

# 基于改进型蜻蜓算法的车辆路径问题研究

陶文瀚<sup>1</sup>, 赵晨聪<sup>1</sup>, 孙翌博<sup>2</sup>, 刘晨磊<sup>3</sup>, 孙知信<sup>1</sup>, 孙哲<sup>1</sup>

(1. 南京邮电大学 现代邮政学院 & 现代邮政研究院, 江苏 南京 210003;

2. 常州工学院 计算机信息工程学院, 江苏 常州 213032;

3. 南京邮电大学 宽带无线通信与传感网技术教育部重点实验室, 江苏 南京 210003)

**摘要:**随着现代物流业的高速发展,物流配送过程中的车辆路径问题已经成为影响物流行业发展的关键因素。为了实现在物流配送过程中有效地提高配送效率,减少车辆的空车行驶率和行驶距离,降低运输成本,提出了一种改进型蜻蜓算法。将随机学习优化的思想融入到传统蜻蜓算法中,优化了原算法存在的收敛精度低、最优解容易陷入局部收敛等缺陷,并将该算法应用到带软时间窗约束的车辆路径问题上。首先根据配送货物的运输成本、仓库的驻留成本、超时惩罚成本等因素,构建出一种综合成本最小化的车辆路径问题的数学模型,并用该算法对该问题进行求解。然后通过系统仿真模拟构建最优路径,并与其他智能优化算法进行对比分析,证实了该算法的有效性和可行性,同时也证明了该算法在求解带软时间窗约束的车辆路径问题上有着较好的性能。

**关键词:**物流配送; 车辆路径问题; 时间窗; 蜻蜓算法

中图分类号: TP301

文献标识码: A

文章编号: 1673-629X(2020)12-0170-06

doi: 10.3969/j.issn.1673-629X.2020.12.030

## Research on Vehicle Routing Problem Based on Improved Dragonfly Algorithm

TAO Wen-han<sup>1</sup>, ZHAO Chen-cong<sup>1</sup>, SUN Yi-bo<sup>2</sup>, LIU Chen-lei<sup>3</sup>, SUN Zhi-xin<sup>1</sup>, SUN Zhe<sup>1</sup>

(1. School of Modern Posts & Institute of Modern Posts, Nanjing University of Posts and

Telecommunications, Nanjing 210003, China;

2. School of Computer Science and Information Engineering, Changzhou Institute of Technology,

Changzhou 213032, China;

3. Key Lab of Broadband Wireless Communication and Sensor Network Technology of Ministry of Education,

Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

**Abstract:** With the rapid development of the modern logistics industry, the problem of vehicle routing in the logistics distribution process has become a key factor affecting the development of the logistics industry. In order to effectively improve the distribution efficiency, reduce the empty rate and distance of vehicles, and reduce transportation costs during the logistics distribution process, we propose an improved dragonfly algorithm that incorporates the idea of random learning optimization into the traditional dragonfly algorithm. The disadvantage of the original algorithm such as low convergence accuracy and easily falling into local convergence of optimal solution is optimized. The algorithm is applied to vehicle routing problems with soft time window constraints. Firstly, a mathematical model of the comprehensive cost minimization of vehicle routing problem is constructed based on factors such as the transportation cost of the delivered goods, the resident cost of the warehouse and the cost of overtime penalties, and the proposed algorithm is used to solve the problem. Then the optimal path is constructed through system simulation, and compared with other intelligent optimization algorithms, the effectiveness and feasibility of the proposed algorithm are verified. At the same time, it also proves that the proposed algorithm has better performance in solving vehicle routing problems with soft time window constraints.

**Key words:** logistics distribution; vehicle routing problem; time window; dragonfly algorithm

## 0 引言

车辆路径问题( vehicle routing problem, VRP)<sup>[1]</sup>是物流配送优化环节中至关重要的一环。在物流配送过程中,通过为配送车辆选择最优的运输路径,合理安排车辆的调度顺序,可以有效减少车辆的空车行驶率和行驶距离。该问题最早由国外学者 Dantzig 和 Ramser 在 1959 年提出,属于 NP-Hard 问题<sup>[2]</sup>。目前,国内外学者对 VRP 问题已经进行了广泛而深入的研究,VRP 已经应用到垃圾车的线路优化、连锁商店的送货线路优化等<sup>[3]</sup>众多社会领域。

但是,传统的 VRP 问题往往只考虑到配送距离而忽视了客户对配送时间的要求,进而影响客户对物流配送服务的满意度。带时间窗约束的车辆路径问题是原车辆路径问题的一种拓展,在经典组合优化车辆路径问题上,引进了客户对货物到达时间的时间窗约束。其中,时窗约束可以分为两种,一种是硬时间窗,要求车辆必须要在时窗内到达,早到必须等待,而迟到则拒收;另一种是软时窗,不一定要在时窗内到达,但是超出时间窗到达则需要处罚。

该文总结众多新型智能算法应用到车辆路径问题上的优缺点,分析当前路径规划上存在的问题和需求,提出了一种改进型蜻蜓算法,并将该算法应用到带软时间窗约束的车辆路径问题中,发现该算法在求解 VRP 问题上具有较高的可行性。

## 1 相关研究

目前,已有许多新型智能算法用于求解带时间窗约束的车辆路径问题<sup>[4]</sup>。文献[5]通过使用禁忌搜索算法和自适应大邻域搜索算法来解决具有时间依赖性的软时间窗和随机行驶时间的车辆路径问题,并说明该方法可用于解决其他一些大规模问题。文献[6]利用人工蜂群激发蜜蜂的觅食行为,运用混合超启发式算法求解带软时间窗的多目标车辆路径问题。文献[7]提出了一种使用分布估计算法的方法,通过使用广义的 Mallows 分布作为概率模型来描述解空间的分布,以解决带时间窗的车辆路径问题。文献[8]探索了时变多行车辆路径问题(TDMTVRP),使用了具有改进行进速度的模型,与容量较大的 VRPTW<sup>[9]</sup>相比,减少了行车时间和行进距离,并通过使用最近邻启发式算法获得初始解和禁忌搜索启发式算法搜索最优解。文献[10]提出了一种基于改进的头脑风暴优化算法(IBSO-ACO)的新型蚁群优化算法,以解决带有软时间窗的车辆路径问题。文献[11]开发了一种混合遗传算法的求解方法,对遗传算法和变量邻域搜索方法(VNS)进行了杂交,以解决带有软时间窗的静态和动态车辆路径问题。这些学者对带有时间窗的车辆

路径问题进行了系统而深入的研究,但他们的研究大多使用改进型的传统启发式算法,例如遗传算法、蚁群算法和禁忌搜索算法等。这些算法在带时间窗约束 VRP 的求解上取得了可喜的成果,但也存在不少问题,如禁忌搜索算法对初始解的依赖性较高,遗传算法存在局部搜索能力不强、易陷入早熟、总体上可行解的质量不是很高等缺点。

蜻蜓算法(dragonfly algorithm, DA)自从 2015 年被 Mirjalili<sup>[12]</sup>提出就得到了广泛的研究与应用。该新型算法已被成功用于医疗疾病预测诊断<sup>[13]</sup>、太阳能热系统优化<sup>[14]</sup>、小麦碰撞声信号检测与识别<sup>[15]</sup>等诸多领域。其中,Abdelaziz I. Hammouri 等<sup>[16]</sup>将蜻蜓算法运用到旅行商问题(TSP)的求解之中,验证了蜻蜓算法在求解类路径规划问题的可行性。文献[17]使用 DA 来估计随机部署在指定区域中的节点的位置,通过仿真实验来表明 DA 可以为基于范围的定位产生低误差。文献[18]提出了一种用于预测问题的带有极限学习机(ELM)系统的混合蜻蜓算法,通过利用 DA 在隐藏层中选择较少数量的节点,以加快 ELM 的性能。

然而,在软时间窗的约束下,针对原蜻蜓算法解决大规模问题时存在收敛速度慢、运算时间长、容易陷入局部最优解等缺陷<sup>[19]</sup>,该文提出一种改进型的蜻蜓算法应用于车辆路径问题,从而更好地提高物流运输车辆的配送效率。

## 2 问题描述及数学模型

带时间窗约束的车辆路径问题可以描述为:一个配送中心,拥有一定数量的同种车辆,车辆容量有限且已知,车辆满载货物由配送中心出发,向区域内若干客户配送同种商品,要求在完成配送任务的总路程最短的基础上派出的车辆数最少,并考虑在客户点的驻留成本、未在规定时间内送达的惩罚成本。

该文从实际物流车辆运输特点和自身实验条件出发,对构建的带时间窗约束的车辆路径问题数学模型进行如下假设:

- (1) 车辆必须从固定的配送站点出发,运输途中不经停该站点,在完成一趟运输过程之后才能回到配送站点;
- (2) 规定车辆以一种恒定的行驶速度在各站点之间行驶;
- (3) 各服务点的需求量、服务时间窗、服务时间不会临时变动;
- (4) 各服务点之间的路径都可达,并且不会产生任何突发事件影响车辆的行驶;
- (5) 一辆车只可以同时服务一个服务点、配送一

条路线。

根据以上数学模型假设,对构建的带时间窗的车辆路径问题数学模型中的参数和相关的变量进行如下定义:

$V = \{1, 2, \dots, n\}$  表示站点编号集合,  $n$  为站点总个数,编号 1 为起始站点。

$Pos$  表示配送站点和客户服务站点的位置集合,其中  $(Pos_i, Pos_i)$  表示站点  $i$  的横、纵坐标位置。

$Dis$  表示各个站点之间的距离,站点  $i$  与站点  $j$  之间的距离  $Dis_{ij}$  由式(1)给出。

$$Dis_{ij} = \sqrt{(Pos_{i_x} - Pos_{j_x})^2 + (Pos_{i_y} - Pos_{j_y})^2} \quad (1)$$

$Ser\_Time$  为站点的服务时间集合,其中  $Ser\_Time_i$  表示站点  $i$  的服务时间。

$Demand$  为站点的服务需求量集合,其中  $Demand_i$  表示站点  $i$  的货物需求量。

$TW$  表示各站点规定的服务时间窗,其中  $TW_{i_{last}}$  表示站点  $i$  服务时间窗的最大非处罚服务时间。

$duty$  记录车辆已经服务过的站点序列。

$v\_vel$  表示车辆行驶的速度。

$v\_cap$  表示车辆的最大载重量。

$\alpha_{ij}$  判断车辆是否经过站点  $i$  和站点  $j$  之间的路线。当  $\alpha_{ij} = 1$  时,表示车辆经过;当  $\alpha_{ij} = 0$  时,表示车辆没有经过。

$\beta_i$  判断站点  $i$  是否已经服务过。当  $\beta_i = 1$  时,表示站点  $i$  已经服务过;当  $\beta_i = 0$  时,表示没有服务过。

$\gamma_i$  判断车辆在到达站点  $i$  时时间是否超出该站点的时间窗上限。当  $\gamma_i = 1$  时,表示超过;当  $\gamma_i = 0$  时,表示没有。

$time$  记录车辆刚到达某一站点时的当前时间,其计算公式由式(2)给出。

$$time_i = \sum_{j=1}^n \sum_{k=1}^n \alpha_{jk} * Dis_{jk} / v\_vel + \sum_{j=1}^n \beta_j * Ser\_Time_j \quad (2)$$

$c_{trans}$ 、 $c_{ser}$  和  $c_{pun}$  分别表示车辆单位行驶路程、单位服务时间和单位惩罚时间成本。

综上所述,该文构建的数学模型目标函数由式(3)给出。

$$\begin{aligned} \min(cost) = & \sum_i \sum_j \alpha_{ij} * Dis_{ij} * c_{trans} + \sum_i \\ & Ser\_Time_i * c_{ser} + \sum_i \gamma_i * (time_i + \\ & Ser\_Time_i - TW_{i_{last}}) * c_{pun} \end{aligned} \quad (3)$$

约束条件如下所示。

$$\alpha_{ij} = \begin{cases} 1 & \text{车辆经过站点 } i \text{ 和站点 } j \text{ 之间的路线} \\ 0 & \text{车辆未经过站点 } i \text{ 和站点 } j \text{ 之间的路线} \end{cases} \quad (4)$$

$$\beta_i = \begin{cases} 1 & \text{站点 } i \text{ 已经服务过} \\ 0 & \text{站点 } i \text{ 没有服务过} \end{cases} \quad (5)$$

$$\gamma_i = \begin{cases} 1 & \text{站点 } i \text{ 服务时间已经超出时间窗} \\ 0 & \text{站点 } i \text{ 服务时间没有超出时间窗} \end{cases} \quad (6)$$

$$0 \leq d\_n \leq n \quad (7)$$

$$v\_cap \geq \sum_i demand_i \quad (8)$$

$$\sum_i \beta_i = n \quad (9)$$

### 3 模型实现

#### 3.1 蜻蜓算法概述

蜻蜓算法是受蜻蜓行为启发而提出的一种新型群智能算法。与其他的群智能优化算法类似的是,蜻蜓算法也有着较好的局部最优解避免能力和精确近似全局最优解的能力。

蜻蜓算法像大多数群智能算法一样皆是遵循“求生”的原则,蜻蜓个体有两个行为:寻找食物和躲避天敌,蜻蜓群体的位置移动由以下五种行为组成:

(1) 分离,即蜻蜓与相邻个体之间避免碰撞。

$$S_i = - \sum_{j=1}^n X - X_j$$

式中,  $X$  是当前个体的位置;  $X_j$  是第  $j$  个附近个体的位置;  $N$  是附近个体的个数。

(2) 结队,即相邻个体之间倾向于保持相同的速度。

$$A_i = \frac{\sum_{j=1}^n V_j}{n}$$

式中,  $V_j$  是第  $j$  个附近个体的速度。

(3) 聚集,即蜻蜓倾向于向相邻个体中心聚集。

$$C_i = \frac{\sum_{j=1}^n X_j}{n} - X$$

(4) 觅食,即蜻蜓对食物的倾向度。

$$F_i = X^+ - X$$

式中,  $X^+$  是蜻蜓食物的位置。

(5) 躲避天敌,即蜻蜓避免被天敌捕食,对其产生排斥。

$$E_i = X^- + X$$

式中,  $X^-$  是蜻蜓天敌的位置。

除以上五种行为之外,为了较为准确地模拟出蜻蜓的移动过程, Mirjalili 又引入了两个量:步长向量 ( $dX$ ) 和位置向量  $X$ 。步长向量计算公式如下(这是逐维定义的步长):

$$\Delta X_{i+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + w\Delta X_i$$

式中,  $s$  表示分离权重,  $a$  表示结队权重,  $c$  表示聚集权重,  $f$  表示食物因子,  $e$  表示天敌因子;  $S_i$  表示第  $i$  个体

分离之后的位置,  $A_i$  表示第  $i$  个体结队之后的位置,  $C_i$  表示第  $i$  个体聚集之后的位置,  $F_i$  表示第  $i$  个体蜻蜓食物的位置,  $E_i$  表示第  $i$  个体蜻蜓天敌的位置;  $w$  表示惯性权重;  $t$  表示当前迭代次数。

位置更新如下:

(6) 附近存在蜻蜓个体, 则按照如下方式更新:

$$X_{i,t+1} = X_i + \Delta X_{i,t+1}$$

(7) 附近不存在蜻蜓个体, 则执行莱维飞行:

$$X_{i,t+1} = X_i + \text{Levy}(d) \times X_i$$

式中,  $d$  是求解问题的维数。

### 3.2 改进型蜻蜓算法

通过研究发现, 采用传统蜻蜓算法求解带时间窗的车辆路径问题时, 存在收敛精度较低、最优解容易陷入局部收敛等缺陷。因此, 为了针对求解带时间窗的车辆路径问题, 该文根据以上设计的数学模型, 将随机学习优化<sup>[20]</sup>的思想融入到蜻蜓算法中, 重新设计了一种能够更有效应用于带时间窗约束的车辆路径问题研究的改进型蜻蜓算法。

该算法的主要改进点在于将传统蜻蜓算法结队、觅食、避敌、避撞、聚集五种行为的实值计算形式, 修改为随机学习目标位置对, 即利用随机学习优化方法完成以上行为的实现:

(1) 通过向同期蜻蜓种群中位置最优的蜻蜓随机学习一组排序, 更新位置序列, 优化局部最优解, 实现结队行为;

(2) 向历史蜻蜓种群中位置最优的蜻蜓随机学习一组排序, 优化位置序列, 趋向于全局最优解, 实现觅食行为;

(3) 通过远离同期蜻蜓种群中位置最差的蜻蜓位置, 避免陷入局部最优, 实现避敌行为;

(4) 发生碰撞时, 随机长度随机打乱一只蜻蜓位置, 保证种群多样性, 实现避撞行为。

该算法的具体实现步骤如下:

步骤 1: 初始化蜻蜓算法和数学模型的相关参数。其中, 每一只蜻蜓的位置向量表示一种选路方式, 即为除编号 1 外的  $n-1$  个站点编号的随机排列组合;

步骤 2: 计算初代各解, 即站点间的距离矩阵、每只蜻蜓位置向量所对应的成本值、初代最优解和最优成本值等;

步骤 3: 开始迭代计数, 令迭代标识符  $iter = 1$ ;

步骤 4: 通过结队、觅食、避敌、避撞等行为, 进行随机学习优化排序, 优化各蜻蜓的位置向量;

步骤 5: 利用目标函数计算每只蜻蜓位置向量所对应的运输成本值, 并且记录最优值和最优蜻蜓位置向量, 即记录最佳的成本值和对应的车辆路径规划方案;

步骤 6: 将本代最优值和历史最优值进行比较, 筛选得出最优成本值和最优蜻蜓位置向量;

步骤 7: 迭代标识符加 1;

步骤 8: 判断是否达到最大的迭代次数。若是, 进入步骤 9; 若否, 返回步骤 4;

步骤 9: 得出实验结果, 生成实验图表, 结束。

### 3.3 基于改进蜻蜓算法的模型实现

基于改进型蜻蜓算法的带时间窗约束的车辆路径问题模型实现流程如图 1 所示。

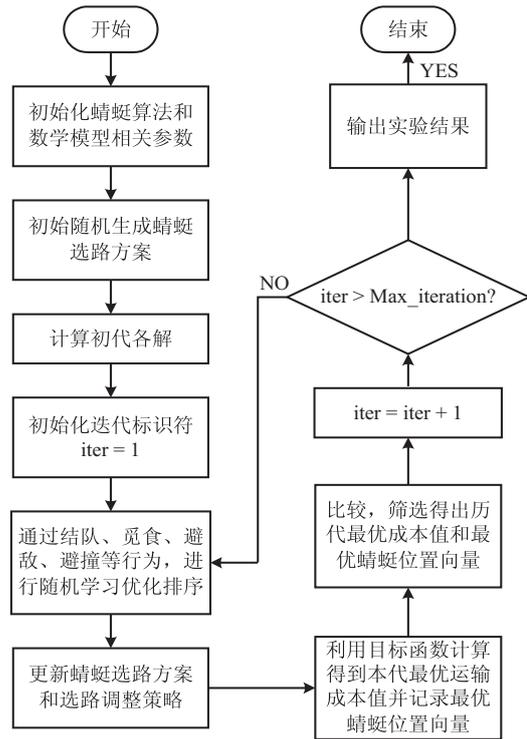


图 1 改进型蜻蜓算法实现流程

## 4 算例实验

### 4.1 实验参数设置

服务站点初始参数设置如表 1 所示。

表 1 服务站点参数设置

编号	站点位置 $x$	站点位置 $y$	货物需求量	时间窗	服务时间
1	35.00	35.00	0.00	0.00	230.00
2	41.00	49.00	10.00	161.00	171.00
3	35.00	17.00	7.00	50.00	60.00
4	55.00	45.00	13.00	116.00	126.00
5	55.00	20.00	19.00	149.00	159.00
6	15.00	30.00	26.00	34.00	44.00
7	25.00	30.00	3.00	99.00	109.00
8	20.00	50.00	5.00	81.00	91.00
9	10.00	43.00	9.00	95.00	105.00
10	55.00	60.00	16.00	97.00	107.00

车辆初始参数设置如表 2 所示。

表 2 车辆参数设置

参数名称	数值
车辆数量 v_n	1
车辆速度 v_vle	5
车辆最大载重量 v_cap	150

改进蜻蜓算法的初始参数设置如表 3 所示。

表 3 蜻蜓算法参数设置

参数名称	数值
蜻蜓数量 SearchAgents_no	50
最大迭代次数 Max_iteration	5 000
蜻蜓种群 Fly. route	[]
最优解迭代次数 Best. iter	0
最优解成本值 Best. cost	[]
最优解蜻蜓位置 Best. pos	[]
历史最优路径 Best. route	[]
当前最优路径 MinPos	0

实验设计的拓扑网络环境如图 2 所示。

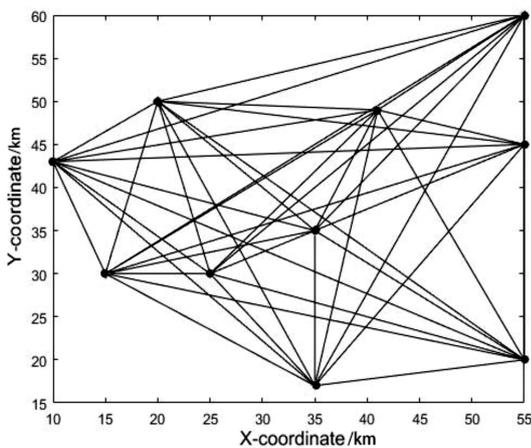


图 2 实验物流网络拓朴

根据该文设计的带时间窗车辆路径问题数学模型约束条件,该物流拓扑网络一共拥有 3 628 800 种车辆路径规划方案。

### 4.2 实验结果

针对文中的车辆路径问题所提出的数学模型,将遗传算法(GA) 禁忌搜索算法(TSA)、传统蜻蜓算法(DA)和改进后的蜻蜓算法(RDA)一同进行实验测试并进行比较。为保证实验的公平性和客观性,将以上四种算法的迭代次数设为 1 000 次,采用相同的成本函数进行运算,得到的算法对比结果如图 3 所示。

由实验结果可以看出,无论是在求解精度还是收敛速度上,改进型蜻蜓算法都展现出优越的性能,充分说明改进型蜻蜓算法在求解车辆路径问题时具有更加

稳定的性能。

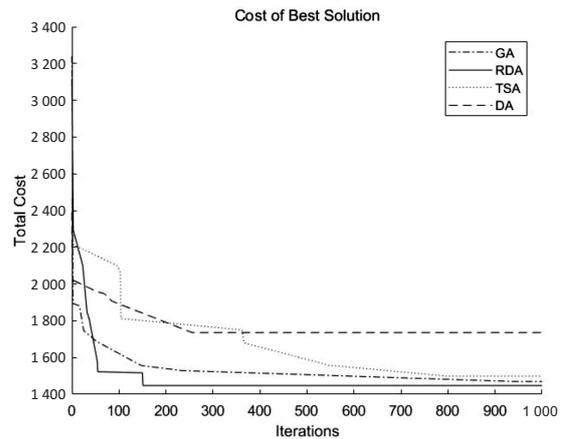


图 3 算法对比

经过实验得到最优的综合成本为 1 447.23。其对应的车辆路径规划方案如图 4 所示。

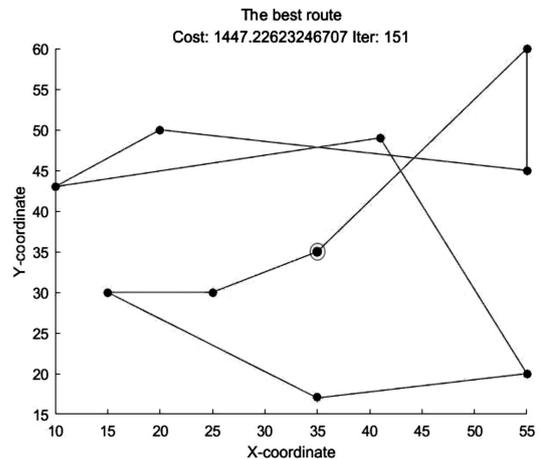


图 4 最优路线规划

路线规划方案为 1→7→6→3→5→2→9→8→4→10。其中,加圈位置为站点 1,即始发站点。

## 5 结束语

讨论了带软时间窗约束的车辆路径问题,构建了一种更符合配送中心与顾客服务目标优化的带时间窗约束的车辆路径问题数学模型。通过设将随机学习优化的思想融入到原先的蜻蜓算法之中,针对车辆路径问题数学模型的求解,重新设计了一种能够更有效应用于该问题求解的改进型蜻蜓算法。

该算法的主要改进之处为:将传统的蜻蜓算法结伴、觅食、避敌、避撞、聚集五种行为原则从原先的实值计算形式修改成为随机学习目标位置对,即利用随机学习优化方法完成以上行为的实现,使改进型蜻蜓算法更符合解的特性,从而得到一个更优的结果。

通过 Matlab 仿真实验证明了将蜻蜓算法应用到带软时间窗约束的车辆路径问题求解的可行性,并且验证了改进型蜻蜓算法用于该数学模型实现的有效

性。下一步将在该研究的基础上,提高算法收敛速度和收敛精度,改进蜻蜓随机学习的优化方式,并将该算法应用于其他领域的优化。

#### 参考文献:

- [1] 张文博,苏 秦,程光路. 基于动态需求的带时间窗的车辆路径问题[J]. 工业工程与管理,2016,21(6):68-74.
- [2] DANTZIG G B,RAMSER J H. The truck dispatching problem[J]. *Management Science*,1959,6(1):80-91.
- [3] KIM G,ONG Y S,HENG C K, et al. City vehicle routing problem (city VRP): a review[J]. *IEEE Transactions on Intelligent Transportation Systems*,2015,16(4):1654-1666.
- [4] 李珍萍,张煜炜. 带时间窗和服务顺序约束的多需求车辆路径问题[J]. 控制与决策,2019,34(7):1565-1570.
- [5] TAS D,DELLAERT N,VAN WOENSEL T, et al. The time-dependent vehicle routing problem with soft time windows and stochastic travel times[J]. *Transportation Research Part C: Emerging Technologies*,2014,48:66-83.
- [6] IQBAL S, KAYKOBAD M, RAHMAN M S. Solving the multi-objective vehicle routing problem with soft time windows with the help of bees [J]. *Swarm and Evolutionary Computation*,2015,24:50-64.
- [7] PÉREZ-RODRÍGUEZ R, HERNÁNDEZ-AGUIRRE A. A hybrid estimation of distribution algorithm for the vehicle routing problem with time windows[J]. *Computers & Industrial Engineering*,2019,130:75-96.
- [8] SUN Y, WANG D, LANG M, et al. Solving the time-dependent multi-trip vehicle routing problem with time windows and an improved travel speed model by a hybrid solution algorithm [J]. *Cluster Computing*, 2019, 22: 15459 - 15470.
- [9] 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.
- [10] WU L, HE Z, CHEN Y, et al. Brainstorming-based ant colony optimization for vehicle routing with soft time windows [J]. *IEEE Access*,2019,7:19643-19652.
- [11] BOUCHRA B,BTISSAM D,MOHAMMAD C. A hybrid genetic algorithm for the static and dynamic vehicle routing problem with soft time windows[C]//2016 3rd international conference on logistics operations management (GOL). Fez, Morocco;IEEE,2016:1-9.
- [12] MIRJALLI S. Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems [J]. *Neural Computing and Applications*,2016,27(4):1053-1073.
- [13] YASEN M, AL-MADI N, OBEID N. Optimizing neural networks using dragonfly algorithm for medical prediction [C]//2018 8th international conference on computer science and information technology (CSIT). Amman; IEEE,2018:71-76.
- [14] SURESH V, SREEJITH S. Generation dispatch of combined solar thermal systems using dragonfly algorithm[J]. *Computing*,2017,99(1):59-80.
- [15] 马彧廷,郭 敏. 基于极限学习与蜻蜓算法的小麦碰撞声信号检测与识别[J]. 电子设计工程,2016,24(5):8-11.
- [16] HAMMOURI A I, SAMRA E T A, AL-BETAR M A, et al. A dragonfly algorithm for solving traveling salesman problem [C]//2018 8th IEEE international conference on control system, computing and engineering (ICCSCE). Penang, Malaysia; IEEE,2018:136-141.
- [17] DAELY P T, SHIN S Y. Range based wireless node localization using dragonfly algorithm[C]//2016 eighth international conference on ubiquitous and future networks (ICUFN). Vienna, Austria; IEEE,2016:1012-1015.
- [18] SALAM M A, ZAWBAA H M, EMARY E, et al. A hybrid dragonfly algorithm with extreme learning machine for prediction[C]//2016 international symposium on innovations in intelligent systems and applications (INISTA). Sinaia, Romania; IEEE,2016:1-6.
- [19] 吴伟民,吴汪洋,林志毅,等. 基于增强个体信息交流的蜻蜓算法[J]. 计算机工程与应用,2017,53(4):10-14.
- [20] 陶 卿,马 坡,张梦晗,等. 机器学习随机优化方法的个体收敛性研究综述[J]. 数据采集与处理,2017,32(1):17-25.