

Mesh 网络 p-cycle 的快速配置方法¹

石 兵 闵 凡 周明天

(电子科技大学计算机科学与工程学院 成都 610054)

摘 要: 该文提出了 mesh 冗余网络中快速配置 p-cycle 保护环的方法, 算法从两个方面有效地提高配置速度: 一方面在一次比较搜索过程中, 配置一个多容量 p-cycle 环, 而不是仅配置一个单位容量环; 另一方面, 将已有算法基于圆的搜索策略, 改进为基于通路的搜索策略, 这样可以有效地防止大量的搜索信息汇聚到一个节点, 降低配置速度。当改为通路搜索后, 原来汇聚到一个节点的处理, 可分散到多个相邻节点进行并行处理, 这将有效地加快配置速度。通过仿真表明, 该算法具有同已有算法相当的恢复率, 但所需的搜索趟数性能却能获得大幅度的改善, 同时也不显著增加备用容量的消耗。结果表明网络规模越大, 本文算法的优势越明显。

关键词: mesh(网状) 网络, 预配置环, 评价函数

中图分类号: TN913.2 **文献标识码:** A **文章编号:** 1009-5896(2004)02-0291-07

Pre-configuration-Cycle Fast Configuration Techniques in Mesh Network

Shi Bing Min Fan Zhou Ming-tian

(School of Computer Sci. & Eng., UEST of China, Chengdu 610054, China)

Abstract A new fast distributed p-cycle pre-configuration searching algorithm is proposed in this paper. It can speed up the searching process convergence from two aspects. (1) The multi-p-cycle found by the algorithm within one searching wave should be completed configuration through multi searching waves in classical algorithm. This is the reason why this method faster than classical one. (2) Our algorithm has been changed over from the cycle-based searching to the path-based searching. Using this technique, searching information converged on one node in classical algorithm will be split to its multiple adjacent nodes so reduce the performance requirement of node equipment. The p-cycles searched by the distributed algorithm have been filtered within adjacent node hence the searching process becomes faster. The simulation result also shows that the penalty to spare capacity consumed is very small in our algorithm as well as the obvious benefit on the number of searching wave descent.

Key words Mesh networks, Pre-configuration cycle, Score function

1 前言

现代通信网络, 特别是骨干网络的生存性能已成为网络运营和规划的重要组成部分。目前已有大量文献介绍电信网络生存性问题的研究成果。很久以来一直存在两种基本的网络恢复结构: 环网和 mesh 抗毁网络。基于环的抗毁策略包括: 双向线路倒换环 (BLSRs) 和单向通路倒换环 (UPSRs)。在 UPSRs 中, 每个点到点的业务在其绕环的主用和备用路径中传送。每个接收节点进行 1:1 的选择性接收, 相当于对每个支路具有 1:1 不同路由的保护。在 BLSR 中, 同故障跨段 (节点) 相邻的节点检测到信号丢失后, 便去检测保护信道的状态, 如果空闲, 则将其传输

¹ 2002-08-30 收到, 2003-04-30 改回
电子科学基金资助课题 (51415010101DZ0233)

的信号反向倒换到保护信道上。因为 BLSR 中的备用容量由多个跨段共享使用。所以比 UPSR 有更高的资源利用率。Mesh 抗毁网络一般是基于 DCS(数字交叉系统) 设备作为节点, 通过集中或分布式控制进行业务疏导。经过故障跨段的信号可通过拆分为单个的或成组的信号然后经过不同的备用路由进行恢复, 每条备用通路的容量可以相对较小。每个单位的备用容量依据不同的故障可以共享分配到不同的恢复路由中。而在环网中, 备用带宽只能保护在同一环网中的跨段。Mesh 网络恢复虽然需要更复杂的控制处理, 但它需要的备用容量比 BLSR 要少。环网可以提供更快速 (<50ms) 的恢复时间和更简单的控制机制。Mesh 网络的解决方案则有更高的效率以及更易优化且操作方便。鉴于这两种方式各自的特点, 研究人员提出了将这两种方式进行结合的新思路, 使得 mesh 抗毁网络既具有自愈环网的快速恢复速度又保持传统 mesh 网络的带宽效率。

本文对已有的基于 p-cycle(preconfiguration-cycle) 的恢复策略进行比较, 然后提出一种具有更好运行效率的算法。全文内容安排如下: 先对已有的算法进行简介; 然后对改进的算法进行描述并给出对比实验结果; 最后对全文进行总结。

2 p-cycle 的概念及构造算法

p-cycle 利用 mesh 网络的空闲链路来构造保护环路, 它模仿环型网的自愈机制, 同时保持了网状网的带宽效率。对于某些类型的故障, p-cycle 从整体网络结构考虑进行更广泛的保护。在 p-cycle 恢复机制中, 恢复路由在故障发生前通过程序非实时地预先设置完成, 恢复路由的激活由故障链路端节点 DCS 设备做路由桥接和倒换完成。故障恢复迅速, 控制结构简单。图 1 说明了当单个链路失效时, 如何利用 p-cycle 进行恢复。

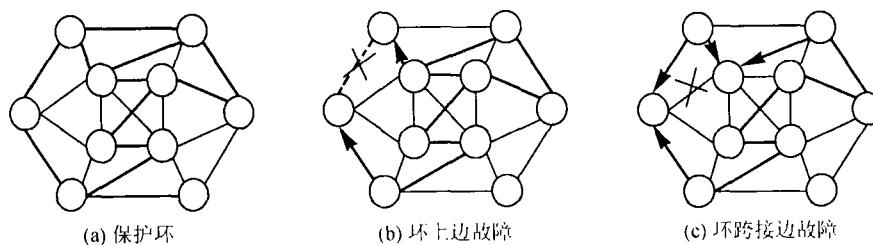


图 1 mesh 网 p-cycle 示意图

从图 1 看出, p-cycle 可为环上故障链路提供一条保护路径, 而对跨接于环上的链路提供两条保护路径。而 BLSRs 和 UPSRs 只能对环上链路进行保护, 因此 p-cycle 方式具有更好的带宽利用率。文献 [1, 2] 中对传统的基于链路恢复的 mesh 抗毁网络的备用容量使用和基于 p-cycle 方式的备用容量的使用通过不同的实验网络进行了验证。结果表明在达到 100% 恢复的前提下, p-cycle 方式所需要的备用容量只比传统的方式高出不到 10%, 网络规模越大, 业务量规模越大, 两者之间的差距越小。从而有力地说明了基于 p-cycle 的网络恢复策略的可行性。对于在已知网络业务分配和备用容量分布的前提下, 文献 [1, 2] 提出了求解在备用容量网络中的 p-cycle 覆盖, 并使得能够达到最大的网络恢复率问题的算法。恢复率定义为在所有预计故障情况下, 通过 p-cycle 恢复的工作通道总数同受到影响的工作通道总数的比率。并使用了如下的 p-cycle 评价函数。评价函数是用于评估不同 p-cycle 恢复性能的指标。文献 [1, 2] 中单位 p-cycle 环的评价函数:

$$S_g = \text{numpaths}/\text{hopcount} \quad (1)$$

式 (1) 中 numpaths 表示为环提供的有效恢复路径数, hopcount 代表环的路径长度 (边数)。对于一个单位容量的 p-cycle 环而言, numpaths=hopcount+2×straddlecount; straddlecount 代表跨接于该 p-cycle 的链路条数。当然, 如果某条环链路上的工作通路数为 0, 则需要从总的 numpaths 中减 1。如果某跨段上的工作通路数 $N < 2$, 则需从 numpaths 中减去 $(2 - N)$ 。详情请参阅文献 [1, 2]。

文献 [1, 2] 同时给出了分布式优化 p-cycle 搜索算法, 算法每次按式 (1) 从备用容量网络中选择一个单位容量的最优 p-cycle, 即 S_g 值最大的 p-cycle, 然后循环搜索至没有 p-cycle 存在为止。如果网络的业务分配发生变化, 再重新触发 p-cycle 搜索构成算法, 在全网范围内重构恢复结构。此算法在寻找一个 p-cycle 的过程中, 对所有的节点进行遍历, 每一节点顺序计算通过本节点的最优环, 然后在全网范围内进行选择。此算法的弱点是收敛较慢。但是, 一旦保护环建立, 故障情况下的恢复速度则很快。

为了克服上述算法的缺点, 下面我们提出一种新的改进算法——快速搜索及配置算法。

3 快速搜索及配置算法

算法描述:

我们重新定义 p-cycle 的评价函数为

$$S_s = \alpha \frac{\text{p-cycle 环可以保护的工作链路容量}}{\text{p-cycle 的环容量} \times \text{环长度}} + (1 - \alpha) \frac{\text{p-cycle 总容量}}{\text{环路上的总备用容量}} \quad (2)$$

如果 S_s 的值越大, p-cycle 越优。表达式的前一项表明要搜索备用容量利用率尽量高的环, 而后一项希望一次搜索可以配置一个多容量的环, 以便有效降低搜索过程的循环次数。具体的算法描述如下:

```

1 exitSR = true; // 设置循环变量, 判断是否还能寻找到 p-cycle;
2 while( exitSR ) { // 如果上次遍历搜索到 p-cycle 存在, 继续搜索;
3   for(网络中每一节点 K) { // 在全网范围内, 遍历搜索一个最优的 p-cycle;
4     并发搜索从 k 到其邻接点的所有满足条件的通路集合;
5     在 k 的每个邻接点 Nk, 利用 (2) 式对经过  $L_{k,nk}$  p-cycle 环进行筛选;
6     邻接点将筛选结果传送给节点 k, 由 k 节点从侯选环中筛选 s 值最大的 p-cycle;
7   } // 一次遍历结束;
8   在全网范围内发布每个节点选择的最优环信息;
9   由最优环的发起节点对 p-cycle 进行配置;
10 } // 算法结束。

```

在 3~7 步的循环中, 将文献 [1, 2] 中基于环的搜索变成了基于路径的搜索, 这样便将集中到一个节点 k 上的搜索汇聚信息分散到节点 k 的多个邻接点上, 可以有效地降低发起搜索节点 k 的性能要求, 并且各邻接点对搜索到的 p-cycle 进行一次筛选过滤, 便能有效加快一次搜索的速度。另外, 通过 (2) 的定义, 我们在一次搜索中找到的 p-cycle 的容量可能要在文献 [1] 的算法中进行多次循环方能完成。这也有效地减低了算法的遍历次数, 从而有效地加快 p-cycle 的配置, 提高了算法的收敛速度。

搜索发起节点 sender, 将其邻边备用资源不为零的邻接点选为搜索的目标节点 chooser, 构造搜索消息, 并向其经过邻边备用容量不为 0 的邻接点扩散搜索消息。当搜索消息每经过一个非目标节点的中间节点时, 节点便检查其经过的路径, 如果消息已经过此节点, 或消息经过的路径长度超过规定值便丢弃此消息。否则在消息路径中增加此节点的标识, 并检查此节点是否同路径上的其它节点有邻接关系; 如有, 便将此边上工作通路带宽信息加入到消息中。然后,

继续向此节点的邻接点扩散消息。当一个 chooser 节点收到一条以自己为宿节点的消息时,说明存在着一个经过它与 sender 节点邻边的一个备用容量环。

全网范围内的各节点的候选环的比较选择,使用文献 [1, 2] 的方法。通过各节点向全网分别广播候选环的信息,然后各节点按照相同的选择方式,确保在一次候选环比较的过程中,选择到唯一的一个候选环。如果两个候选环的评价函数值相同,则按某种方式对节点的标识进行比较(或其它方法),保留下唯一的优胜环。在此过程中,每个节点对收到的候选环信息同本节点保存的最优环信息进行比较,如果收到的环的评价函数值更大,则节点向其邻接点广播新的候选环信息。

当网络业务层发生改变时,可以触发此过程重新对网络的 p-cycle 集合状态进行改变,重构全网的恢复结构。

4 模拟实验结果

为了评估新的分布式算法的性能,现使用 11 个实例网络进行性能对比,这 11 个网络的特性见表 1。

表 1 实例网络拓扑信息

| 网络编号 | 节点数 | 跨段数 | 平均度数 | 总工作容量 | 总备用容量 |
|------|-----|-----|------|-------|-------|
| 1 | 10 | 22 | 4.40 | 710 | 268 |
| 2 | 14 | 21 | 3.00 | 788 | 591 |
| 3 | 15 | 28 | 3.73 | 912 | 660 |
| 4 | 19 | 40 | 4.21 | 1131 | 543 |
| 5 | 20 | 31 | 3.10 | 1046 | 963 |
| 6 | 20 | 32 | 3.20 | 1046 | 926 |
| 7 | 24 | 43 | 3.58 | 1620 | 1313 |
| 8 | 39 | 58 | 2.97 | 3382 | 3746 |
| 9 | 53 | 79 | 2.98 | 7044 | 9371 |
| 10 | 26 | 42 | 3.23 | 9789 | 6564 |
| 11 | 11 | 23 | 4.18 | 1252 | 615 |

注:总工作容量表示包含在所有跨段中工作链路的总和,总备用容量表示包含在所有跨段中备用链路的总和。

1, 3, 5, 9 号网是应用于文献 [2] 中的网络拓扑,业务量的分布为节点对间的均匀分布,业务量大小分别为: 10, 4, 2, 1 单位。2, 4, 6, 7, 8 号网分别是以下网络: NSF, “European community project model”, ARPANET, “US long hual network”, “Belgian nation transport network”, 节点对间的业务量大小分别为: 4, 3, 2, 2, 1 单位。10, 11 号网分别为从相关文献中获得的网络拓扑信息。1~9 号网的业务通过最短路径进行安排,运用文献 [3] 提供的算法进行规划,使网络的备用容量可对单跨段故障进行 100% 的恢复。在进行 p-cycle 搜索时,将圈的最大长度分别设定为 3~12,以此来考察与恢复率的关系。设定 α 值从 0.0 到 0.9 10 个值,以此考察与达到最优解的关系见图 2~图 4。为方便起见,将文献 [1, 2] 中的算法称为 Grover 算法和 G 算法。本文算法也称为 SB 算法。

图 2 给出了利用文献 [1] 中算法进行 p-cycle 配置后,能够快速恢复的全网恢复率和 p-cycle 跳数之间的关系,基本是随跳数的增加而提高。为了清楚,仅图示了 2, 6, 9, 11 号网络的数据。图 3, 图 4 给出两个实例网络运用本文算法在不同参数条件下所获得的恢复率与文献 [1] 中算法的比较。从图中可以看出,本文算法的恢复率并不一定小于文献 [1] 中算法的恢复率。在不同跳数限制的情况下,本文的最优恢复率与文献 [1] 算法的恢复率相差很少。这也说明了本算法的有效性。在表 2 中给出了所有模拟的结果统计,从中可以得出,在达到相当的网络恢复率的情况下,本算法所需要的循环次数要少很多,并且网络的业务量规模和备用容量网络规模越大,两

种算法所需循环次数的差距越大,越能显示本算法快速收敛的特性。图 5,图 6 显示了在获得循环次数显著减少的情况下,本算法所需占用的备用资源与文献 [1] 中算法的比较。

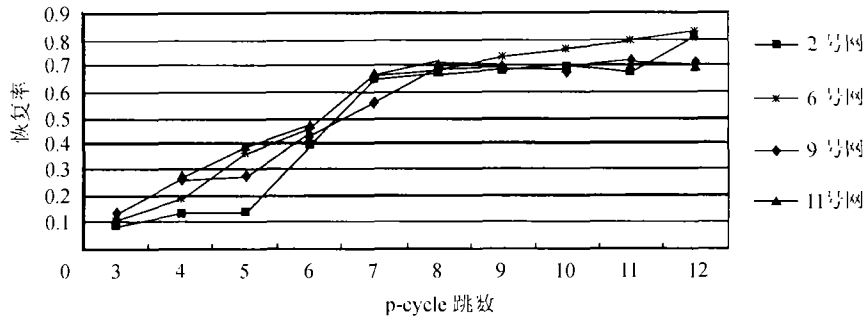


图 2 Grover 算法恢复率与跳数的关系

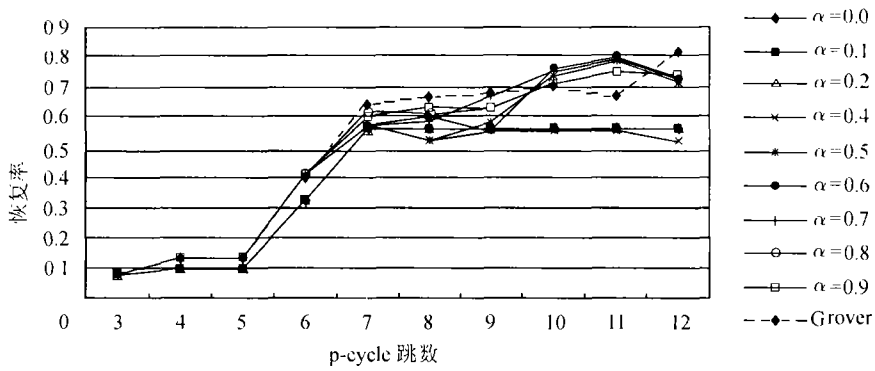


图 3 2 号网, 参数 α 的取值与恢复率之间的关系以及与 Grover 算法的比较

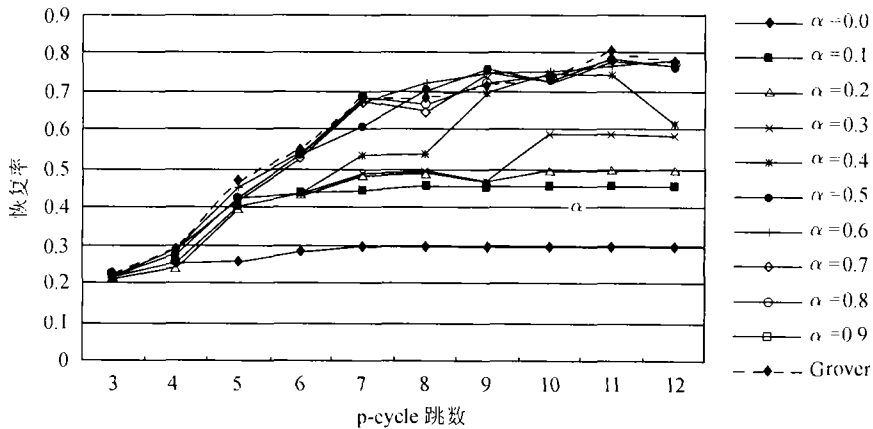


图 4 6 号网, 参数 α 的取值与恢复率之间的关系以及与 Grover 算法的比较

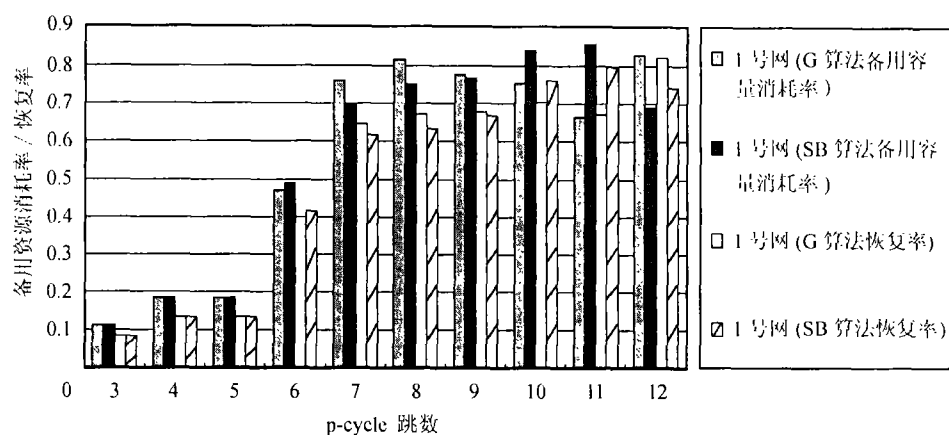


图 5 1号网, 本文算法最大恢复率情况下备用容量消耗率与 Grover 算法的比较

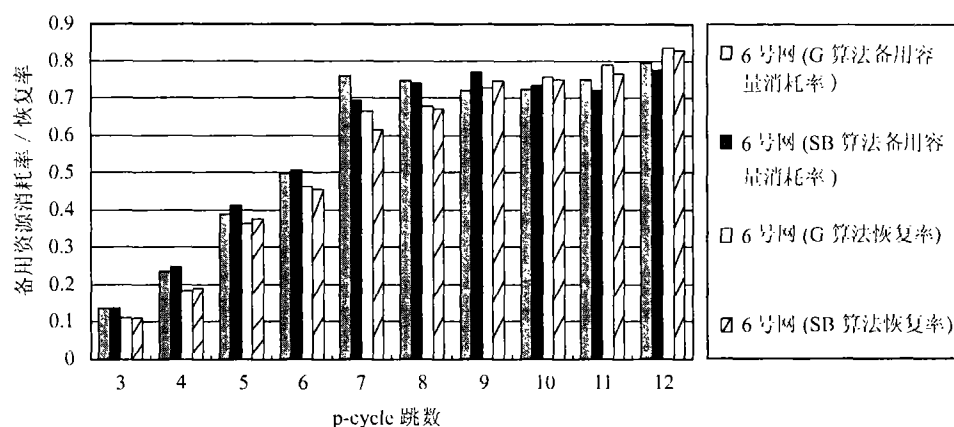


图 6 6号网, 本文算法最大恢复率情况下备用容量消耗率与 Grover 算法的比较

从图 5、图 6 可以看出, 在达到相近程度的恢复率的情况下, 本文算法所使用的备用容量与文献 [1] 中算法大体相当。另外, 在给定 p-cycle 跳数限定的前提条件下, α 值的选取是通过分布式的方式进行协商得到的, 但如果每次网络的业务量发生了小幅度的变化, α 值的选取都经过从 0.0 到 0.9 的逐步取值, 那么本文的算法在循环次数上的优势将不再存在。因此, 需要考察在业务量小幅度的变化情况下, α 值的稳定性。在实验中, 我们对每个网络进行单个业务量的添加和减少进行测试, 发现 α 值是稳定的, 即可以继续使用已有的 α 值对 p-cycle 进行重新配置。

5 结论

如何利用 mesh 网络的空闲链路来构造保护环, 使其模仿环型网的自愈机制, 具有自愈环的恢复速度, 同时又保持 mesh 网络的带宽效率, 是网络恢复技术研究的一个热点领域。p-cycle 方式可提供更广泛的保护。本文提出的在 mesh 备用容量网络中搜索优化 p-cycle 的算法, 在保持文献 [1] 算法的恢复效率的前提下, 具有更好的算法收敛性。

表 2 Grover 算法与本文算法性能比较

| 网络 编号 | Grover 算法恢复率 (上) 与本文算法最高恢复率 (下) 跳数 3~12 | Grover 算法 循环次数 | 本算法 循环次数 |
|----------|--|-------------------|-------------|
| 1 | 0.139 0.291 0.408 0.442 0.569 0.680 0.736 0.908 0.908 0.908 0.126 0.287 0.405 0.498 0.509 0.566 0.616 0.812 0.812 0.812 | 22-38 | 5-10 |
| 2 | 0.083 0.135 0.135 0.399 0.644 0.670 0.678 0.703 0.670 0.819 0.083 0.135 0.135 0.414 0.615 0.634 0.752 0.760 0.796 0.742 | 22-72 | 2-15 |
| 3 | 0.192 0.246 0.456 0.502 0.547 0.542 0.667 0.626 0.641 0.748 0.192 0.243 0.447 0.529 0.542 0.558 0.667 0.691 0.698 0.729 | 47-82 | 6-19 |
| 4 | 0.188 0.321 0.407 0.527 0.587 0.542 0.633 0.682 0.677 0.748 0.189 0.325 0.418 0.460 0.611 0.621 0.674 0.666 0.663 0.693 | 38-86 | 9-28 |
| 5 | 0.202 0.349 0.401 0.590 0.686 0.731 0.762 0.816 0.816 0.847 0.202 0.349 0.401 0.557 0.698 0.742 0.762 0.747 0.764 0.808 | 73-106 | 7-26 |
| 6 | 0.110 0.184 0.364 0.462 0.664 0.677 0.729 0.758 0.792 0.836 0.110 0.188 0.377 0.454 0.617 0.673 0.746 0.752 0.767 0.828 | 42-109 | 5-26 |
| 7 | 0.226 0.295 0.466 0.551 0.682 0.683 0.720 0.740 0.804 0.777 0.226 0.29 0.451 0.547 0.683 0.722 0.759 0.756 0.785 0.787 | 108-142 | 16-26 |
| 8 | 0.102 0.178 0.213 0.434 0.473 0.735 0.767 0.783 0.782 0.807 0.102 0.175 0.214 0.431 0.482 0.734 0.779 0.790 0.789 0.808 | 141-417 | 12-42 |
| 9 | 0.133 0.261 0.271 0.437 0.556 0.690 0.691 0.685 0.708 0.701 0.133 0.261 0.271 0.437 0.556 0.688 0.688 0.683 0.708 0.697 | 442-1038 | 24-50 |
| 10 | 0.128 0.226 0.346 0.385 0.401 0.414 0.455 0.472 0.516 0.561 0.128 0.226 0.348 0.387 0.424 0.451 0.494 0.479 0.484 0.554 | 417-578 | 11-29 |
| 11 | 0.119 0.273 0.392 0.472 0.664 0.710 0.698 0.698 0.698 0.698 0.119 0.297 0.400 0.482 0.563 0.730 0.730 0.828 0.828 0.828 | 49-84 | 2-13 |

参 考 文 献

- [1] Grover W D, Stamatelakis D. Cycle-oriented distributed preconfiguration: ring-link speed with mesh-like capacity for self-planning network restoration. Proc. ICC'98, Georgia, USA, 1998: 537-543.
- [2] Grover W D, Stamatelakis D. Self-organizing closed path configuration of restoration capacity in broadband mesh transport networks. CCB'98, June 21-24, Ottawa, Canada, 1998: 145-156.
- [3] Grover W D, Bilodeau T D, Venables B D. Near optimal spare capacity planning in a mesh restorable network. Globecom'91, Phoenix, AZ, USA, Dec. 1991: 2007-2012.

石 兵: 男, 1969 年生, 博士生, 目前从事通信网络抗毁性能的研究.

闵 凡: 男, 1973 年生, 博士生, 目前从事分布式计算的研究.

周明天: 男, 1939 年生, 博士生导师, 从事计算机网络、分布式计算、人工智能研究.