

IP网络弹性路由层拓扑生成优化算法

伍文^{1,2}, 孟相如¹, 刘芸江¹, 康巧燕¹

(1. 空军工程大学信息与导航学院 西安 710077; 2. 解放军93868部队 银川 750025)

【摘要】针对弹性路由层快速重路由技术存在的重路由路径偏长的问题,提出了一种基于单亲遗传算法的弹性路由拓扑子层生成算法。介绍了弹性路由层基本原理,给出了弹性路由层的矩阵表示定义,分别建立了以平均最短重路由路径长度和重要度加权的最短重路由路径长度为优化目的的弹性路由层生成数学规划模型,给出了单亲遗传算法求解该优化问题的方法步骤。实验结果表明,优化算法可以得到最佳的弹性路由层拓扑结构,与Minimum算法、Rich算法相比,明显缩短了重路由路径长度,提高了流量转发效率。

关键词 快速重路由; IP网络; 弹性路由层; 可生存性

中图分类号 TP393

文献标志码 A

doi:10.3969/j.issn.1001-0548.2014.05.025

Optimizing Algorithm for Resilient Routing Layers Topology Building of IP Networks

WU Wen^{1,2}, MENG Xiang-ru¹, LIU Yun-jiang¹, and KANG Qiao-yan¹

(1. School of Information and Navigation, Air Force Engineering University Xi'an 710077; 2. Unit 93868 of PLA Yinchuan 750025)

Abstract The fast rerouting paths of resilient routing layers are not short enough to deliver traffic efficiently. To solve this problem, a resilient routing sub-layer building algorithm based on Partheno-genetic algorithm is proposed. The matrix expression of resilient routing layers is given. The optimizing models respectively with average minimum rerouting path length and weighted minimum rerouting path length are put forward. The method and steps to solve the optimizing problem based on Partheno-genetic algorithm are given. Experimental results show that the optimal topology structure of resilient routing layers could be acquired by the optimizing algorithm. Compared with minimum algorithm and Rich algorithm, the optimizing algorithm can reduce the rerouting path length and improve the efficiency of traffic forwarding.

Key words fast rerouting; IP network; resilient routing layers; survivability

因特网在政治、经济、军事、文化等领域发挥着越来越重要的作用,然而作为一个开放性系统,其不可避免遭受来自网内、网外的各种攻击、故障及意外事件的干扰和伤害^[1]。为了保证网络业务的持续性,IP路由协议在设计之初就被赋予了很好的鲁棒性,任意的网络链路或节点中断在一定时间内均可通过路由的重新收敛得到恢复^[2]。近些年,随着网络承载业务量的增加,及VoIP、视频会议等新兴业务的推广,使得网络对中断越来越敏感,一个小的中断便会造成大量的数据丢失^[3]。原有的路由协议收敛慢、效率低,已经无法满足这些新兴业务的生存需求,因此IP快速重路由技术被广泛关注^[4]。

弹性路由层(resilient routing layers, RRL)是IP快速重路由技术的一种,通过计算全连接拓扑子集

来实现受故障影响流量的无中断转发^[5-6]。路径次优问题和额外状态信息是RRL的劣势所在。这是RRL技术平台在设计的时候就存在的问题,但是在RRL技术平台之上可以设计不同的路由子层生成算法,好的生成算法一定程度上可改善路径次优问题,也可以从应用角度协调路径长度和状态信息的关系。已有的路由子层生成算法有Minimum算法、Sparse算法、Rich算法等^[7-9],其中,Minimum算法以生成最小层数为目的,但是路径次优问题明显;Sparse算法是针对多故障问题提出的,对多故障同时保护的概率远远大于其他算法,但是以牺牲备份路径长度为代价的;Rich算法固定产生K层路由子层,每层保护链路数相当,与Minimum算法和Sparse算法相比较,易于生成备份路径更短的路由子层方案,但是

收稿日期: 2013-05-24; 修回日期: 2014-03-10

基金项目: 国家自然科学基金(61003252,61201209); 全军军事学研究生课题(2011JY002-524,2012JY002-563)

作者简介: 伍文(1985-),女,博士生,主要从事网络可生存性方面的研究。

Rich算法生成目的简单,没有包含优化的思想,还有很大的提升空间。

为了最大限度地解决路径次优问题,缩短重路由路径长度,本文提出了一种以生成平均最短重路由路径为目的的弹性路由层优化算法(shortest rerouting path resilient routing layers optimization algorithm, SRPRO)。在介绍了RRL理论基本原理的基础上,给出了SRPRO算法的核心优化思想和基于单亲遗传算法的求解步骤,最后通过仿真验证了算法的合理性和可行性。

1 RRL基本原理

弹性路由层可以归类为一种特殊的多拓扑路由方式^[10-11],其基本思想为:以IP层全拓扑为基础建立若干个全连接生成子拓扑,这些生成子拓扑称为“层”,每一层中不包含一定的链路和节点,从而达到对这些链路或节点进行保护的的目的,当故障发生时直接将受影响的流量转移到安全的子拓扑中进行传输^[7]。

图1给出了弹性路由层生成的一个例子,仅以保护链路为目标。对全拓扑建立了3个弹性路由层对所有链路进行保护,每一层分别保护3条链路,如图1中层1、层2、层3所示。当网络在正常状态时,从节点1~7的分组通过最短路径1→3→6→7进行传输。当链路1→3故障时,节点1首先发现故障,不需要将故障信息通知其他节点,而是直接将传输分组切换到layer1中继续传输,并在IP分组头部标注当前所在路由层,其他节点也按照该层的路由信息传递这些被切换的分组,重路由的路径为1→2→5→7。当链路3→6故障时,分组传输到节点3时被切换到layer1进行传输,重路由路径为3→4→6→7。当链路6→7故障时,分组传输到节点6时被切换到layer3中进行传输,重路由路径为6→4→5→7。未受故障影响的分组则继续在原全拓扑中传输。

网络中每一个链路或节点都至少对应一个弹性路由层,为了减少路由层数及路由切换的复杂性,一般单个链路或节点由单个路由层进行保护。一个弹性路由层可对多个链路或节点进行保护,每一层保护的元素越多,则网络结构越稀疏,备份路径越长;保护的元素越少,则弹性路由层数越多,备份路由表中需要保存的信息越多。具体选择多少层及保护链路或节点在每一层的分布则根据实际情况进行确定。RRL的生成可通过人工手动完成,也可通过算法自动完成。通过算法生成RRL速度快,可考

虑更多的因素,得到的RRL更优,更加适合于大型网络。

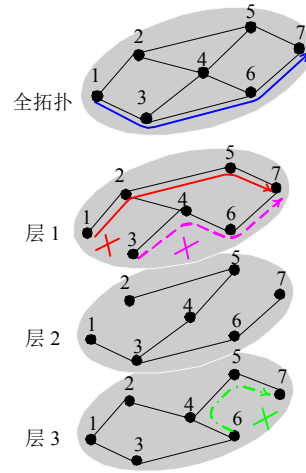


图1 RRL示例

2 SRPRO算法

给定一个网络拓扑,按照RRL的要求可以有多种RRL生成结果,每一种RRL结构差异很大,寻找到重路由路径最短的RRL结构作为流量转发的依据将大大缩短重路由路径长度,提高数据的转发效率。当弹性路由层数等于保护的链路数时,重路由路径最短,但是该情况下路由子层的层数也最多,过多的层数会严重消耗存储和计算资源。文献[9]指出,对于大型网络,4~5层的路由子层可以满足应用需求,因此在进行路由子层寻优中,会首先限定路由子层的层数。寻找到该限定层数下的最佳RRL结构作为RRL生成算法的最终目标。

2.1 RRL的矩阵表示

RRL理论在创建之初是以图的形式表示的,虽然形象但不便于分析与计算,因此首先给出RRL的矩阵表示形式。

网络IP层拓扑是由节点和链路构成的无向图,记为 $G(V, E)$,其中, V 表示顶点(对应拓扑中的节点)的集合, E 表示边(对应拓扑中的链路)的集合。若图的顶点个数为 n ,则可表示为一个 $n \times n$ 的矩阵,用 $D = (d_{ij})_{n \times n}$ 表示,其中:

$$d_{ij} = \begin{cases} 1 & \text{如果 } (i, j) \in E \\ 0 & \text{其他} \end{cases} \quad (1)$$

式中, (i, j) 表示连接顶点 i 和 j 的一条边。

定义 1 若拓扑矩阵 D_r 与 D_1 、 D_2 、 \dots 、 D_l 满足如下关系:

$$D_r = \frac{1}{l-1} \sum_{k=1}^l D_k \quad (2)$$

且满足矩阵 $\sum_{i=1}^{n-1} \mathbf{D}_k^i$ 中的元素全部为非零元素(该条件表示各拓扑子层必须是连通的, n 为矩阵的阶数), 则矩阵 \mathbf{D}_1 、 \mathbf{D}_2 、 \dots 、 \mathbf{D}_l 所表示的网络拓扑组合为矩阵 \mathbf{D}_F 所表示的网络拓扑生成的一组弹性路由层。其中, l 为拓扑子层的层数。

2.2 RRL优化问题

最短路径矩阵 $\mathbf{P} = (p_{ij})_{n \times n}$ 表示一个 n 阶图中各顶点间的最短路径, 其中, 元素 p_{ij} 为顶点 i 与顶点 j 之间的最少边数。第 k 个路由层与全拓扑的最短路径差值矩阵为:

$$\Delta \mathbf{P}_k = \mathbf{P}_k - \mathbf{P}_{FT} \quad (3)$$

式中, \mathbf{P}_k 为第 k 个路由层的最短路径矩阵; \mathbf{P}_{FT} 为全拓扑的最短路径矩阵。

定义 l 个矩阵 $\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_l$, 其中, 第 k 个矩阵 $\mathbf{A}_k = (a_{ij})_{n \times n}$, 若原拓扑中节点 i 到节点 j 最短路径上的第一条链路 (i, i_{next}) 包含于弹性拓扑子层 \mathbf{D}_k , 则 $a_{ij} = 1$, 其他情况则 $a_{ij} = 0$ 。

RRL重路由的路径增加矩阵 $\Delta \mathbf{P}$ 为:

$$\Delta \mathbf{P} = (\Delta p_{ij})_{n \times n} = \sum_{k=1}^l \mathbf{A}_k * \Delta \mathbf{P}_k \quad (4)$$

式中, 符号 $*$ 表示矩阵对应元素相乘。

则, 重路由路径增加值为:

$$\Delta s = \sum_{i=1}^n \sum_{j=1}^n \Delta p_{ij} \quad (5)$$

式(5)给出的重路由路径增加值均等考虑了网络上每条路径的重路由路径增加情况。但是在实际应用中, 每条路径的重要性是不同的, 如有些链路相比其他链路更易损毁, 有些链路的利用率更高一些, 那么这些易损毁的以及利用率更高的链路就显得更重要, 赋予每条路径不同的重要度权重, 加权得到的重路由路径增加值能够反映出RRL对更重要链路的保护程度。

定义路径重要度矩阵 $\mathbf{B} = (b_{ij})_{n \times n}$, 其中, b_{ij} 为原拓扑中节点 i 到节点 j 最短路径上的第一条链路 (i, i_{next}) 的重要度。则路径重要度加权的重路由路径增加矩阵为:

$$\mathbf{Q} = (q_{ij})_{n \times n} = \mathbf{B} * \Delta \mathbf{P} \quad (6)$$

则路径重要度加权的重路由路径增加值为:

$$\Delta s_B = \sum_{i=1}^n \sum_{j=1}^n q_{ij} \quad (7)$$

对RRL结构的优化可以归纳为以下规划问题:

$$\min \quad \Delta s \text{ or } \Delta s_B \quad (8)$$

$$\text{s.t.} \quad l \leq L \quad (9)$$

$$\mathbf{D}_F = \frac{1}{l-1} \sum_{k=1}^l \mathbf{D}_k \quad (10)$$

$$\sum_{i=1}^{n-1} \mathbf{D}_k^i = (d_{kij})_{n \times n} \cap \prod_{i=1}^n \prod_{j=1}^n d_{kij} \neq 0 \quad (11)$$

$$\mathbf{D}_F - \mathbf{D}_k = (d_{(F-k)ij})_{n \times n} \cap \sum_{i=1}^n \sum_{j=1}^n d_{(F-k)ij} \leq \text{ceil} \left(\frac{\sum_{i=1}^n \sum_{j=1}^n d_{Fij}}{l} \right) \quad (12)$$

$$(k = 1, 2, \dots, l)$$

上述规划问题中, 目标函数为最小化重路由路径增加值 Δs 或者路径重要度加权的重路由路径增加值 Δs_B , 具体选择哪一个取决于是否考虑路径重要度问题; 条件式(9)约束了生成的弹性路由层的最大层数 L ; 条件式(10)和式(11)根据定义1给出, 确保了生成的路由层为弹性路由层; 条件式(12)的给出使得每一个弹性路由层保护的链路数相当, 不会差别太大, 这样更有利于生成恢复路径短的重路由弹性路由层结果。

2.3 基于单亲遗传算法的SRPRO算法

上一节描述的规划问题为NP-完全问题, 通过常规方法很难得到精确解, 因此采用一种启发式智能算法——遗传算法对其进行求解^[12]。将上述规划问题的问题空间映射到遗传算法的表达空间, 发现单亲遗传算法适合求解该问题。

首先对应用遗传算法求解RRL问题中的编码方法、适应度函数及遗传算子的选择进行讨论。

2.3.1 编码方法

采用有序串编码方法进行编码。将原拓扑中的每条链路用符号①、②、③……进行编号, 不同的弹性路由层生成结果可以看成这些链路在不同弹性路由层的分布, 这些链路分布在哪个子层, 则表示该子层对这些链路进行保护。确定每一个路由层的保护链路数后, 将编码符号以块的形式依次排列, 每一块表示一个弹性路由层的保护链路情况。根据规划问题条件式(12)的要求, 这些被保护链路在各子层平均分布, 因此每一个块内的链路数目基本相同, 仅可能有一个编码块位数与其他不同。

2.3.2 适应度函数

适应度是个体之间进行比较选择的依据, 在RRL生成规划问题的遗传算法求解中, 个体的重路由路径增加值 Δs 或者路径重要度加权的重路由路

径增加值 Δs_B 越小, 则认为适应度越高, 因此适应度函数为:

$$f = \begin{cases} \frac{1}{\Delta s} & \text{不考虑链路重要度} \\ \frac{1}{\Delta s_B} & \text{考虑链路重要度} \end{cases} \quad (13)$$

2.3.3 遗传算子

1) 交叉算子

交叉算子是为了产生新的个体, 扩大搜索范围。本文中个体编码采取的是有序串编码方式, 若采用传统的遗传操作将产生非法的后代, 因此交叉操作采用单亲遗传的方式进行。每一个个体的交叉操作仅由单个个体完成, 采取基因换位的方式进行。

为了原来群体中的优良特性能够在一定程度上保存, 同时又使得算法能够更快探索新的基因空间, 一并采用单点换位和两点换位两种算子。具体选取方式如下:

- 1) 若 $\frac{f_i}{\sum f_i} > \frac{1}{n}$, 则进行单点基因换位;
- 2) 若 $\frac{f_i}{\sum f_i} \leq \frac{1}{n}$, 则进行两点基因换位。

方式1)中, 第 i 个个体的适应度 f_i 在整个个体适应度总和中所占比例大于平均值, 则说明该个体在群体中具有更好的适应性, 为了尽可能保存该个体中的优良基因, 对其采取单点基因换位; 方式2)中, 个体在群体中的适应性较差, 则采用两点基因换位, 促使更快的更新基因图谱。

单点基因换位和两点基因换位在实际操作中要避免在同一个编码块中进行, 因为这样得到的新的基因与原基因在本文应用中是同一个个体。单点基因换位具体操作方法为: 从基因块中任意选取两块, 再从选出的两个基因块中任意选取两个基因位进行交换。两点基因换位则再一次重复上述过程。

2) 选择算子

选择是通过竞争来实现的, 采取父子竞争选择模式, 经过家庭竞争和社会竞争两轮竞争来完成选择。父代个体首先繁衍出若干子代个体形成一个家庭, 这些子代个体与父代个体一起进行竞争选择, 只存留少量适应度高的个体, 这一步称为家庭竞争。各个家庭竞争存留下的个体再展开社会竞争, 淘汰大多数适应度低的个体, 只存活少数适应度高的个体, 这些存活的个体就构成了新一代的群体。

采取两轮竞争选择是为了通过调节繁殖个体的

个数及家庭选择的数目来控制算法的搜索能力和收敛速度。每一个父代个体繁殖的子代个体数目越多, 则算法的搜索能力越强, 若直接从子代个体中选出适应度好的若干个体, 算法收敛速度快, 容易陷入局部最优, 因此限制从每个家庭中选择个体的数目, 限制的个数越少, 算法收敛越慢。社会竞争选择出与群体 i 数目一致的新的群体 $i+1$, 再循环进行新一轮选择直至找到符合条件的解。

2.3.4 算法步骤

下面给出单亲遗传算法的步骤:

1) 令进化代数 $g=0$, 按照编码规则给出初始群体, 群体中个体数目为 n_0 , 其中 n_0 取值越大, 则算法的搜索能力越强, 但同时也会使得算法的计算量成倍增加, n_0 的合适取值根据网络拓扑复杂度进行确定。

2) 按照2.3.2节给出的适应度函数计算每个个体的适应度, 再根据本文给出的交叉算子对每一个个体进行繁殖, 每繁殖一个子代个体按照定义1中给出的验证连通性的方法对该个体进行连通性验证, 若不满足则淘汰, 直到繁殖的个数达到子代个体数目总值 n_p , 其中 n_p 的取值也根据网络拓扑复杂度进行确定。

3) 按照2.3.3节给出 n_p 的选择算子选择出新一代群体, 包含个体数目为 n_0 。令 $g=g+1$ 。

4) 如果终止条件满足 (g 大于终止代数 G), 则算法结束。否则, 转到步骤2)。

3 仿真与性能分析

为了验证本文所给出的RRL生成方法的正确性和有效性, 选择一个实际的网络NSFNet网络^[13]进行仿真, 如图2所示。

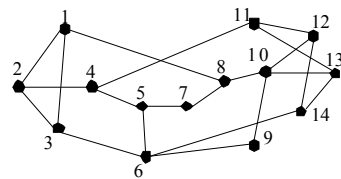


图2 NSFNet网络拓扑结构

以生成3层RRL为例进行说明, NSFNet网络拓扑包含14个节点, 21条边, 取 $n_0=2$, $n_p=5$ 即可满足迭代要求, 按照2.3.4节给出的算法步骤进行仿真, 仿真过程如图3所示。

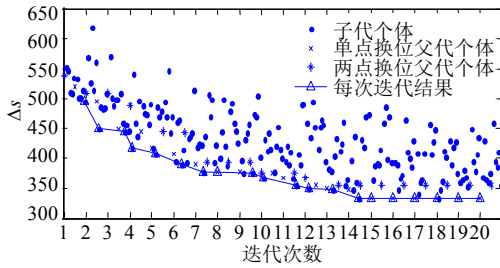


图3 $n_o = 2, n_p = 5$ 时遗传算法仿真过程

从图3中可以看出,遗传算法经过15次迭代后收敛至最小值 $\Delta s = 332$ 。此时对应的RRL结构如图4所示。

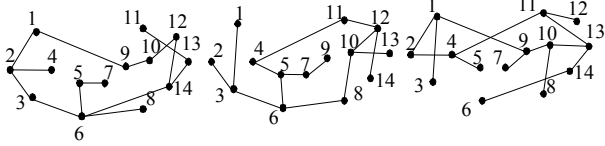


图4 3层最佳RRL结构

变换仿真参数,分别将 n_o 、 n_p 增加1倍,观察迭代进程,令 $n_o = 4, n_p = 10$,再次仿真3层RRL的生成过程,如图5所示。

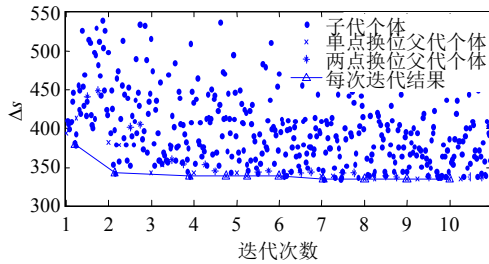


图5 $n_o = 4, n_p = 10$ 时遗传算法仿真过程

从图5可以看出,在迭代4次以后,仿真结果就非常接近最佳值,当迭代到第7次后得到了最小值 $\Delta s = 332$,与参数为 $n_o = 2, n_p = 5$ 的仿真结果一致,但是仿真速度快了一倍左右。这是因为 n_o 、 n_p 值的增加提高了算法的搜索能力,因此会更快搜索到最佳值。

赋予链路重要度权值,如图6所示。

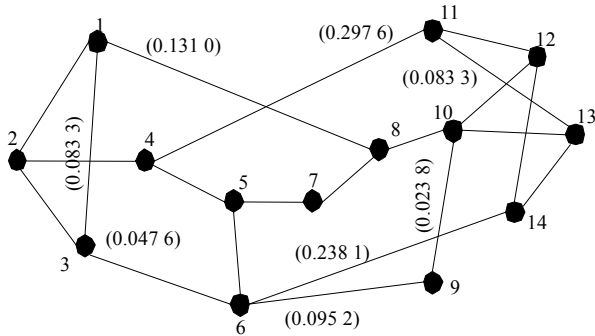


图6 重要度加权的NSFNet网络拓扑结构

假定链路(1,3)、(3,6)等标有权值的链路是网络

中极易损毁的链路,常常需要切换到RRL上来保证流量正常传输,根据其易损毁程度赋予不同的权值,其他未标权重链路则认为可靠性较好,赋予权重0。

取 $n_o = 2, n_p = 5$,得到遗传算法的仿真过程图如图7所示。

从图7可以看出,经过17次迭代,遗传算法便可收敛至最小值 $\Delta s_b = 19.77$,此时对应的RRL结构如图8所示。

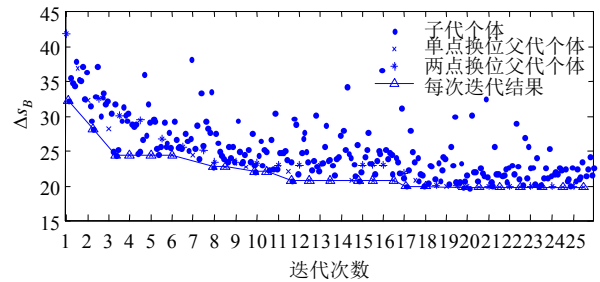


图7 赋予重要权重的遗传算法仿真过程

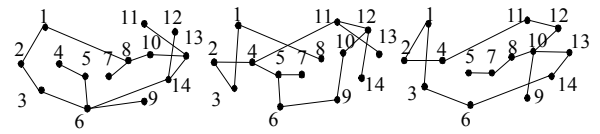


图8 3层加权重最佳RRL结构

比较图4和图8的RRL结构,可以看出两种RRL结构是不同的,但是两种结构又有较大的关联性。这是由于两种结构的目标是一致的:寻找重路由路径最短的RRL结构,只是图8中的RRL结构考虑了链路重要度。

为了验证SRPRO算法的有效性,将SRPRO算法与其他算法进行比较。图9分别给出了SRPRO算法、Minimum算法、Rich算法及全拓扑情况下的 Δs 值。

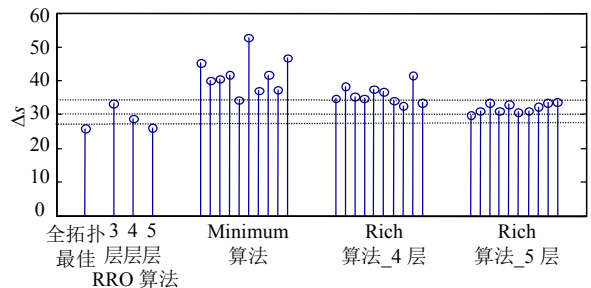


图9 Δs 值比较图

全拓扑情况下的 Δs 值是该网络最佳的重路由路径增加值,与SRPRO算法得到的 Δs 值进行比较,可以看出随着RRL层数的增加,SRPRO算法的 Δs 值越来越接近全拓扑情况下的 Δs 值,当SRPRO算法采用5层RRL结构时,已经与最佳情况相差无几。

Minimum算法得到的是3层RRL,因此与SRPRO算法的3层RRL得到的结果进行比较。可以看出,Minimum算法得到的 Δs 值明显大于SRPRO算法得

到的 Δs 值。

Rich算法是计算特定RRL层数,仿真中分别计算了4层和5层RRL两种结果,与SRPRO算法的4层和5层RRL结果分别进行比较,可以看出SRPRO算法明显优于Rich算法。

4 结束语

本文提出了一种优化RRL重路由路径的弹性路由层生成算法,可以有效改善RRL方法中存在的路径次优问题,提高快速重路由中路径传输效率。首先将RRL的生成过程以矩阵的形式表示,基于此建立了优化RRL生成的数学规划模型,最后采用单亲遗传算法对该模型进行求解。将SRPRO算法与全拓扑最佳重路由结果、Minimum算法、Rich算法进行比较,可以发现SRPRO算法得到的结果非常接近全拓扑最佳结果,且明显优于Minimum算法和Rich算法得到的结果。在下一步工作中,将考虑流量重新转发的负载均衡问题,提高RRL重路由的可靠性。

参 考 文 献

- [1] STERBENZ J P G, HUTCHISON D, CTINKAYA E K, et al. Resilience and survivability in communication networks: strategies, principles, and survey of disciplines[J]. *Computer Networks*, 2010, 54(8): 1245-1265.
- [2] SU Hui-kai. A local fast-reroute mechanism for single node or link protection in hop-by-hop routed networks[J]. *Computer Communications*, 2012, 35(8): 970-979.
- [3] XU Ming-wei, YANG Yuan, LI Qi. Selecting shorter alternate paths for tunnel-based IP fast reroute in linear time[J]. *Computer Networks*, 2012, 56(2): 845-857.
- [4] 苏金树, 胡乔林, 赵宝康. 互联网无中断转发的生存性路由协议[J]. *软件学报*, 2010, 21(7): 1589-1604.
SU Jin-shu, HU Qiao-lin, ZHAO Bao-kang. Disruption-free forwarding survivable routing protocols on internet[J]. *Journal of Software*, 2010, 21(7): 1589-1604.
- [5] 徐明伟, 杨芫, 李琦. 域内自愈路由研究综述[J]. *电子学报*, 2009, 37(12): 2753-2761.
XU Ming wei, YANG Yuan, LI Qi. Survey of intra-domain self-healing routing[J]. *Acta Electronica Sinica*, 2009, 37(12): 2753-2761.
- [6] KVALBEIN A, HANSEN A F, CICIC T, et al. Fast recovery from link failures using resilient routing layers[C]//Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC 2005). Cartagena, Spain: IEEE Press, 2006: 554-560.
- [7] HANSEN A F, KVALBEIN A, CICIC T, et al. Resilient routing layers for recovery in packet networks[C]//Proceedings of the 2005 International Conference on Dependable Systems and Networks. Washington, DC: IEEE Press, 2005: 238-247.
- [8] CICIC T, KVALBEIN A, HANSEN A F, et al. Resilient routing layers and p-cycles: tradeoffs in network fault tolerance[C]//Proceedings of the 2005 Workshop on High Performance Switching and Routing. Hongkong, China: IEEE Press, 2005: 278-282.
- [9] KVALBEIN A, HANSEN A F, CICIC T, et al. Multiple routing configurations for fast IP network recovery[J]. *IEEE/ACM Transactions on Networking*, 2009, 17(2): 473-486.
- [10] QIU Jian, GURUSAMY M, KEE C C, et al. Local restoration with multiple spanning trees in metro ethernet networks[J]. *IEEE/ACM Transactions on Networking*, 2011, 19(2): 602-613.
- [11] CHO S, ELHOURANI T, RAMASUBRAMANIAN S. Independent directed acyclic graphs for resilient multipath routing[J]. *IEEE/ACM Transactions on Networking*, 2012, 20(1): 153-162.
- [12] 熊彦铭, 毛凌, 杨战平. 基于遗传算法的时间决策系统标定优化方法[J]. *电子科技大学学报*, 2012, 41(1): 80-84.
XIONG Yan-min, MAO Lin, YANG Zhan-pin. Novel calibration optimization method for delay-control systems based on genetic algorithm[J]. *Journal of University of Electronic Science and Technology of China*, 2012, 41(1): 80-84.
- [13] LEE S S W, TSENG P K, CHANG C C, et al. A non-weighted load balanced fast local protection scheme for IP networks[C]//Proceedings of the 2010 IEEE Conference on Computer Communications Workshops (INFOCOM 2010). San Diego, CA: IEEE Press, 2010: 1-5.

编辑 蒋晓