

doi: 10.3969/j.issn.1002-0268.2024.12.006

# 基于改进樽海鞘算法的高速公路应急资源调度

于涵诚, 王长华, 倪双静\*, 朱熙豪, 刘海萍  
(浙江省机电设计研究院有限公司, 浙江 杭州 310051)

**摘要:**为解决当前高速公路救援不及时、救援效果差及容易引发二次事故等问题,结合高速公路环境下应急资源调度的实际需求,充分考虑突发事件及其对潜在二次事故的影响,研究了一种基于改进樽海鞘算法的高速公路应急救援资源调度方法,以高效应对高速公路的紧急情况。首先,从技术角度深入研究分析最优路径选择方法和救援物资调度分配方法,将时间和成本作为高速公路应急救援调度的目标,并建立调度目标函数。然后,针对高速公路应急救援的最佳路径选择方法,研究蚁群最短路径算法,并结合高速公路自身的路网特征对蚁群算法中的全局更新规则进行改进,提高算法的路径搜索效率。随后,为求解潜在事故下的应急资源调度问题,研究樽海鞘算法,利用该算法计算量小、具有全局探索等优势对资源调度目标函数进行求解。同时,结合烟花算法中的爆炸、变异等操作对樽海鞘算法进行改进,进一步避免算法陷入局部最优的情况。最后,通过试验结果表明,所提出的调度方法能提供更快的路径规划,改进的樽海鞘算法能有效提高收敛速度和计算精度,而且相比不考虑潜在事故的调度,一旦发生二次事故,该方法能有效降低应急救援的资源调度总成本。

**关键词:**智能交通; 应急资源调度; 樽海鞘算法; 高速公路; 烟花算法; 蚁群算法

中图分类号: U491

文献标识码: A

文章编号: 1002-0268 (2024) 12-0048-10

## Expressway Emergency Resource Scheduling Based on Improved Salp Swarm Algorithm

YU Han-cheng, WANG Chang-hua, NI Shuang-jing\*, ZHU Xi-hao, LIU Hai-ping  
(Zhejiang Institute of Mechanical & Electrical Engineering Co., Ltd., Hangzhou, Zhejiang 310051, China)

**Abstract:** To solve the current issues of untimely expressway rescue, poor rescue effectiveness and secondary accidents easily caused on expressways, combining with the actual needs for emergency resource scheduling in the expressway environments, the emergencies and their potential influences on secondary accidents were fully considered. The expressway emergency resource scheduling method based on the improved salp swarm algorithm was studied to efficiently cope with the emergency situations on expressway. First, the optimal path selection method and rescue material scheduling method were in-depth studied and analyzed from a technical perspective. Time and cost were taken as the goals of expressway emergency resource scheduling. The scheduling objective functions were established. Then, in accordance with the optimal path selection method for expressway emergency rescue, the ant colony shortest path algorithm was studied. The global update rules for ant colony algorithm were improved based on the expressway network characteristics to enhance the algorithm path search efficiency. Subsequently, to solve the emergency resource scheduling issues related to potential accidents, the salp swarm algorithm was studied. The algorithm's advantages of low computational complexity and global exploration were utilized to solve the resource scheduling objective function. Simultaneously, the salp swarm algorithm was improved with explosion, mutation, and other operations in the

收稿日期: 2022-04-06 修改日期: 2024-04-01

基金项目: 2022年度“尖兵”“领雁”研发攻关计算项目(2022C01042)

作者简介: 于涵诚(1981-),男,浙江杭州人,硕士,高级工程师。(634299715@qq.com)

\*通讯作者: 倪双静(1988-),女,浙江杭州人,硕士,高级工程师。(335213925@qq.com)

fireworks algorithm to further avoid the algorithm falling into local optima. Finally, the test results show that the proposed scheduling method can provide faster path planning. The improved salp swarm algorithm can effectively improve the convergence speed and computational precision. Moreover, compared with the scheduling without considering potential accidents, the proposed method can effectively reduce the total cost of resource scheduling for emergency rescue in the event of a secondary accident.

**Key words:** intelligent transport; emergency resource scheduling; salp swarm algorithm; expressway; fireworks algorithm; ant colony algorithm

## 0 引言

近年来,中国高速公路里程数和机动车保有量不断增加,导致高速公路交通事故时常发生。高速公路因车速快、全线封闭、相对外界道路隔绝等特点,导致一旦发生事故将带来严重后果。因此迫切需要建立一个完备的交通事故紧急救援体系,从而保障道路畅通、生命财产安全。当前,中国应对高速公路突发事件时,一般由当地政府和应急部门对事故进行指挥救援。救援过程以人为方式为主,先根据周边救援物资的分布情况对物资进行分配和调度,然后再安排车辆进行运送。随后车辆驾驶员通常仅凭经验和导航对运送路线进行规划,而没有考虑当前事故可能对交通带来的影响,因此容易耽误救援并引发二次交通事故,加重事故影响程度并扩大影响范围。当高速公路突发交通事故时,如果能实现资源共享,运用合理的应急救援调度方法,将有效提高资源利用率,加快响应速度,实现最佳救援效果。这对减少高速公路因交通事故带来的生命财产损失、提高高速公路应急管理水平等具有重要现实意义。

目前关于应急救援调度的研究成果颇多,主要集中在时间和成本 2 个指标上<sup>[1]</sup>。应急救援调度的主要任务首先是确定救援点需要的物资情况,可根据事故具体情况对救援物资的种类和数量做基本的范围确定;然后在资源共享的前提下,根据周边可提供的物资分布位置和数量,选择参与救援的施救点及其需要运送的物资种类和数量;最后规划运送的最佳路径。因此,应急救援问题可以转化为在已知可选择的救援点及其救援物资情况下<sup>[2]</sup>,需要选择合适的救援点及其提供的救援物资种类和数量,通过最佳路径到达事故现场,使得救援时间和成本最低。从技术角度来说,应急救援调度问题主要从最优路径选择和救援物资调度分配 2 个方面进行研究。

高速公路应急救援的最优路径选择主要考虑时间因素,以最短时间为目标。目前国内外对应急救援最优路径选择算法的研究颇多。王薇等<sup>[1]</sup>通过贪

婪算法求解出救援点到达目的位置的最优路径,并根据施工区环境选择合适的调度方案,但是该算法考虑的因素不够完善。Potvin 等<sup>[3]</sup>考虑动态时间问题,提出一种带时间窗的路径规划方法,并通过设置单个容差参数,定义和比较不同的无功调度策略,但是该方法也同时增加了算法的复杂度。Nikoo 等<sup>[4]</sup>通过建立多个影响因素评价体系模型求解最佳路径,但并未考虑各因素之间的关系。郝正博等<sup>[5]</sup>在求解最短路径问题上,虽然充分考虑了车道数量、路段限速、路段长度、平均车速等多个因素,但是在现实中很多数据较难获取,而且在复杂路网环境下计算的误差也会较大,容易造成错误预估。

救援物资调度分配主要是指针对每种资源选择合适的救援点和物资数量<sup>[6]</sup>。根据救援点到各事故点的最佳路径,以时间和成本最小化为目的,建立调度模型,通过算法求解,获得最佳资源调度分配方案。孙秀巧等<sup>[7]</sup>在求解调度模型时,采用模拟退火算法和动态变异交叉方法,改进了遗传算法中容易陷入局部最优的问题,提高了算法的寻优结果事故响应时间,但是模拟退火算法需要更多的参数调整,针对实际问题调参较为困难。张奕等<sup>[8]</sup>将事故等级和类型作为参考,建立资源调度模型,并通过改进粒子群算法进行求解,但是该算法还是存在容易陷入局部最优的情况。Sung 等<sup>[9]</sup>针对资源分配中集合划分问题,采用列生成的方法来有效处理资源调度,相比固定优先级的资源分配更优,但是该方法也增加了资源分配的复杂度。万孟然等<sup>[10]</sup>将备灾和应急处理进行双层规划,设计了相应的选址路径优化模型,并通过改进的双层樽海鞘遗传算法模型进行求解,但是该模型未考虑人为因素。Wang 等<sup>[11]</sup>通过元胞自动机中引入辅助种群和邻域结构,开发了多目标元胞遗传算法对提出的多目标资源分配模型进行求解,然而随着目标增加,容易导致算法求解时间较长,不符合实际应用。邵泽军等<sup>[12]</sup>对粒子群算法进行全局和局部混合模式改进,用于多资源时间-成本调度模型的求解,由于 2 种模式混合

求解,自然增加了算法求解时间。

综上,在应急救援的路径选择和资源调度方面都有相当多的研究,但是针对高速公路环境下的研究较少,对二次事故影响的考虑也很少。因此,本研究从路径选择和资源调度2方面进行研究,提出一种针对高速公路环境下的应急救援资源调度方法,并通过Sioux Falls路网仿真试验验证了该方法能为高速公路应急资源调度提供更快的路径规划,有效降低应急救援的资源调度成本。

## 1 应急救援路径选择的研究

高速公路救援路径选择主要考虑避开拥挤路段到达事故点实施救援,必要时也可选择到达事故点的对称路段实施救援。而蚁群算法正好是一种智能仿生算法,具有很强的发现较好解的能力,且具有较强的鲁棒性。高速公路救援路径选择符合蚁群算法的路径选择原则,因此本研究将蚁群算法用于高速公路应急救援最佳路径选择。

### 1.1 蚁群最短路径算法

蚁群算法是一种启发式算法,其原理在于蚂蚁在寻找食物的过程中,会在其走过的路上留下一种分泌物,并吸引其他蚂蚁也沿着该路径行走<sup>[13]</sup>。这种分泌的物质就是信息素,且浓度越高,吸引力越大,最终使大部分的蚂蚁都会走上同一条最优路径。不难看出,蚂蚁能快速找到最短路径,主要是通过路径上不断积累的信息素。信息素越多,走这条路的蚂蚁就越多,从而使这条路上的信息素更多,这就是一种正反馈过程。而在算法开始时,各条路径上的信息素浓度是一样的,蚂蚁通过一次次路径选择,使轨迹浓度开始变化。算法通常采用反馈方式<sup>[14]</sup>,在每次迭代后计算最优路径并释放更多信息素,最终引导整个系统向最优解发展。

蚁群算法的关键主要有2个。第1个关键点是访问路径的选择。假设有 $m$ 只蚂蚁,每只蚂蚁从各自起点出发,从未访问过的 $N$ 个地点中按轮盘赌的方式选择一个地点前往。设蚂蚁 $k$ 在当前所在的位置在地点 $i$ ,下一可选地点集合为 $J_i^k$ ,则选择到地点 $j$ 的轮盘概率为:

$$P_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta}{\sum_{\mu \in J_i^k} \tau_{i\mu}^\alpha \eta_{i\mu}^\beta}, & j \in J_i^k \\ 0, & \text{其他} \end{cases}, \quad (1)$$

$$\eta_{ij} = \frac{1}{d_{ij}}, \quad (2)$$

式中, $\tau_{ij}(t)$ 为 $t$ 时刻地点 $i$ 到下一可选地点 $j$ 之间的信息素量; $i\mu$ 为蚂蚁从当前所在位置 $i$ 到下一可选地点 $\mu$ ;  $d_{ij}$ 为地点 $i$ 和 $j$ 之间的距离; $\eta_{ij}$ 为地点 $i$ 到地点 $j$ 的启发函数; $\alpha$ 和 $\beta$ 分别为调节信息素浓度和距离影响程度的启发因子。

由式(1)可见,到备选地的信息素越大,则选择的可能性越大,备选地点与当前位置距离越远,则选择的可能性越小。将备选地点放到轮盘上,根据概率比例得出所占比例,转动选出下一个地点,蚂蚁便前往该地点。然后进行下一轮,直到最后一个地点选出,完成一轮迭代。

第2个关键点是信息素的更新。残留信息素过多容易影响启发信息,为此需要在一次遍历后对其进行更新。新一轮迭代结束后,每只蚂蚁又在地点 $i$ 和 $j$ 之间留下了浓度为 $\Delta\tau_{ij}$ 的信息素。信息素释放机制模型主要有蚁周模型、蚁量模型和蚁密模型,其中蚁周模型较优<sup>[13]</sup>。因为蚁量模型和蚁密模型更新信息素时,选择的是局部信息素,而蚁周模型选择的是全局信息素,不宜陷入局部最优解。蚁周模型信息素的更新为:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \rho \sum_{k=1}^m \Delta\tau_{ij}^k(t), \quad (3)$$

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q_k}{L_k}, & \text{蚂蚁 } k \text{ 经过路径 } (i, j) \\ 0, & \text{其他} \end{cases}, \quad (4)$$

式中, $\rho$ 为介于 $(0, 1)$ 的一个小数,表示以前信息素保留的比例,其值越小,则信息素更新受当前轮迭代情况影响越大; $Q_k$ 为常数,表示蚂蚁 $k$ 在此次循环中留下的信息素总量; $L_k$ 为蚂蚁 $k$ 在此次循环中所走的总长度。

### 1.2 基于改进蚁群算法的最优路径选择

由于高速公路自身的路网特征,蚁群算法在求解最优路径的问题上存在一些不足,本研究对这些问题进行分析,并相应改进算法,得到适合高速公路应急救援的最优路径选择算法。

(1) 在基本蚁群算法的初始时刻,每条路径上的信息素都是相同的,使得蚂蚁在路径选择上有较大的盲目性,容易影响最佳路径搜索的效率。高速公路路网具有节点密度低,节点间路线距离长等特点,一旦方向选择错误,驶离事故点,则需要花费更多的时间返回。因此如果盲目搜索,反而导致效率低下。为此,可以对算法进行信息素的初始化。在确定了救援点和事故点后,救援路线的走向基本也确定了,如果是靠近事故点方向的路径,则将其

初始信息素量相对提高, 这样在搜索初期, 蚂蚁就会倾向于事故点方向的路径, 更快找到最优解。

(2) 高速公路上一旦出现交通事故, 将会影响该路段的车辆通行时间。事故持续时间越长, 该车道上甚至周边车道上可能会出现车辆排队现象, 车辆通过的时间也会越久。而基本蚁群算法并不考虑时间的动态问题, 因此也不能很好地模拟该场景下的最优路径选择。为此, 需要增加交通事件对路段通行时间的影响<sup>[5]</sup>。高速公路上资源调度以车辆通行总时间最短的行车路径作为最优路径, 所以假设从地点  $i$  到地点  $j$  的车辆通行时间为  $m_{ij}$ , 则蚂蚁  $k$  选择到地点  $j$  的轮盘概率为:

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha}{[m_{ij} + f_{ij}(u^k)]^\beta}, & j \in C_{\text{allowed}}^k, \\ 0, & \text{其他} \end{cases} \quad (5)$$

式中,  $f_{ij}$  为事故对地点  $i$  到地点  $j$  道路上车辆通行时间的增量函数; 参数  $u^k$  为事故持续时间, 即蚂蚁  $k$  到达地点  $i$  时所花费的时间, 如果事故对该方向路段不造成影响, 则  $f_{ij}(u^k) = 0$ ;  $C_{\text{allowed}}^k$  为蚂蚁  $k$  可选择能通行的地点集合。

(3) 蚁群算法通过正反馈更新信息素浓度, 引导整个系统向最优解的方向进化, 但是也正是因为这个特点, 使得算法容易出现局部最优的情况。为此, 对算法中信息素的全局更新规则进行改进。如式 (6) 所示, 每次迭代后, 将所有蚂蚁走过的路径总长度按从小到大排序, 取前  $n$  个蚂蚁的集合, 得到集合  $K$ :

$$\tau_{ij}(t+1) = \begin{cases} (1-\rho)\tau_{ij}(t) + \sum_{k \in K} \Delta\tau_{ij}^k(t), & k \in K \\ (1-\rho)\tau_{ij}(t), & \text{其他} \end{cases} \quad (6)$$

对旅行商问题 (Traveling Salesman Problem, TSP), 进行算法改进前后试验比较, 其中取集合  $K$  的个数  $n$  为 10, 迭代 500 次, 得到结果如图 1 所示。由图可见, 算法改进后虽然初始时搜索到的路径更长, 但是随着迭代计算, 在 12 次迭代后搜索到长度为 2 042 的较短路径, 随后在 120 次迭代后搜索到长度为 1 105 的更短路径, 而原算法在 46 次迭代后才能搜索到长度为 2 791 的较短路径, 随后在 135 次迭代后搜索到长度为 1 138 的更短路径。因此证明本研究改进算法能有效降低计算的迭代次数, 提高算法收敛速度。

## 2 应急救援资源调度的研究

### 2.1 潜在事故对高速公路应急救援资源调度的影响

高速公路一旦发生交通事故, 通常会引起二次

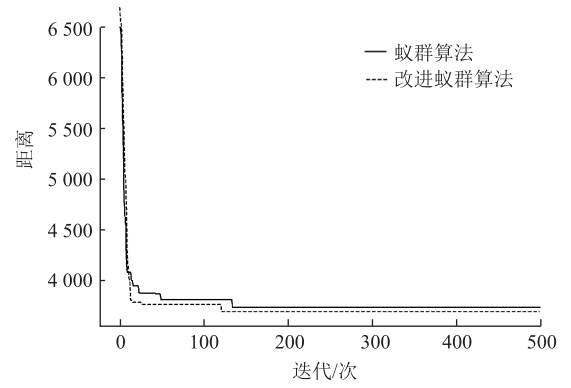


图 1 算法改进前后迭代次数比较

Fig. 1 Iteration comparison before and after algorithm improvement

事故的发生, 而该事故通常发生在相同或者相近路段上。需要提供救援物资的救援点一般都会与前一次事故的救援点重叠, 因此潜在事故的发生对制订救援资源的最优调度方案有一定的影响。同时, 根据潜在事故发生的严重程度, 调度方案需要做不同的调整。在潜在事故的影响下, 需要建立基于潜在事故的紧急救援资源调度模型, 该模型既要考虑传统调度模型中的确定性约束, 还需要考虑潜在事故对救援资源需求的不确定性约束, 解决在发生多起事故情况时, 为决策者提供多救援点的联合救援方案。

首先, 对参数进行假设: (1) 路网中救援点集合为  $L$ , 救援点  $i$  可选的救援车辆数为  $r_i$ ; (2) 当前事故点集合为  $F$ , 其中事故点  $f$  需要的救援车辆数为  $n_f$ ; (3) 潜在事故节点集合为  $H$ , 其中潜在事故点  $h$  的发生概率为  $P_h$ , 需要的救援车辆数为  $n_h$ ; (4) 救援点  $i$  到达当前事故点  $f$  的时间为  $T_{if}$ , 派出的救援车辆数为  $x_{if}$ ; (5) 救援点  $i$  到达潜在事故点  $h$  的时间为  $T_{ih}$ , 派出的救援车辆数为  $x_{ih}$ 。然后, 建立救援资源调度模型, 救援总成本  $Z$  为:

$$Z = \sum_{i \in L} \sum_{f \in F} T_{if} x_{if} + \sum_{i \in L} \sum_{h \in H} P_h T_{ih} x_{ih}, \quad (7)$$

式中, 前半部分表示当前所有事故点的应急救援总成本; 后半部分表示当前所有潜在事故点的应急救援总成本。假设当前事故和潜在事故发生的间隔较短, 不足以共同使用一辆救援车, 因此救援点  $i$  配置的救援资源既要满足当前事故  $f$  的救援需求, 又要满足潜在事故  $h$ , 需要符合约束为:

$$\sum_{f \in F} x_{if} + \sum_{h \in H} x_{ih} \leq r_i, \quad (8)$$

$$\sum_{i \in L} x_{if} \geq n_f, \quad \forall f \in F, \quad (9)$$

$$\sum_{i \in L} x_{ih} \geq n_h, \quad \forall h \in H. \quad (10)$$

## 2.2 樽海鞘算法

2017年, Mirjalili等<sup>[15]</sup>在研究樽海鞘的群体行为时,发现其行为很特别,经常由领头的樽海鞘引导方向,后面的樽海鞘跟随其后,依次首尾相连,像链条一样移动和觅食。根据这一发现,提出了一种较新颖的群体智能优化算法,也就是樽海鞘算法(Salp Swarm Algorithm, SSA)。该算法通过樽海鞘领导者的全局探索和追随者的局部探索,能有效避免陷入局部最优的情况。樽海鞘算法的关键在于3步操作。

(1) 初始化种群。在空间  $D \times N$  区域内,初始化领导者和追随者,其中  $D$  为空间维度,  $N$  为种群数量:

$$X_{D \times N} = \text{rand}(D, N) \times (b_u - b_l) + b_l, \quad (11)$$

式中,  $x_d^i \in X_{D \times N}$ , 追随者为  $X_d^i$ ,  $i=2, 3, 4, \dots, N$ ; 种群中领导者为  $X_d^1$ ;  $d=1, 2, 3, \dots, D$ ;  $b_u$  为搜索上界;  $b_l$  为搜索下界。

(2) 更新领导者位置。计算种群中每个个体位置的适应度,并进行排序,将首位适应度最优的樽海鞘位置确定为当前要搜寻的食物位置,然后计算领导者位置:

$$X_d^1 = \begin{cases} F_d + c_1[(b_u - b_l)c_2 + b_l], & c_3 \geq 0.5 \\ F_d - c_1[(b_u - b_l)c_2 + b_l], & c_3 < 0.5 \end{cases}, \quad (12)$$

$$c_1 = 2e^{-\left(\frac{4l}{L}\right)^2}, \quad (13)$$

式中,  $F_d$  为食物原位置向量的第  $d$  维;  $c_1$  为收敛因子;  $l$  为当前迭代次数;  $L$  为最大迭代次数; 控制参数  $c_2, c_3 \in [0, 1]$  用来增强  $X_d^1$  的随机性。

(3) 更新追随者位置。群体中剩余的  $N-1$  个樽海鞘视为追随者,并按照式(12)更新追随者的位置。然后根据新的群体位置重新计算适应度,重复步骤(2)和步骤(3)直到达到终止条件,输出当前食物位置,即模型的最优解:

$$X_d^{i'} = \frac{X_d^i + X_d^{i-1}}{2}, \quad i > 1, \quad (14)$$

式中  $X_d^{i'}$  为第  $d$  维中更新后的追随者位置。

## 2.3 基于改进樽海鞘算法的资源调度

在小型路网中,路径选择时可以使用穷举方法列出所有解的情况,再根据最优目标得到最佳路径。但是高速公路路网复杂,增加一条路段会使得路径选择情况呈指数增长,因此,穷举法并不适合解决本研究提出的问题。由于樽海鞘算法具有计算量小,且移动结构能有效避免陷入局部最优的特点,本研究借鉴该优化算法解决高速公路

路应急资源调度问题。虽然樽海鞘算法在很多方面有很好的效果,但是仍然存在一定的不足,容易出现收敛速度降低、计算精度差的问题<sup>[16]</sup>,而且到了后期,还是容易陷入局部最优的情况。为此,针对以上问题,本研究基于樽海鞘算法,借鉴烟花算法中的爆炸和变异操作,加大全局搜索的范围,跳出局部最优的陷阱。同时改进跟随者的位置变更方式,避免盲目追随。改进后的算法流程如图2所示。

### 2.3.1 领导者位置更新

由于樽海鞘算法中领导者的位置是根据最优适应度的位置来确定的,容易受初始种群的影响。如果一开始进入到次优的环境中,根据领导者更新位置公式,很难跳出局部最优的陷阱,为此,需要对领导者位置更新进行改进。本研究对烟花算法进行研究并借鉴该算法优点,应用于樽海鞘算法中。烟花算法是一种具有较强优化能力的算法<sup>[17]</sup>,每个烟花在爆炸之后选择最好的烟花作为下一次爆炸的烟花,而且在多个烟花爆炸的同时,每个烟花又都是相互独立的,只在本身爆炸范围内寻找最优爆炸烟花。烟花算法会根据适应度生成不同爆炸半径和数量的火花,并通过高斯变异跳出局部最优的环境,然后根据不同的火花计算适应度,可得到更优的解作为樽海鞘算法中新的食物位置。因此,在樽海鞘算法中对计算的  $N$  个适应度进行排序后,将排在前面一半的樽海鞘作为领导者。在每个领导者位置生成一个食物  $Fd_{i0}$ ,并将该位置作为烟花爆炸点,以烟花算法的爆炸半径  $R_i$  和产生的火花数量  $S_i$  分别得到多个食物  $\{Fd_{i1}, Fd_{i2}, \dots, Fd_{is_i}\}$ ,爆炸半径和产生的火花数量为:

$$R_i = \hat{R} \times \frac{f(Fd_{i0}) - f_{\text{best}} + \varepsilon}{\sum_{i=1}^{N/2} [f(Fd_{i0}) - f_{\text{worst}}] + \varepsilon}, \quad (15)$$

$$S_i = \hat{S} \times \frac{f_{\text{best}} - f(Fd_{i0}) + \varepsilon}{\sum_{i=1}^{N/2} [f_{\text{worst}} - f(Fd_{i0})] + \varepsilon}, \quad (16)$$

式中,  $f_{\text{worst}}$  和  $f_{\text{best}}$  分别为樽海鞘领导者位置生成的食物中适应度最差值和最优值,当前食物位置的适应度越好,则烟花半径越小,否则烟花半径越大,这样使得算法能更快找到最佳适应度的位置;  $\hat{R}$  和  $\hat{S}$  为分别用来调节爆炸半径和爆炸火花数目的常数;  $\varepsilon$  为一个极小的值。为了避免好的个体和差的个体产生过多或过少火花,对  $S_i$  进行处理:

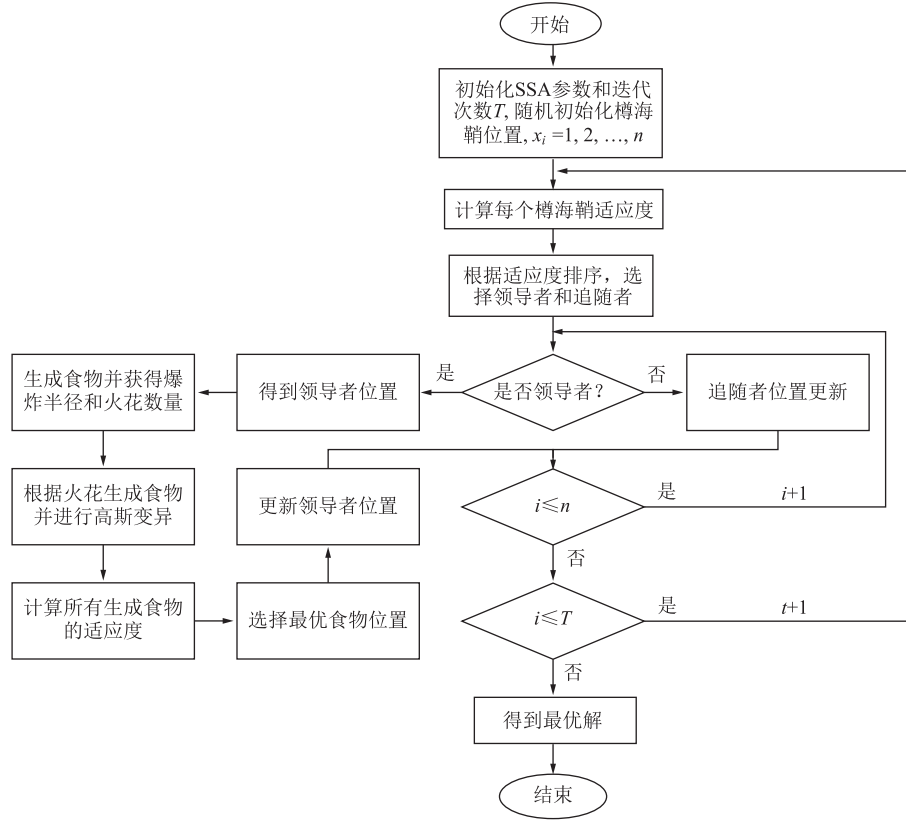


图 2 改进后的算法流程

Fig. 2 Improved algorithm flow

$$S_i = \begin{cases} \text{round}(a \times \hat{S}), & S_i < a \times \hat{S} \\ \text{round}(b \times \hat{S}), & S_i > b \times \hat{S} \\ \text{round}(S_i), & \text{否则} \end{cases} \quad (17)$$

式中  $a$  和  $b$  为常数。随后对爆炸产生的食物位置进行高斯变异, 以提高种群多样性<sup>[18]</sup>。同时当火花在某一维度上超过边界, 将通过映射规则映射到新的位置。变异公式为:

$$Fd_i = Fd_{\text{best}} + c \times (Fd_{\text{best}} - Fd_i), \quad (18)$$

$$c = \begin{cases} \text{randc}(0.5, 0.1), & \text{rand} < P \\ \text{randc}(1.2, 0.1), & \text{否则} \end{cases}, \quad (19)$$

$$P = \frac{t_1}{t_1 + t_2}, \quad (20)$$

式中,  $Fd_{\text{best}}$  为最好的食物位置;  $c$  为学习因子;  $\text{randc}(a, b)$  为柯西分布;  $P$  为限制因子;  $t_1$  为位置参数小的柯西分布次数;  $t_2$  为位置参数大的柯西分布次数。映射规则为:

$$Fd_i^k = Fd_{\text{ub}}^k + \text{rand}(0, 1) \times (Fd_{\text{ub}}^k - Fd_{\text{lb}}^k), \quad (21)$$

式中,  $Fd_i^k$  为第  $i$  个食物的第  $k$  维位置;  $Fd_{\text{ub}}^k$  和  $Fd_{\text{lb}}^k$  分别为食物第  $k$  维位置的上界和下界。随后对烟花爆炸方式生成的食物和  $Fd_{i0}$  计算适应度, 将适应度

最优的食物作为  $Fd_{\text{best}}$ , 而不是将原来适应度最优的樽海鞘作为食物  $Fd_{\text{best}}$ 。最后通过领导者位置变更公式得到新的位置  $X_d^1$ 。其中  $Fd_{\text{best}}$  的选择为:

$$Fd_{\text{best}} = \text{argmin} \{f(Fd_{i0}), f(Fd_{i1}), f(Fd_{i2}), \dots, f(Fd_{is_i})\}. \quad (22)$$

### 2.3.2 追随者位置更新

在基本樽海鞘算法中, 追随者的更新位置是与上一追随者位置的平均值, 这是导致算法收敛速度慢和计算精度差的主要原因<sup>[19]</sup>。如果能让樽海鞘的走向更靠近适应度高的位置, 将加快算法的收敛速度, 更靠近最优解。张严等<sup>[20]</sup>通过有条件的位置更新对追随者位置进行更新, 但是只考虑了当前追随者和上一追随者的位置, 而没考虑后面追随者的位置。因此本研究参考前、后追随者位置, 对当前追随者进行更精确的移动。公式为:

$$X_d^{i'} = \begin{cases} \frac{X_d^{i+1} + X_d^i + X_d^{i-1}}{3}, & 1 < i < n \\ \frac{X_d^i + X_d^{i-1}}{2}, & i = n \end{cases}. \quad (23)$$

### 2.3.3 改进算法比较

为验证改进算法的有效性, 将基本樽海鞘算法

和改进后的樽海鞘算法进行迭代求解，在经过 10 000 次迭代后，最佳适应度变化如图 3 所示，由图可明显看出改进算法能更快收敛，找到最优解。

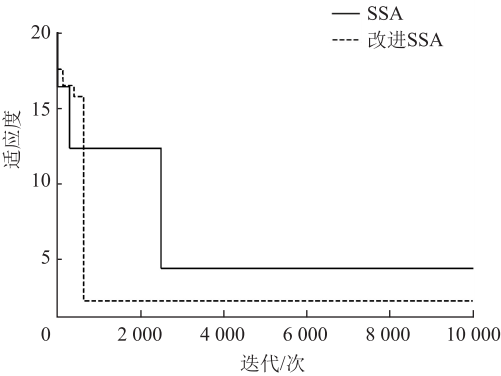


图 3 SSA 改进前后适应度变化结果

Fig. 3 Adaptation variation result before and after SSA improvement

3 试验结果与分析

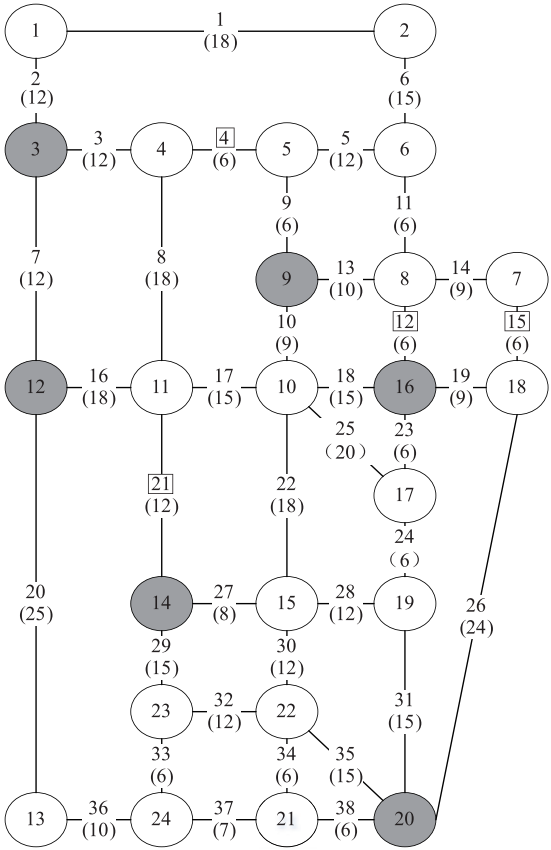
3.1 试验准备

考虑到目前公开的中国国内路网数据较为复杂，较难筛选出合适的高速公路路网，而且本研究的算法可适用于解决国内外各类高速公路路网资源调度问题，因此本研究采用美国公开的高速公路路网数据，即南达科他州最大城市苏瀑市高速公路路网进行分析。该路网有 24 个节点，38 条双向路段，4 个事故点，如图 4 所示。各段通行时间（单位：min）显示在小括号中。假设救援点从最近的路网节点进出高速公路，因此可以认为最近节点就是救援点，其中该路网的救援点和物资情况如表 1 所示，救援点在图 4 中用灰色节点表示。试验平台采用 windows 10 操作系统，CPU 为 Intel(R) Core(TM) i5-8250U，内存为 8 GB，程序采用 python 语言编写，并在 eclipse 开发平台上运行。

假设某个路段出现事故，其事故点在路段中心，且潜在事故发生在该事故的相同位置上，各个事故点所需物资情况如表 2 所示。

3.2 救援路径选择

首先设置算法参数  $\alpha=1$ ， $\beta=2$ ， $\rho=0.5$ 。根据救援点到事故位置走向，初始化信息素矩阵。本研究将走向一致的路径初始值设置为 2，其他通路为 1，得到信息素矩阵。然后执行信息素矩阵改进前后的算法，各迭代 50 次，得到每次救援点到各事故点的总路径时间，比较结果如图 5 所示。



注：[4]，[12]，[15]，[21] 为事故点。

图 4 设计的高速公路路网

Fig. 4 Designed expressway network

表 1 路网的救援点和物资情况

Tab. 1 Rescue points and supplies on expressway network

| 救援点 | 可用消防车数量/辆 |
|-----|-----------|
| 3   | 5         |
| 9   | 5         |
| 12  | 2         |
| 14  | 3         |
| 16  | 6         |
| 20  | 2         |

表 2 事故点所需物资情况

Tab. 2 Supplies required at accident points

| 事故点 | 所需消防车数量/辆 | 潜在事故所需消防车数量/辆 |
|-----|-----------|---------------|
| 4   | 5         | 2             |
| 12  | 4         | 3             |
| 15  | 3         | 2             |
| 21  | 3         | 1             |

由图 5 可见，改进后的算法能更快找到最优路径，这说明信息素的初始化操作能有效提高算法的收敛速度，降低救援路径的搜索时间。假设事故对地点  $i$  到地点  $j$  道路上车辆通行时间的增量函数为：

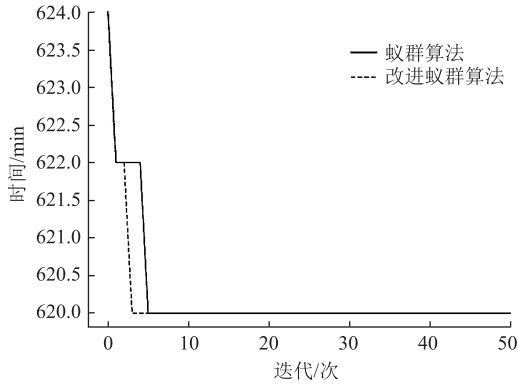


图 5 算法改进前后救援总路径时间比较

Fig. 5 Comparison of total rescue path time before and after algorithm improvement

$$f_{ij}(u) = \begin{cases} 0.3u, & j \text{ 的相邻路段} \\ 0, & \text{其他} \end{cases} \quad (24)$$

将交通事故对车辆通行时间影响函数加入到算法中, 并假设事故仅影响事故路段及相邻路段。本研究以从救援点 20 出发, 到事故点 21 的救援为例,

表 4 最优路径及时间

Tab. 4 Optimal path and time

| 救援点 | 事故点 4                 | 时间/min | 事故点 12              | 时间/min | 事故点 15                 | 时间/min | 事故点 21           | 时间/min |
|-----|-----------------------|--------|---------------------|--------|------------------------|--------|------------------|--------|
| 3   | 3—4—(4)               | 19.5   | 3—4—5—9—8—(12)      | 40.9   | 3—4—5—9—8—7—(15)       | 49.6   | 3—12—11—(21)     | 43.2   |
| 9   | 9—5—(4)               | 11.7   | 9—8—(12)            | 16.9   | 9—8—7—(15)             | 25.6   | 9—10—11—(21)     | 36.3   |
| 12  | 12—3—4—(4)            | 31.5   | 12—3—4—5—9—8—(12)   | 52.9   | 12—3—4—5—9—8—7—(15)    | 61.6   | 12—11—(21)       | 31.2   |
| 14  | 14—11—4—(4)           | 39.3   | 14—15—19—17—16—(12) | 44.7   | 14—15—19—17—16—18—(15) | 54.6   | 14—(21)          | 7.8    |
| 16  | 16—8—9—5—(4)          | 27.7   | 16—(12)             | 3.9    | 16—18—(15)             | 15.6   | 16—10—11—(21)    | 41.4   |
| 20  | 20—19—17—16—8—9—5—(4) | 47.7   | 20—19—17—16—(12)    | 25.7   | 20—18—(15)             | 35.1   | 20—19—15—14—(21) | 45.2   |

### 3.3 救援资源调度

根据本研究最优路径算法得到的最优路径选择结果, 建立调度目标函数为:

$$\begin{cases} f(x, y) = \sum_{i \in L} \sum_{f \in F} (d_{if}x_{if} + d_{if}y_{if}p_f) \\ L = \{3, 9, 12, 14, 16, 20\}, \\ F = \{4, 12, 15, 21\} \end{cases} \quad (25)$$

式中,  $L$  为救援点集合;  $F$  为事故点集合;  $d_{if}$  为救援点  $i$  到事故点  $f$  的最优路径时间;  $x_{if}$  为事故点  $f$  所需救援点  $i$  派出的消防车数量;  $y_{if}$  为事故点  $f$  发生潜在事故需救援点  $i$  派出的消防车数量;  $p_f$  为事故点  $f$  发生潜在事故的概率。约束条件包括:

$$\sum_{f \in F} (x_{if} + p_f y_{if}) \leq r_i, \quad i \in L, \quad (26)$$

$$\sum_{i \in L} x_{if} \geq n_f, \quad f \in F, \quad (27)$$

分别采用传统蚁群算法和本研究算法得出救援路径选择, 结果如表 3 所示。

表 3 算法改进前后对路径的选择比较

Tab. 3 Comparison of path selections before and after algorithm improvement

| 路径选择  | 救援点 20 到事故段 21 的路径选择          |
|-------|-------------------------------|
| 蚁群算法  | 路径选择 Path1: ②0—21—②4—②3—①4—21 |
| 本研究算法 | 路径选择 Path2: ②0—①9—①5—①4—21    |

救援点 20 到事故段 21 的救援路径中, 在不考虑事故影响时, Path1 路径时间为 40 min, Path2 路径时间为 41 min, 选择 Path1 路径更优; 在考虑事故影响时, Path1 路径时间为 46.3 min, Path2 路径时间为 45.2 min, 选择 Path2 路径更优。由此看到交通事故带来的道路通行时间加长会影响到救援路径的选择, 这也说明了本研究提出的改进方法的必要性。根据本研究算法得到各个救援点到事故点的最优路径及时间如表 4 所示。

$$\sum_{i \in L} y_{if} \geq m_f p_f, \quad f \in F, \quad (28)$$

式中,  $r_i$  为救援点消防车数量;  $n_f$  为事故点  $f$  需要的消防车数量;  $m_f$  为事故点  $f$  发生潜在事故需要的消防车数量。

然后进行本研究改进的樽海鞘算法, 得到最优调度方案。假设方案 1 不考虑潜在事故发生, 则在第 1 次事故发生后, 得到首次事故调度方案, 如表 5 所示。不久后发生了潜在事故, 则再次调动应急响应, 剩余资源中除去原来事故已调动的车辆, 得到潜在事故调度方案, 如表 6 所示。最终得到的最终救援总成本为 435.8 min · 辆。方案 2 考虑潜在事故发生, 则在事故发生后, 得到首次事故调度方案, 如表 7 所示。在发生潜在事故后, 调度方案如表 8 所示。最终得到的救援总成本为 390.2 min · 辆。2

表 5 方案 1 的首次事故调度方案 (单位: 辆)  
Tab. 5 First accident scheduling scheme for scheme 1  
(unit: veh)

| 调度来源          | 事故点 4<br>需要 5 辆 | 事故点 12<br>需要 4 辆 | 事故点 15<br>需要 3 辆 | 事故点 21<br>需要 3 辆 |
|---------------|-----------------|------------------|------------------|------------------|
| 救援点 3 剩余 5 辆  | 1               | —                | —                | —                |
| 救援点 9 剩余 5 辆  | 4               | —                | 1                | —                |
| 救援点 12 剩余 2 辆 | —               | —                | —                | —                |
| 救援点 14 剩余 3 辆 | —               | —                | —                | 3                |
| 救援点 16 剩余 6 辆 | —               | 4                | 2                | —                |
| 救援点 20 剩余 2 辆 | —               | —                | —                | —                |

表 6 方案 1 的潜在事故调度方案 (单位: 辆)  
Tab. 6 Potential accident scheduling scheme for scheme 1  
(unit: veh)

| 调度来源          | 事故点 4<br>需要 2 辆 | 事故点 12<br>需要 3 辆 | 事故点 15<br>需要 2 辆 | 事故点 21<br>需要 1 辆 |
|---------------|-----------------|------------------|------------------|------------------|
| 救援点 3 剩余 4 辆  | 1               | 1                | 2                | —                |
| 救援点 9 剩余 0 辆  | —               | —                | —                | —                |
| 救援点 12 剩余 2 辆 | 1               | —                | —                | 1                |
| 救援点 14 剩余 0 辆 | —               | —                | —                | —                |
| 救援点 16 剩余 0 辆 | —               | —                | —                | —                |
| 救援点 20 剩余 2 辆 | —               | 2                | —                | —                |

表 7 方案 2 的首次事故调度方案 (单位: 辆)  
Tab. 7 First accident scheduling scheme for scheme 2  
(unit: veh)

| 调度来源          | 事故点 4<br>需要 5 辆 | 事故点 12<br>需要 4 辆 | 事故点 15<br>需要 3 辆 | 事故点 21<br>需要 3 辆 |
|---------------|-----------------|------------------|------------------|------------------|
| 救援点 3 剩余 5 辆  | 5               | —                | —                | —                |
| 救援点 9 剩余 5 辆  | —               | 2                | —                | —                |
| 救援点 12 剩余 2 辆 | —               | —                | —                | —                |
| 救援点 14 剩余 3 辆 | —               | —                | —                | 3                |
| 救援点 16 剩余 6 辆 | —               | —                | 3                | —                |
| 救援点 20 剩余 2 辆 | —               | 2                | —                | —                |

表 8 方案 2 的潜在事故调度方案 (单位: 辆)  
Tab. 8 Potential accident scheduling scheme for scheme 2  
(unit: veh)

| 调度来源          | 事故点 4<br>需要 2 辆 | 事故点 12<br>需要 3 辆 | 事故点 15<br>需要 2 辆 | 事故点 21<br>需要 1 辆 |
|---------------|-----------------|------------------|------------------|------------------|
| 救援点 3 剩余 0 辆  | —               | —                | —                | —                |
| 救援点 9 剩余 3 辆  | 1               | —                | 2                | —                |
| 救援点 12 剩余 2 辆 | 1               | —                | —                | 1                |
| 救援点 14 剩余 0 辆 | —               | —                | —                | —                |
| 救援点 16 剩余 3 辆 | —               | 3                | —                | —                |
| 救援点 20 剩余 0 辆 | —               | —                | —                | —                |

种方案的救援成本 (目标值) 对比, 方案 2 比方案 1 成本降低了 10.5%。试验证明: 调度方案的设计中考虑潜在事故的发生能有效降低实际调度过程中

的救援成本。

4 结论

考虑到潜在二次事故对高速公路应急资源调度的影响, 本研究结合高速公路环境下应急资源调度的实际需求, 从路径选择和资源调度 2 方面进行研究。提出了一种基于改进樽海鞘算法的高速公路应急救援资源调度方法, 有效提高了路径搜索效率, 提供了最优资源调度方案。

(1) 本研究针对高速公路应急救援资源调度中实际存在的问题, 提出了一种适合于实际应用的资源调度方法。针对蚁群算法在高速公路自身路网特征下的求解最优路径问题上存在的一些不足, 对蚁群算法信息素的全局更新规则进行改进, 从而更快地搜索到最佳救援路径。

(2) 本研究考虑潜在事故对救援资源调度的影响, 建立基于潜在事故的紧急救援资源调度模型, 并采用改进的樽海鞘算法对该模型进行求解。针对传统樽海鞘算法存在容易陷入局部最优解的问题, 借鉴烟花算法中的爆炸、变异等操作, 对算法进行改进, 提高算法求解效率。

(3) 通过美国南达科他州苏瀑市路网仿真试验验证了本研究对蚁群算法和樽海鞘算法改进的有效性, 同时证实了本研究提出的方法更符合实际的高速公路应急救援资源调度情况, 能有效降低救援时间和成本。

(4) 本研究在事故对道路通行的影响函数上采用了线性函数, 在实际环境中, 该函数需要根据历史数据和实际交通情况进行建模。在下一步工作中, 将针对实际情况, 采用动态仿真数据对本研究模型进行优化调参, 建立最佳应急资源调度模型。

参考文献:

References:

[1] 王薇, 武毅, 赵阳, 等. 施工环境下高速公路应急资源调度方法研究 [J]. 重庆交通大学学报 (自然科学版), 2019, 38 (2): 102-108.  
WANG Wei, WU Yi, ZHAO Yang, et al. Dispatching Method of Expressway Emergency Resources Under Construction Environment [J]. Journal of Chongqing Jiaotong University (Natural Sciences), 2019, 38 (2): 102-108.

[2] FAN Y, JIA P, GUO W, et al. Emergency Dispatch Optimization of Engineering Vehicles in Disaster Rescue Activities [C] // 2019 Chinese Control Conference (CCC). New York: IEEE, 2019: 2001-2005.

[3] POTVIN J Y, XU Y, BENYAHIA I. Vehicle Routing and

- Scheduling with Dynamic Travel Times [J]. *Computers & Operations Research*, 2006, 33 (4): 1129-1137.
- [4] NIKOO N, BABAEI M, MOHAYMANY A S. Emergency Transportation Network Design Problem: Identification and Evaluation of Disaster Response Routes [J]. *International Journal of Disaster Risk Reduction*, 2018 (27): 7-20.
- [5] 郝正博, 杨晓光, 王一喆, 等. 考虑时空协同优先的城市应急车辆路径优化方法 [J]. *公路交通科技*, 2024, 41 (3): 169-178, 198.
- HAO Zheng-bo, YANG Xiao-guang, WANG Yi-zhe, et al. Dynamic Optimization on Urban Emergency Vehicle Route Considering Collaborative Spatio-temporal Prioritization [J]. *Journal of Highway and Transportation Research and Development*, 2024, 41 (3): 169-178, 198.
- [6] CHAKRABORTY S P, TIWARI A, SINHA R P. Adaptive and Optimized Emergency Vehicle Dispatching Algorithm for Intelligent Traffic Management System [J]. *Procedia Computer Science*, 2015, 57 (C): 1384-1393.
- [7] 孙秀巧, 王健, 巫威眺. 基于改进遗传退火算法的高速公路巡逻车路径优化调度 [J]. *科学技术与工程*, 2019, 19 (21): 296-302.
- SUN Xiu-qiao, WANG Jian, WU Wei-tiao. Routing and Scheduling Optimization of Freeway Service Patrols Based on the Improved Algorithm Combined Genetic with Simulated Annealing [J]. *Science Technology and Engineering*, 2019, 19 (21): 296-302.
- [8] 张奕, 卜凡亮. 高速公路突发事件应急资源调动系统 [J]. *信息技术*, 2020 (2): 1-6.
- ZHANG Yi, BU Fan-liang. Emergency Resource Scheduling System of Expressway [J]. *Information Technology*, 2020 (2): 1-6.
- [9] SUNG I, LEE T. Optimal Allocation of Emergency Medical Resources in a Mass Casualty Incident: Patient Prioritization by Column Generation [J]. *European Journal of Operational Research*, 2016, 252 (2): 623-634.
- [10] 万孟然, 叶春明, 董君, 等. 考虑备灾的双层规划应急资源调度选址—路径优化模型与算法 [J]. *计算机应用研究*, 2021, 38 (10): 2961-2967.
- WAN Meng-ran, YE Chun-ming, DONG Jun, et al. Bi-level Programming Location-routing Optimization Model and Algorithm for Emergency Resource Scheduling Considering Preparedness [J]. *Application Research of Computers*, 2021, 38 (10): 2961-2967.
- [11] WANG F Y, PEI Z W, DONG L J, et al. Emergency Resource Allocation for Multi-period Post-disaster Using Multi-objective Cellular Genetic Algorithm [J]. *IEEE Access*, 2020, 8: 82255-82265.
- [12] 邵泽军, 陈凡红, 吕晓娜, 等. 基于改进粒子群算法的多资源应急调度研究 [J]. *价值工程*, 2020, 39 (10): 243-245.
- SHAO Ze-jun, CHEN Fan-hong, LÜ Xiao-na, et al. Research on Multi-resource Emergency Scheduling Based on Improved Particle Swarm Optimization [J]. *Value Engineering*, 2020, 39 (10): 243-245.
- [13] 陈鑫, 王海宝, 罗强, 等. 基于改进蚁群算法的柑橘采摘最优路径 [J]. *安徽大学学报 (自然科学版)*, 2022, 46 (1): 68-74.
- CHEN Xin, WANG Hai-bao, LUO Qiang, et al. Optimal Path of Citrus Sinensis L. Picking Based on Improved Ant Colony Algorithm [J]. *Journal of Anhui University (Natural Science Edition)*, 2022, 46 (1): 68-74.
- [14] 刘欣. 基于蚁群算法的医疗人力资源应急调度设计 [J]. *信息技术*, 2021 (2): 142-146.
- LIU Xin. Design of Emergency Dispatch of Medical Human Resources Based on Ant Colony Algorithm [J]. *Information Technology*, 2021 (2): 142-146.
- [15] MIRJALILI S, GANDOMI A H, MIRJALILI S Z, et al. Salp Swarm Algorithm: A Bio-inspired Optimizer for Engineering Design Problems [J]. *Advances in Engineering Software*, 2017, 114: 163-191.
- [16] 俞家珊, 吴雷. 双领导者樽海鞘群算法 [J]. *计算机科学*, 2021, 48 (4): 254-260.
- YU Jia-shan, WU Lei. Two Types of Leaders Salp Swarm Algorithm [J]. *Computer Science*, 2021, 48 (4): 254-260.
- [17] HE L J, LI W F, ZHANG Y, et al. A Discrete Multi-objective Fireworks Algorithm for Flowshop Scheduling with Sequence-dependent Setup Times [J]. *Swarm and Evolutionary Computation*, 2019, 51: 100575. <https://doi.org/10.1016/j.swevo.2019.100575>.
- [18] 许德刚, 李凡, 王露, 等. 优化烟花算法在医疗物资应急调度中的应用 [J]. *计算机工程与应用*, 2021, 57 (24): 249-258.
- XU De-gang, LI Fan, WANG Lu, et al. Application of Optimized Fireworks Algorithm in Emergency Dispatching of Medical Supplies [J]. *Computer Engineering & Science*, 2021, 57 (24): 249-258.
- [19] 谢聪, 郑洪清. 一种新型的樽海鞘群算法及其应用 [J]. *计算机工程与科学*, 2022, 44 (1): 184-190.
- XIE Cong, ZHENG Hong-qing. A Novel Salp Swarm Algorithm and Its Application [J]. *Computer Engineering and Science*, 2022, 44 (1): 184-190.
- [20] 张严, 秦亮曦. 基于 Levy 飞行策略的改进樽海鞘群算法 [J]. *计算机科学*, 2020, 47 (7): 154-160.
- ZHANG Yan, QIN Liang-xi. Improved Salp Swarm Algorithm Based on Levy Flight Strategy [J]. *Computer Science*, 2020, 47 (7): 154-160.