

# Internet 路由抖动的抑制算法研究

蔡家鹏, 吴建平, 尹 霞

(清华大学计算机科学与技术系, 北京 100084)

**摘要:**路由抖动指 Internet 上路由的反复声明与撤销现象。路由抖动起源于网络的一个区域, 并迅速在网络中传播, 造成网络的不稳定。升级网络节点的软件和(或)硬件有利于维护网络的稳定性。为此, 网络中的节点依照它们处理路由抖动能力的不同被分类。该文分析了路由抖动的传播规律, 提出并比较了对网络进行有效升级的算法, 给出了网络升级的优先步骤。

**关键词:**路由抖动; 网络拓扑; 抑制算法

## Research on Restrainable Algorithms of Route Flap over Internet

CAI Jiapeng, WU Jianping, YIN Xia

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

**【Abstract】** Route flap refers to the repeated and often redundant advertisement and withdrawal of Internet routes. Route flap, always origins in one area of the network and spreads to others, thus causes instability of the network. Upgrading network nodes with new software and/or hardware will help suppress such instability. Different types of nodes are distinguished according to their abilities to deal with route flap. Simulations are performed to acquire a quantitative evaluation of the spread. Algorithms are given to acquire an efficient upgrading plan.

**【Key words】** Route flap; Internet topology; Restrainable algorithms

路由抖动指 Internet 上路由的反复声明与撤销现象。由于路由的反复声明与撤销(网络可达性反复变化), 会引起路由器反复计算最优路由, 影响路由器的性能, 也占用了一定的网络带宽; 路由抖动还可能在整个 Internet 上传播, 影响整个 Internet 的性能与应用。为了降低路由抖动对 Internet 的影响, 采取了相应的抑制措施, 目的是要降低路由抖动造成的 CPU 处理负担, 避免路由抖动的传播, 与此同时, 不影响稳定路由的收敛及计算<sup>[1]</sup>。

网络中的节点(如路由器和主机)按照它们处理路由抖动能力的不同被分类。某些节点对于路由抖动没有抵抗能力, 将被路由抖动感染并进一步传播数据, 感染网络中更多的节点。另一类节点可以阻止路由抖动的进一步传播。网络中还存在路由抖动产生的源节点。模拟结果表明, 一个源节点产生的路由抖动也能在网络中大片区域中传播, 对网络稳定性造成严重破坏。

抑制路由抖动传播的一种方法是升级网络中节点的软件及(或)硬件系统。由于资金和时间的限制, 对网络中所有节点进行升级是不现实的。为此, 本文提出了一种有效的升级方案。其主要思想是首先升级能将网络分割成不同区域的节点, 使得路由抖动被限制在它产生的区域内。这些升级的节点是图论中定义的点断集<sup>[2]</sup>。

### 1 路由抖动的传播和模拟

为了保证信息交互的效率和可靠性, 网络被高度连通起来。路由协议决定了源地址到目的地址的最优路径。当网络拓扑发生变化时(路由器或是某个连接不存在时), 路由器将在剩余的网络中重新计算最优路径, 保证信息交互的顺畅。

然而, 这样一个高度连通的网络也为路由抖动的传播提供了便利。由于路由的反复声明与撤销, 会引起路由器反复计算最优路由, 显式地或隐式地撤销非最优路由, 声明最优

路由, 会占用大量 CPU 资源, 影响路由器的性能, 也占用了一定的网络带宽。为了降低消耗, 同时保证路由的一致性, 抑制路由抖动的算法被引入。具有这种机制的路由器将抑制路由抖动, 阻止其传播。

由此可知, 网络中的节点可以分类。首先, 网络中包含了路由抖动产生的源节点(Error Source Nodes, ESNs)。不能抵抗路由抖动而会被感染的节点被称为不抵抗节点(Resistless Nodes, RLNs)。能抵制路由抖动并阻止其进一步传播的节点被称为健康节点(Healthy Nodes, HLN)。S)。

为了定量地分析数据的传播规律, 识别影响数据传播的相关因素, 采用了模拟的方法。

模拟的模型是一个二元组  $\langle G, R \rangle$ 。

$G$ : 连通图。 $G=(V, E)$ 。 $V$ 是顶点的集合(例如个人计算机或路由器),  $E$ 是边的集合(例如连接)。对于任意的  $v_i \in V$ ,  $v_i$ 的邻居  $N(v_i) = \{v_j | (v_i, v_j) \in E\}$ 。从  $v_i$ 到  $v_j$ 的数据被记作  $m_{ij}$ 。

$R$ : 通信规则。在路由抖动中, 如果  $v_i, v_j$ 是邻居, 也就是说  $(v_i, v_j) \in E$ , 那么数据可以在  $v_i$ 和  $v_j$ 之间传递。为了避免路由回路, 路由协议禁止路由信息从接收方直接传给该路由信息的发送方。既然路由抖动  $m_{ij}$ 也是一种路由信息,  $v_j$ 就不能把  $m_{ij}$ 直接发给  $v_i$ 。

模拟基于上述模型, 包含了以下步骤:

(1)产生一个包含节点及其连接的连通图(网络拓扑)。

(2)对于图中的节点, 随机选择它们的类型(ESNs、RLNs以及 HLN)。

**基金项目:**国家自然科学基金资助项目(90104002)

**作者简介:**蔡家鹏(1980-), 男, 硕士生, 主研方向: 协议测试, 路由稳定性; 吴建平, 教授、博导; 尹霞, 副教授

**收稿日期:**2006-05-10 **E-mail:** cjp@csnet1.cs.tsinghua.edu.cn

(3)从 ESNs 开始,按照深度优先(或广度优先)进行搜索。如果一个 RLN 接收了路由抖动,它将会被感染并产生形如 ESNs 的行为。

(4)重复(3)直到所有的 ESNs(包括被感染的 RLNs)被遍历。

(5)计算被感染节点,没有被感染的节点以及没有接收到路由抖动节点的百分比。

随机生成网络拓扑。输入有 3 个参数:(1)Node\_num: 网络中的节点总数,表示网络大小。(2)Per\_esn: ESNs 所占的百分比。这些是路由抖动的来源。(3)Per\_hln: HLN 所占的百分比。

在分析路由抖动传播规律时,设置 Per\_hln=0,表示网络中没有可以抑制路由抖动的节点。由于路由抖动的扩散过程可以有一棵生成树,树的任意结点都会收到且仅收到一次路由抖动的包,对于连通图,网络中所有结点都包含在这棵生成树内,因此根据树的结点与边的关系可以知道,对于 N 个结点的树,必然有 N-1 条边,这 N-1 条边就对应 N-1 个路由抖动的包。而网络中的跳数是和生成树的高度相关的,对于一个高度为 D 的生成树,路由抖动在网络中传播的最大跳数就是 D-1。这也说明随着结点的增加,网络的耦合度并没有提高,生成树的高度在线性增加。节点数与跳数的关系见图 1。

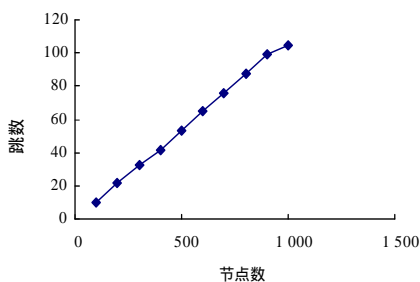


图 1 网络中跳数与结点数关系

随机增加网络中抑制结点的数目,网络中路由抖动的传播参数会发生相应的变化,本文分别研究了网络中包的数目、受感染结点数、收到抖动但没有被感染的结点数、没有收到路由抖动的结点数,以及网络中传播的跳数随着抑制结点的变化规律。首先有以下基本结论:(1)网络中包的数目 + 没有收到路由抖动的结点数 = 网络规模(结点数)-1;(2)网络中包的数目 = 受感染结点数 + 收到路由抖动但没有被感染的结点数。

本文以网络的规模为 400 结点为例,模拟数据结果如图 2 所示。

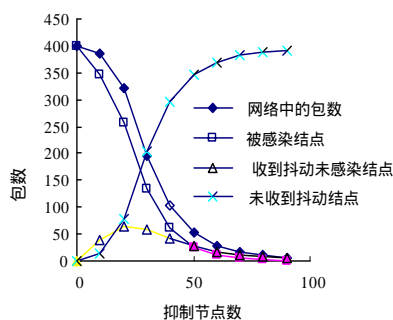


图 2 网络中包的数目和各类结点与抑制结点数目的关系

从图 2 中可看出,网络中包的数目和被感染结点的数量

都是随着抑制结点数的增加而下降的,且在抑制结点比例为 20% ~ 50% 这一区间衰减最快,当网络中抑制结点数达到整个网络结点总数的 50% 时,网路中路由抖动的包可以减少到 15%,基本上达到了抑制路由抖动的目的。而网络中没有收到路由抖动的结点数和网络中包的数目是刚好相反的。此外,通过图 2 中表示网络中收到抖动但没有被感染的结点数曲线可以看出,增加抑制结点并不总是在破坏抖动扩散的生成树的连通性。在抑制结点增加的初期,网络的连通性并未被大量破坏,路由抖动虽然不能够通过抑制结点扩散开,但却可以通过其它非抑制结点绕道扩散给其它结点,使得结点收到感染,而此时抖动传播的跳数是呈现增长的趋势的,即抖动扩散的生成树高度在增加,抖动是以更长的路径传播到了网络的最远端。但当抑制结点达到一定数目(如 20%)后,抑制结点能够较好地破坏图的连通性,生成树的更多分支被抑制结点保护,处于抖动完全不可达的区域,这样抑制结点可以高效地保护其它结点,而网络中未收到路由抖动的结点数也开始迅速上升。这就可以看出在网络中进行结点升级时,升级 20% ~ 50% 这一区域会有较好的收益,这一区间称为有效区间;如果超过 50% 的部分升级,则效果不再十分明显,而如果升级比例低于一定门限(如 20%),则不能很好地起到抑制路由抖动的效果。

## 2 抑制算法

网络管理者关心如何以最有效的方式对网络节点进行升级。或者说,为了维护网络的稳定性,需要升级节点的最小数量是多少?

**定义** 点断集(Set of Vertex Separator, SVS)。G 是一个连通图,S 是 G 顶点的集合。如果  $G \setminus S$  至少分为 2 个连通的子图,或者只有 1 个孤立的顶点,那么 S 是点断集。

由图论知道,如果 HLN 能形成点断集,它们会把路由抖动隔离。因此,最有效、最经济的升级方法是升级网络拓扑中最小的点断集(minimal SVS, MSVS)。为了得到 MSVS,根据图论的定理<sup>[2]</sup>,表示网络拓扑的连通图首先被转化为一个 (0-1) 的网络流图 N,通过计算 N 的最大流量得到 N 的最小切割容量,进而得到 MSVS。

仍以规模为 400 结点的网络为例,并使用图 2 中的网络拓扑,按照最小点断集的方式进行抑制,结果如图 3。

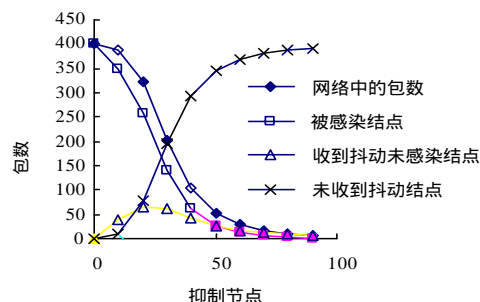


图 3 网络中包的数目和各类结点与抑制结点数目的关系

对于随机生成的网络,图 2 和图 3 二者差异非常微小,这主要是由于随机生成的网络中度都被限制在 1~5 的范围内,因此最大度的算法和随机选择算法取得的抑制结点对网络的抑制效果十分接近,在拟合的关系图上几乎无法分辨。

为此,本文使用实际网络拓扑进行对比,选用 CERNET 的拓扑结构作为研究对象以外输入代替随机生成的网络拓扑,将 CERNET 的拓扑图输入程序,然后分别选择 MSVS

的算法和随机选择的算法,得到网络中的参数对比见图4所示。通过对比最大度和随机选择两种算法得到的参数曲线,可以看到对于实际的网络,随机选择的算法是不可取的,几乎很难达到抑制效果。而实际网络的拓扑通常都是环状和星型的拓扑结构的结合,最大度的算法对这种拓扑结构很有针对性,几乎每升级一个节点都会带来很好的抑制效果。而且实际网络的结构中有大量的叶结点,我们不应忽视这一事实,随机选择时选择叶结点的几率是相当大的,而这些结点作为抑制结点是完全没有意义的,这也是随机选择表现出的效果较差的一个原因。

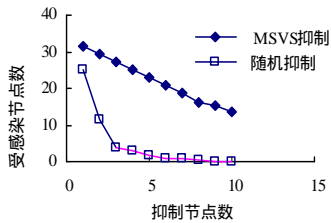


图4 受感染节点数与抑制节点数的关系

### 3 相关工作和结论

我们在路由抖动方面的测试引发了本文工作的研究。在此之前,有关路由抖动危害及其来源的文章陆续发表。在骨干网上进行的测试表明,很多路由器中抖动抑制的选项并未打开。原因在于抑制路由抖动需要路由器为每一条路由维护历史信息,并每隔一定的时间计算路由的“信誉值”(Merit Value)。这样将消耗路由器的CPU和内存资源。由于路由抖动的传播和网络中其它有害数据(如计算机病毒)的传播具有相同的特性,这样促使我们建立一个有效升级网络的方案。

(上接第127页)

到稳态时的结果,从第3类业务第10000个包被处理完时开始统计,统计达到第20000个包被处理完,得到每个包的平均时延、平均服务时间和平均等待时间。考虑到包产生和处理时间的随机性,将上述这一过程再重复10次,然后将得到的时间再次平均,重新得到上述的3个时间。表2给出了仿真得出时间和用平均剩余容量法算出时间的比较。

表2 仿真得出时间和平均剩余容量法算出时间

a1	1020	1080	1200	1320	1380	1500
$\lambda_1$	17	18	20	22	23	25
a2	600	600	600	600	600	600
$\lambda_2$	10	10	10	10	10	10
$\lambda_3$	25	23	20	18	16	15
$t_1$	0.0332277	0.0353921	0.0398605	0.0445157	0.0469135	0.0518497
$t_1'$	0.0293387	0.03280267	0.0365463	0.041716517	0.043886975	0.04832347
$t_3$	0.196255	0.190299	0.19656	0.224015	0.188118	0.233289
$t_3'$	0.20836247	0.19547947	0.1936173	0.21544952	0.19183822	0.21743557
$ t_3 - t_3' /t_3$	6.16925%	2.7224%	1.4971%	3.82363%	1.97759%	6.79565%

表中a1、a2分别是第1类和第2类业务的负荷, $\lambda_1$ 、 $\lambda_2$ 、 $\lambda_3$ 分别为3类业务的到达率, $t_1$ 、 $t_3$ 为平均剩余容量法计算得到的用户的平均处理时间和用户的平均时延, $t_1'$ 、 $t_3'$ 为仿真得到的用户的平均处理时间和用户的平均时延, $|t_3 - t_3'|/t_3$ 为相对误差。

### 5 结束语

从表2很容易看出,无论是用户的平均处理时间还是用户的平均时延,仿真结果与剩余平均信道法中计算出来的结果差别都不大,对平均时延本文还给出了二者的相对误差,

不同的网络节点(除去源节点)对有害数据具有不同的处理能力。一些节点不能抵抗这些数据,被数据感染并进一步传播它们。其它节点可以抑制有害数据,阻止其进一步传播。为了定量地分析路由抖动的传播规律,使用了模拟的方法。结果表明,很少的源节点也会影响网络的很大一部分区域。增加网络中抑制有害数据的节点数目能维护网络的稳定性。然而受资源的限制,升级网络节点是一个困难和漫长的过程。因此,有必要利用有限的资源对网络进行有效的升级。

本文提出了一种有效的升级手段。升级将网络分隔的节点(点断集)能维护网络的稳定性。升级最小的点断集显然是一个有效的方法。为此,本文对比了在随机网络 and 实际网络中分别按照随机方式和MSVS方式进行升级的对比图,结论显示:在限制节点最大度的随机网络中,2种升级方式没有明显差别。但是在实际网络中,由于网络的Hub-Spoke结构,因此最小点断集方式具有明显的优势。

### 参考文献

- 1 Labovitz C, Malan G R, Jahanian F. Internet Routing Instability[C]//Proceedings of the ACM SIGCOMM'97, Nice, France. 1997-08.
- 2 Kurt M. Graph Algorithms and NP-completeness[M]. Berlin: Springer-Verlag, 1984.
- 3 Zhao Y X, Xu K, Wang A P, et al. A Hierarchical Approach to Establish an Efficient Upgrade Plan[C]//Proceedings of IEEE International Conference on Network. 2002.
- 4 Zhao Y X, Xu M W, Yin X, et al. Simulations on the Spread of Route Flap over the Internet[C]//Proceedings of IEEE International Conference on Network. 2002.

最大为6.79%,考虑到仿真序列的随机性,可以认为用平均剩余信道法对数据包的处理时间等排队系统参数进行计算是可行的,较准确的。

下一步的工作将从理论上对随机容量系统其它方面的性能做进一步的研究。

### 参考文献

- 1 Shikh-Bahaei M R, Vamin A H A, Cliara S. Joint Planning of Soft-capacity and Soft-coverage for 3G WCDMA Systems[C]//Proc. of 3G Mobile Communication Technologies. 2001: 78-81.
- 2 Dong In Kim. Selective Relative Best Scheduling for Best-effort Downlink Packet Data[J]. IEEE Transactions on Wireless Communications, 2006, 5(6): 1254-1259.
- 3 Wydrowski B, Zukerman M. QoS in Best-effort Networks[J]. IEEE Communications Magazine, 2002, 40(12): 44-49.
- 4 Chakraborty S, Gries M, Thiele L. Supporting a Low Delay Best-effort Class in the Presence of Real-time Traffic[C]//Proceedings of the 8<sup>th</sup> IEEE Real-time and Embedded Technology and Applications Symposium. 2002: 1-10.
- 5 Kaufman J S. Blocking in a Shared Resource Environment[J]. IEEE Trans. on Commun., 1981, 29(10): 1474-1481.
- 6 李建东. 信息网络理论基础[M]. 西安: 西安电子科技大学出版社, 2001-06: 7-62.
- 7 黄生叶, 叶梧, 冯穗力. 基于准独立近似的巨型资源共享系统呼叫阻塞率的快速算法[J]. 通信学报, 2002, 23(11): 122-128.
- 8 吴新瞻, 吴新垣. 随机模型与计算机模拟[M]. 北京: 电子工业出版社, 1990: 351-360.

