

基于萤火虫算法的应急救援车辆调度^①

王付宇^{1,2}, 王涛¹, 叶春明²

¹(安徽工业大学 管理科学与工程学院, 马鞍山 243032)

²(上海理工大学 管理学院, 上海 200093)

摘要: 针对突发灾害事件情景下交通路网容量限定的特点, 引用 BRP 路阻函数求解各路段车辆行驶时间, 建立救援路径最短和车辆行驶时间最短双目标车辆调度模型; 设计改进离散的萤火虫算法, 构建算例对模型求解, 求解结果与遗传算法的求解结果进行对比, 验证了该算法的可行性和能更好的满足应急救援车辆调度的需要。

关键词: 车辆调度; 路网容量; 萤火虫算法; 应急救援

引用格式: 王付宇, 王涛, 叶春明. 基于萤火虫算法的应急救援车辆调度. 计算机系统应用, 2017, 26(9): 188-194. <http://www.c-s-a.org.cn/1003-3254/5975.html>

Emergency Rescue Vehicle Scheduling Based on Glowworm Swarm Optimization Algorithm

WANG Fu-Yu^{1,2}, WANG Tao¹, YE Chun-Ming²

¹(School of Management Science and Engineering, Anhui University of Technology, Maanshan 243032, China)

²(School of Management, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: In view of the characteristics of traffic networks capacity under the situation of mass emergency, this paper cites the BPR impedance function to solve the vehicle travel time between the various sections. It builds the shortest path and the shortest vehicle scheduling model and designs the improved discrete Glowworm Swarm Optimization Algorithm. It constructs a numerical example to solve the model and the results are compared with the results of genetic algorithm. The feasibility of the algorithm is verified and can better meet the needs of emergency rescue vehicle scheduling.

Key words: vehicle scheduling; road network capacity; glowworm swarm optimization algorithm; emergency rescue

近年来, 世界自然、社会环境急剧变化, 地震等突发灾害事件频繁发生, 社会稳定和社会安全受到严重威胁, 人们正常的生产和生活由于这些频繁发生的突发事件受到了极大的干扰. 突发灾害事件具有不确定性和大规模性等特征, 往往此类事件发生是都伴随着应急物资(如: 食品、医疗人员、设备等)的配送. 早在 1959 年 Dantzig 就提出车辆调度问题(VRP)并提出解决办法来解决这类物流配送的问题, 但随着城市结构和计算机网络等的发展, 该类问题已经上升到一定的高度, 研究对象和解决办法也相对与以前有了较大的改变. 针对应急救援模型的研究 Najafi M 等人^[1]针对

地震发生初期, 资源短缺这个特点, 提出了一种多目标、多模式、多种物资、多周期随机模型来管理地震条件下两种商品的物流, 该模型旨在综合考虑应急救援条件所涉及的不确定性因素.

G. Nikolakopoulou 等人^[2]研究了平衡车辆时间使用时间为目标的车辆路径优化问题. Wohlgemuth S 等人^[3]分析了确定性需求条件下应急救援车辆的动态路径问题. Wang H 等人^[4]考虑行程时间、总成本和可靠性的分配问题, 构造了一个非线性整数开放位置路径优化模型. 谈晓勇等人^[5]以可变双向距离、道路风险和成本最小为目标建立应急救援车辆调度的多目标模型,

① 基金项目: 国家自然科学基金(71271138); 教育部人文社会科学青年基金(14YJC630119); 安徽省高校人文社科研究重大项目(SK2014ZD016); 住建部软科学研究项目(2015-R2-057)

收稿时间: 2017-01-02; 采用时间: 2017-02-13

以此来体现应急救援车辆调度问题与普通车辆调度问题的差别。谢秉磊等人^[6]为优化传统的车辆路径优化问题,提出需求可分的车辆路径优化模型,并加强模型的约束,将原模型转变为等价的改进 SDVRP。孙丽君等人^[7]对车辆路径优化问题进行了综述,将原问题分为两种类型:图模型和数学模型,并分析了两类模型的优缺点。针对求解算法的研究,Nabila Azi 等人^[8]介绍了一种精确算法求解带时间窗和多路径的单车路径问题并将该算法分为两个阶段,第一阶段是路径的生成,第二阶段是路径的选取与排序。Hong Ma 等人^[9]提出了一种带有自适应惩罚机制的禁忌搜索算法,解决带时间窗和车辆吨位容量限制的单车路径问题,并应用于香港某运输公司的危险材料运输项目,经过大量计算,验证了方法的有效性。王晓博等人^[10]采用混合遗传启发式算法求解多车场、多车型的装卸混合车辆调度模型,引入了2-交换变异策略,并结合爬山算法加强染色体的局部搜索能力,最后对混合遗传算法求得的精英种群进行禁忌搜索,以此提高搜索效率。陈建军等人^[11]利用蚁群算法求解物流配送路径优化问题,并利用仿真证明蚁群算法具有搜索速度快的特点。王飞等人^[12]提出一种改进的粒子群算法求解带时间窗车辆调度问题,该算法在惯性权重递减的基础上通过群体极值进行*t*分布变异,克服了标准粒子群算法存在早熟收敛和易陷入局部解的问题。

综上所述,在应急救援车辆调度这个问题的研究中,国内外学者做了深入的研究,积累了丰富的理论基础。但是,对应急救援时间的处理,大多数学者都默认为应急救援车辆行驶路径最短即为车辆行驶时间最短,考虑到车辆行驶时间受到多种不确定因素的影响,为此,本文在路径最短的模型基础上,将路网容量限制应用于车辆行驶时间的预测,建立路径最短与时间最短双目标优化模型,并采用新型的群智能优化算法——离散的萤火虫算法对问题进行求解,以期待更加符合应急救援特殊情况下的车辆调度。

1 路径最短优化模型

1.1 多目标的确定

应急救援具有紧急性、急迫性和不确定性,在最短的时间里以最快的速度将应急救援物资送至待救援点是首要目的,也是减少人员伤亡、财产损失等各种损失的主要途径。对于时间的处理,只考虑车辆行驶路

径最短是远远不够的。

在正常道路运行过程中,交通路网是有车辆流通容量限制的,在突发灾害事件情景下,路网容量的限制对车辆行驶时间的影响更是凸显无疑,例如:地震发生后,路面遭到破坏,交通路网的实际通行能力将受到限制,如果再以交通路网的设计通行能力对车辆行驶时间进行估计,将增大时间估计量的误差,会对应急救援产生巨大影响。

本文为对救援时间进行较为准确的估计,将不再采用通常的时间计算方法,即:时间=距离/速度,而是引进 BPR 路阻函数,路阻函数用于描述车辆在道路上的行程费用(或时间)与道路交通条件之间的关系,能反映道路网络各组成部分的交通容量限制和拥挤效应,是交通量分配预测中的一项十分关键的技术,也是实施交通量分配的前提条件^[13-16]。将路阻函数引用于车辆行驶时间的预测,提高了模型与突发事件发生时实际情况的切合度,符合本文研究的需求。路阻函数公式如下式(1)所示:

$$T_{ij} = t_{ij}[1 + \beta(\frac{y_{ij}}{C_{ij}})^n] \quad (1)$$

其中, T_{ij} 表示路段(*i, j*)的运行时间; t_{ij} 表示该路段的自由流行程时间; C_{ij} 表示实际通行能力; y_{ij} 表示路段(*i, j*)的实际交通流量; β, n 为给定参数,虽然 BPR 函数在国内研究起步较晚,但其具有一定的代表性,为不失一般性,本文中给定参数选择 $\beta=0.15, n=4$,不作调整^[15]。距离最短作为应急救援的代表性目标,将其作为第二个目标函数与时间预测函数一起建立应急救援车辆调度模型,通常距离的求解为待救援点与待救援点的直线距离,即欧氏距离,而本文考虑到道路与道路通常以“井”字型交叉,所以采用绝对值距离,即 (x_1, y_1) 、 (x_2, y_2) 表示两个待救援点,两待救援点之间的距离为: $d_{12}=|x_1-x_2|+|y_1-y_2|$ 。

1.2 应急救援车辆调度模型

定义如下变量:

$$\text{sign}(n_k) = \begin{cases} 1, & n_k \geq 1 \\ 0, & \text{其他} \end{cases};$$

$$x_{ij}^k = \begin{cases} 1, & \text{车辆}k\text{从}i\text{点到}j\text{点} \\ 0, & \text{其他} \end{cases};$$

$$y_i^k = \begin{cases} 1, & \text{点}i\text{的需求由车辆}k\text{满足} \\ 0, & \text{其他} \end{cases}$$

目标函数:

$$f_1 = \min \sum_i^n \sum_j^n \sum_k^K d_{ij} x_{ij}^k \quad (2)$$

$$f_2 = \min \sum_i^n \sum_j^n \sum_k^K T_{ij} x_{ij}^k \quad (3)$$

约束条件:

$$\sum_{i=1}^{n_k} q_i * y_i^k \leq Q_k, k \in \{1, 2, \dots, K\} \quad (4)$$

$$\sum_{k=1}^K y_i^k \geq 1, i \in \{1, 2, \dots, n\} \quad (5)$$

$$\sum_{i=1}^n x_{ij}^k = y_j^k, j \in \{1, 2, \dots, n\}, k \in \{1, 2, \dots, K\} \quad (6)$$

$$\sum_{j=1}^n x_{ij}^k = y_i^k, i \in \{1, 2, \dots, n\}, k \in \{1, 2, \dots, K\} \quad (7)$$

$$\sum_{i=1}^{n_k} D_{k_j k_{j+1}} + D_{k_0 k_n} \text{sign}(n_k) \leq D \quad (8)$$

其中, K 表示配送车辆数; Q_k 表示第 k 辆车的载重量 ($k=1, 2, \dots, K$); n 表示待救点总数; q_i 表示第 i 个待救点需求量; d_{ij} 表示第 i 个待救点与第 j 个待救点之间的距离; D 表示车辆行驶上限; (x_1, y_1) 表示表示救援中心位置; (x_i, y_i) 表示表示待救点位置; n_k 表示第 k 辆车服务的待救点数; k_j 表示第 k 辆车历经的第 j 个待救点所对应路线中待救点集合中的顺序。

式(2)、(3)表示目标函数为总路程最短和总行驶时间最短; 式(4)表示每辆车在救援过程中都不超出其最大容量限制; 式(5)表示每个待救援点都有一辆车进行配送; 式(6)、(7)表示每个待救援点有且只有一辆车进行配送; 式(8)表示每辆车都不能超过其最大运输距离限制。

2 改进萤火虫算法求解

2.1 标准萤火虫算法

2.1.1 萤火虫算法原理

萤火虫算法^[17-19](glowworm swarm optimization algorithm)最早由两位印度学者 Krishnanand 和 Ghose 于 2005 年提出的一种新型群智能仿生算法。自然界中, 萤火虫通过发光吸引同伴求偶或进行觅食行为, 萤火虫的发光现象是因为萤火虫自携带一种叫荧光素的物质, 并且携带的荧光素越多, 发光越亮, 萤火虫个体的吸力越强, 越容易使周围的萤火虫向其靠拢。基于此, 我们可以理解为: 萤火虫发光越亮, 其所处的位置就越

好, 在此位置感知周围的能力越强, 即感知范围越大, 置差的萤火虫就会向位置好的萤火虫靠拢。但是, 在感知范围内, 萤火虫个体的集合是有一定数量限制的, 以保证有适当的邻居数。萤火虫算法就是基于这种思想提出的。

2.1.1 算法描述

在利用 GSO 算法求解问题时, 萤火虫个体随机分布在搜索空间 V 中, 设每个萤火虫个体携带相同数量的荧光素 l_0 , 迭代过程中, 第 i 只萤火虫在第 t 次迭代时位置用 $x_i(t)$ 表示, 萤火虫的荧光素值由适应度函数值求得, 用 $f(x_i(t))$ 表示适应度函数, 并且在每次迭代时都伴随着萤火虫位置的更新, 位置更新取决于个体的荧光素差值。具体萤火虫算法步骤如下:

1) 第 i 只萤火虫在 t 迭代次数时所处位置 $x_i(t)$ 对应的适应度函数值 $f(x_i(t))$ 转化为荧光素浓度值 $l_i(t)$ 。

$$l_i(t) = (1 - \rho)l_i(t-1) + \gamma f(x_i(t)) \quad (9)$$

其中, ρ 表示荧光素挥发因子, γ 表示荧光素增强因子, $\rho, \gamma \in [0, 1]$ 。

2) 在 t 迭代次数时第 i 只萤火虫的动态决策半径 $r'_d(t)$ 内, 选择荧光素浓度值大于 i 的个体组成其邻域集合 $N_i(t)$ 。

$$N_i(t) = \{j : \|x_j(t) - x_i(t)\| < r'_d(t) : l_j(t) > l_i(t)\} \quad (10)$$

其中: $\|x_i(t)\|$ 表示 x 的范数, 即距离。

3) 计算 t 迭代次数时第 i 只萤火虫向其邻域中个体 j 的移动概率 $p_{ij}(t)$ 。

$$p_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} (l_k(t) - l_i(t))} \quad (11)$$

4) 对移动后的第 i 只萤火虫位置进行更新。

$$x_i(t+1) = x_i(t) + s * \left[\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \right] \quad (12)$$

其中, s 表示移动步长。

5) 对动态决策域半径进行更新。

$$r'_d(t+1) = \min\{r_s, \max\{0, r'_d(t) + \beta(n_t - |N_i(t)|)\}\} \quad (13)$$

其中: r_s 表示萤火虫个体的最大感知半径; n_t 表示个体邻域集合内种群数量的阈值, β 表示动态决策更新率。

2.2 改进的萤火虫算法

2.2.1 编码与解码

(1) 编码

萤火虫算法在求解连续型问题中已得到应用, 由于路径优化问题是组合优化问题, 在编码的过程中必

须满足对应路径编码的唯一性和必须满足不同的路径组合. 所以, 本文利用文献[17]提出的编码规则进行编码, 其优势在于将萤火虫算法应用于离散的问题求解中, 利用待救点直接排列的方法, 将所有待救点用整数 $1 \sim n$ 编号构成全排列形成一个序列 $(0, x_1, x_2, \dots, x_n)$, 其中 0 表示救援中心, x_i 表示第 i 个待救点被访问的序号. 例如编码, 024135, 2 表示编号为 1 的待救点在第二个被救援, 所以救援顺序为 031425.

(2) 解码与二次解码

利用文献[17]对路径解码, 在利用文献中提出的方法时, 解码得到的路径只能显示待救援点被访问的顺序, 无法看出有几辆车进行救援行动和每辆救援车辆访问待救援点的数量和承担的救援任务. 这里, 本文利用救援车辆最远运输距离限制和最大载重量限制对每一可行编码进行配送中心插入操作, 此为编码的二次解码过程, 具体操作如下:

假设救援过程中所使用的车辆是统一的, 每台车的最远运输距离 (D_k) 和最大载重量 (Q_k) , 即 $D_k=D$, $Q_k=Q$. 将一次解码得到的可行编码记为 $road$, 可行编码代表的可行路径中待救援点间的距离用 D_road 表示, 每个待救援点的救援物资需求量用 Q_road 表示, 按照可行路径中待救援点的顺序将每个待救援点间的距离和救援物资需求量累计向下求和, 得到 D_road_l 和 Q_road_l , 其中 $D_road_l(i)$ 和 $Q_road_l(j)$ 表示第 i, j 个点的相关指标. 执行以下操作:

1) 计算 D_road_l 和 Q_road_l , 将 D 与 D_road_l 逐个比较, Q 与 Q_road_l 逐个比较, 若比较过程中发现 $D_road_l(i-1) < D$ 且 $D_road_l(i) > D$, 记 $i_1=i$, 同时找到 $Q_road_l(j-1) < Q$ 且 $Q_road_l(j) > Q$, 记 $j_1=j$, 比较 i_1 与 j_1 , 取 $z=\min\{i_1, j_1\}$, 在 z 位置插入 0 作为救援中心, 更新 $road, D_road, Q_road_l$.

2) 对 D_road_l 和 Q_road_l 归零操作, 并并进行累计向下求和计算, 每遇到编码中的 0, 就将 D_road_l 和 Q_road_l 归零并在此处再次重新进行累计向下求和计算.

3) 重复以上两步操作直至遍及所有待救点, 最终得到的编码中, 0 的个数即为救援车辆的台数.

2.2.2 产生初始最优解

采用邻域搜索算法产生初始解. 即: 第一步搜索距离救援中心最近的待救援点作为第一个被访问的点;

第二步以第一个被访问的点作为搜索起始点, 搜索其邻域空间内距离最近的点作为第二个被访问的点, 以救援车辆的最大载重量作为约束条件, 满足车辆载重限制时返回救援中心, 进行第二辆车的搜索, 最终得到的即为初始最优解.

2.2.3 初始化种群

算法进行开始后, 需要对搜索空间内的萤火虫进行初始化, 本文采用的方法是对待救援点进行随机排列的方法.

2.2.4 萤火虫个体间间距的确定

根据萤火虫算法的基本原理, 在算法运行过程中, 位置差的萤火虫需要向位置好的萤火虫移动, 此时, 需要确定两萤火虫间的间距. 由于本文所求的问题不是连续型函数问题, 萤火虫个体的位置以编码形式确定的, 无法直接计算出两萤火虫间的间距. 本文利用编码差异度来代替距离, 具体计算如下:

设个体 i, j 在 t 迭代过程中的编码为 $x_i(t)=[0, x_{i1}, x_{i2}, \dots, x_{im}]$, $x_j(t)=[0, x_{j1}, x_{j2}, \dots, x_{jm}]$, 用 $\sigma_{ij}(t)$ 表示编码差异度, 则:

$$\sigma_{ij}(t) = \frac{\sum_{i=1}^n |d_{ij}(t, m)|}{M} \quad (14)$$

$$D_{ij}(t) = c\sigma_{ij}(t) \quad (15)$$

其中, $|d_{ij}(t, m)|$ 表示编码对应位置的绝对值差值.

M 表示 $|d_{ij}(t, m)|$ 可能取得的最大值, c 为常数, $c=20$.

$$M = \begin{cases} \frac{(n-1)(n+1)}{2} & n \text{ 为奇数} \\ \frac{(n-1)^2}{2} & n \text{ 为偶数} \end{cases}$$

2.2.5 萤火虫位置的更新

一般的萤火虫算法(GSO)利用固定步长 s 进行位置更新只适用于求解连续函数问题, 所以本文利用轮盘赌法的思想对位置更新公式作如下改变:

萤火虫个体 i 向萤火虫 j 移动:

$$x_i(t+1, m) = \begin{cases} x_i(t, m) & r_m < p_1 \\ x_j(t, m) & p_1 \leq r_m < p_2 \\ \text{randscr}(1, 1, [1, n]) & r_m \geq p_2 \end{cases} \quad (16)$$

其中, $m \in \{1, 2, \dots, n\}$, r 表示 $\{0, 1\}$ 之间的随机数 $r_0=0$, $r_s=\{r_0, r_1, \dots, r_n\}$, p_1, p_2 为更新参数, $p_1, p_2 \in [0, 1]$.

此时, 萤火虫个体 i 在 t 迭代次数更新编码时, 以概率 p_1 保留第 m 维上的数字, 以概率 p_2-p_1 将个体

j 上第 m 为数字转移到个体 i 上, 以概率 $1-p_2$ 将第 m 维编码转变为 $[1, n]$ 之间的随机数.

2.2.6 不可行编码的处理

若个体 i 在 t 迭代过程中的编码 $x_i(t)=[0, x_{i1}, x_{i2}, \dots, x_{in}]$ 为不可行编码, 则保留前两维编码 $0, x_{i1}$, 以 x_{i1} 为第一个被访问的点并进行最近邻域搜索, 以此方法将不可行编码改成可行编码.

3 算例

3.1 算例背景

现假设某地区发生地震, 政府组织了一个应急救援中心囤放应急救援物资, 有 20 个应急救援待救点需要救援物资(救援中心和待救援点依据某地区交通道路网络图随机选取), 配送中心和待救点分布在边长为 20 km 的正方形区域内, 救援中心有 5 辆配送车辆可以利用. 现截取待救援点与救援中心所在范围内的交通道路网络图, 建立合适坐标系, 随机产生待救点坐标和需求量如表 1 所示, 现利用新建立的模型一合理安排救援物资车辆运输路径, 使救援车辆最快到达应急待救点. 在该问题中设定每辆救援车辆的最大载量是 8 t, 车辆一次配送最大行程是 60 km. 随机产生救援中心坐标(10.0 km, 10.0 km).

表 1 待救援点坐标及需求量

需求点编号	横坐标 x (km)	纵坐标 y (km)	需求量 q (t)
1	8.8	15.0	1.6
2	7.6	5.1	1.8
3	15.3	10.1	0.7
4	15.9	14.0	1.6
5	3.7	17.8	1.3
6	9.8	19.2	0.2
7	8.9	10.9	0.6
8	12.9	2.8	1.1
9	14.2	3.0	1.9
10	15.1	5.2	0.7
11	5.5	16.8	0.3
12	13.6	5.1	1.9
13	13.1	16.3	1.2
14	3.3	4.9	1.0
15	2.4	18.6	1.6
16	10.0	7.0	0.5
17	19.2	3.9	0.8
18	6.8	5.0	1.8
19	11.7	12.3	1.6
20	4.5	9.5	1.7

3.2 路段间车辆行驶时间

在案例的特殊背景下, 随机产生交通路网实际容量如表 2 所示, 假设交通路段设计流量和实际通行能力均为 1000 辆/h, 路段 (i, j) 的自由流行程时间 t_{ij} 由路段 (i, j) 的距离与车辆在该路段上的速度之商得到, 即 $t_{ij}=d_{ij}/v$, 车辆的自由流行驶速度在这里统一用 $v=40$ km/h 表示.

表 2 交通路网实际交通流量

序号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	0
1	0	36	77	699	92	602	8	123	58	81	21	262	52	779	49	454	37	128	38	179	78
2	91	0	80	891	29	263	40	184	6	929	30	603	23	715	17	432	20	999	58	423	42
3	13	934	0	959	76	654	26	240	23	776	47	711	49	904	98	825	49	171	25	94	9
4	91	679	49	0	75	689	80	417	35	487	23	222	62	891	71	84	34	33	29	599	27
5	63	758	45	139	0	748	43	50	82	436	84	117	68	334	50	133	95	561	62	471	15
6	10	743	65	149	57	0	91	903	2	447	19	297	40	699	47	173	92	882	27	696	28
7	28	392	71	258	8	84	0	945	4	306	23	319	37	198	6	391	5	669	82	700	44
8	55	656	75	841	5	229	26	0	17	509	17	424	99	31	68	831	74	190	98	639	53
9	96	171	28	254	53	913	15	489	0	511	23	508	4	744	4	803	27	369	73	34	46
10	96	706	68	814	78	152	14	338	73	0	44	86	89	500	7	61	42	461	34	69	88
11	16	32	66	244	93	826	87	900	65	795	0	263	91	480	52	399	55	982	58	320	52
12	97	277	16	929	13	538	58	369	45	644	92	0	80	905	10	527	94	156	11	531	94
13	96	46	12	350	57	996	55	111	55	379	43	29	0	610	82	417	42	856	91	654	64
14	49	97	50	197	47	78	15	780	30	812	18	929	26	0	82	657	98	645	88	408	96
15	80	824	96	251	1	443	85	390	74	533	90	730	34	859	0	628	30	376	82	820	24
16	14	695	34	616	34	107	62	242	19	351	98	489	68	806	15	0	70	191	26	718	68
17	42	317	59	473	16	962	35	404	69	939	44	579	14	577	66	432	0	428	59	969	29
18	92	950	22	352	79	5	51	97	18	876	11	237	72	183	52	16	54	0	2	531	67
19	79	34	75	831	31	775	40	132	37	550	26	459	11	240	97	984	70	121	0	325	70
20	96	439	26	585	53	817	8	942	63	623	41	963	65	887	65	167	67	590	31	0	7
0	66	382	51	550	17	869	24	956	78	587	59	547	49	29	80	106	18	226	16	611	0

3.3 算法求解步骤

1) 在萤火虫算法运行之前,对萤火虫的参数进行初始化.设置萤火虫种群个数 a ,最大迭代次数 b ,初始荧光素 $I_i(0)$,初始决策半径 $r'_d(0)$,最大感知半径 r_s ,荧光素挥发因子 ρ ,荧光素增强因子 γ ,动态决策更新率 β ,个体邻域集合内种群数量的阈值 n_i ,求出初始最优解和初始化萤火虫群,设置更新参数 p_1, p_2 ;

2) 解码和二次解码,以路径的距离的倒数为适应度函数,计算适应度函数值 $f(x_i(t))$,根据式(9)将适应度函数值转化为荧光素值 $I_i(t)$;

3) 根据式(14)、(15)计算个体间距离,求出邻域集合 $N_i(t)$;

4) 利用式(11)计算移动概率,根据轮盘赌法选择决定移动的个体;

5) 利用式(16)更新编码;

6) 对不可行编码进行处理;

7) 利用式(13)更新萤火虫个体的动态决策半径;

8) 判断是否达到迭代次数,若是,则输出结果,若不是,则返回第二步继续进行算法.

3.4 求解结果

在 matlab 运行环境下,输入初始参数:萤火虫种群个数 $a=60$,最大迭代次数 $b=200$,初始荧光素 $I_i(0)=5$,初始决策半径 $r'_d(0)=4$,最大感知半径 $r_s=20$,荧光素挥发因子 $\rho=0.4$,荧光素增强因子 $\gamma=0.6$,动态决策更新率 $\beta=0.08$,个体邻域集合内种群数量的阈值 $n_i=5$,更新参数 $p_1=0.7, p_2=0.9$.对算例求解 10 次,对路径最短的三组方案进行时间比较,求解结果如表 3 所示.

表 3 算例求解结果

方案	距离(km)	时间(min)
1 a: 0-3-10-12-14-18-2-0; b: 0-1-6-15-5-11-20-0; c: 0-9-17-4-13-19-7-0; d: 0-16-8-0;	140.4	1609.6
2 a: 0-2-12-3-4-7-0; b: 0-1-11-5-15-6-13-19-0; c: 0-8-9-17-10-18-14-16-0; d: 0-20-0;	135.2	1345.7
3 a: 0-1-11-15-5-13-19-0; b: 0-20-14-10-17-9-8-16-0; c: 0-3-4-12-18-2-0; d: 0-7-6-0;	142.2	1667.7

通过表 3,首先取时间最短,我们得到方案 2: a: 0-3-10-12-14-18-2-0; b: 0-1-6-15-5-11-20-0; c: 0-9-17-4-

13-19-7-0; d: 0-16-8-0.最短时间接受范围是[1108.3, 1413.0](由 $[T, T(1+5\%)]$ 计算得出),在时间接受方案内没有满意解,所以这里选择方案 2,即: a: a: 0-3-10-12-14-18-2-0; b: 0-1-6-15-5-11-20-0; c: 0-9-17-4-13-19-7-0; d: 0-16-8-0 作为应急救援车辆的行驶路径.其优化过程和救援路径图如图 1、图 2 所示.

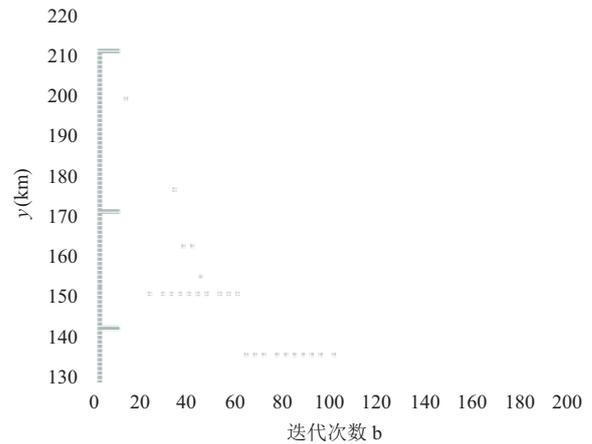


图 1 萤火虫算法优化过程

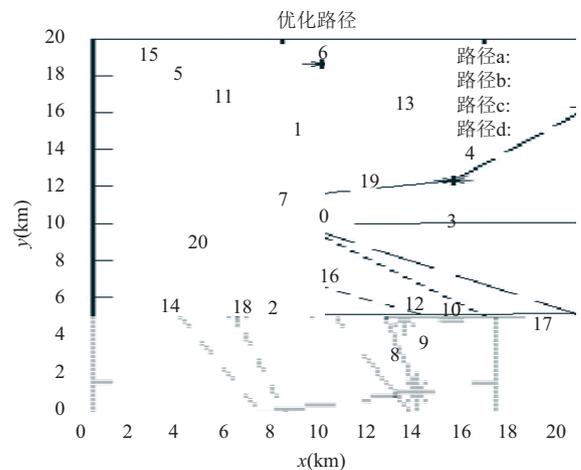


图 2 救援车辆行驶路径

3.5 结果对比

利用遗传算法对本文的算例进行求解,设置遗传算法的交叉概率为 0.9,变异概率为 0.1,求解问题 10 次选择最优值,结果与本文算法求解的结果进行对比,得到下列对比表格如表 4 和遗传算法寻优过程如图 3 所示.

由表 4 的结果对比可以看到,萤火虫算法在求解应急救援车辆路径优化问题的过程中,改变算法的编

码和解码规则,使之适用于求解离散问题,利用本文改进的萤火虫算法,对本文模型进行求解,得到的最终结果要优于基本遗传算法求解的结果并且在求解过程中算法更加稳定。

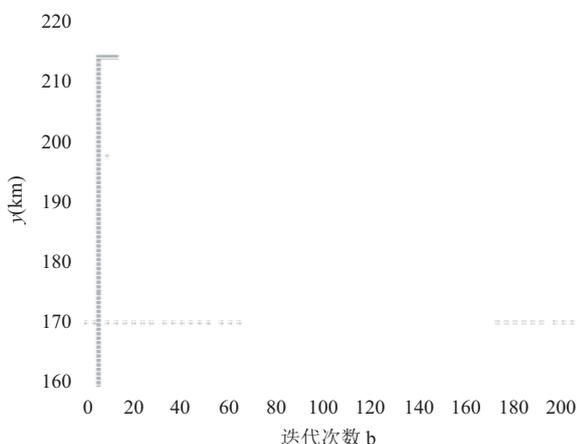


图3 遗传算法寻优过程

表4

	车辆行驶方案	距离(km)	时间(min)
遗传算法	a: 0-1-11-6-16-8-14-15-5-0; b: 0-20-18-2-12-10-0; c: 0-13-4-3-17-9-19-0; d: 0-7-0;	142.4	1363.8
萤火虫算法	a: 0-3-10-12-14-18-2-0; b: 0-1-6-15-5-11-20-0; c: 0-9-17-4-13-19-7-0; d: 0-16-8-0;	130.2	1136.8

4 结论

本文研究了路网容量对应急救援时间的影响,提出救援车辆行驶时间最短和行驶路径最短双目标应急救援车辆调度优化模型,并设计离散的萤火虫算法。最后通过设计算例,对模型求解,对比验证了本文算法在求解过程中要优于一般的遗传算法,并得到应急救援车辆行驶路径的满意解。

参考文献

- Najafi M, Eshghi K, Dullaert W. A multi-objective robust optimization model for logistics planning in the earthquake response phase. *Transportation Research Part E: Logistics and Transportation Review*, 2013, 49(1): 217–249. [doi: 10.1016/j.tre.2012.09.001]
- Nikolakopoulou G, Kortesis S, Synefaki A, *et al.* Solving a vehicle routing problem by balancing the vehicles time utilization. *European Journal of Operational Research*, 2004,

- 152(2): 520–527. [doi: 10.1016/S0377-2217(03)00042-0]
- 3 Wohlgenuth S, Oloruntoba R, Clausen U. Dynamic vehicle routing with anticipation in disaster relief. *Socio-Economic Planning Sciences*, 2012, 46(4): 261–271. [doi: 10.1016/j.seps.2012.06.001]
- 4 Wang HJ, Du LJ, Ma SH. Multi-objective open location-routing model with split delivery for optimized relief distribution in post-earthquake. *Transportation Research Part E: Logistics and Transportation Review*, 2014, 69: 160–179. [doi: 10.1016/j.tre.2014.06.006]
- 5 谈晓勇, 林鹰. 基于混沌蚁群算法的应急救援车辆调度优化. *计算机应用研究*, 2014, 31(9): 2640–2643.
- 6 谢秉磊, 胡小明, 张一喆. 需求可分的车辆路径问题模型与算法. *运筹与管理*, 2012, 21(3): 72–76.
- 7 孙丽君, 胡祥培, 王征. 车辆路径规划问题及其求解方法研究进展. *系统工程*, 2006, 24(11): 31–37. [doi: 10.3969/j.issn.1001-4098.2006.11.006]
- 8 Azi N, Gendreau M, Potvin JY. An exact algorithm for a single-vehicle routing problem with time windows and multiple routes. *European Journal of Operational Research*, 2007, 178(3): 755–766. [doi: 10.1016/j.ejor.2006.02.019]
- 9 Ma H, Cheang B, Lim A, *et al.* An investigation into the vehicle routing problem with time windows and link capacity constraints. *Omega*, 2012, 40(3): 336–347. [doi: 10.1016/j.omega.2011.08.003]
- 10 王晓博, 李一军. 多车场多车型装卸混合车辆路径问题研究. *控制与决策*, 2009, 24(12): 1769–1774. [doi: 10.3321/j.issn:1001-0920.2009.12.002]
- 11 陈建军. 蚁群算法在物流配送路径优化中的研究. *计算机仿真*, 2011, 28(2): 268–271.
- 12 王飞. 带时间窗车辆调度问题的改进粒子群算法. *计算机工程与应用*, 2014, 50(6): 226–229.
- 13 王锐. 基于智能交通的路阻函数的改进研究. *计算机光盘软件与应用*, 2012, (21): 75–76.
- 14 刘宁, 赵胜川, 何南. 基于 BPR 函数的路阻函数研究. *武汉理工大学学报(交通科学与工程版)*, 2013, 37(3): 545–548.
- 15 周彪, 智路平, 李彬. 改进 BPR 路阻函数及其在 EMME 中的应用. *上海海事大学学报*, 2013, 34(4): 67–70, 90.
- 16 边霞, 米良. 遗传算法理论及其应用研究进展. *计算机应用研究*, 2010, 27(7): 2425–2429, 2434.
- 17 周永权, 黄正新, 刘洪霞. 求解 TSP 问题的离散型萤火虫群优化算法. *电子学报*, 2012, 40(6): 1164–1170.
- 18 周永权, 黄正新. 求解 TSP 的人工萤火虫群优化算法. *控制与决策*, 2012, 27(12): 1816–1821.
- 19 李雪竹. 基于免疫萤火虫算法的 RFID 仓储车辆动态调度. *计算机工程与应用*, 2014, 50(6): 235–239.