分布式环境下的路由器级互联网抗毁性研究

朱凯龙 陆余良 杨 斌
(中国人民解放军电子工程学院网络工程系 合肥 230037)

摘 要 基于MapReduce分布式计算框架对路由器级互联网拓扑的抗毁性进行研究,从连通性和传输效率两个角度衡量网络的拓扑抗毁性,提出了两个抗毁性新测度:网络连通率和网络传输效率比。基于MapReduce设计并实现了互联网抗毁性分析算法(AIIMR),算法在分布式环境下采用不同的攻击策略对互联网拓扑进行仿真攻击。实验对比分析了传统测度在衡量路由器网络时存在的问题,证明了所提测度的有效性。在不同网络上的实验结果表明,路由器级互联网在遭受随机攻击时表现出很强的抗毁性,而在面对蓄意攻击时则表现得十分脆弱。最后,在不同规模的Hadoop集群上进行实验,结果验证了算法的高效性和扩展性。

关键词 路由器级Internet,抗毁性,分布式计算,AIIMR
中图法分类号 TP301.6 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2017.11.025

Study on Invulnerability of Router-level Internet Based on MapReduce

ZHU Kai-long LU Yu-liang YANG Bin
(Department of Network Engineering, Electric Engineering Institute of PLA, Hefei 230037, China)

Abstract We used MapReduce to study the invulnerability of router-level Internet. In order to evaluate network invulnerability, two new measures of invulnerability were proposed: network connectivity ratio and efficiency ratio. We designed the analysis algorithm of the Internet invulnerability based on MapReduce (AIIMR) using the two new measures. The algorithm simulates the different attack on the Internet. The experimental results show the problem of traditional measure in measuring the router network and prove the validity of the proposed measure. The experimental results also show that the attacked router-level Internet is robust at random attack and fragile at targeted attack. Finally, the experiment proves that the proposed MapReduce algorithm effectively improves the efficiency, and has good scalability.

Keywords Router-level Internet, Invulnerability, Distributed computing, AIIMR

1 引言

随着网络覆盖范围的不断扩大,互联网在人类社会生活中起着越来越重要的作用。如今,互联网服务已经覆盖金融、交通、能源等众多领域[1-4],是世界各国真正的"大动脉"。然而,互联网上每天都会出现各种各样的故障和黑客的攻击,在这种情况下,互联网要承受各种攻击并保持良好的可靠性无疑是一个重要的课题。

互联网是一个由大量路由器连接而成的复杂网络,我们可以使用复杂网络理论对其进行研究。针对复杂网络的抗毁性研究最早始于2000年Albert等[5]的工作,他们比较了ER随机图和BA无标度网络的抗毁性,实验表明无标度网络对随机节点故障具有极高的鲁棒性,而对蓄意攻击具有高度的脆弱性。针对不同领域的复杂网络,研究者们进行了实证分析,Broder等[6]在多个大型万维网的子集合上进行研究,发现若要破坏万维网的连通性则必须删除所有度数大于5的节点。Jeong等[7]研究了蛋白质网络,Dunne等[8]研究了食物链网络,Newman等[9]研究了P2P网络,这些研究发现不同网络都具有相似的抗毁性。

复杂网络抗毁性测度研究是网络抗毁性研究的一个热门课题,研究者们一直致力于寻找能够客观且全面描述网络抗毁性的测度方法。常用的方法包括网络直径[5]、网络平均路径长度[10-12]、最大连通子图大小等。在国内,吴俊等[13]通过综合考虑最大连通子图大小和平均路径长度,提出使用连通系数来测度网络拓扑变化,并在世界贸易网络上进行了实验,取得了较好的效果。最近,他们又提出了复杂网络抗毁性的谱测度方法[14-15],其利用自然连通度来刻画复杂网络的抗毁性,引起了国内外研究者的广泛关注。

在互联网的抗毁性研究中,Holme等[10]所做的工作较为全面,他们不仅考虑了节点攻击的情况,还考虑了边攻击的情况,同时还考虑了基于介数的攻击策略。但是,包括Albert和Holme在内的研究者都仅针对AS级互联网进行了研究,

到稿日期:2016-10-13 返修日期:2016-12-27
朱凯龙(1991-),男,博士生,主要研究方向为计算机网络抗毁性、大规模图数据处理,E-mail:471801698@qq.com;陆余良(1964-),男,教授,博士生导师,主要研究方向为计算机网络安全;杨 斌(1989-),男,博士生,主要研究方向为计算机网络安全。

对路由器级互联网抗毁性的研究还比较少。

相比于 AS 级互联网，路由器级互联网拓扑数据规模巨大、变化速度快，对海量路由器级互联网拓扑数据进行处理是研究路由器级互联网所要解决的问题。同时，网络抗毁性分析算法的计算复杂度较高，单台计算机内存受限，在单机上运行的网络抗毁性分析算法不能有效处理大规模路由器级互联网拓扑数据，因此考虑使用分布式计算框架。Google 提出的 MapReduce[16] 是一种简洁的抽象并行计算模型，Apache 在此基础上实现了著名的 Hadoop 计算平台，该平台已凭借其高可靠性、高扩展性以及开源的优越性成为了分布式计算平台的标准框架。

本文在 MapReduce 分布式计算框架下对互联网拓扑的抗毁性进行研究，主要工作如下：从连通性和传输效率两个方面衡量了网络的抗毁性，提出了两个新的抗毁性测度指标；根据所选测度指标设计了基于 MapReduce 的互联网抗毁性分析算法；利用该算法在互联网实测数据上进行仿真实验，对互联网抗毁性进行了分析，同时验证了算法的性能；最后总结了本文的研究成果和应用价值。

2　MapReduce 计算框架

MapReduce 是运行在大规模集群上的分布式数据处理模型，隐藏了并行化编程的繁琐细节。MapReduce 将编程抽象为 Map 和 Reduce 两个过程，在整个过程中，数据以键/值对的形式进行处理。MapReduce 的处理流程如图 1 所示，分布式计算平台先对大规模数据进行分片，然后为每个分片创建一个 map 任务并进行处理。数据经过 map 函数处理后再进行排序和混排，将键相同的数据发送给同一个 reduce 进行处理，reduce 函数按照用户要求对输入数据进行聚合操作，输出结果键/值对，并将其写入到磁盘中。

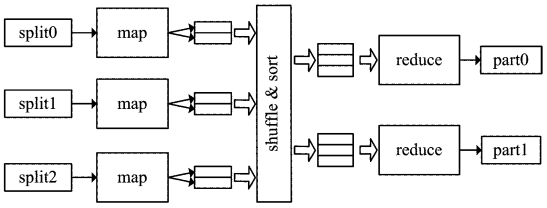


图 1　MapReduce 处理过程

3　网络抗毁性测度

本文从网络连通性和网络传输效率两个方面来测度网络的抗毁性。网络连通性用于评估路由器节点之间的可达性，传输效率用于评估数据在路由器网络中传输的速度。

设原路由器网络 $G=\{V,E\}$ 包含 n 个路由器节点和 m 条链接。从网络中移除 $f\cdot n$（f 为移除节点占原始网络节点数目的比例）个节点后得到网络 $G^f=\{V^f,E^f\}$。

3.1　网络连通性测度

用于衡量网络在遭受攻击时的连通性的指标有很多，文献[12,17]和文献[18]分别采用最大连通分量大小和其余连通分量的平均大小来研究网络在遭受攻击时的分裂过程，取得了很好的效果。参照其定义，本文针对路由器网络定义了以下测度。

网络 G^f 被划分成了 q 个互不相连的连通分量（C_1^f，C_2^f，…，C_q^f），其中 $C_i^f=\{V_i^f,E_i^f\}$。

定义 1（网络核心规模 S）　G^f 中最大连通分量大小与初始网络大小的比值定义为 G^f 的核心规模。

$$S(f)=\frac{\max\{|V_1^f|,|V_2^f|,\cdots,|V_q^f|\}}{|V|} \tag{1}$$

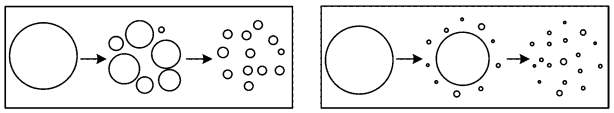
网络核心规模衡量了网络在分裂过程中网络核心部分连通性的保持情况。

定义 2（网络分离粒度 g）　网络 G^f 的分离粒度 g 是指除最大连通分量外的其他连通分量规模的平均值。

$$g(f)=\frac{\sum_{i=1}^{q}|V_i^f|-S(f)}{q-1} \tag{2}$$

网络分离粒度衡量了分裂过程中从网络中分离出去的平均分片的大小。

根据网络核心规模和分离粒度，可以从宏观上来分析网络在遭受攻击时的分裂过程。下面利用这两个参数来描述两种典型的网络分裂情况。1）若网络在遭受攻击时 S 下降得很快，而 g 有较大的波动，则表明网络的核心部分受到了明显的破坏，网络被分割成多个互不相连的子网，如图 2(a)所示；2）若 S 下降较慢，而 g 保持在一个较小的范围内，则说明网络核心部分仍然保持着较好的连通性，节点是以很小的规模从网络中分离出去的，如图 2(b)所示。



(a)　　(b)

图 2　网络分裂的两种过程

在网络遭受攻击时，原本可以通信的节点对可能会断开连接，网络相互连通的节点对数目（可以作为衡量网络连通性的指标）会不断减少。基于连通节点对的数目，下面提出了一个新的测度：网络连通率。

定义 3（网络连通率 ρ）　设网络 G^f 会不断减少中连通的节点对数目为 $p(f)$，则网络连通率定义为：

$$\rho(f)=\frac{p(f)}{p(0)} \tag{3}$$

网络连通率衡量了网络可连通的路由器节点对数目的变化情况，当 $f=0$ 时，ρ 取最大值 1，表明网络连通率没有遭到破坏；随着 f 的增大，ρ 会不断减小，表明越来越多的节点间的通信被切断；当 ρ 取得最小值 0 时，表明路由器间已经无法通信，网络被分割成孤立的节点。

3.2　网络传输效率测度

在攻击过程中存在这样的情况：攻击网络中少量的节点后，网络的连通性没有遭到破坏，但是节点间原有的通信路径不可用，需要选择其他较长的链路进行通信。在这种情况下，网络的传输效率受到影响，但是利用上述连通性测度无法发现网络的性能变化。为了衡量网络性能的这种变化，研究者们基于最短路径提出了一些网络传输效率的测度。设节点 i 和节点 j 之间的距离为 $d(i,j)$，它表示两节点间最短路径的长度。

定义 4(网络直径 D) G^f 的直径 $D(f)$ 是指网络中所有节点对之间距离的最大值。

$$D(f)=\max\{d(i,j)\mid i\neq j\in V^f\} \tag{4}$$

网络直径表示网络中距离最远的两个节点间进行通信所需要经过的跳数,没有使用网络的全部信息,容易受特殊值影响而产生较大的误差。因此,研究者提出了平均路径长度来衡量网络的传输效率。

定义 5(平均路径长度 l) 网络的平均路径长度是指网络 G^f 中所有可达节点间距离的平均值。

$$l(f)=\frac{1}{p(f)}\sum_{i\neq j\in V^f}d(i,j) \tag{5}$$

其中,$p(f)$ 为网络中连通节点对的数目。一般情况下,随着网络不断遭到攻击,网络直径和平均路径长度会不断增大,网络的传输效率不断降低。但是,网络直径和平均路径长度的定义都没有考虑网络的连通性,随着攻击节点比例的增加,这两个测度不再适用。

针对以上测度存在的问题,本文基于最短路径长度提出了传输效率比来衡量网络的传输效率。一般而言,节点间的传输效率与节点间的传输距离反相关,使用 $1/d(i,j)$ 表示节点 i 到节点 j 这条链路的传输效率,利用链路传输效率定义网络的传输效率。

定义 6(网络传输效率 E) 将 G^f 中所有节点间的链路传输效率之和定义为网络传输效率 $E(f)$。

$$E(f)=\sum_{i\neq j\in V^f}\frac{1}{d(i,j)} \tag{6}$$

若两节点不可达,则 $d(i,j)\to\infty$,$1/d(i,j)$ 趋于 0。很显然,网络传输效率 E 的大小受网络规模的影响。为对比不同规模网络的传输效率,定义了网络传输效率比。

定义 7(网络传输效率比 γ) 网络传输效率比是指当前网络 G^f 的传输效率与原始网络传输效率的比值。

$$\gamma(f)=\frac{E(f)}{E(0)} \tag{7}$$

网络传输效率比衡量了当前网络相比于初始网络传输效率的变化情况,当 $f=0$ 时,γ 取得最大值 1,表示网络效率未受到影响;随着攻击比例的增加,γ 呈递减趋势变化;当 γ 取得最小值 0 时,说明数据在网络中的传输速度为零,网络已经完全丧失了传输能力。

4 AIIMR 算法

利用上述网络抗毁性测度,本文提出了 AIIMR 算法,采用仿真攻击的方法对路由器网络的拓扑抗毁性进行研究。本节首先给出了算法的流程,然后针对算法的关键步骤给出伪代码。

4.1 算法流程

AIIMR 算法采用不同的攻击策略对网络拓扑进行仿真攻击,通过观察攻击过程中的网络连通性和网络传输效率来衡量网络抗毁性,算法步骤如算法 1 所示。

算法 1 AIIMR 算法

输入:网络拓扑探测数据(文件 A)

输出:移除不同比例的节点后网络的连通性和传输效率测度值(文件 R)

1. 对文件 A 中的数据进行清理,删除探测结果中的错误节点信息,包括探测中出现的私有 IP 地址和为了探测而设计的虚拟 IP 地址,获得文件 B。
2. 根据文件 B 中每条 traceroute 数据得到网络中的边关系,去重合并后获得网络的边集 E(文件 C)。
3. 根据 E 构造网络的拓扑图 G(V,E)的邻接表(文件 D)。
4. 根据攻击策略移除网络中的 $\Delta f\cdot n$ 个节点后获得文件 E,其中 $\Delta f\in(0,1)$,表示每次攻击的粒度。
5. 计算当前网络的网络核心规模 S(f)、网络分离粒度 g(f)、网络连通率 $\rho(f)$ 和网络传输效率比 $\gamma(f)$,并将结果添加到文件 R 中。
6. 若网络中还有存活节点,则跳转到第 4 步继续执行,否则算法结束,输出文件 R。

算法中使用的网络拓扑探测数据是来自 CAIDA 对全球路由器进行主动探测的数据。算法所使用的攻击策略包括随机攻击和蓄意攻击。随机攻击策略是从网络中随机挑选节点进行移除,蓄意攻击是针对性地移除网络中度较大的节点。算法中第 4 步(计算网络抗毁性测度)是算法的核心步骤,下文给出计算过程的伪代码。

4.2 计算网络核心规模和分离粒度

计算网络核心规模和网络分离粒度时首先需要找出网络中所有的连通分量,本文采用标签传播的方法[19]获得网络的连通分量。基于 MapReduce 获取连通分量的算法如算法 2 所示,开始时所有节点以自身 id 为标签,在 Map 阶段向所有邻居节点广播自身标签;在 Reduce 阶段,每个节点从收到的标签中选择最小的标签作为自身标签,完成一轮的标签传递。不停地迭代该 MapReduce 过程,直到网络中所有节点的标签都不再发生变化为止,此时标签相同的节点就属于同一连通分量。

算法 2 获取网络连通分量

```
Mapper ComponentMap(id n,node N)
输入:第 i 轮标签传播后的节点信息(包括邻接关系和节点标签)
输出:节点信息和候选标签
1. Emit(id n,node N);
2. for all m∈N. neighbor do
3.     Emit(id m,n. label);
4. end for
Reducer ComponentReduce(id m,[p1,p2,…])
输入:节点信息和节点候选标签
输出:第 i+1 轮标签传播后的节点信息
1. find node M from [p1,p2,…];
2. for all p∈[p1,p2,…]-M do
3.   if p<M. label
4.     M. label=p;
5.   end if
6. end for
7. Emit(id m,Node M)
```

对获得的连通分量大小进行统计,最大连通分量大小即为网络的核心规模,除去该连通分量外其余分量大小的平均值即为网络的分离粒度。

4.3 计算网络连通率和传输效率比

计算网络连通率和传输效率比的关键步骤是计算网络中所有节点对之间的距离,本文采用所有节点同时开始并发进行宽度遍历的方法求解节点间的距离。求解节点间距离的伪

代码如算法 3 所示，每个节点保存多条记录，每条记录中包含了宽度遍历的源节点及其与该源节点的距离。在 Map 阶段，每个节点进行宽度遍历，将本节点到源节点的距离加 1 后作为更新消息发送给所有邻居节点；在 Reduce 阶段，邻居节点根据收到的消息来更新自己到源节点的距离，完成一轮宽度遍历。不停地迭代 MapReduce 过程，直到所有节点都完成宽度遍历为止，此时每个节点的记录中都包含了其与其他所有节点的距离。

算法 3　计算节点间距离

```
Mapper DistanceMap(id n,node N)
输入:第 i 轮遍历后的节点信息(包括邻接关系、距离源节点的距离)
输出:节点信息和更新距离消息
1. Emit(id n,node N);
2. for all record∈N. neighbor do
3.    if record. distance=round
4.       message. source←record. source;
5.       message. distance←record. distance+1;
6.       Emit(id m,Message message);
7.    end if
8. end for
Reducer DistanceReduce(id n,[p1,p2,…])
输入:节点信息和更新距离消息
输出:第 i+1 轮遍历后的节点信息
1. find node M from [p1,p2,…];
2. for all p∈[p1,p2,…]-M do
3.    if p. source=record. source
4.       record. distance←message. distance;
5. Emit(id m,node M)
```

5　实验及分析

搭建 Hadoop 集群并进行实验，选用的 Hadoop 版本为 2.6.4。集群包括 1 个 master 节点和 16 个 slave 节点，各个节点的配置如表 1 所列。

表 1　Hadoop 计算节点配置

节点	CPU	内存/GB	硬盘	网卡/MB
master	i7-4770 (4 核 3.4GHz)	16	500GB/7200RPM	1000
slave	i7-4770 (4 核 3.4GHz)	4	1TB/7200RPM	1000

实验所使用的是 CAIDA 在 2015 年 8 月的互联网拓扑探测数据[20]。为研究不同规模路由器网络的抗毁性，从全球互联网中选取了 5 个不同大小的自治域网络，这些网络的特征参数如表 2 所列。其中，*ASno* 表示自治域号，n 表示网络中的路由器节点数，m 表示网络的边数，$\langle k\rangle$ 表示路由器节点的平均度，*CLU* 表示路由器节点的平均聚类系数[21]。

表 2　5 个不同规模的网络的特征参数

ID	*ASno*	n	m	$\langle k\rangle$	*CLU*
G_1	5617	99316	99957	2.01	0.0031
G_2	18881	47117	50492	2.14	0.0117
G_3	33287	13852	13985	2.02	0.0021
G_4	17444	4322	4461	2.06	0.0095
G_5	13345	501	510	2.04	0.005

使用 AIIMR 算法对 5 个不同大小的实验网络进行仿真攻击，为减少实验中的偶然误差，所有实验结果均为进行 100 次重复实验的平均值。

5.1　传统测度分析

为分析本文算法中所用测度的优越性，使用网络直径和网络平均路径长度进行对比实验。使用节点度优先攻击对 5 个实验网络进行攻击，记录蓄意攻击条件下网络的直径和平均路径长度，如图 3 和图 4 所示。其中，横坐标 f 表示攻击网络节点的比例。

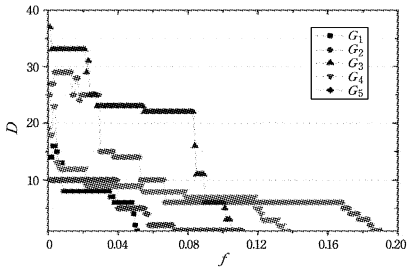


图 3　蓄意攻击下的网络直径

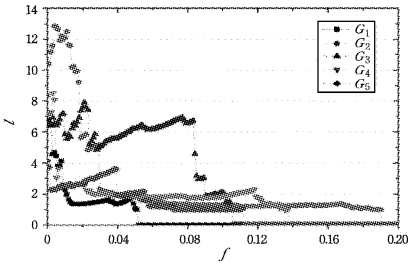


图 4　蓄意攻击下的网络平均路径长度

通过图 3 可以看出，随着网络中攻击节点比例的增加，虽然网络的连通性和传输效率都遭到了破坏，但是各个网络的直径表现出不同的变化规律。这是因为，网络直径仅衡量了网络距离最远的两个节点之间的距离，不能准确地衡量网络整体的拓扑情况。当攻击网络中少量(小于 2%)节点时，若攻击的节点处于距离最远的两个节点之间的路径上，则会导致网络直径的增加，如图中的网络 G_1，G_2 和 G_4 所示；若攻击的节点不在距离最远的两个节点之间的路径上，则网络的直径将保持不变，如图中的 G_5 所示；若攻击的节点导致距离最远的两个节点不连通，网络的直径将由距离较小的另一对节点决定，则网络的直径将减小，如图中的 G_3 所示。攻击条件下网络的直径变化存在多种情况，不能直观地对网络的状态进行描述。此外，攻击过程中存在这样的情况：随着攻击节点比例的增加，网络的直径保持不变。显然，在这种情况下不便于我们对网络的连通性和效率进行分析。

蓄意攻击条件下网络的平路径长度如图 4 所示。可以看出，平均路径长度相比于网络直径有了改进，在攻击过程中网络的平均路径不会出现保持不变的情况。但是，该测度仍然不是单调变化的，不同的网络仍然表现出不同的变化规律。G_1，G_2 和 G_4 的网络平均路径长度先增加后减小，G_3 和 G_5 网络的平均路径长度呈现出震荡变化并最终减小。这是因为网络的平均路径长度受两种情景影响：1)在攻击过程中，网络中的一些节点仍然保持连通，但是它们之间的最短路径发生了变化，这些节点将导致网络的平均路径长度增加；2)网络中一

些节点在攻击过程中变得不再连通，它们之间的距离用 0 来表示，这将导致网络的平均路径减小。当第一种影响大于第二种影响时，网络的平均路径增加；反之，网络的平均路径减小。因此这种非单调的变化也为评价网络的连通性和效率带来了不便。本文所提出的网络连通率和网络传输效率比解决了这个问题。

5.2 网络连通性分析

通过分析网络核心规模和分离粒度可以了解网络在遭到攻击时的分裂过程。采用两种攻击策略对 5 个实验网络进行攻击，结果如图 5 和图 6 所示，图 5 给出了蓄意攻击条件下 5 个网络的核心规模 S 和分离粒度 g 的变化曲线。横坐标 f 表示攻击节点的比例，纵坐标表示测度的取值。其中测度取值小于 1 的曲线是网络的核心规模 S 的变化曲线，测度取值大于 1 的曲线是网络的分离粒度 g 的变化曲线，图中仅显示了网络完全变为孤立节点前的部分。

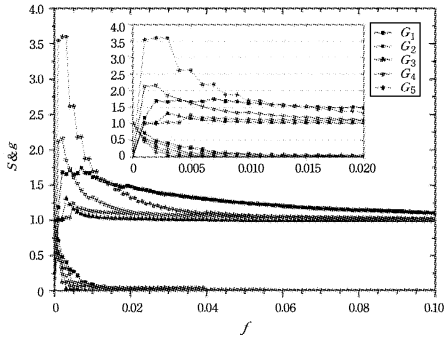


图 5 蓄意攻击下的网络核心规模和分离粒度

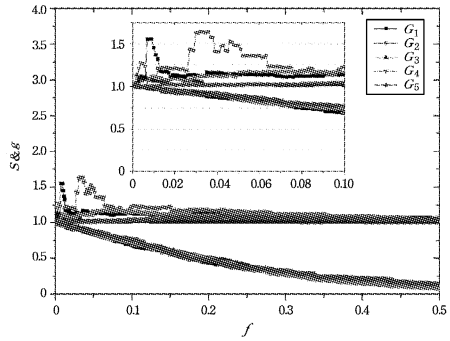


图 6 随机攻击下的网络核心规模和分离粒度

采用蓄意攻击策略的实验结果如图 5 所示。可以发现，当从网络中移除少量度较大的节点后导致了网络核心规模急剧下降，同时，网络的分离粒度不断增大。这种结果表明，有针对性地攻击网络中小部分节点会导致大规模的分量从网络核心中分离出去，使得网络迅速被分裂成为多个分片(见图 2(a))。采用节点度优先的攻击策略时，当移除 0.1%的重要节点后，网络的核心规模缩小了大约一半；当移除比例达到 0.7%之后，网络核心几乎被完全破坏。

采用随机攻击策略的实验结果如图 6 所示。从网络中随机移除部分节点后，网络核心规模线性缓慢减小，网络分离粒度一直保持在较低的水平。这表明，面临随机攻击时近似于将节点逐个从网络核心中分离出去，网络的分裂过程如图 2(b)所示。采用随机攻击策略时，当移除网络中 10%的节点后，网络核心规模仍能保持 70%；若要完全破坏网络的核心规模，则需移除网络中一半以上的节点。

下面进一步分析在攻击过程中网络连通性的变化。采用蓄意攻击策略时，网络连通率的变化情况如图 7 所示，可以发现，蓄意攻击对网络的连通性造成了巨大的破坏，仅移除网络中度最大的 0.1%的节点，网络的连通率就减小到 1/4，这意味着有 75%的节点的之间的链路被切断；当攻击 1%的节点后，网络被分割成为孤立的节点。

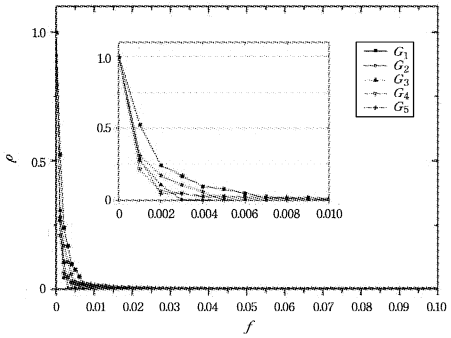


图 7 蓄意攻击下的网络连通率

采用随机攻击时，网络连通率的变化情况如图 8 所示，可以看出，相比于蓄意攻击，面对随机攻击时网络连通率下降得很慢，随机攻击 10%的节点后仍有一半以上的链路能保持通信，当移除比例达到 50%时网络才会被分割成孤立节点。

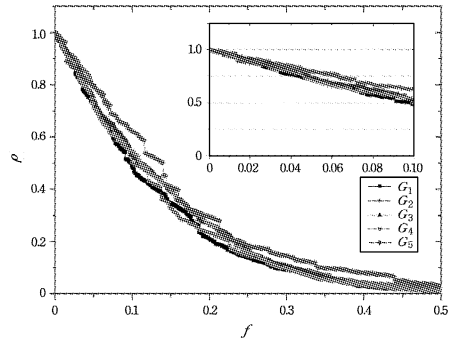


图 8 随机攻击下的网络连通率

5.3 网络传输效率分析

通过分析网络传输效率来了解在面临不同攻击时网络中数据传输速度的变化趋势。面对蓄意攻击时，网络的传输效率比如图 9 所示。可以看出，移除网络中少量重要节点就会对网络的传输速度造成巨大影响，当移除 0.1%的节点时，网络的传输效率就下降了约 60%；当移除 1%的节点时，网络的传输效率几乎为零，这说明节点间无法进行有效的通信。

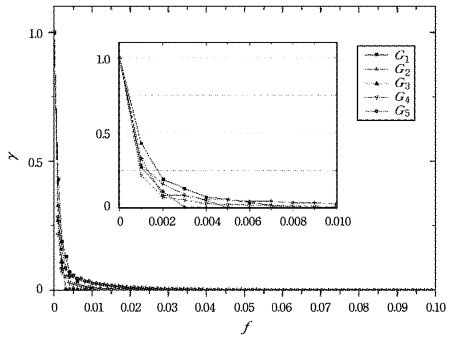


图 9 蓄意攻击下的网络传输效率比

面对随机攻击时，网络传输效率比如图 10 所示，可以看出，随机移除网络的节点对传输效率的影响较小。只有移除 12%以上的节点才能降低网络 50%的传输效率；想要完全破坏网络的传输能力，至少要移除网络中一半的节点。

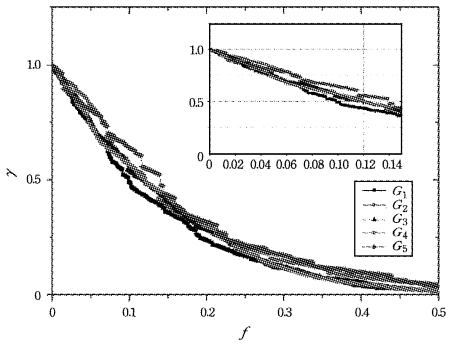


图 10　随机攻击下的网络传输效率比

5.4　算法效率验证

在不同规模的计算集群上执行 AIIMR 算法，并记录算法执行的时间。不同条件下算法的执行时间如图 11 所示，其中 N 表示 Hadoop 集群的计算节点个数。可以看出，对于同一网络，AIIMR 算法的执行时间随着计算节点的增加而明显缩短，尤其是针对大规模的网络时，算法执行时间缩短的幅度更大。

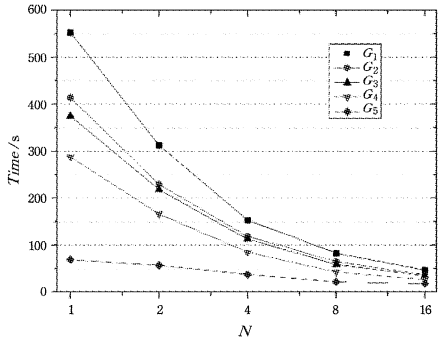


图 11　互联网拓扑脆弱性分析算法的耗时

不同计算集群下算法的加速比[22]如图 12 所示。可以看出，随着计算集群规模的增长，算法的加速比不断增加。当计算节点个数为 16 时，加速比达到了 11.2，计算效率提高了 11 倍以上。实验表明，本文所提算法在 Hadoop 平台上具有良好的扩展性，与传统单机算法相比在效率上有巨大优势。

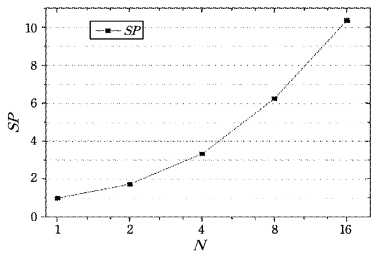


图 12　互联网拓扑脆弱性分析算法的加速比

结束语　本文提出了两个网络抗毁性的测度，在分布式环境下设计并实现了 AIIMR 算法，利用该算法对路由器级互联网抗毁性进行了研究。实验结果表明，不同规模的路由器网络在面对蓄意攻击时都表现出高度的脆弱性，而面对随机攻击时则表现出很强的抗毁性。当蓄意攻击网络中 0.1%的重要节点时，网络的核心规模下降一半，超过 75%的节点对之间丧失了传输能力，网络传输效率降低约 60%，网络发生快速的分裂现象。而少量节点的随机攻击则对网络的性能影响很小。对算法效率分析的实验证明了 AIIMR 算法在 Hadoop 平台上具有良好的扩展性。

本文的结论说明了个别关键节点在网络中的重要地位，可以为提高网络抗毁性提供参考。通过在网络中添加冗余节点或者边来减缓网络在面对攻击时性能的下降速度，并对网络中重要的路由器进行针对性的保护或者提供备份。同时，本文利用分布式计算框架来处理大规模互联网拓扑数据，提升了对网络拓扑数据分析的效率，为互联网拓扑分析提供了新思路。

参 考 文 献

[1] HUANG A Q, CROW M L, HEYDT G T, et al. The future renewable electric energy delivery and management (FREEDM) system: the energy internet [J]. Proceedings of the IEEE, 2011, 99(1): 133-148.

[2] SU W, HUANG A Q. Proposing an electricity market framework for the energy internet [C]// IEEE Power & Energy Society General Meeting. Vancouver, British Columbia, Canada, 2013: 1-5.

[3] HERNANDEZ B, JIMENEZ J, MARTIN M. Customer Behavior in Electronic Commerce: The Moderating Effect of E-purchasing Experience [J]. Journal of Business Research, 2010, 63(9/10): 964-971.

[4] LIU J S, ZHOU C Q, GUO L S. Application of "Internet plus" era of big data technology in military field [J]. National Defense Science & Technology, 2011, 36(6): 35-41. (in Chinese)
刘金山，周朝谦，郭连升. "互联网＋"时代大数据技术在军事领域的应用[J]. 国防科技，2015，36(6)：35-41.

[5] ALBERT R, JEONG H, BARABASI A L. Attack and error tolerance in complex networks [J]. Nature, 2000, 406(6794): 387-482.

[6] BRODER A, KUMAR R, MAGHOUL F, et al. Graph structure in the web [J]. Computer Networks, 2000, 33(1-6): 309-320.

[7] JEONG H, MASON S, BARABASI A L, et al. Lethality and centrality in protein networks [J]. Nature, 2001, 411(6833): 41-42.

[8] DUNNE J A, WILLIAMS R J, MARTINEZ N D. Network structure and biodiversity loss in food webs: robustness increases with connectance [J]. Ecology Letters, 2002, 5(4): 558-567.

[9] NEWMAN M E J, FORREST S, BALTHROP J. Email networks and the spread of computer viruses [J]. Physical Review E Statistical Nonlinear & Soft Matter Physics, 2002, 66(3): 1162-1167.

[10] ZHOU S, MONDRAGON R J. Redunadancy and Robustness of the AS-level Internet topology and its models [J]. Electronics Letters, 2004, 40(2): 151-152.

[11] WANG T, WU L L. Research on invulnerability of urban transit network based on complex network[J]. Application Research of Computers, 2010, 27(11): 4080-4086. (in Chinese)
汪涛，吴琳丽，基于复杂网络的城市公交网络抗毁性分析[J]. 计

算机应用研究,2010,27(11):4080-4086.

[12] HOLME P,KIM B J,YOON C N,et al. Attack vulnerability of complex networks [J]. Physical Review E Statistical Nonlinear & Soft Matter Physics,2002,65(5):634-649.

[13] WU J,TAN Y J. Study on measure of complex network invulnerability [J]. Journal of Systems Engineering,2005,20(2):128-131. (in Chinese)
吴俊,谭跃进. 复杂网络抗毁性测度研究[J]. 系统工程学报,2005,20(2):128-131.

[14] WU J,BARAHANA M,TAN Y J,et al. Natural connectivity of complex networks [J]. Chinese Physics Letters,2010,27(7):078902.

[15] WU J,BARAHANA M,TANY J,et al. Spectral measure of structural robustness in complex networks [J]. IEEE Transactions on Systems Man and Cybernetics Part A,2011,41(6):1244-1252.

[16] DEAN J,GHEMAWAT S. MapReduce:Simplified data processing on large clusters[J]. Communication of the ACM,2008,51(1):107-113.

[17] OUYANG M,FEI Q,YU M. Survey on Efficiency and Vulnerability of Complex Network [J]. Computer Science,2008,35(6):1-4. (in Chinese)
欧阳敏,费奇,余明. 复杂网络的功效性与脆弱性研究综述[J]. 计算机科学,2008,35(6):1-4.

[18] TAN Y J,WU J,DENG H Z. Progress in invulnerability of complex network [J],University of Shanghai for Science and Technology,2011,33(6):643-668. (in Chinese)
谭跃进,吴俊,邓宏钟. 复杂网络抗毁性研究进展[J]. 上海理工大学学报,2011,33(6):643-668.

[19] DU Y H. The research of graph algorithm based on cloud computing [D]. Beijing:Beijing University of Posts and Telecommunications,2011. (in Chinese)
杜雅红. 基于云计算平台的图算法研究[D]. 北京:北京邮电大学计算机学院,2011.

[20] CAIDA. Topology : Macroscopic Internet Topology Data Kit (Ark ITDK;restricted) [EB/OL]. (2015-10-01) [2016-03-15] https://topo-data.caida.org.

[21] 汪小帆,李翔,陈关荣. 复杂网络理论及其应用[M]. 北京:清华大学出版社,2006:10-11.

[22] XIE C,MAI L D,DU Z H,et al. Research and analysis of Parallel computing system speedup [J]. Computer Engineering and Applications,2003,39(26):66-68. (in Chinese)
谢超,麦联叨,都志辉,等. 关于并行计算系统中加速比的研究与分析[J]. 计算机工程与应用,2003,39(26):66-68.

(上接第 163 页)

[17] KOUNEV S. Performance Modeling and Evaluation of Distributed Component-Based Systems Using Queueing Petri Nets[J]. IEEE Trans. on Softw. Eng.,2006,32(7):486-502.

[18] TRIBASTONE M. A fluid model for layered queueing networks [J]. IEEE Trans. on Softw. Eng.,2013,39(6):744-756.

[19] DISTEFANO S,SCARPA M,PULIAFITO A. From UML to Petri Nets:The PCM-Based Methodology[J]. IEEE Trans. on Softw. Eng.,2011,37(1):65-79.

[20] TRIBASTONE M,GILMORE S,HILLSTON J. Scalable Differential Analysis of Process Algebra Models[J]. IEEE Trans. on Softw. Eng.,2012,38(1):205-219.

[21] KOZIOLEK A,ARDAGNA D,MIRANDOLA R. Hybrid multi-attribute QoS optimization in component based software systems [J]. J. Syst. Softw.,2013,86(10):2542-2558.

[22] HUANG X,WANG W,ZHANG W B,et al. Automatic performance modeling approach to performance profiling of Web applications[J]. Journal of Software,2012,23(4):786-801. (in Chinese)
黄翔,王伟,张文博,等. 面向性能剖析的 Web 应用自动性能建模方法[J]. 软件学报,2012,23(4):786-801.

[23] LI C H,WANG W M,SHI Y Y. Performance prediction method for UML software architecture and its automation[J]. Journal of Software,2013,24(7):1512-1528. (in Chinese)
李传煌,王伟明,施银燕. 一种 UML 软件架构性能预测方法及其自动化研究[J]. 软件学报,2013,24(7):1512-1528.

[24] NAVARRO E,CUESTA C E,PERRY D E,et al. Antipatterns for architectural knowledge management[J]. Int. J. Inf. Technol. Decis. Mak.,2013,12(3):547-589.

[25] SMITH C U,WILLIAMS L G. More new software performance antipatterns:Even more ways to shoot yourself in the foot[C]// Computer Measurement Group Conference. 2003:717-725.

[26] MEIER J D,VASIREDDY S,BABBARA,et al. Mackman,Improving. NET Application Performance and Scalability[OL]. http://www.bokus.com/bok/9780735618510/Improving-net-application-performance-and-scalability.

[27] CORTELLESSA V,MARTENS A,REUSSNER R,et al. A process to effectively identify 'guilty' performance antipatterns [C] // Fundamental Approaches to Software Engineering. Springer,2010:368-382.

[28] CORTELLESSA V,DI MARCO A,TRUBIANI C. An approach for modeling and detecting software performance antipatterns based on first-order logics[J]. Softw. Syst. Model.,2014,13(1):391-432.

[29] TRUBIANI C,KOZIOLEK A. Detection and solution of software performance antipatterns in palladio architectural models [J]. ACM SIGSOFT Software Engineering Notes,2011,95(9):19-30.

[30] CORTELLESSA V,FRITTELLA L. A Framework for Automated Generation of Architectural Feedback from Software Performance Analysis[C]//Formal Methods and Stochastic Models for Performance Evaluation. Springer Berlin Heidelberg,2007:171-185.

[31] BACHMANN F,BASS L,BIANCO P,et al. Using ArchE in the Classroom:One Experience[OL]. http://repsitory.cmu.edu/sei/3441.

[32] CADORET F,BORDE E,GARDOLL S,et al. Design Patterns for Rule-Based Refinement of Safety Critical Embedded Systems Models[C]// 17th International Conference on Engineering of Complex Computer Systems (ICECCS). 2012:67-76.

[33] DEB K,PRATAP A,AGARWAL S,et al. A fast and elitist multiobjective genetic algorithm:NSGA-II[J]. IEEE Trans. on Evol. Comput.,2002,6(2):182-197.