao Erbao 等^[14]设计了随机模拟算法来估算额外行驶里程, 并针对可能产生的不可行解, 引入了相应的惩罚函数, 利用差分进化算法进行求解; Yang Peng 等^[15]提出了基于整数编码的粒子群算法,

收稿日期: 2017-04-01; 修回日期: 2017-04-24。

基金项目: 湖州市自然科学基金 (2015YZ07)。

作者简介: 朱 颢 (1980-) 男, 湖北监利人, 硕士, 主要从事车辆路径问题的研究。

对粒子更新的过程中产生的实数, 利用基于升序的排序规则, 将实数转换成整数。

蝙蝠算法作为一种新型的仿生智能算法, 由剑桥大学学者 Yang^[16]于 2010 年提出, 该算法利用自然界中蝙蝠运用超声波来搜寻猎物这一生物学特性, 将蝙蝠随机分布在解空间, 通过蝙蝠不断变化的脉冲频率来搜寻猎物。截至目前, 蝙蝠算法在 VRP 问题的应用尚不多见, 只有马祥瑞^[17-18]等针对有能力约束的确定性 VRP 问题和带时间窗的确定性 VRP 问题, 以 2L 维向量编码, 运用蝙蝠算法进行了求解; Yongquan Zhou^[19]等针对确定性的 VRP 问题, 运用基于贪婪随机自适应搜索算法和基本蝙蝠算法的混合算法进行了求解。目前蝙蝠算法在带模糊需求的 VRP 问题中的应用尚未出现, 本文基于可信性理论, 建立相应的模糊规划模型, 并运用改进的蝙蝠算法求解带模糊需求量的车辆路径问题。

1 问题描述

带模糊需求量的 VRP 可以描述为: 某运输网络中有一个车场 (用 0 表示) 和 n 个需要取货的客户 (用 $1, 2, \dots, n$ 表示); 该车场共有 m 辆车 (用 $1, 2, \dots, m$ 表示) 可以利用; 每辆车的标记载重量相同, 用 Q 表示; 车辆从车场出发, 为一定数量的客户提供取货服务后需返回车场; 每个客户一次只能由一辆车服务; 每个客户的货物需求量 (在取货环境下, 可以理解为客户需要寄送的货物量) 不确定, 用一个三角模糊数 $d = (d_1, d_2, d_3)$ 表示; 客户 i 和客户 j 之间的距离用 D_{ij} 表示; 本文的目标是在一系列约束条件下, 为车辆选择合适的取货顺序, 使得总运输距离最短 (包括计划行驶距离和额外行驶距离)。

由于每个客户的需求量为三角模糊数, 对于某一个给定的客户 i , 其需求量为 $d_i = (d_{1i}, d_{2i}, d_{3i})$ 。当某一辆车为 k 个客户提供取货任务后, 其剩余载重能力为 $Q_k = Q - \sum_{i=1}^k d_i$, 由于 d_i 为三角模糊数, 因此 Q_k 也为一个三角模糊数, 令 $Q_k = (Q_{1k}, Q_{2k}, Q_{3k})$, 则有 $(Q_{1k}, Q_{2k}, Q_{3k}) = (Q - \sum_{i=1}^k d_{3i}, Q - \sum_{i=1}^k d_{2i}, Q - \sum_{i=1}^k d_{1i})$, 令下一个客户的需求量 d_{k+1} 小于剩余载重能力 Q_k 的可信性为 $Cr\{d_{k+1} < Q_k\}$, 根据模糊理论, $Cr\{d_{k+1} < Q_k\}$ 可以表示为下式 (1):

$$Cr\{d_{k+1} < Q_k\} = \begin{cases} 0 & d_{1,k+1} \geq Q_{3k} \\ \frac{d_{1,k+1} - Q_{3k}}{2(d_{1,k+1} - Q_{3k} - d_{2,k+1} + Q_{2k})} & d_{1,k+1} \leq Q_{3k}, d_{2,k+1} \geq Q_{2k} \\ \frac{2(d_{2,k+1} - Q_{2k}) - (d_{3,k+1} - Q_{1k})}{2(d_{2,k+1} - Q_{2k} - d_{3,k+1} + Q_{1k})} & d_{2,k+1} \leq Q_{2k}, d_{3,k+1} \geq Q_{1k} \\ 1 & d_{3,k+1} \leq Q_{1k} \end{cases} \quad (1)$$

根据模糊理论, 车辆的剩余载重能力 Q_k 越大, 第 $k+1$ 个客户的需求量 d_{k+1} 越小, 则 $Cr\{d_{k+1} < Q_k\}$ 的值越大, 在此设置决策者的主观偏好值 Cr^* , 当 $Cr\{d_{k+1} < Q_k\} \geq Cr^*$ 时, 车辆继续为第 $k+1$ 个客户服务, 否则, 该车辆返回车场, 新开一辆车完成剩余客户的取货任务, 依此策略, 针对某一客户排列, 可以将其中的所有客户依次分配给相应的车辆, 得到若干子路径, 并计算所有车辆的计划行驶距离。

然而, 在实际的取货过程中, 当车辆按上述策略所计划的路线到达某个客户处时, 客户的模糊需求变为“实际需求”, 而车辆的剩余载重能力也变为一个确定值, 此时, 可能会出现车辆剩余载重能力不能满足该客户的“实际需求”而导致原计划路线“失败”, 此时, 车辆需要先返回车场, 卸货完后再空驶至该“失败点”, 然后继续完成剩余的运输任务, 从该“失败点”到车场的往返即构成了车辆的额外行驶距离。因此, 评估一个车辆路线安排的优劣时, 既要考虑计划行驶距离, 还要考虑由于路线“失败”而产生的额外行驶距离。由于客户需求具有模糊性, 在按照上述可信性条件安排计划路线时, 无法明确路线“失败”发生的地点、次数及由此产生的额外行驶距离, 因此, 采用随机模拟的方式产生客户的“实际需求”, 并对可能产生的额外行驶距离进行估算。

2 构建问题的模型

定义问题的变量为 y_{ik} 、 x_{ijk} 。其含义如下:

$$y_{ik} = \begin{cases} 1 & \text{客户 } i \text{ 由车辆 } k \text{ 提供服务} \\ 0 & \text{否则} \end{cases}$$

$$x_{ijk} = \begin{cases} 1 & \text{车辆 } k \text{ 由客户 } i \text{ 行驶至客户 } j \\ 0 & \text{否则} \end{cases}$$

该问题的模型如下:

$$\min Z = \sum_{k=1}^m \sum_{i=0}^n \sum_{j=0}^n D_{ij} x_{ijk} + D' \quad (2)$$

$$Cr\left\{\sum_{i=1}^n q_i y_{ik} \leq Q\right\} \geq Cr^* \quad \forall k = 1, 2, \dots, m \quad (3)$$

$$\sum_{k=1}^m y_{ik} = 1 \quad \forall i = 1, 2, \dots, n \quad (4)$$

$$\sum_{i=0}^n x_{ijk} = y_{jk} \quad \forall j = 1, 2, \dots, n \quad k = 1, 2, \dots, m \quad (5)$$

$$\sum_{j=0}^n x_{ijk} = y_{ik} \quad \forall i = 1, 2, \dots, n \quad k = 1, 2, \dots, m \quad (6)$$

$$\sum_{i \in R} \sum_{j \in R} x_{ijk} \leq |R| - 1 \quad R \in \{1, 2, \dots, n\} \quad \forall k = 1, 2, \dots, m \quad (7)$$

$$\sum_{j=1}^n x_{0jk} + \sum_{j=1}^n x_{j0k} = 2y_{0k} \quad \forall k = 1, 2, \dots, m \quad (8)$$

该模型中: (2) 表示模型的目标函数, $\sum_{k=1}^m \sum_{i=0}^n \sum_{j=0}^n D_{ij} x_{ijk}$ 为计划行驶距离, D' 为额外行驶距离; (3) 表示车辆装载能力约束, 即每辆车所装货物的总重量小于其标记载重量的可信性大于给定的主观偏好值 Cr^* ; (4) 表示每个客户一次只由一辆车提供取货服务; (5) 和 (6) 表示 y_{ik} 和 x_{ijk} 之间的关系, 且每个客户只能被服务一次; (7) 为消去支路约束; (8) 表示每辆车从车场出发, 取完货后需要返回车场。

3 基本蝙蝠算法

蝙蝠是一种神奇的动物, 拥有令人惊异的回声定位能力, 通过向四周发出一定频率的声音脉冲, 然后聆听从周围物体反射回来的回声波, 利用双耳接收回声波的时间差、回声波音强的变化来建立周围环境的三维场景, 以此来搜寻猎物或者避开障碍物^[20]。

在基本蝙蝠算法中, 第 i 只蝙蝠搜寻猎物时的飞行速度更新公式为^[20]:

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

式 (9) 中 X_i^t 表示第 i 只蝙蝠在 t 时刻所在的位置, X^* 为当前蝙蝠群体中的最优位置, V_i^t 和 V_i^{t+1} 分别表示第 i 只蝙蝠在 t 时刻和 $t+1$ 时刻的飞行速度, f_i 为第 i 只蝙蝠搜寻猎物时所使用的脉冲频率, 且有:

$$f_i = f_{\min} + (f_{\max} - f_{\min}) \times rand \quad (10)$$

式 (10) 中 f_{\max} 和 f_{\min} 分别为蝙蝠搜寻猎物时使用的最大脉冲频率和最小脉冲频率, $rand$ 为区间 $[0,1]$ 内服从均匀分布的随机数。

根据蝙蝠的飞行速度公式, 第 i 只蝙蝠在第 t 时刻的位置为:

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (11)$$

在利用蝙蝠算法寻优的过程中, 还可以进行局部搜索, 局部搜索时, 蝙蝠的位置更新公式如下:

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

式 (12) 中 X_{new} 为蝙蝠的一个待定解, X_{old} 一般为当前蝙蝠群体中的最优位置, ϵ 为区间 $[-1,1]$ 内的一个随机数, $A^t = [A^t]$ 为所有蝙蝠在同一时刻的平均响度。

蝙蝠在搜寻猎物的过程中发射脉冲的响度和脉冲发射率更新公式为:

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

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

式 (13) 中 A_i^{t+1} 为 $t+1$ 时刻蝙蝠 i 发射脉冲的响度, α 为响度衰减系数, 且 $\alpha \in [0,1]$; 式 (14) 中 r_i^{t+1} 为 $t+1$ 时刻蝙蝠 i 的脉冲发射率, r_i^0 为最大脉冲发射率, γ 为脉冲发射率增加系数, 且 $\gamma > 0$ 。显然, 对于任意的 $\alpha \in [0,1]$ 和 $\gamma > 0$, 有 $\lim_{t \rightarrow \infty} A_i^t = 0$, $\lim_{t \rightarrow \infty} r_i^t = 1$ 。

4 改进蝙蝠算法求解模糊车辆路径问题

4.1 生成初始种群

随机生成 pop_size 只蝙蝠。每只蝙蝠用客户编号 $1, 2, \dots, n$ 的一个重排表示。对于每只蝙蝠, 采用如下方法进行解码, 以表示一个路径安排^[5]:

Step1: 选择第 1 辆车出列, 准备执行任务;

Step2: 选择排列中最左边的客户, 按公式 (1) 计算该客户的模糊需求量小于当前车辆剩余载重能力的可信性, 若该可信性值大于给定的主观偏好值 Cr^* , 则将客户安排给当前车辆; 否则, 新派一辆车, 将客户安排给新开的车;

Step3: 将该客户从排列中删除;

Step4: 重复 Step2 和 Step3, 若所有的客户均安排完毕, 则获得了一个可行的路径安排。

4.2 随机模拟算法计算额外行驶距离

如前所述, 对于任意一个可行的路径安排, 由于客户的需求量为三角模糊数, 车辆按原计划抵达客户处时, 客户的“实际需求”可能超过车辆剩余载重能力, 从而导致计划“失败”, 车辆需返还车场, 卸完货后再重新回到该客户处, 此时额外行驶距离的计算方法如下。

1) 对某个客户 i , 先随机模拟其“实际需求”, 采用如下方法:

(1) 随机产生一个位于区间 $[d_{1i}, d_{3i}]$ 内的值 d' , 代表该客户的需求, 计算 d' 的隶属度 μ ;

(2) 随机产生一个位于区间 $[0,1]$ 内均匀分布的随机数 α ;

(3) 若 $\mu > \alpha$, 则接受 d' 作为该客户的“实际需求”, 否则, d' 暂时不接受, 重复 (1) 和 (2), 直到条件 $\mu > \alpha$ 成立, 能接受 d' 作为该客户的“实际需求”为止。

2) 重复 Step1, 直至生成所有客户的“实际需求”。

3) 让车辆依照此路径安排行驶, 计算所有客户的需求变为“实际需求”的前提条件下, 所有车辆的总额外行驶距离。

4) 重复上述步骤 (1) 至 (3) 共 M 次, 取其平均值, 作为该路径安排由于可能发生计划“失败”而产生的额外行驶距离的估计值。

4.3 基于升序的排序规则

在利用蝙蝠算法的速度和位置更新公式 (9) ~ (11) 更新蝙蝠的位置时, 生成的新位置中元素会出现实数, 而本文算法中每只蝙蝠的位置应该处于离散空间中, 即每个位置元素应该为 $\{1, 2, \dots, n\}$ 中的整数, 此时需要采取基于升序的排序规则进行调整, 具体表述如下: 将新位置中取值最小的元素用 1 代替, 取值第二小的元素用 2 代替, 依次类推, 取最大值的元素用 n 代替。例如, 经过速度和位置更新公式更新后, 某只蝙蝠的位置为 $[-0.8, 3.4, 2.5, 6.7, -5.4, 9.7, 4.8]$, 则利用基于升序的排序规则进行调整后, 该蝙蝠对应的位置应为 $[2, 4, 3, 6, 1, 7, 5]$ 。若不采用基于升序的排序规则, 而采用其它的取整方法, 如向下取整、向上取整或四舍五入取整等方法, 均不能得到 5.1 节所要求的编码形式。

4.4 基于子路径的局部搜索

针对车辆路径问题这一类的组合优化问题, 由于问题解空间是离散的, 不能直接运用公式 (14) 的方法来定义蝙蝠的局部搜索。因此, 在当前最优蝙蝠 X^* 的周围进行局部搜索时, 首先按照 5.1 节的方式进行解码, 然后随机选择一条子路径 (需保证该子路径内客户数大于 1), 随机选择子路径内的两个客户, 然后采用如下 3 种操作:

1) 交换操作: 交换其位置;

2) 插入操作: 将后一个客户插入到前一个客户的前面;

3) 逆序操作: 将它们之间的客户序列根据原来的顺序逆序排列。

很显然, 上述任何一种搜索策略均未改变子路径内的客户模糊需求量之和的值, 新位置依然满足约束条件

$$Cr\{\sum_{i=1}^n q_i y_{ik} \leq Q\} \geq Cr^*。$$

基于子路径的局部搜索策略步骤如下。

1) 获取当前最优蝙蝠 X^* , 并按照 5.1 节的方式进行解码。令 $ct = 1$, 设置最大尝试次数 ct_{\max} 。

2) 令 $l \leftarrow 1$ 。

3) 若 $l = 1$, 执行交换操作; 若 $l = 2$, 执行插入操作; 若 $l = 3$, 执行逆序操作; 若得到的新位置 X_{new} 优于 X^* , 则接受 X_{new} , 直接返回 5); 否则, 令 $l = l + 1$, 重新执行 3) 直至 $l > 3$, 并不断更新三种操作所获得的最好的 X_{new} , 返回 4)。

4) 令 $ct = ct + 1$, 若 $ct < ct_{\max}$, 则返回 2)。

5) 结束局部搜索。

4.5 惯性权重的调整

基本蝙蝠算法通过向处于最优位置的蝙蝠学习, 从而进行速度和位置更新, 达到快速收敛, 但这容易产生早熟现象, 导致算法过早地处于停滞阶段。为提高算法在运行前期的全

局搜索能力和后期的局部搜索能力，本文在速度更新公式 (9) 中引入惯性权重，见下式 (15)。

$$V_i^{t+1} = \omega V_i^t + (X_i^t - X^*) \times f_i \tag{15}$$

其中 ω 为惯性权重。在迭代初期，为使算法拥有较强的全局搜索能力，需要一个较大的 ω ；在迭代后期，需要在当前最优解周围进行精确搜索，需要较小的 ω 。为此，对惯性权重 ω 采用基于非线性调整的策略，见下式 (16)。

$$\omega = \frac{(\max_iter - iter)^\beta}{(\max_iter)^\beta} \times (\omega_{\max} - \omega_{\min}) + \omega_{\min} \tag{16}$$

式 (16) 中 ω_{\max} 和 ω_{\min} 分别表示最大惯性权重和最小惯性权重， \max_iter 为最大迭代次数， $iter$ 为当前迭代次数， β 为区间 $[0,1]$ 内的常数。

4.6 交叉操作

为维持种群多样性，避免算法过早地陷入局部极小，在每一次迭代完成后，采用精英保留策略，对种群内蝙蝠进行优劣排序，排在前一半的蝙蝠直接进入下一次迭代，而排在后一半的蝙蝠两两之间随机进行交叉，随机产生两个交叉点，将第一个蝙蝠的交叉段移到另一个蝙蝠的首部，消去后面相同的元素，得到两个新的位置，进入下一次迭代，如图 1 所示为两只蝙蝠进行交叉的过程，虚线框内的元素为交叉段。不同于 5.4 节的所述的局部搜索，局部搜索为某条子路径内的客户调整，而通过交叉操作，在很大程度上可以看作子路径之间的客户调整，这将有利于扩大搜索空间，避免算法早熟。

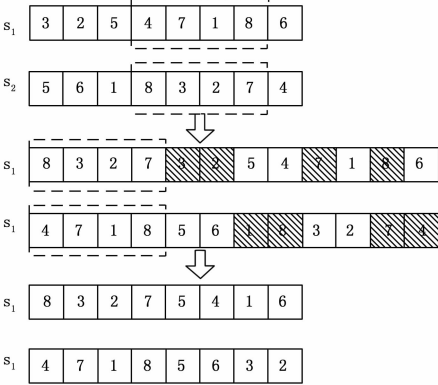


图 1 蝙蝠的交叉操作

5 算法步骤

Step1: 参数的初始化。设置主观偏好值 Cr^* ，随机模拟次数 M ，最大迭代次数 \max_iter ，蝙蝠种群规模 pop_size ，最大惯性权重 ω_{\max} 和最小惯性权重 ω_{\min} ，最大脉冲频率 f_{\max} 和最小脉冲频率 f_{\min} ，最大脉冲发射率 r^0 ，最大脉冲响度 A^0 ，脉冲发射率增加系数 γ ，响度衰减系数 α 。

Step2: 随机生成蝙蝠种群和每只蝙蝠的速度向量，计算其目标值，并找出当前群体中的最优位置 X^* 及其对应的目标值 Z^* 。

Step3: 对每只蝙蝠 X_i ，生成随机数 $rand_1$ ，若 $rand_1 > r_i$ ，则对处于当前群体中最优位置的蝙蝠 X^* 按照 5.4 节方式进行局部搜索，产生一个新位置 X_{new} ，否则，按照公式 (10)、(11)、(15)、(16) 进行更新，产生一个新位置 X_{new} 。

Step4: 对新位置 X_{new} 进行评估。产生随机数 $rand_2$ ，

若 $rand_2 < A_i$ 且新位置 X_{new} 对应的目标值比 X^* 更优，即 $Z_{X_{new}} < Z^*$ ，则 X_i 跳至 X_{new} ，并按公式 (13)、(14) 更新脉冲响度 A_i 和脉冲发射率 r_i ；否则， X_i 位置保持不变。

Step5: 对所有蝙蝠进行排序，更新当前最优蝙蝠 X^* 和其目标值 Z^* 。

Step6: 判断是否满足迭代终止条件，若满足，则返回 Step7。否则，保留排在前一半的蝙蝠，对排在后一半的蝙蝠进行交叉操作，代替原来的父代蝙蝠，并返回 Step3。

Step7: 终止迭代，输出最优蝙蝠 X^* 和其目标值 Z^* 。

6 仿真实验

随机生成一组实验数据，包含 40 个客户和 1 个车场，其中每个客户的横坐标和纵坐标均在范围 $[100 \times 100]$ 内随机产生，车场的横坐标和纵坐标均为 50。车辆的标记载重量 Q 均为 100 吨，每个客户的模糊需求在车辆标记载重量 Q 范围内随机产生，相关参数设置如下表 1：

表 1 模型参数及蝙蝠算法参数设置

pop_size	M	ct_{\max}	f_{\max}	f_{\min}	ω_{\max}	ω_{\min}
40	100	5	10	-10	0.9	0.1
β	r^0	A^0	γ	α	\max_iter	
0.5	0.9	0.9	0.05	0.95	200	

6.1 仿真结果

选取主观偏好值 Cr^* 为 0.5，利用改进的蝙蝠算法进行仿真，利用 Matlab 编程得到如下的路径安排（见表 2 和图 2）。

表 2 最优解对应的路径安排

子路径序号	路径安排	子路径序号	路径安排
1	0-13-39-24-0	10	0-29-9-25-0
2	0-19-1-0	11	0-35-20-0
3	0-27-23-0	12	0-14-7-15-0
4	0-34-21-5-37-0	13	0-3-8-0
5	0-36-0	14	0-17-0
6	0-31-0	15	0-12-0
7	0-26-28-0	16	0-40-30-2-0
8	0-4-0	17	0-33-32-38-0
9	0-10-16-22-0	18	0-18-11-6-0

此时该路径安排对应的总行驶距离为 2 151.8944，额外行驶距离为 134.9335，计划行驶距离为 2 016.960 9。

6.2 主观偏好值 Cr^* 对结果的影响分析

以上基本参数保持不变，让主观偏好值 Cr^* 在区间 $[0,1]$ 内变动，每个 Cr^* 下随机运行 10 次，并统计额外行驶距离、计划行驶距离和总行驶距离的均值，表 3 为主观偏好值 Cr^* 对结果的影响分析。

从表 3 所统计的结果来看，当主观偏好值 Cr^* 为 0 时，其额外行驶距离最大，计划行驶距离最小，此时决策者希望充分利用车辆的剩余载重能力，同时甘愿冒车辆剩余载重能力不能满足下一客户的“实际需求”而导致原计划路线“失败”的风险。随着主观偏好值 Cr^* 的增加，额外行驶距离单调递减，而计划行驶距离单调递增。当 Cr^* 取值小于 0.5 时，随着 Cr^* 取值增加，额外行驶距离的减小量大于计划行驶距

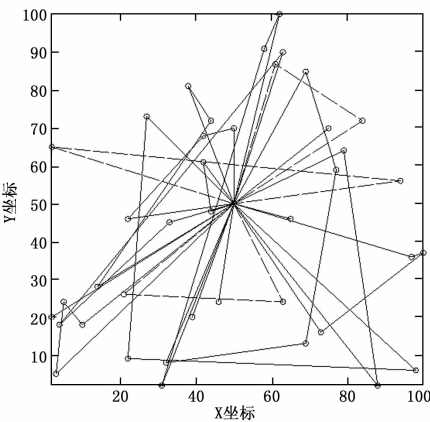


图 2 最优解对应的路径示意图

表 3 主观偏好值 Cr^* 对结果的影响

Cr^*	额外行驶距离	计划行驶距离	总行驶距离
0	784.815	1 688.17	2 472.985
0.1	630.6943	1 829.99	2 460.6843
0.2	493.3363	1 926.88	2 420.216
0.3	388.4686	2 028.65	2 417.119
0.4	214.9754	2 119.81	2 334.785
0.5	113.1742	2 171.58	2 284.754
0.6	38.99493	2 262.06	2 301.055
0.7	7.69294	2 313.21	2 320.903
0.8	0.052612	2 388.99	2 389.05
0.9	0	2 406.9	2 406.9
1	0	2 503.91	2 503.91

离的增加量,因而总行驶距离逐步递减;而当 Cr^* 取值大于 0.5 时,随着 Cr^* 取值增加,额外行驶距离的减小量小于计划行驶距离的增加量,因而总行驶距离逐步递增。综合分析看,当 Cr^* 取值为 0.5 时,总行驶距离最小。

6.3 本文算法与其它算法的对比分析

针对本节所选取的仿真实例,设定 $Cr^* = 0.5$,种群规模 $pop_size = 40$,最大迭代次数 $\max_iter = 200$,随机模拟次数 $M = 100$,分别采用本文算法、基本蝙蝠算法、文献 [15] 所介绍的粒子群算法、文献 [10] 所介绍的遗传算法、文献 [7] 所介绍的差分进化算法进行仿真,对仿真实例各运行 10 次,其中基本蝙蝠算法参数中最大脉冲频率 f_{\max} 、最小脉冲频率 f_{\min} 、最大脉冲发射率 ρ^0 、最大脉冲响度 A^0 、脉冲发射率增加系数 γ 、响度衰减系数 α 等参数与本文一致;粒子群算法中惯性权重为 0.5,加速度常数 $c_1 = 1.5, c_2 = 1.5$;遗传算法中交叉概率为 0.3,变异概率为 0.2;差分进化算法中最小交叉概率为 0.3,最大交叉概率为 0.9,变异缩放因子为 0.5。对 10 次运行结果的总行驶距离最优值、最差值和平均值进行比较,见下表 4 所示。

表 4 五种算法运行结果对比

	最优值	最差值	平均值
本文算法	2 151.8944	2 333.105	2 284.754
基本蝙蝠算法	2 249.8123	2 357.2181	2 334.7874
粒子群算法	2 235.1252	2 352.2008	2 330.0194
遗传算法	2 165.5338	2 341.9322	2 291.0297
差分进化算法	2 131.9367	2 310.6029	2 282.4266

从表 4 可以看出,本文算法的最优值、平均值明显优于基本蝙蝠算法、粒子群算法和遗传算法,稍弱于差分进化算法。另外,五种算法取得最优值时算法的迭代过程见下图 3 所示。

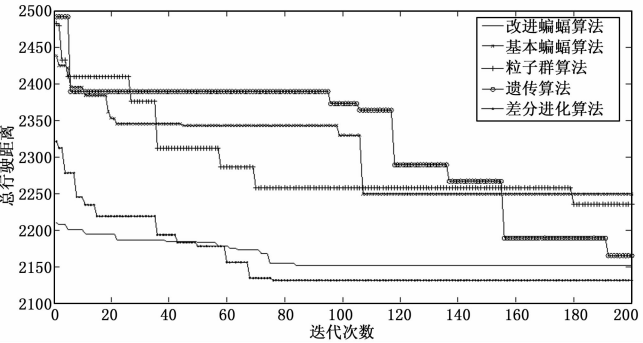


图 3 五种算法的迭代过程比较

从迭代过程来看,本文提出的改进蝙蝠算法对比基本蝙蝠算法、粒子群算法和遗传算法,能较快地搜寻到较优解,这得益于该算法引入了基于非线性调整的惯性权重,并对处于较差位置的蝙蝠进行交叉,有效地扩大了搜索范围,同时使用了基于子路径的局部搜索策略,使算法又具备一定的局部搜索能力。不过,从最终解的质量来看,该算法稍劣于差分进化算法,对该算法的改进还需要进一步的研究。

7 结论

本文针对带模糊需求的车辆路径问题,运用改进的蝙蝠算法进行了求解。在算法迭代过程中,为了有效地扩大搜索范围,避免算法陷入早熟,引入非线性调整的惯性权重,对位置较差的蝙蝠进行交叉操作,同时,为了提高算法的局部搜索能力,提出了基于子路径的局部搜索策略。最后给出了仿真实验,分析了决策者主观偏好值对目标值的影响,并将改进的蝙蝠算法与其他算法进行比较,经过实验证明,该算法对于解决此类模糊 VRP 问题是有效的。

参考文献:

[1] Dantzing G, Ramser J. The truck dispatching problem [J]. Management Science, 1959, 10 (6): 80-91.

[2] Teodorovic D, et al. The fuzzy set theory approach to the vehicle routing problem when demand at nodes is uncertain [J]. Fuzzy Sets and Systems, 1996, 82: 307-317.

[3] 祝崇隽, 刘 民, 吴 澄, 等. 针对模糊需求的 VRP 的两种 2-OPT 算法 [J]. 电子学报, 2001, 29 (8): 1-2.

[4] 张建勇, 郭耀煌, 李 军. 模糊需求信息条件下的车辆路径问题研究 [J]. 系统工程学报, 2004, 19 (1): 74-78.

[5] 张建勇, 李 军. 模糊车辆路径问题的一种混合遗传算法 [J]. 管理工程学报, 2005, 19 (2): 23-26.

[6] 张建勇, 李 军. 模糊需求 VRP 的一种 Sweeping 启发式算法 [J]. 中国管理科学, 2007, 15 (10): 71-75.

[7] 曹二保, 赖明勇, 张汉江. 模糊需求车辆路径问题研究 [J]. 系统工程, 2007, 25 (11): 14-18.

[8] 曹二保, 赖明勇, 李董辉. 基于混合差分进化算法的模糊需求车辆路径问题 [J]. 系统工程理论与实践, 2009, 29 (2): 106-113.

[9] 彭北青. 第三方物流配送车辆路径问题模型及算法研究 [D]. 武

汉：华中科技大学，2009.

[10] 戎丽霞. 模糊需求条件下车辆路径问题的模糊模拟 [J]. 计算机工程与应用, 2010, 46 (18): 209-210.

[11] 吴天昇, 许继恒. 基于混合遗传算法的模糊需求车辆路径问题 [J]. 解放军理工大学学报 (自然科学版), 2014, 15 (5): 475-482.

[12] Xue L. Fuzzy Simulation on the Vehicle Routing Problem [J]. Information Technology Journal, 2013, 12 (21): 6098-6102.

[13] Kuo R J, Zulvia Ferani E, Kadarsah Suryadi. Hybrid particle swarm optimization with genetic algorithm for solving capacitated vehicle routing problem with fuzzy demand - A case study on garbage collection system [J]. Applied Mathematics and Computation, 2012, 219: 2574-2588.

[14] Cao Erbao, Lai Ming yong. The open vehicle routing problem with fuzzy demands [J]. Expert Systems with Application, 2010, 37 (3): 2405-2411.

为突出方面，但总体来说本文的数字图像处理技术的人脸检测算法综合性较强。但从表 1 中显示的人脸检测实验结果可以看出，想要达到比较稳定的人脸检测准确率，这 4 种算法最少选取特征数目依次为 80、75、45 和 34 个。在 4 种方法中人脸检测准确率相差无几，但本文方法提出的数字图像处理技术的人脸检测算法需要提取的人脸特征数目最少，所以人脸检测速度最快。

从图 3 图 4 的检测结果可以看出，想要得到较为理想的人脸检测效果人脸特征数目应控制在 30 个之后。

以相同的实验样本对文献 [6] 文献 [7] 文献 [8] 所提算法进行人脸实验，并对 4 种方法的性能进行比较。使用相同的实验数据，重复上述上人脸检测实验。在试验中使用本文方法时，对人脸特征相关度设定阈值，设该阈值为 $\epsilon = 0.25$ 。本实验将对这 4 中算法的结果进行评估，用 ROC 曲线表示 4 种算法的特征曲线图。人脸检测的实验结果将由两种方式表达，即阳性（人脸）以及阴性（非人脸），阳性样本又可以叫真阳性，阴性样本又被称为假阴性。假阴性用 X 轴表示，真阳性率用 Y 轴表示。实验比较结果如图 5 所示。

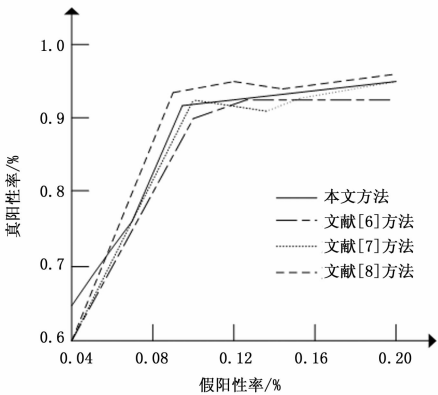


图 5 各种方法数字图像人脸特征的曲线图

表 1 显示出这 4 种算法的算法性能稳定时，使用的人脸特征数目，TPR、FPR、人脸检测时间等数据。

表 1 各种方法的结果对照

算法	本文方法	文献[6]	文献[7]	文献[8]
特征数目	80	75	45	34
真阳性率(%)	94	92.3	95	95.4
假阳性率(%)	2.4	3.1	2.2	1.9
平均时间(%)	0.547	0.194	0.137	0.083

从图 5 的数字图像人脸特征曲线中可以看出，在人脸检测性能方面，本文的方法效果最好，文献 [5-7] 的算法都有较

[15] Yang Peng, Ye—mei Qian . A particle swarm optimization to vehicle routing problem with fuzzy demands [J]. Journal of Convergence Information Technology, 2010, 8 (5).

[16] Yang X S. A new metaheuristic bat—inspired algorithm [C]. Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer, 2010: 65-74.

[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): 79-86.

[18] 马祥丽, 张惠珍, 马 良. 带时间窗物流配送车辆路径问题的蝙蝠算法 [J]. 计算机工程与应用, 2016, 52 (11): 254-264.

[19] Zhou Y Q, Xie J, Zheng H Q. A hybrid bat algorithm with path relinking for capacitated vehicle routing problem [J]. Mathematical Problems in Engineering, 2013: 1-10.

[20] 刘长平, 叶春明. 具有 Lévy 飞行特征的蝙蝠算法 [J]. 智能系统学报, 2013, 8 (3): 240-246.

3 结论

针对当前方法在人脸检测的过程中需要提取的人脸特征数目较多、检测速度较慢，从而降低了人脸检测效率。提出一种基于数字图像处理技术的人脸检测算法。仿真实验结果表明，所提检测方法所需人脸特征数据较少，可以快速准确对人脸进行识别检测。

参考文献：

[1] 马小虎, 谭延琪. 基于鉴别稀疏保持嵌入的人脸识别算法 [J]. Acta Automatica Sinica, 2014, 40 (1): 73-82.

[2] 张宏星, 邹 刚, 赵 键, 等. 基于 Gabor 特征与协同表示的人脸识别算法 [J]. 计算机工程与设计, 2014, 35 (2): 666-670.

[3] 林国军, 解 梅. 一种鲁棒协作表示的人脸识别算法 [J]. 计算机应用研究, 2014, 31 (8): 2520-2522.

[4] 周 全, 魏 昕, 陈建新, 等. 一种基于稠密 SIFT 特征对齐的稀疏表达人脸识别算法 [J]. 电子与信息学报, 2015, 37 (8): 1913-1919.

[5] 谢 佩, 吴小俊. 基于 Shearlet 变换和均匀局部二值模式特征的协作表示人脸识别算法 [J]. 计算机应用, 2015, 35 (7): 2056-2061.

[6] 唐雨佳, 周李威, 陈 耿, 等. 一种自适应加权 SpPCA 单样本人脸识别算法 [J]. 计算机应用研究, 2014, 31 (9): 2856-2858.

[7] 陈立潮, 张秀琴, 潘理虎, 等. 煤矿考勤系统中人脸识别算法的研究 [J]. 工矿自动化, 2015, 41 (4): 69-73.

[8] 张顺岚, 莫建文. 基于双目视觉的三维人脸识别算法 [J]. 电视技术, 2014, 38 (9): 214-217.

[9] 王庆伟, 应自炉. 一种基于 Haar—Like T 特征的人脸检测算法 [J]. 模式识别与人工智能, 2015, 28 (1): 35-41.

[10] 贾海鹏, 张云泉, 袁 良, 等. 基于 OpenCL 的 Viola—Jones 人脸检测算法性能优化研究 [J]. 计算机学报, 2016, 39 (9): 1775-1789.