

灾害救援实时最快通行路径算法^①

喻德旷, 杨 谊

(南方医科大学 生物医学工程学院, 广州 510515)

摘 要: 灾害救援需要物资人力的快速运输, 而突发灾害常会影响到道路的通行状态, 研究道路网络动态变化情况下救援车辆的实时最快通行路径算法, 具有重要的经济和社会意义. 针对灾害发生后道路状况多变突变的情况, 提出一种实时最快通行路径求解算法 ARFTP(Algorithm of Real-time Fastest Traffic Path), 将结点进行分类筛选, 依据相应准则进行运算, 减少了需要重新计算的结点和路径数量. 当车辆行驶在原定救援最快通行路径上时, 实时收到路段变化信息, 根据 ARFTP 求解策略可快速求出新的最快通行路径. 通过仿真验证了算法的有效性和效率, 对提高灾害救援运输效率具有一定的意义.

关键词: 路况变化; 实时信息; 最快通行路径; 分类筛选; 灾害救援

Algorithm of Real-Time Fastest Traffic Path for Disaster Rescue

YU De-Kuang, YANG Yi

(School of Biomedical Engineering, Southern Medical University, Guangzhou 510515, China)

Abstract: Disaster rescue needs rapid transportation of both supplies and staff. The abrupt disasters often affect the traffic states, and research on the real-time fastest path under dynamic road conditions has important economic and social values. Aiming at dealing with the abrupt and frequent road variation after disaster, a fast algorithm of real-time Fastest Traffic Path(ARFTP) is proposed, which reduces the amount of re-calculation on the nodes and the sub-paths by classifying the nodes into different types and filtering them, then putting them into corresponding calculation modules, so as to avoid redundant calculations. When the vehicles are on the original shortest path to the disaster area and receive real-time road variation information, the up-to-date fastest shortest path can be calculated by ARFTP rapidly. Experiments have been made to test the correctness and efficiency of ARFTP with many cases of road cases, and the results proved its effect in improving the efficiency which has some guidance significance to disaster relief transportation.

Key words: road variation; real-time information; fastest traffic path; classification and selection; disaster rescue

近年来, 大规模突发事件、多种自然灾害和人为灾害, 正以越来越频繁的速度袭击我们生存的世界. 灾区的正常生活被中断, 人们生存所必需的资源遭到破坏, 救灾物资和营救人员需要尽早送达灾害发生现场, 以减少各类伤亡和损失.

突发灾害和伴生事件常常会影响到道路的通行状态, 研究道路网络动态变化情况下救援车辆的实时最快通行路径算法, 对提高运输效率、减少时间延误、降低灾害损失、优化运输管理, 具有重要的经济和社

会意义.

目前对于动态网络环境下最短路径搜索问题的研究主要有三类方法: (1)概率理论(probability theory): 运用概率分布来描述道路通行状况的变化性^[1], 当掌握了网络中路段通行时间的概率分布时, 可以依据此规律进行新的最短路径搜寻. 这种方法适合日常交通监控与疏导, 而突发灾害时道路通行时间往往表现为离散、突变性的而非连续性的概率分布, 并不适用于灾害救援. (2)模糊最短路径(fuzzy shortest path): 运用

^① 基金项目:广东省科技计划项目(2013B051000054)

收稿时间:2015-09-12;收到修改稿时间:2015-11-02 [doi:10.15888/j.cnki.csa.005158]

模糊度^[2,3]概念进行路径长度(通行时间)的估计;根据不确定理论(uncertainty theory)^[4-6],结合置信度(belief degree)描述不确定现象,具有一定的灵活性和实用性^[7]。但是模糊度及置信度的设置与经验值有关,误差可能导致结果偏差较大。(3)多条可靠最快通行路径(multiple reliable paths)^[8-10]:车辆出发时先求出多条最短路径,一旦首选最快通行路径不可用,即选择其他的备选路径。这种方案实质上属于静态最短路径算法,难以适合动态变化的网络。

搜求突发事件下的道路交通网络中的最快通行路径,直接目标是寻找源点(物资供应点)与目标点(受灾点)的最快通行路径,属于动态最短路径范畴。它与普通动态最短路径问题的区别是:突发事件下的道路交通网络状态具有复杂变化性,多种因素使得路段通行时间的变化具有随机特征。文献[11,12]改进了不完全信息下道路突发中断时的最短路径动态算法,可看做本文研究内容的特例。文献[13-15]分别提出了网络中结点/边的增加与删除、边的权值改变的最短路径算法,为问题求解提供了思路。

本文针对突发事件下的道路变化情况提出一种实时最快通行路径求解算法 ARFTP(Algorithm for Real-time Fastest Traffic Path),结合模糊度量方法,对路径变化量进行归一化处理,并不重新计算所有结点和路段,而是对结点和路段进行分类,选取可能受到影响的结点和路段进行重新计算,从而减少冗余计算量,达到快速求出新的最快通行路径的目的。本文的内容安排如下:第1节为问题描述和有关定义,第2节为实时最快通行路径ARFTP算法思路,并分析算法的时间复杂度,第3节基于交通网络实例进行求解,显示多个案例结果来验证算法的正确性和有效性;第4节为总结和工作展望。

1 问题描述

将现实中的道路交通网络抽象成以道路交叉口为结点,两结点间的道路为边的网络。将道路交通网络表示为带权无向图 $G(V, E, C)$, 其中 $V = (s, v_1, v_2, \dots, v_{n-2}, t)$ 为结点集, n 为结点的个数, s 为起始点, t 为终止点, $E = (e_1, e_2, \dots, e_m)$ 为边的集合, m 为边的条数。边用两个关联结点 v_i 和 v_j 表示: (v_i, v_j) , 边的权重在不同环境下取不同的属性,如距离、行驶时间、费用等,本文指行驶时间。

定义 1. 边的权值 W : 代表通过该边需要花费的时间,边 (v_i, v_j) 的权值记为 $W_{(i,j)}$ 。

灾害情况下无法确定道路改变的分布,权值采用实时统计速度来计算行驶时间。设在时间窗口 ΔT 中边 (v_i, v_j) 上的车辆统计平均车行速度记为 $\overline{\mu}_{i,j}$, 边长度为 $len(v_i, v_j)$, $W_{(i,j)} = len(v_i, v_j) / \overline{\mu}_{i,j}$, 把道路(边)权值表示为随时间改变的动态值,拥堵中断等使得权值增大,新开辟路径使得权值减小。

定义 2. 最快通行路径 P : 网络中结点 u 和 v 之间的最快通行路径是指从 u 到 v 花费时间之和最少的路径,记做 $P(u, v)$, 为最快通行路径上的结点的集合。

定义 3. 最快通行路径长度: 等于 $P(u, v)$ 所有边的权值之和。

定义 4. 最快通行路径树 SPT : $G(V, E, C)$ 中,从结点 s 到 u 的最快通行路径树 $SPT(s, u)$ 是 G 的子图 $G'(V', E', C')$, 且满足: $V' = V$, G' 是连通图,图中没有回路,结点 u 到任一结点 $v \in \{V' - u\}$ 的路径都是 G 中的最快通行路径。

定义 5. 权值变化率: 边的权值改变量占原有权值的比率 $\eta = (W' - W) / W \times 100\%$, 其中 W' 为新权值, W 为原权值。

本文研究的假设条件为:

① 设初始最快通行路径为 $SD(s, t) = \{s, v_1, v_2, \dots, v_k, t\}$, 当行驶到 v_i ($1 \leq i < k$) 之后收到变更的道路信息,则仅对 $SD(s, v_{i+1})$ 执行变更(已行驶路段不允许回退);

② $W_{(p,q)}$ ($p, q \in V$) 为非恒定值(道路状况随时间变化);

③ $W_{(p,q)}$ ($p, q \in V$) 不作用于 $SD(q, t)$ (不根据车辆所在路段信息,但可根据其他路段的信息重新求解最快通行路径)。

2 实时最快通行路径算法(ARFTP)

2.1 ARFTP 算法思路

ARFTP 的主要思路是:救援车辆初始根据 Dijkstra 算法计算出来的最快通行路径前进,并且实时接收道路状况变化信息,实时计算出最快通行路径。

分析 Dijkstra 算法过程:

将图中顶点状态分为未处理或已处理 2 种,建立一个队列 Queue 存放未处理结点,初始化包含所有顶点;

每次循环中选择 Queue 中未标记顶点中权值最小的顶点, 将它设置为已确定结点; 从队列 Queue 中删除该顶点后修改其邻接点(如果存在的话)的最短路径, 并对队列 Queue 进行重构; 当被选择结点是目标点时算法结束.

如果每次根据道路状况变化信息, 重新调用 Dijkstra 算法来计算新的最快通行路径, 每次都要搜索网络中所有的结点, 时间耗费很大.

实际上, 并不是所有道路变动都会影响到已经生成的最快通行路径, 许多路段的变化对原有最快通行路径不会造成影响, 因此没有必要把所有结点都代入算法. 对结点分类处理可避免重复求解, 节省时间.

Dijkstra 求解过程保证了最短路径的子路径是最短路径, 否则, 如果某一段子路径不是最短路径, 则根据求解过程, 必定会有别的最短子路径代替它, 从而保持最短子路径的性质, 源点到每个目标点的最短路径是唯一且确定的, 本文的最快通行路径也具有这一性质. 因此, 对于结点分类分析如下:

(1) 车辆已经行驶经过了的结点, 由于不允许回退重走, 所以从源点到它们的最快通行路径不会改变;

(2) 权值没有变化(或变化很小)的边所连接的两个结点, 从源点到它们的最快通行路径几乎不变;

(3) 权值发生变化的边分为四种情况:

① 在原有 SPT 上的边, 如果权值变小, 则 SPT 上从起点到该边终点的结点都不做处理. 原因是大部分权值变小的 SPT 边仍为更新后的 SPT 上的边, 无需更新; 但这一类边可能被非 SPT 上权值减少更多的边所取代(取决于③的处理结果).

② 在原有 SPT 上的边, 如果权值变大, 则该边终点及所有后续关联结点都执行更新例程. 原因是当 SPT 上的边的权值变大时, 有可能被非 SPT 上的权值较小的边所取代.

③ 非原有 SPT 上的边, 如果权值变小, 则执行更新例程. 原因是这类边可能取代原有 SPT 上权值较大的边;

④ 非原有 SPT 上的边, 如果权值变大, 则不做处理. 原因是这类边取代原有 SPT 上的边的可能性很小(取决于②的处理结果).

据此改进算法, 可缩小搜索范围, 只对部分边和结点进行处理, 减少重新计算的结点.

2.2 算法过程与时间复杂度分析

ARFTP 算法过程如下:

(1) 最快通行路径初始化: 调用 Dijkstra 算法生成以起点 s 为根结点的到指定终点 t 最快通行路径树 $SPT(s, t)$.

(2) 对于 $SPT(s)$ 上的各个结点 v_i , 设置数组 SD 记录起点 s 到该点的最快通行路径 $SD(s, v_i)$.

(3) 产生(或接收)边的变化信息集合: $I = \{(p, q) | W'_{(p,q)} = newWeight\}$, 并进行权值过滤: 权值变化率 $\eta < \epsilon$ 时, 认为权值变化对最快通行道路的影响较小, 可以忽略, 将该边从集合 I 中删除. 一般取 ϵ 为 $\pm 10\%$.

(4) 设置两个队列 Q_1 和 Q_2 , 分别存储无需处理的结点和需要处理的结点, 初始状态均为空.

(5) 循环结构依次处理 I 中的每条边, 将该边的起点记为 p , 终点记为 q , 循环结束后转(6):

<5-1> 如果该边属于原 SPT, 则继续判断:

<5-1-a> 如果改变后的权值比原来减小, 把 SPT 上从 s 到 q 之间的所有结点以及 q 的所有子孙结点存入 Q_1 , 把网络中其余结点存入 Q_2 ;

<5-1-b> 如果改变后的权值比原来增大, 将 q 及 q 的所有子孙结点存入 Q_2 , 把网络中其余结点存入 Q_1 ;

<5-2> 如果该边不属于原 SPT, 则继续判断:

5-2-a. 如果改变后的权值比原来减小, 继续判断:

<5-2-a-a1> 如果 $SD(s, p) < SD(s, q) + W'_{(p,q)}$ 且 $SD(s, q) < SD(s, p) + W'_{(p,q)}$, 表示 p 和 q 不同时出现在以 s 为起点的某条最快通行路径上, 则退出本次循环;

<5-2-a-a2> 如果 $SD(s, p) < SD(s, q)$, 则修改 q 的父节点为 p , 把 SPT 上从 s 到 p 之间的所有结点、 q 以及 q 的所有子孙结点存入 Q_1 , 把网络中其余结点存入 Q_2 ; 否则, 修改 p 的父节点为 q , 把 SPT 上从起点 s 到结点 q 之间的所有结点、 p 以及 p 的所有子孙结点存入 Q_1 , 把网络中其余结点存入 Q_2 ;

5-2-b. 如果新权值比原来增大, 则跳出本次循环.

(6) 对于 Q_2 中的所有结点, 调用 Dijkstra 算法计算该结点到 Q_1 中结点的最快通行路径.

(7) 循环执行下列代码, 直到 Q_2 为空: 取出 Q_2 中 fastest 通行路径最小的结点 v_{\min} , 将其父节点修改为该最小权值变在 Q_1 中的结点, 并把 v_{\min} 加入 Q_1 , 把 v_{\min} 从 Q_2 中删除.

ARFTP 时间耗费主要取决于从队列 Q_1 和 Q_2 中选取结点的效率, 此外, 和图中结点的检索效率也有关. 为了快速寻找最小权值的顶点, 可采用二叉堆存储 Q_1

和 Q_2 . 为了便于查找结点的双亲和子孙, 可采用十字链表法存储图结构. 为了便于回溯各结点在最快通行路径上的状态, 可采用邻接表存储最快通行路径. ARFTP 算法时间复杂度分析如下(m 为边数, n 为顶点数, 二叉堆存储):

- (1)最快通行路径初始化: $O((m+n)\lg n)$
- (2)设置数组 SD 记录 $SD(s, v_i)$: $O(n)$
- (3)产生(或接收)边的变化信息集合: $O(m)$
- (4)初始化队列: Q_1 和 Q_2 , $O(1)$
- (5)循环处理 I 中的每条边, 涉及的边数不超过顶点数: $O(n)$

(6)调用 Dijkstra 算法计算该结点到 Q_1 中结点的最快通行路径: $O(n(m+n)\lg n)$

- (7)循环修改最快通行路径, 直到 Q_2 为空: $O(\lg n)$

总的复杂度为上述 7 个步骤的求和, 结果为 $O(n(m+n)\lg n)$. ARFTP 是基于 Dijkstra 算法的改进, 与 Dijkstra 算法的时间复杂度相当. 但实际处理时, 由于 ARFTP 已经剔除了部分无需更改的结点(边), 又将变化率较小的边忽略不计, 步骤(2)、(3)、(5)、(6)、(7)需要处理的结点(边)的个数往往远小于图中的结点(边)的总数.

3 实验结果与分析

生成一个具有 8 个结点 15 条边的带权交通网络如图 1 所示, 由 ARFTP 和 Dijkstra 算法求出从 v_1 到 v_6 的初始最快通行路径相同, 如图 2 所示.

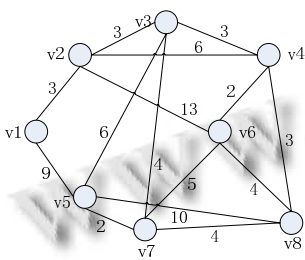


图 1 一个具有 8 个结点、15 条边的带权交通网络

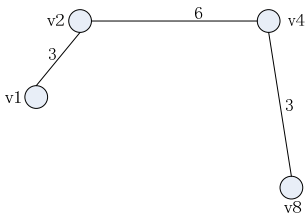


图 2 从 v_1 到 v_8 的初始最快通行路径(总长为 12)
假设车辆正行进在 v_1 到 v_2 的路段上, 实时收到

边的权值变化信息如表 1 所示($\epsilon=10\%$).

表 1 边的权值变化信息(第一次)

边	原权值	现权值	变化率(%)	处理
(v_1, v_5)	9	8.7	+3.33	忽略
(v_2, v_3)	3	3.2	+6.67	忽略
(v_2, v_4)	6	8	+33.3	5-1-b
(v_2, v_6)	13	12	-7.7	忽略
(v_3, v_4)	3	4	+33.3	5-2-b
(v_3, v_5)	6	5	-16.7	5-2-a-a1
(v_3, v_7)	4	3	-25.0	5-2-a-a2
(v_4, v_6)	2	2.2	+10.0	忽略
(v_4, v_8)	3	7	+133.3	5-1-b
(v_5, v_7)	2	3	+50	5-2-b
(v_5, v_8)	10	9	-10	忽略
(v_6, v_7)	5	4	-20	5-2-a-a1
(v_6, v_8)	4	3.5	-12.5	5-2-a-a1
(v_7, v_8)	4	3	-33.3	5-2-a-a1

此时调用 ARFTP, 寻找到目前新的最快通行路径如图 3 所示.

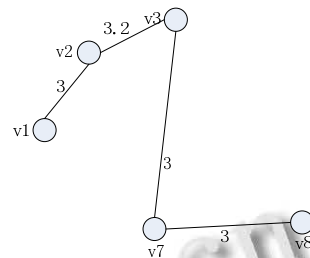


图 3 收到表 1 信息, ARFTP 得到新的最快通行路径
(总长为 12.2)

假设车辆正行进在 v_2 到 v_3 的路段上, 实时收到边的权值变化信息如表 2 所示.

表 2 边的权值变化信息(第二次)

边	原权值	现权值	变化率(%)	处理
(v_1, v_2)	3	3.6	+20.0	忽略
(v_1, v_5)	8.7	8	-8.8	忽略
(v_2, v_4)	8	7.6	-5.0	忽略
(v_2, v_6)	12	10	-16.7	5-2-a-a1
(v_3, v_4)	4	4	0	忽略
(v_3, v_5)	5	3	-40.0	5-2-a-a1
(v_3, v_7)	3	7	+133.3	5-1-b
(v_4, v_6)	2.2	3	+36.4	5-2-b
(v_4, v_8)	7	6	-14.2	5-2-a-a1
(v_5, v_7)	3	2.6	-13.3	5-2-a-a1
(v_5, v_8)	9	9.1	+1.1	忽略

(v_6, v_7)	4	5	+25	5-2-a-1
(v_6, v_8)	3.5	3.3	+5.7	忽略
(v_7, v_8)	3	3	0	忽略

此时调用 ARFTP, 寻找到目前新的最快通行路径如图 4 所示.

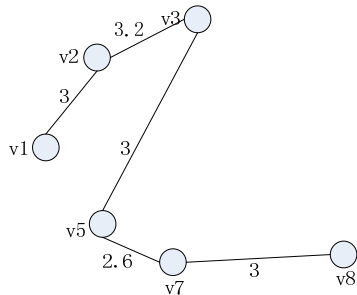


图 4 收到表 2 信息, ARFTP 得到新的最快通行路径 (总长为 14.8)

经验证, 以上相应情况下 Dijkstra 算法产生相同的最快通行路径或总长度相同的最快通行路径, 表明 ARFTP 求解正确.

根据广东省多个市县的道路交通调查数据, 大型(省级范围)、中型(市级范围)、小型(县级及以下范围)交通网络结点个数量级分别为 10000-30000、1000-10000、100-1000, 救灾运输通常为中型交通网络, 结点度分布范围为[1,8], 80-90%的度在 3~4 之间. 按照结点与边 1:3 的比例关系生成多个模拟网络, 对于每个配置各产生 20 个拓扑结构不同的测试案例, 分别执行 ARFTP 和 Dijkstra 算法, 所得路径结果见表 3. 以 Dijkstra 算法结果为标准, ARFTP 总长度误差率定义为: $\text{总长度误差率} = (\text{ARFTP 长度} - \text{Dijkstra 长度}) / \text{Dijkstra 长度} \times 100\%$

表 3 ARFTP 和 Dijkstra 算法结果

结点数	边数	最快通行路径总长度相同率(%)	最快通行路径总长度误差率均值	最快通行路径相同率(%)
100	300	80	3.58%(±0.67%)	60
500	1500	80	3.21%(±0.54%)	70
1000	3000	75	4.82%(±0.75%)	60
2000	6000	70	4.67%(±1.13%)	60
5000	15000	70	4.09%(±0.83%)	55
10000	30000	55	4.93%(±1.26%)	45
15000	45000	60	5.44%(±0.91%)	50
20000	60000	55	5.62%(±0.84%)	40
30000	90000	55	5.87%(±1.68%)	45

在上述实验中, ARFTP 与正确结果的误差率均值在 8%以内. 部分结果比 Dijkstra 略长, 是由研究假设

条件①: “车辆沿着最快通行路径前进, 不允许回退”所决定的, 导致遗漏某些耗时更少一些的路径.

两个算法所求得的最快通行路径(结点序列)相同率差异较大, 主要是由于 ARFTP 中采用了两个队列, 而 Dijkstra 算法只使用一个队列, 且结点进入队列的次序不尽相同, 当存在多条最快通行路径时, 两个算法按照各自规则选择第一条最快通行路径.

表 4 比较了两种算法的效率, 机器配置为 Win7, Intel® Core™Intel i3-3220, 主频 3200MHz, 内存 8G, 仿真平台为 Visual Studio 2010. 实验结果表明, 当道路网规模较小时, ARFTP 耗时与 Dijkstra 算法接近, 而当道路网规模较大时, ARFTP 在执行速度和时间代价方面明显比 Dijkstra 算法占优, 表明 ARFTP 能够减少大量重复计算, 明显节约时间.

表 4 ARFTP 和 Dijkstra 算法的运行时间比较

结点数	边数	ARFTP 平均耗时(s)	Dijkstra 平均耗时(s)	速度比
100	300	0.319	0.400	1.254
500	1500	2.614	3.509	1.342
1000	3000	5.604	8.220	1.467
2000	6000	11.704	19.266	1.646
5000	15000	30.432	58.207	1.913
10000	30000	59.283	137.836	2.325
15000	45000	80.272	240.668	2.998
20000	60000	90.204	376.587	4.175
30000	90000	107.229	687.749	6.414

4 结论

本文针对灾害救援时的路况可能发生频繁变化或突然变化而无法掌握通行时间准确概率分布问题, 提出一种实时最快通行路径的求解算法 ARFTP, 将结点和路段进行分类, 设定判断和处理规则, 减少重复求解结点数量, 从而显著减少计算时间, 保证了很高的正确率. 除道路网络外, ARFTP 可用于无法掌握概率分布函数的随机网络, 如电路网络、计算机网络、通讯网络等, 求解具有相似结构的动态最短路径. 本文中仅考虑速度因子来反映路况变化, 下一步可纳入多种权值, 例如道路等级、限行要求等, 开展多种因素影响下的实时最快通行路径问题探索.

参考文献

1 Yang LX, Zhou XS. Constraint reformulation and lagrangian

- relaxation-based solution algorithm for a least expected time path problem. *Transportation Research Part B*, 2014, 59: 22-44.
- 2 Yamani OE, Mouncif H, Rida M. A fuzzy TOPSIS approach for finding shortest path in multimodal transportation networks. *Int. J. Comput. Optimization*, 2014, 1(2): 95-111.
- 3 Yang LX, Zhou XS, Gao ZY. Credibility-based rescheduling model in a double-track railway network: A fuzzy reliable optimization approach. *Omega*, 2014, 48(C): 75-93.
- 4 Baidya A, Bera U K, Maiti M. A solid transportation problem with safety factor under different uncertainty environments. *Journal of Uncertainty Analysis & Applications*, 2013, 1(1): 1-22.
- 5 Zhang B, Peng J. Uncertain programming model for uncertain optimal assignment problem. *Applied Mathematical Modelling*, 2013, 37(9): 6458-6468.
- 6 Zhu AD, Ma H, Xiao XK, et al. Shortest path and distance queries on road networks: Towards bridging theory and practice. *Proc. of the ACM SIGMOD International Conference on Management of Data(SIGMOD)*. Scottsdale, AZ, USA. ACM. 2013. 857-868.
- 7 张波良,张瑞昌,关侗红.道路网上最短路径算法综述. *计算机应用与软件*,2014,31(10):1-9.
- 8 李敬贤,厉小润.求解震后最优路径的改进 Dijkstra 算法. *计算机工程*,2012,38(6):271-272.
- 9 徐涛,丁晓璐,李建伏. K 最短路径算法综述. *计算机工程与设计*,2013,34(11):3900-3906,3911.
- 10 毛少武,张焕国,黄崇超,吴万青.改进的 K 最短路径算法在通信网络中的应用. *武汉大学学报(理学版)*,2013,59(6): 534-538.
- 11 Xiao P, Xu YF, Su B. Finding an anti-risk path between two nodes in undirected graphs. *Journal of Combinatorial Optimization*, 2009, 17(3): 235-246.
- 12 Mahadeokar J, Saxena S. Faster algorithm to find anti-risk path between two nodes of an undirected graph. *Journal of Combinatorial Optimization*, 2014, 27(4): 798-807.
- 13 梁德恒,姚国祥,官全龙.基于路由最短路径树的动态多节点删除算法. *计算机工程*,2011,37(5):121-123.
- 14 安红岩,胡光岷,何永富.网络最短路径的动态算法. *计算机工程与应用*,2003,39(1):173-174,180.
- 15 程远.网络最短路径的一种更新策略. *计算机应用与软件*,2013,30(1):171-175.