基于复杂网络的网络系统脆弱点发现方法研究

赵小林, 徐 浩, 薛静峰*, 宋天凌, 胡晶晶, 闫怀志
北京理工大学 北京 中国 100081

摘要 利用复杂网络寻找网络系统中的脆弱点可以从网络拓扑结构的角度出发, 利用节点的拓扑性质研究其脆弱性, 这可以有效的解决攻击图等脆弱性评估手段无法处理规模过大的网络的问题。通过对李鹏翔等的节点删除方法进行改进, 计算动态删除节点后网络平均最短路径变化, 模拟网络中节点在受到攻击后无法使用, 从而导致的网络整体性能的变化。使得评估时不仅考虑删除节点对网络破坏程度, 同时兼顾了对网络的效率的影响, 从而可以更有效的针对脆弱点布置防御措施。

关键词 复杂网络; 脆弱点; 节点删除法
中图法分类号 TP393.0 DOI 号 10.19363/J.cnki.cn10-1380/tn.2019.01.04

Research on network system vulnerability detection method based on complex network

ZHAO Xiaolin, XU Hao,XUE Jingfeng*, SONG Tianling, HU Jingjing, YAN Huaizhi
Beijing Institute of Technology, Beijing 100081, China

Abstract Using complex networks to find vulnerable points in network systems can be carried out from the perspective of the topology of the network. We can research the vulnerability of nodes based on their topological characteristics. This can effectively solve the problem that the vulnerability assessment methods such as attack graphs cannot handle large-scale networks. Through the improvement of the node deletion method of Li Pengxiang, the average shortest path change of the network after the dynamic deletion of the node is calculated, and the nodes in the simulated network cannot be used after being attacked, thereby causing the change of the overall performance of the network. The evaluation not only considers the degree of network damage caused by deleting nodes, but also takes into account the impact on the efficiency of the network, so that defensive measures can be deployed more effectively against the vulnerable points.

Key words complex network; vulnerable points; the node deletion method

1 序言

随着微型计算机、手机等网络设备的全面普及, 互联网逐渐成为人们日常生活中必不可缺的一部分, 人们已经无法离开网络空间而生存。图 1 是互联网系统协会(Internet Systems Consortium, ISC)统计的互联网接入主机数[1], 该图反映了世界上上网人群的数量, 由图中可以看出, 早在 2014 年, 接入互联网的主机数就已经突破 10 亿, 直到 17 年 7 月, 这个数量还在稳步上涨。2018 年接入主机数为 10.16 亿, 这与 93 年 1 月的 131 万相比, 增长了近 776 倍。与世界互联网的高速发展相呼应, 中国国内的发展同样迅猛, 图 2 是由中国互联网信息中心(China Internet Network Information Center, CNNIC)发布的第 42 次《中国互联网络发展状况统计报告》[2]中给出的网民规模及互联网普及率的统计数据, 截至 2018 年 6 月, 我国网民规模为 8.02 亿, 互联网普及率达到 57.7%, 且网民数量仍呈现稳定高速增长的趋势。

互联网用户的增多必然会引起越来越多的安全问题。因此, 网络空间的安全问题正在越发的引人注目, 网络空间已经成为关乎国家安全的兵家必争之地。如今, 企业和政府大量运用计算机技术和网络技术来构筑业务运行的基础设施, 但是黑客攻击、蠕虫病毒、后门漏洞等安全威胁的存在, 使得用户的关键业务暴露在危险中, 尤其是军工企业的涉密网络更是黑客最为关注的对象。图 3 是由美国国家标准和

通讯作者: 薛静锋, 教授, Email:xuejf@bit.edu.cn。
本课题得到国家重点研发计划项目(No. 2016YFB0800700)资助。
收稿日期: 2018-09-30; 修改日期: 2018-12-04; 定稿日期: 2018-12-14

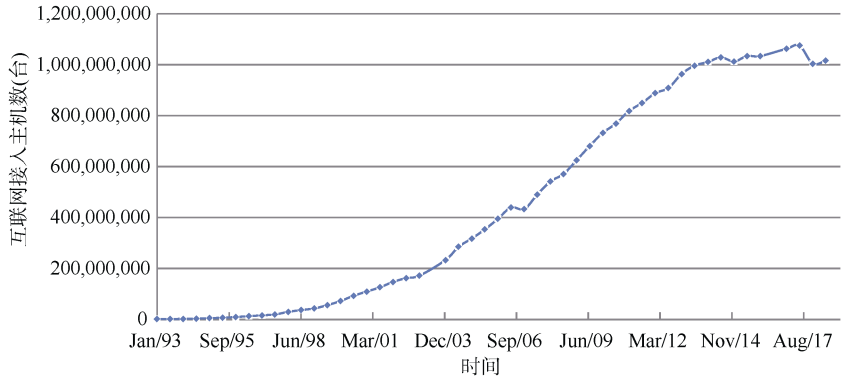

图 1　互联网接入主机数

Figure 1　Number of Internet access hosts

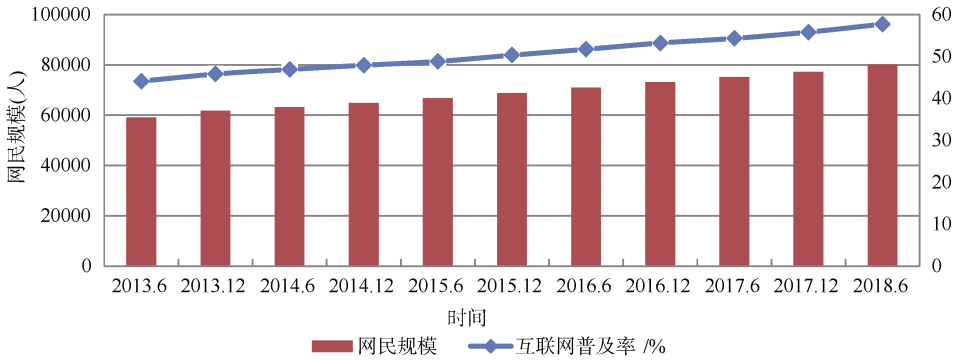

图 2　中国网民规模及互联网普及率

Figure 2　Internet users and Internet penetration in China

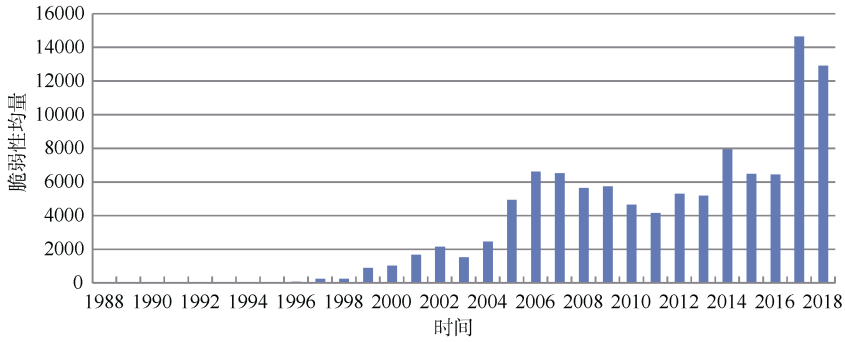

图 3　NVD 收录脆弱性数量

Figure 3　The number of vulnerabilities included in NVD

技术研究院(National Institute of Standard and Technology, NIST)的国家脆弱性数据库(National Vulnerability Database, NVD)历年发布的脆弱性数量[3]的统计图, 由图 3 中可以看出, 每年都有大量新的漏洞被发现, 脆弱性数量不断上涨。与国际形势类似, 国家信息安全漏洞共享平台(CNVD)所收录的脆弱性数量也在持续走高。如图 4 所示, 是从 2013 年到 2017 年 CNVD 所收录的安全漏洞与高危漏洞统计图[4], 自 2013 年以来, CNVD 收录的漏洞数量年增长率为 21.6%, 2017 年与 2016 年相比, 更是增长了 47.4%。高危漏洞的数量也在持续增长[4]。

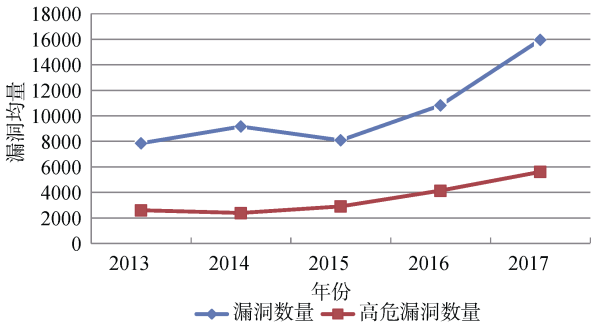

图 4　CNVD 收录漏洞及高危漏洞数量

Figure 4　CNVD vulnerabilities and high risk vulnerabilities

网络安全问题的日益严重, 一方面是由于网络的应用越来越广泛, 网络规模越来越大, 另一方面是越来越多的简单易用的攻击工具被开发出来, 而网络脆弱性无法彻底消除, 这使得网络安全事件不断发生。网络安全的从业人员在面临攻击时将要付出高出攻击者数倍的努力才能实现漏洞的修补与抢救, 但依旧总是处于被动。所以为了保障网络和信息的安全, 构建一个安全的网络环境, 除了必要的防火墙、入侵检测、杀毒软件等安全措施, 还需要料敌于先, 提前发觉网络系统的脆弱点, 及时修复, 或针对该脆弱点重点布防, 减少被攻破的可能。

2 网络系统脆弱点研究现状

在网络安全领域, 脆弱点的寻找即脆弱性评估最开始是对黑客攻击的防范技术, 后来随着网络规模的扩大和各种应用技术的增多, 网络脆弱性的评估方法为了满足当前的需求也在不断的发展。如图 5 所示, 在风险评估的初期, 防护者们利用经验分析脆弱点, 保护网络系统, 主要是采用基于规则的评估策略, 后来利用 Nessus、Nmap、OpenVAS 等扫描工具对本地或远程主机进行漏洞扫描, 结合已知的漏洞信息, 寻找网络脆弱点。在近些年, 随着网络规模的不断扩大, 网络环境越来越复杂, 基于模型的评估方法在逐渐兴起, 研究人员开始用树、图等分析工具进行分析, Cunningham 等[5]于 1985 年提出攻击图技术, 利用攻击图技术可以有效利用网络中各个节点的信息, 关联分析网络中节点脆弱性的关系, 从而分析得出完整的攻击路径, 找出网络的脆弱点。从这之后, 各种基于模型的分析技术得到发展, 其中最具代表的攻击图技术已经成为最受欢迎的评估手段之一。如今常用的攻击图主要有以下几种, Sheyner 等[6]提出了状态攻击图, 可以简洁地展示所有可达目标的攻击路径, 贾炜[7]利用状态攻击图评估网络脆弱性; Lingyu 等[8]提出了由原子攻击节点和属性节点组成的属性攻击图, 陈锋等[9]利用属性攻击图解决状态爆炸的问题; 文献[10]利用贝叶斯网络对网络攻击的不确定性进行建模, 王秀娟[11]等将属性攻击图转化为贝叶斯网络并更好的消除了环路; Ou[12-13]提出了逻辑攻击图, 该方法以逻辑惯性图表示原子攻击之间的依赖关系,他们实现的逻辑推理机 MulVAl[12-13]仍是如今备受欢迎的脆弱点分析工具, 孙哲等[14]基于 Datalog 语言提出了新的逻辑攻击图。贾炜等[15]提出基于网络中心性的方法对计算机网络进行评估。

图 5 网络脆弱点发现方法的发展

Figure 5 Development of network vulnerability discovery method

3 各研究方法分析

基于规则的评估策略仅对单一主机或服务进行扫描并进行风险评估, 却忽略了各个脆弱点之间的联系; 同时该方法也只能对已知类型的漏洞进行扫描, 无法度量未知的风险。攻击图技术虽然可以关联分析网络中节点的脆弱性关系, 但计算复杂度高, 尽管很多人做了很多工作, 依然难以解决当面临大规模网络时, 大量的节点会加大攻击图的计算量, 从而会产生状态爆炸的问题, 这限制了攻击图技术在现实中的互联网等大规模网络上的应用。因此, 为了解决这一问题, 本文用复杂网络的相关研究方法, 寻找可能的脆弱点。

复杂网络可以看作是由一些拥有属于自己的独立特征、但又与其他个体有相互连接关系的节点的集合, 每个个体在图中可以视为一个顶点, 节点间相互连接的关系被看作图中的边。复杂网络就是呈现高度复杂性的网络, 钱学森先生曾经给出复杂网络的一个较为严格的定义: 如果一个网络具有自组织性、自相似性、吸引子、小世界性质、无标度性质中部分或全部性质的网络称为复杂网络。脆弱性表示系统容易受攻击或容易被破坏的趋势[16], 脆弱性的一个本质特征就是在脆弱部位受到攻击后, 能够引起网络性能的大幅度下降[17], 而这个脆弱部位, 就是需要寻找的脆弱点。而网络性能, 主要是指网络的连通性, 网络平均最短路径等。网络的连通性代表着网络能否正常工作, 网络的平均最短路径则反映了网络执行的效率。而在复杂网络中寻找脆弱点主要有以下两种思路。

第一种是网络中心性反映脆弱性, 节点的中心性反映了节点在网络中的重要程度, 因此, 当重要节点受到攻击时必会产生较大影响。通过网络的静态拓扑信息, 利用网络中节点的统计性质(如度、最短路径、集聚系数、介数等)来表征各个节点的中心性, 用各个节点的中心性来表示其不同的重要性, 而重要性最高的节点集就被认为是脆弱点。常用的中心性指标主要包括度中心性[18]、介数中心性[19]、

接近度中心性[20]、特征向量中心性[21]等。而每个指标都从某一特定方面分析脆弱点。

度中心性是基于度的概念进行中心性测度, 节点的度是与节点相连的其他节点个数, 其计算方法为

$$D_i = \left|\left\{\left(v_i, v_j\right): v_i, v_j \in V, \left(v_i, v_j\right) \in E\right\}\right|, \tag{1}$$

其中 $V = \{v_1, v_2, \cdots, v_n\}$ 是一图中节点集合, $E \subseteq \left\{\left(v_i, v_j\right): v_i, v_j \in V\right\}$ 表示图中节点之间的边。节点 v_i 的度中心性就是其度 D_i 除以最大可能的度 $\left(N-1\right)$, 也就是归一化度值

$$C_d = D_i / \left(N-1\right), \tag{2}$$

显然 C_d 值越大, 节点越重要, 因此我们可以用度中心性作为节点脆弱性评分标准。

同度中心性类似, 介数中心性是节点的归一化介数, 节点的介数指网络中经过该节点的最短路径占所有最短路径的比例, 计算方法为

$$B_i = \sum_{j<k} g_{jk}\left(i\right) / g_{jk} \tag{3}$$

其中 $g_{jk}\left(i\right)$ 表示节点 v_j 与 v_k 节点间最短路径经过 v_i 的路径条数。在一个节点数为 N 的网络中, 最极端的情况是任意两个其他节点之间的最短路径均经过节点 v_i, 此时节点的介数达到最大值 $C_{N-1}^2 = \dfrac{\left(N-1\right)\left(N-2\right)}{2}$, 由此求得节点 v_i 的介数中心性为

$$C_b = \frac{2B_i}{\left(N-1\right)\left(N-2\right)}, \tag{4}$$

介数中心性反映了节点在网络中的枢纽地位, 显然通过该节点的最短路径越多, C_b 值越大, 节点越重要, 因此我们可以用介数中心性作为节点脆弱性评分标准。

接近度是拓扑空间里的基本概念, 而这个概念可以推广到图论中。可以将接近度定义为节点 v_i 到其他节点最短路径之和的倒数

$$C_C\left(v_i\right) = \left[\sum_{j=1}^{N} l\left(v_i, v_j\right)\right]^{-1}, \tag{5}$$

其中 $l\left(v_i, v_j\right)$ 为节点间最短距离。考虑到在含 N 个节点的网络中, 节点到其他节点的距离之和不会小于 $N-1$, 因此接近度中心性为

$$C_c\left(v_i\right) = \left(N-1\right) C_C\left(v_i\right), \tag{6}$$

这个指标反映了节点通过网络对其他节点的影响能力, 它同时考虑了节点的度值与节点在网络中的位置, 可以综合这两个重要信息, 得出更合理的关键节点。显然 C_c 值越小, 节点越重要, 因此我们可以用接近度中心性作为节点脆弱性评分标准。

特征向量指标反应了节点间的相互影响, 但由于一个大规模网络的特征值过于麻烦, 我们可以考虑利用特征向量指标的本质意义, 即网络中节点的中心性是由邻居节点的中心性给出。因此, 可以采用以下算法近似特征向量中心性:

步骤 1: 设一个网络中有 N 个节点, 先忽视节点的连接情况, 将重要程度看作一致, 初始化每个节点的中心值为 1, 即 $C_0 = 1$, $i = 1, 2, \cdots, N$。设 n 表示迭代次数。

步骤 2: 每个节点的新中心值 $C_{n+1}\left(v_i\right)$ 等于它所有邻居节点的旧中心值之和。显然 C_1 即为度中心指标。

步骤 3: 求所有节点中心值 $C_{n+1}\left(v_i\right)$ 之和 S, 将

$$C_{n+1}\left(v_i\right) = C_{n+1}\left(v_i\right) / S, \tag{7}$$

作为当前迭代次数下节点 v_i 的归一化中心值。

步骤 4: 用网络节点个数 N 表示迭代的规模, 令 n=n+1, 若 n<N, 则转去步骤 2, 否则停止。

综合上述步骤, 设节点 v_i 的邻居节点的下标的集合为 U_i, 则该指标的迭代方程可表示为

$$C_n\left(v_i\right) = \sum_{j \in U_i} C_{n-1}\left(v_j\right) / \sum_{j=1}^{N} C_{n-1}\left(v_j\right), \quad n = 1, 2, \cdots, N. \tag{8}$$

定义该网络的特征向量中心性指标为

$$C_e\left(v_i\right) = C_N\left(v_i\right), \tag{9}$$

特征向量中心性反映了节点的重要性不仅与其连接的边数目有关, 也与连接的节点的重要性有关系, 连接重要的节点可以提升自身节点的重要性。因此我们可以用特征向量中心性作为节点脆弱性评分标准。

在上述中心性计算方法基础上, 严栋等[22]提出了 AHP-熵权法确定网络中的重要节点, 该方法综合利用了上述中心性算法用熵权法刻画权重, 用层次分析法(AHP)量化确定各个节点的脆弱性评分; 赵凤花等[23]提出构建多属性决策的综合评价体系; 郭晓成等[24]提出用余弦相似度的方法, 结合多个指标进行综合评价; 这些方法避免了单一指标只从某一方面刻画脆弱点可能造成的较大误差。

第二种思路是基于“节点脆弱性等价于删除节点后对网络系统的破坏性”的思想, 许进等[25]人提出

的“核与核度”理论也为该思想提供了理论基础。该理论给出了核与核度的定义[26]: 对于一个给定的系统, 假如去掉或者破坏这个系统中若干个主要素, 对这个系统的破坏性最大, 那就将这若干个主要素叫做这个系统的核。因此, 可以通过研究删除节点后对网络系统的破坏来确定系统的核, 从而反馈节点的脆弱性。李鹏翔等[27]人基于这个思想, 提出了“节点删除”的计算方法, 他们把节点删除对网络连通性造成的影响分两部分: 第一, 被删除节点不能分别与剩余节点相连接造成的直接影响; 第二被删除节点会导致剩余节点之间缺少“桥梁”而无法连接造成的间接影响。将节点对之间距离(最短路)的倒数作为权数,对所有产生的不连通节点对进行加权求和, 用以度量对网络连通性的破坏程度。而刘浪等[28]提出了优先等级法, 他们提出一个节点的重要度等于将其删除后形成的连通图分支数与该节点直接或间接相连节点数与其各自优先等级系数之积的和, 并且没有追求每个节点的量化评分, 在计算量上对“节点删除”法进行了优化。

表 1 为对各个脆弱点评估方法优缺点分析, 通过表格中的对比, 可以明显的发现在较大规模的网络中寻找脆弱点, 方法 3、4 更为合适。本文选择复杂网络的方法主要是从兼顾准确性的基础上减轻计算量的角度出发, 利用复杂网络的拓扑统计性质, 寻找网络的脆弱点, 以此达到可以对更大规模网络(1000/10000 节点以上)进行脆弱点找寻的目的。

表 1 各脆弱点评估方法分析

Table 1 Analysis of vulnerability assessment methods

方法	优点	缺点
1. 基于规则的漏洞扫描技术	针对单一节点进行较为准确的漏洞扫描, 寻找脆弱点	无法考虑节点间的关联关系; 无法对未知的漏洞进行扫描; 面临大量节点时束手无策
2. 攻击图技术	关联分析网络中节点的关系, 求出攻击路径与脆弱点	同样无法对未知漏洞进行扫描; 算法复杂度高, 若节点个数过大会产生状态爆炸的情况
3. 基于复杂网络中心性算法寻找脆弱点	简化了计算方法, 从网络拓扑结构的角度寻找脆弱点, 计算量小, 可用于大规模复杂网络寻找脆弱点	没有考虑节点自身的身份信息; 没法直接说明找到的节点为脆弱点; 围绕节点进行计算, 没有考虑节点被攻击后对网络整体性质的影响
4. 删除节点法	同样从网络拓扑结构角度寻找脆弱点, 计算量相对攻击图技术要小, 可用于大规模网络; 通过删除节点后对网络系统的破坏来说明了节点的脆弱性, 考虑了节点被攻击后对网络整体性质的影响	没有考虑节点自身的身份信息; 没有考虑因删除节点后导致节点间距离变大的情况, 这使得部分情况结果不够准确; 计算复杂度高于中心性算法

从表 1 可以看出, 方法 3、4 也存在比较明显的缺点, 方法 3 没有考虑网络脆弱点被攻击后, 网络拓扑已经变化的事实, 方法 4 考虑到了这一点, 但是当网络规模增大后计算中无法体现出来, 即方法 3、4 没有反映出在大规模网络中, 网络攻防中脆弱点动态变化带来的对网络的影响。本文提出的对节点删除法的改进就是针对网络攻防中网络拓扑可能因为脆弱点受到攻击而变化的情况, 可以寻找到更符合网络实际情况的脆弱点。

4 删除节点方法改进

表一中提到的方法 3、4 两种思路均是复杂网络中寻找脆弱点的有效方法, 其中方法 3 计算更加简便, 但中心性算法大多只是以被评估节点为中心展开计算, 没有办法体现节点被攻击后对整个网络的影响; 而方法 4 虽然计算更加复杂, 但可以观察节点被攻击后对整个网络系统的影响, 更切合寻找的脆弱点的本质。然而, 在现实生活中, 当网络规模足够大且复杂, 各节点之间的联系繁杂密切, 若只考虑删除节点后产生的不连通节点对的变化, 是难以准确寻找脆弱点的。如图 6 所示, 虽然删除了节点 1 或节点 2, 除了与他们本身与其他节点无法相连接, 其他节点之间还是相连的, 但其连通路径实则已经发生了变化, 这利用原节点删除法是无法分析其中的变化的。因此, 本文在这里提出对该方法进行改进, 通过计算删除节点后对网络平均最短路径的影响来模拟当网络中节点受到攻击导致节点不可用时对网络整体性质的影响。

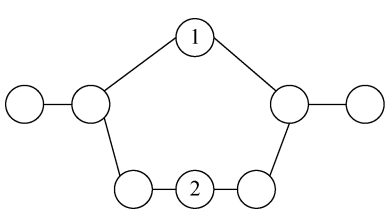

图 6 算法分析示例

Figure 6 Example for algorithm analysis

4.1 算法设计

显然, 通过删除节点后, 可能产生一个或多个连通分支, 也可能改变某些节点对的最短路径, 因此本文认为主要影响的网络性质分别是平均最短路径和连通节点对的变化。

本文首先考虑对平均最短路径的影响。删除节点后, 若不产生新的连通分支, 则通过该节点的最短路径必然会改变, 并且有较大的可能会变大。但若产生了新的连通分支, 会使部分节点之间的距离变为无限大, 从而无法定量的对产生的影响进行分析。因此, 本文计算时将不连通的节点对间的距离设为图的直径大小, 如此不仅考虑到了产生不连通节点对对网络整体脆弱性评分的影响, 同时可以消除节点距离无限大对网络平均最短路径的影响。

假设对网络平均最短路径的影响为

$$D(v_i)=\left(L_{after}-L_{before}\right)/L_{before}, \tag{10}$$

其中 L_{before}, L_{after} 分别表示节点删除前后网络的平均最短路径,其计算方法为

$$L=\frac{2\sum_{v_i,v_j\in V} l\left(v_i,v_j\right)}{N\times(N-1)}, \tag{11}$$

其中 $l\left(v_i,v_j\right)$ 为两节点间的最短路径。设网络的直径即网络任意节点间最短路径的最大值为

$$Dia=\max_{0<i,j<N}\left(l\left(v_i,v_j\right)\right), \tag{12}$$

当产生不连通节点对 $\left(x_i,x_j\right)$ 时, 其最短路径更新为 Dia。而 $D\left(v_i\right)$ 即为衡量节点 v_i 的脆弱性指标。

4.2 算法流程

可以通过以下步骤来实现本文的算法:

步骤 1: 计算网络 A 初始的平均最短路径 L_{before}, 以及网络直径 Dia, 令 $i=1$.

步骤 2: 删除节点 v_i, 即断开节点 v_i 与其他节点的所有连接路径.

步骤 3: 计算删除节点 v_i 后网络的平均最短路径 $L_{after}\left(v_i\right)$.

步骤 4: 若产生了新的不连通节点对, 则将其节点间距离赋值为 Dia.

步骤 5: 计算删除节点对网络平均最短路径的影响。

步骤 6: 重新连接原与 相连的路径, $i=i+1$.

步骤 7: 重复进行步骤 2~ 6, 共 N 次.

算法流程如图 7 所示:

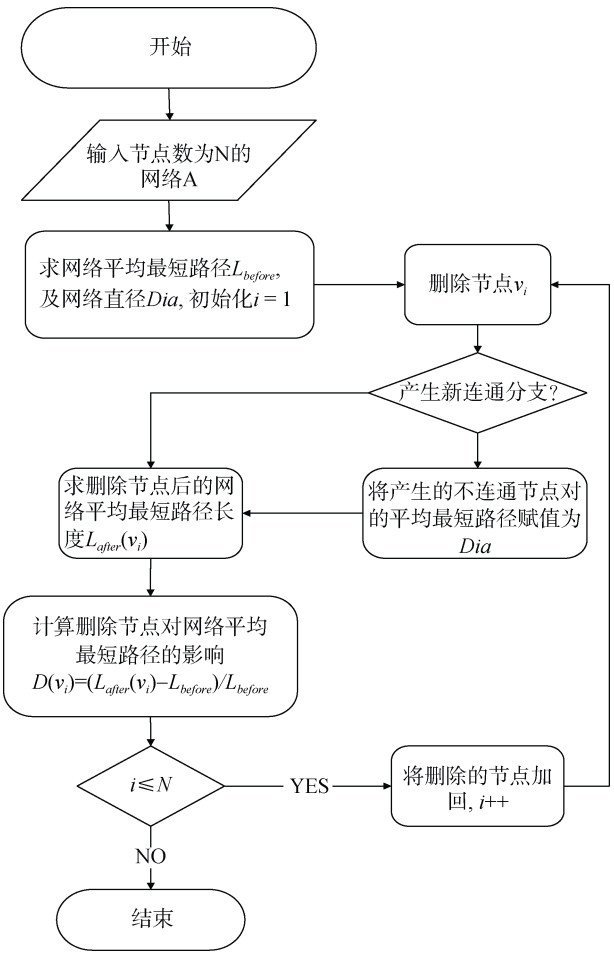

图 7 改进节点删除法算法流程图

Figure 7 Flow chart of improved node deletion algorithm

5 实验设计与分析

5.1 实验环境与准备

实验过程采用 MATLAB 语言, 在 Windows 7 操作系统下的 matlab 2014a 应用程序进行。因为 MATLAB 对矩阵运算的方便性与可拓展性, 本文选用 MATLAB 可以方便的用邻接矩阵清晰明了的表示复杂网络, 并可以方便的对其统计性质进行计算, 即使面对较大规模的网络时, 在硬件条件允许的情况下也可以方便的进行数据存储与计算。

小世界网络是一种特殊的复杂网络, 该网络随机性介于随机网络和规则耦合网络之间, 适合模拟真实网络系统。其概念, 简单的说就是描述这样的一个事实: 尽管一些网络系统有很大的尺寸, 但其中任意两个节点之间却有一个相对小的距离。小世界特征除了有比较短的平均距离外, 还表现出相对较大的集聚系数。目前最常见的有两种小世界网络。

Newman 和 Watts 于 1999 年提出的 NW 小世界

网络[29], 采用在规则耦合网络上进行随机加边的规则实现小世界网络, 其具体构造方法为:

1) 从规则图开始: 首先从一个含有 N 个节点的最近邻耦合网络开始考虑, 假设它们围成一个圈, 其中每个节点与它左右相邻的各 $K/2$ 个节点相连, K 是偶数。参数满足 $N >> K >> \ln N >> 1$。

2) 随机化加边: 以大小为 p 的概率在随机选取的一对未直接相连的不同节点之间加上一条边。从而保证任意两个不同节点之间至多只能有一条边, 并且每一个节点都不会有与自身相连的边。

如图 8 所示, 在该模型中, 当 p=1 时, 得到的网络为随机网络; 当 p=0 时, 得到的网络是规则的耦合网络; 当 0<p<1 时, 得到的即为 NW 小世界网络。

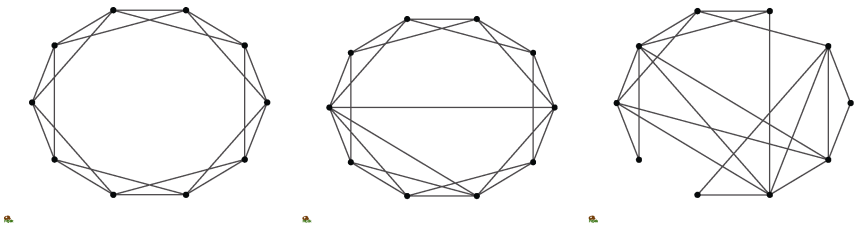

图 8 NW 小世界网络

Figure 8 NW Small-World Network

Watts 和 Strogatz 于 1998 年提出 WS 小世界网络[30],采用在规则耦合网络上进行随机重连的规则实现小世界网络, 其具体构造方法为:

1) 从规则图开始: 首先从一个含有 N 个节点的最近邻耦合网络开始考虑, 假设它们围成一个圈, 其中每个节点与它左右相邻的各 $K/2$ 个节点相连, K 是偶数, 参数满足 $N >> K >> \ln N >> 1$。

2) 随机化重连: 以大小为 p 的概率随机地重新连接网络的每条边, 即将边的一个端点保持不变, 随机选取一个网络中的其他节点作为网络另一个端点, 其中规定, 任意两个两不同节点之间至多只能有一条边, 且每个节点都不能有与自身相连的边。

如图 9 所示, 在该模型中, 当 p=1 时, 得到的网络为随机网络; 当 p=0 时, 得到的网络是规则的耦合网络; 当 0<p<1 时, 得到的即为 WS 小世界网络。

5.2 实验设计

本文选用 NW 小世界网络作为实验分析的模型, 通过对 2 近邻耦合网络以 0.05 的概率进行随机加边, 这也符合现实生活中网络节点间会不断产生新的连接关系的特性, 从而可以使模拟实验结果更加真实有效。为便于比较, 本文同时采用了度中心性, 介数中心性, 接近度中心性, 特征向量中心性等中心性测度以及原删除节点算法来寻找网络脆弱点, 以此作为验证本文提出的改进的节点删除方法所寻找的脆弱点的准确性, 以及对比分析改进删除节点方法的特点。本文同时设计了 41 个节点的某真实实验室网络, 100 节点、100~200 节点、1000 节点三种模拟小世界网络, 总共四种规模的实验, 分别用来验证改进节点删除方法的可行性、验证说明该方法在小规模复杂网络中的有效性、研究该方法的时间复杂度、验证算法在大规模网络中的有效性。具体目标如表 2 所示。

具体实验流程如图 10 所示。

5.3 实验过程

实验中首先对某实验室网络拓扑结构进行模拟; 然后构造 100 节点的 NW 小世界网络, 该网络具有明显的小世界性质, 是典型的复杂网络。然后分别计算两个网络的度中心性、介数中心性、接近度中心性、特征向量中心性以及改进前后的节点删除法得到的节点脆弱性, 并通过折线图直观对比分析各个方法于不同规模的网络中, 在寻找脆弱点时的异同。同时

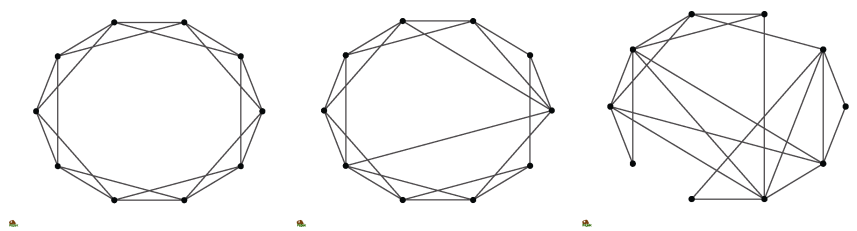

图 9 WS 小世界网络

Figure 9 WS Small-World Network

表 2 各实验目标

Table 2 Experimental Objectives

实验规模	网络生成	实验目标
41 节点	实际实验室网络	在实际网络环境验证算法的可行性
100 节点	模拟小规模网络	在小规模复杂网络中验证算法的有效性
100~200 节点	模拟小规模网络	逐渐增加网络规模, 研究算法的时间复杂度
1000 节点	模拟大规模网络	在大规模网络中, 验证算法的有效性

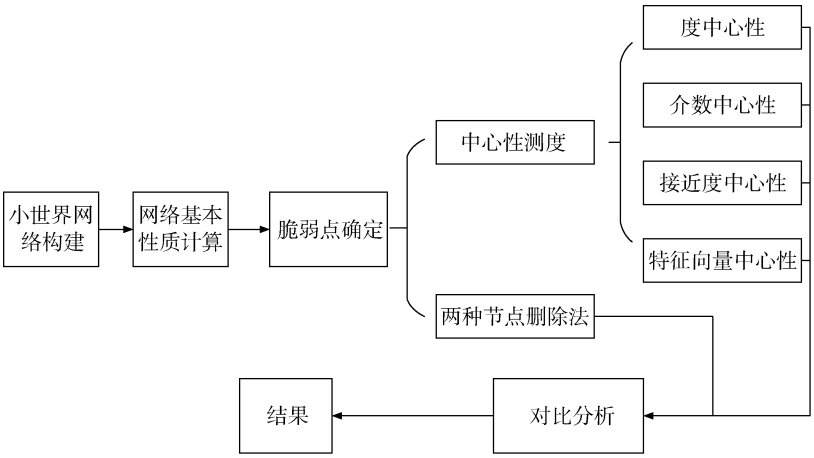

图 10 实验流程

Figure 10 Experimental flow chart

在 100 节点小世界网络的基础上不断增加网络中的节点, 测试计算不同算法的时间复杂度并通过折线图直观反映。最后是对一个 1000 节点的大规模小世界网络进行分析, 寻找网络中的脆弱点。

5.4 实验结果

图 11 是某实验室忽略各个节点身份后的网络拓扑图。图 12 是利用不同算法对图 11 进行节点的脆弱性评分计算的折线图, 该图直观的反应了不同算法得到的脆弱点的分布情况。为便于分析, 已经将脆弱性评分做了归一化处理, 使结果都落于区间$(0,1)$, 其中0代表了脆弱性评分最低的节点, 1代表脆弱性评分最高的节点。表 3 是不同算法得到的图 11 中脆弱性排在前 5 的节点顺序及脆弱性评分。其中C_d表示度中心性测度, C_b表示介数中心性测度, C_c表示接近度中心性测度, C_e表示特征向量中心性测度, DELpre 表示原删除节点法, DELpro 表示改进删除节点法。

从图 12 可以看出, 不同方法在该实际网络中寻找到的脆弱点有高度的相似性, 也说明改进节点删除法是可行的, 同时, 改进前后的删除节点法在评分折线上有惊人的一致性, 这也说明了改进后在小规模网络中没有改变原算法本身的优势。由表 3 中可以看出, 对于 DELpro 所寻找到的脆弱点, 评分排前 3 的所有方法均能找到, 排第 4 的有 5 个方法找到, 排第 5 的有 4 个方法找到, 说明 DELpro 准确度较高, 与 DELpre 一样。

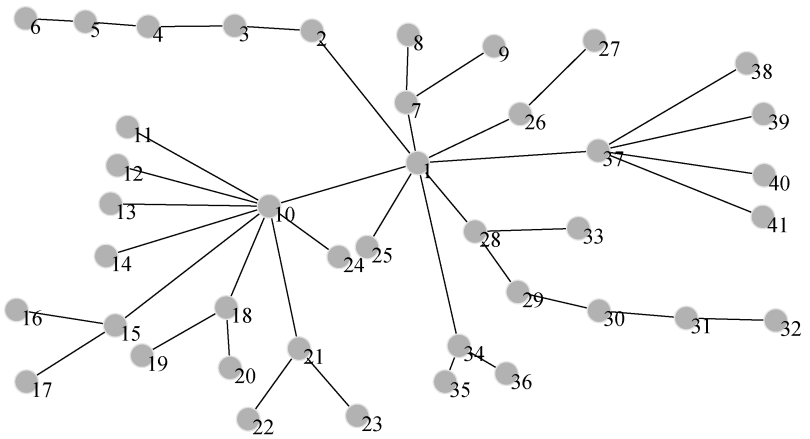

图 11 某实验室网络忽略节点身份的拓扑图

Figure 11 Topology map ignoring node identity of a laboratory network

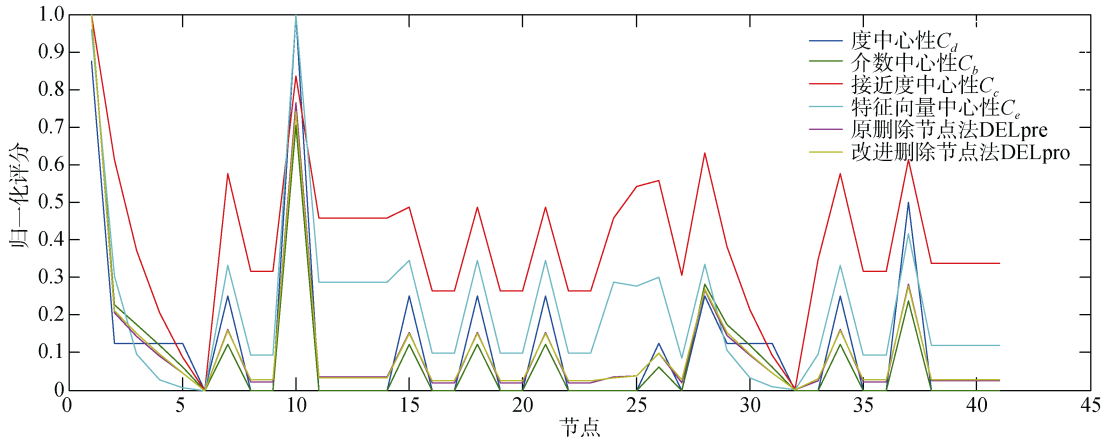

图 12　图 11 中各算法节点脆弱性评分

Figure 12　Node vulnerability score using each algorithm in Figure 11

表 3　图 11 中节点 脆弱点排序及评分

Table 3　Ranking and scoring vulnerability of node in Figure 11

排序	C_d		C_b		C_c		C_e		DELpre		DELpro	
\	节点	评分	节点	评分	节点	评分	节点	评分	节点	评分	节点	评分
1	10	0.2250	1	0.8628	1	0.4444	10	0.1146	1	179.2300	1	0.4663
2	1	0.2000	10	0.6231	10	0.3960	1	0.1100	10	138.6700	10	0.3506
3	37	0.1250	28	0.2808	28	0.3361	37	0.0479	37	55.7762	37	0.1409
4	28	0.1250	37	0.2436	2	0.3306	15	0.0399	28	53.0972	28	0.1380
5	34 等	0.1250	2	0.2359	37	0.3306	18	0.0399	2	42.8639	2	0.1116

图 13 是用软件 MATLAB 构造, pajek 绘制的一个 100 节点的 NW 小世界网络图。

图 14 是不同算法对图 13 进行节点的脆弱性评分计算的折线图, 该图直观地反应了不同算法得到的脆弱点的分布情况。为便于分析, 同样将脆弱性评分做了归一化处理, 其中 0 代表了脆弱性评分最低的节点, 1 代表脆弱性评分最高的节点。而表 4 则是图 13 中脆弱性排在前 10 的节点顺序及脆弱性评分。从图 14 中可以看出, 即使到了 100 个节点、连接情况更为复杂的复杂网络, 不同方法在该网络中寻找到的脆弱点依然有高度的相似性, 说明了在小规模复杂网络中依旧具有可用性。而从表 4 中可以发现, 删除节点算法改进前后展现了较大的区别, 而原删除节点算法与接近度中心性有很高的一致性, 这与表 3 中数据反应的情况不同, 这是因为当删除节点后没有产生新的连通分支的话, 在计算过程上与接近度中心性基本一致。DELPro 寻找到的节点中, 编号 87、11、58 得到 6 种方法全部支持, 编号 86、73 得到 5 种方法支持, 编号 99、67 得到 3 种方法支持, 编号 28、10 得到两种方法支持, 但也有编号 48 没有得到其他方法支持, 是本算法计算删除节点后网络的平均最短路径所致, 这是考虑了删除节点后网络整体性质变化得到的结论, 是其他算法无法找出的节点。并且该节点有很大几率是脆弱点。

图 15 是各个算法计算 100~200 个节点的小世界网络的脆弱性时所用的时间曲线。其中从上到下依次是度中心性测度 C_d, 介数中心性测度 C_b, 接近度中心性测度 C_c, 特征向量中性测度 C_e, 原删除节点法 DELpre, 改进删除节点法 DELpro 的曲线。由图 15 可以看出, 其中本文所提改进删除节点算法耗时最多, 度中心性耗时最少。

相对中心性算法的低耗时, 不论是原删除节点法还是本文所提改进删除节点算法时间复杂度均较高, 还有很大的改进空间。

最后一个实验是模拟大规模网络实验, 节点数量为 1000 个。表 5 是上述 6 种方法对 1000 节点的较大规模网络分析后得到的前 10 位脆弱点及其评分。表 5 中, DELPro 寻找到的节点中, 编号 468、934、934、13 未得到其他方法支持, 而 DELPre 寻找到的

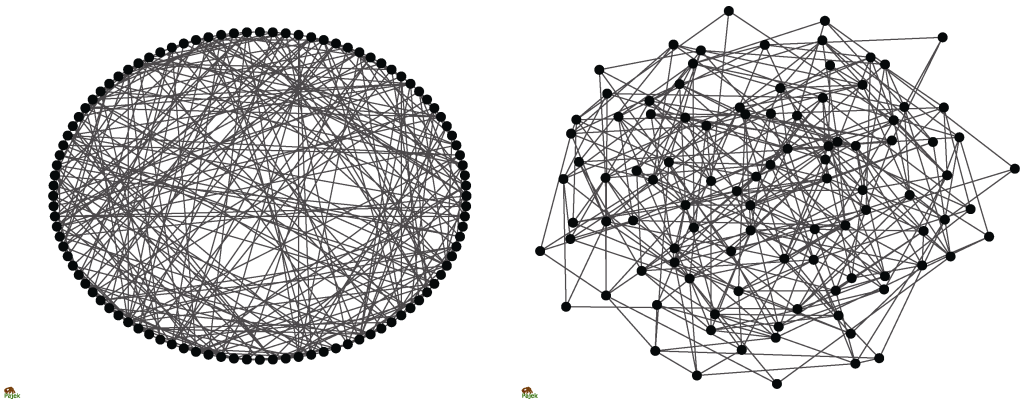

图 13　100 节点小世界网络

Figure 13　100-node small world network

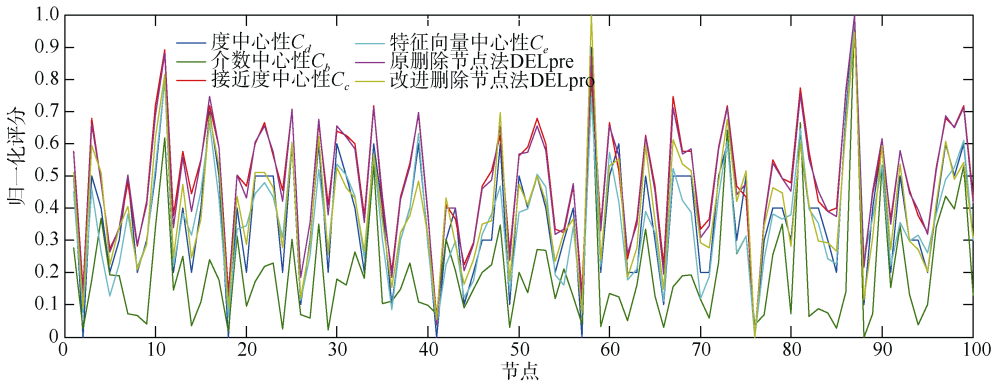

图 14　图 13 中各算法节点脆弱性评分

Figure 14　Node vulnerability score using each algorithm in Figure 13

表 4　图 13 中脆弱点排序及评分

Table 4　Ranking and scoring vulnerability of node in Figure 13

排序	C_d		C_b		C_c		C_e		DELpre		DELpro	
\	节点	评分	节点	评分	节点	评分	节点	评分	节点	评分	节点	评分
1	87	0.1313	87	0.0996	87	0.4562	87	0.0220	87	50.6667	58	0.0143
2	58	0.1212	58	0.0849	11	0.4420	11	0.0182	11	48.8333	87	0.0138
3	11	0.1212	81	0.0730	58	0.4342	58	0.0172	58	48.7500	11	0.0126
4	16	0.1212	73	0.0711	81	0.4267	86	0.0158	86	47.0833	48	0.0115
5	86	0.1111	11	0.0693	86	0.4249	16	0.0157	81	46.8333	16	0.0114
6	25	0.1111	34	0.0627	67	0.4431	81	0.0154	16	46.7500	73	0.0113
7	28	0.1111	90	0.0625	10	0.4195	39	0.0151	99	46.2500	86	0.0110
8	30	0.1111	99	0.0625	16	0.4195	99	0.0147	67	46.2500	28	0.0108
9	34	0.1111	97	0.0550	34	0.4195	73	0.0147	34	46.2500	10	0.0107
10	39	0.1111	98	0.0519	73	0.4195	25	0.0146	73	46.2500	67	0.0107

节点均得到其他方法的支持, 由此可以看出, 当网络规模足够大时, 原删除节点法更倾向于中心性算法, 而改进的删除节点法则体现了其独有的特点,除了与其他算法相同的节点之外, 还认为当编号 468、934、934、13 受到攻击时会对网络性能产生较大影响。但本实验并不能说明这几个节点就一定是脆弱点, 但是其几率很高, 需要进一步的实验验证。

5.5　实验结果分析

图 12、图 14 分别反应了图 11、图 13 网络中节点脆弱性分布评分, 同时反应了该网络的脆弱点分布情况。结合图 14 与表 4 可以看出, 不同算法在节点的脆弱性评分分布上大体一致, 算法间的相互印证也

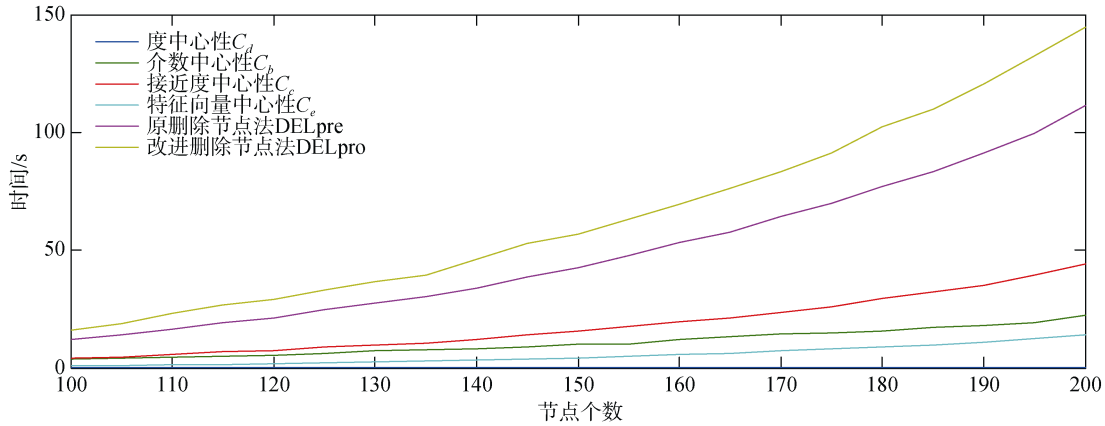

图 15　各算法时间复杂度

Figure 15　Time complexity of each algorithm

表 5　1000 节点小世界网络脆弱性前 10 排序及评分

Table 5　Top 10 ranking and scoring of 1000-node small world network vulnerability

排序	C_d		C_b		C_c		C_e		DELpre		DELpro	
\	节点	评分	节点	评分	节点	评分	节点	评分	节点	评分	节点	评分/10^{-4}
1	682	0.0741	283	0.0084	430	0.5157	682	0.0014	682	534.333	430	8.2082
2	715	0.0731	682	0.0075	682	0.5157	927	0.0014	430	533.667	682	8.1682
3	430	0.0721	209	0.0070	927	0.5152	715	0.0014	927	533.333	300	8.1348
4	927	0.0721	513	0.0670	786	0.5139	430	0.0014	715	532.333	927	8.1215
5	269	0.0711	671	0.0066	269	0.5131	269	0.0014	269	531.667	715	8.1014
6	422	0.0711	672	0.0066	715	0.5131	156	0.0014	786	530.833	468	8.0747
7	156	0.0701	451	0.0065	300	0.5123	422	0.0014	300	530.500	786	8.0747
8	40	0.0691	596	0.0065	694	0.5123	80	0.0014	156	530.333	934	8.0480
9	80	0.0691	378	0.0065	581	0.5120	40	0.0014	422	530.167	819	8.0414
10	126	0.0691	962	0.0064	999	0.5118	126	0.0013	694	530.167	13	7.9950

反应了本文的改进删除节点算法的有效性，其中删除节点法改进前后评分大体相同，找到的脆弱点排序也完全一样，这说明了改进之后并没有改变该方法寻找脆弱点的有效性。从图 14 中可以看出，即使将网络规模扩大到 100 节点，各算法在脆弱性评分上也展现了一致性。

而通过表 4 具体的展示了各个算法对节点脆弱性的评估排序与评估值，反应了各个算法在具体评估上的异同点。由表中可以看出不同的评估方式对于选取脆弱点有自己的“偏好”。虽然排序结果不尽相同，但脆弱性较高的节点集有非常高的重合率。而不重合的部分，也体现了各自算法的偏向性。其中删除节点算法改进前后展现了较大的区别，而原删除节点算法与接近度中心性有很高的一致性，这是因为当删除节点后没有产生新的连通分支的话，在计算过程上与接近度中心性基本一致。这样就没有办法体现出动态删除节点的方法的优越性，这也是要对该方法进行改进的原因。而改进之后的删除节点法则不仅考虑了删除节点后对网路连通性的影响，同时考虑了对网络性能的影响。表 4 中如标号为 87、58、11 的节点有较高的认可度，可以被认为一定是脆弱点，是需要重点保护的节点。而 48 虽然只在某一个算法中有较高的脆弱性，但这从某个角度来说明了节点的脆弱性，也应该是不容错过的脆弱点。比如 48 号节点，在改进的删除节点法中表现出较高的脆弱性，而在静态的中心性指标中没有表现，说明该节点静态统计数据并不突出，但是删除后会导致许多节点对之间路径大幅度增加，影响了网络性能，说明了该节点在网络中的不可替代性。

图 15 直观的反应了不同算法在计算的时间复杂度上的差异，由图 15 可知，节点删除算法改进前后均有多次计算网络的平均最短路径的问题而导致消耗的时间倍增；度中心性由于其简单的计算方法在时间效率上有其绝对的优势；特征向量中心性虽然

经过多次迭代, 但由于没有计算网络平均最短路径, 节省了大量的时间; 介数中心性与接近度中心性都计算了网络平均最短路径, 但由于接近度中心性在计算到单个节点平均最短路径时出现了大量的重复, 因而消耗的时间也要比介数中心性多出许多。

表 5 的数据表明即使在 1000 节点的较大规模复杂网络上, 这几种算法依旧可以较好地找到脆弱点, 同时各个算法之间的差异性也被扩大。其中改进的删除节点算法因计算方式与其他 5 种方法差距较大, 因此寻找到的脆弱点也不尽相同, 这也体现了改进之后考虑节点被攻击后网络整体性能变化这一算法的特点。但要确定这些节点是否是脆弱点, 还需要进一步实验验证。

6 结论

通过表 3、表 4、表 5 中不同算法在不同规模复杂网络中对脆弱点的找寻能力, 可以发现不同算法所寻找的脆弱点有很大相似之处, 成功验证了改进的删除节点法在脆弱点的寻找过程中的可行性与准确度。同时, 表中不同算法所得脆弱点排序也不尽相同, 这体现了不同算法也有自己的特点, 本文改进删除节点算法特别地考虑了删除节点后对网络效率的影响, 解决了当网络规模过大, 节点之间连通情况复杂时, 通过删除节点无法产生新的连通分支而失去动态寻找网络脆弱点的优势的问题, 从而可以结合网络整体性质, 动态的寻找网络中的脆弱点。这实际上是考量了当节点受到攻击而不可用时, 网络整体性能受到影响的程度。同时该算法虽然在时间复杂度上不如基于中心性的测度方法, 也略差于原删除节点法, 但仍可在短时间内计算较大规模的复杂网络并找出脆弱点, 这是攻击图等其他方法无法完成的。

由于本文只通过纯图论的原理对网络拓扑结构进行分析, 实际上忽略了节点本身的特征, 没有进行实际攻防模拟实验, 但从另一个的角度解决了利用攻防模拟实验等方式无法对超大规模网络分析的困难。而如何综合复杂网络性质与节点自身的性质的研究也是以后需要进行深入研究的方向。

致谢 本课题得到国家重点研发计划资助(项目编号 No.2016YFB0800700)。

参考文献

[1] "ISC Internet Domain Survey", Internet System Consortium, https://www.isc.org/network/survey/, Sept 2018.

[2] "The forty-second statistical report on China's Internet development", China Internet Network Information Center, http://www.cnnic.net.cn/hlwfzyj/hlwxzbg/hlwtjbg/201808/P020180820630889299840. pdf, Sept 2018.
("第 42 次中国互联网络发展状况统计报告", 中国互联网络信息中心, http://www.cnnic.net.cn/hlwfzyj/hlwxzbg/hlwtjbg/201808/P020180820630889299840. pdf, Sept 2018.)

[3] "National Vulnerability Database", NVD, http://nvd.nist.gov/, Sept 2018.

[4] "China Internet network security report 2017", CNCERT, http://www.cert.org.cn/publish/main/upload/File/2017annual(1). pdf, Sept 2018.
("2017 年中国互联网网络安全报告", 国家互联网应急中心, http://www.cert.org.cn/publish/main/upload/File/2017annual(1). pdf, Sept 2018.)

[5] W.H. Cunningham. "Optimal attack and reinforcement of a network". *J.assoc.comput.mach*, vol. 32, no. 3, pp. 549-561, 1985.

[6] O. Sheyner, J. Haines, S. Jha, R.Lippmann, J.M. Wing. "Automated Generation and Analysis of Attack Graphs". *IEEE Security & Privacy Magazine* , vol.1971, pp. 273, 2002.

[7] W. Jia, "Research on vulnerability assessment method of computer network [Ph.D. dissertation]," University of Science and Technology of China, 2012.
(贾炜. "计算机网络脆弱性评估方法研究 [博士学位论文]," 中国科学技术大学, 2012.)

[8] L. Wang, C. Yao, A. Singhal, and S. Jajodia. "Interactive Analysis of Attack Graphs Using Relational Queries". *Lecture Notes in Computer Science*, vol.4127, pp. 119-132, 2008.

[9] F. Chen, Y. Zhang, A.H. Bao and J.S. Su. "Research on quantitative assessment of network vulnerability based on attack graph", *Computer Engineering and Science*, vol.32, no. 10, pp. 8-11, 2010.
(陈锋, 张怡, 鲍爱华, 苏金树. "基于攻击图的网络脆弱性量化评估研究". *计算机工程与科学*, 2010, 32(10): 8-11.)

[10] H. Kim, J. Reich, A. Gupta, M. Shahbaz, and N.Feamster. "Kinetic: verifiable dynamic network control". *Usenix NSDI*. 2015.

[11] X.J. Wang, B. Sun, Y.W. Liao, and C.B. Xiang. "Vulnerability assessment of Bayes attribute attack graph network". *Journal of Beijing University of Posts and Telecommunications*, vol.38, no.4, pp. 110: 116, 2015.
(王秀娟, 孙博, 廖彦文, 相从斌. "贝叶斯属性攻击图网络脆弱性评估". *北京邮电大学学报*, 2015, 38(4): 110-116.)

[12] X. Ou, W.F. Boyer, M.A. Mcqueen. "A scalable approach to attack graph generation" in *ACM Conference on Computer and Communications Security* (CCS'06), pp. 336-345, 2006.

[13] X. Ou, S. Govindavajhala,A.W. Appel. "MulVAL: a logic-based network security analyzer". *Usenix Security Symposium*, pp. 8-8, 2005.

[14] Z. Sun, Z.Z. Wu and Q.M. Li. "Modeling and generation method of traffic attack graph", *software*, no.4, 2018.
(孙哲, 巫中正, 李千目. "流量攻击图的建模与生成方法". *软件*, 第 4 期, 2018.)

[15] W. Jia, D.G. Feng and Y.F. Lian. "Computer network vulnerability assessment method based on network centrality", *Journal of the Chinese Academy of Sciences,* vol. 29, no. 4, pp. 529-535, 2012.

(贾炜, 冯登国, 连一峰. “基于网络中心性的计算机网络脆弱性评估方法”. *中国科学院大学学报*,, 2012, 29(4):529-535.)

[16] W.X. Zhang, Q. Li, W.P. Wang, and H.B. Li. “Comprehensive analysis method for vulnerability of complex systems”, *Journal of National University of Defense Technology*, vol.38, no.2, pp. 150-155, 2016.
(张旺勋, 李群, 王维平, 李海兵. “复杂系统脆弱性综合分析方法” . *国防科技大学学报*, 2016, 38(2): 150-155.)

[17] J. Guo. “Vulnerability analysis of power communication network based on complex network theory [master dissertation], ” North China Electric Power University, 2010.
(郭静. “基于复杂网络理论的电力通信网脆弱性分析[硕士学位论文], ”华北电力大学, 2010.)

[18] S. Wassermen, K. Faust. “Social network analysis: methods and applications”. Ca-mbridge; Camridge University Press, 1994.

[19] P. Bonacich. “Factoring and weighting approaches to status scores and clique identification”. *Journal of Mathematical Sociology*, vol.2, no.1, pp. 113-120, 1972.

[20] P. Bonacich. “Technique for analyzing overlapping memberships”. *Sociological Methodology*, vol.4, no.4, pp. 176-185, 1972.

[21] K. Stephenson, M. Zelen. “Rethinking Centrality: Methods and Applications”. *Social Networks*, vol.11, pp.1-37, 1989.

[22] D. Yan, S.B. Zhang, K. Zong and Z.H. Hu. “Identification method of key nodes in complex networks based on AHP- entropy weight method”. *Journal of Guangxi University: Natural Science Edition*, vol.41, no.6, pp. 1933-1939, 2016.
(严栋, 张世斌, 宗康, 胡志华. “基于 AHP-熵权法的复杂网络关键节点识别方法”. *广西大学学报:自然科学版*, 2016, 41(6): 1933-1939.)

[23] F.H. Zhao and B. Yang. “Comprehensive evaluation method of node importance in complex networks”, *Journal of Wuhan University of Technology (information and Management Engineering Edition)*, no.4, pp. 461-464, 2015.
(赵凤花, 杨波. “复杂网络节点重要性的综合评价方法”. *武汉理工大学学报(信息与管理工程版)*, 2015(4): 461-464.)

[24] X.C. Guo, R.N. Ma and G. Wang. “Comprehensive evaluation method of node importance in complex networks”, computer simulation, vol.34, no.7, pp. 264-268, 2017.
(郭晓成, 马润年, 王刚. “复杂网络中节点重要性综合评价方法研究”. *计算机仿真*, 2017, 34(7): 264-268.)

[25] J. Xu. ” A new method of research system -- core and core degree method”, *Systems Engineering and Electronics,*no.6, pp. 1-10, 1994.
(许进. “一种研究系统的新方法—核与核度法”. *系统工程与电子技术*, 第 6 期, 1994(6): 1-10.)

[26] J. Xu. “core and core degree theory of systems and its application”, Xidian University Press, 1994.
(许进. “系统核与核度理论及其应用”. 西安电子科技大学出版社, 1994.)

[27] P.X. Li, Y.Q. Ren and Y.M. Xi. “A measure of the importance of network nodes (sets)”, *Systems engineering*, vol.22, no.4, pp. 13-20, 2004.
(李鹏翔, 任玉晴, 席酉民. “网络节点(集)重要性的一种度量指标”. *系统工程*, 2004, 22(4): 13-20.)

[28] L. Liu, W. Deng, F. Cai, L. Chen. “A new method for calculating node importance -- priority method”. Chinese Journal of Management Science, vol.15, no.s1, pp.162-165, 2007.
(刘浪, 邓伟, 采峰, 陈玲. “节点重要度计算的新方法——优先等级法”. *中国管理科学*, 2007, 15(s1): 162-165.)

[29] M.E.J. Newman, D.J. Watts. “Renormalization group analysis of the small-world network model”. *Physics Letters A*, vol.263, no.4-6, pp. 341-346, 1999.

[30] D.J. Watts, S.H. Strogatz. “Collective dynamics of ‘small-world’networks”. *nature*, vol.393, no.6684, pp.440, 1998.

赵小林 于 2000 年在北京理工大学计算机应用技术专业获得硕士学位。现任北京理工大学计算机学院副教授, 网络空间安全和计算机科学与技术学科硕士生导师, 软件工程专业责任教授。2004—2005 年美国加州大学圣迭戈分校(UCSD)访问学者。研究领域为网络空间安全、数据安全、人工智能。研究兴趣包括: 区块链、网络攻击检测。Email: zhaoxl@bit.edu.cn

薛静锋 于 2003 年在北京理工大学计算机应用技术专业获得博士学位。现任北京理工大学计算机学院教授, 网络空间安全博士生导师。2005—2006 年美国加州大学圣迭戈分校(UCSD)访问学者。研究领域为网络空间安全, 人工智能。研究兴趣包括: 高可信软件、网络攻击检测。Email: xuejf@bit.edu.cn

徐浩 于 2018 年在中国海洋大学计算机科学与技术专业获得工学学士学位。现在北京理工大学网络空间安全专业攻读硕士学位。研究领域为网络空间安全。研究兴趣包括: 网络风险评估、区块链。Email: xhao0403@163.com

宋天凌 于 2018 年在北京理工大学软件工程专业获得学士学位。在本科毕业设计期间研究复杂网络中脆弱点分布规律。Email: 1002911561@qq.com

胡晶晶 于 2005 年在北京理工大学计算机应用专业获得博士学位。现任北京理工大学计算机学院副教授, 网络空间安全硕士生导师。研究领域为软件工程理论、人工智能。研究兴趣包括: 人工游戏智能。Email: hujingjing@bit.edu.cn

闫怀志 于 2003 年在北京理工大学机械电子工程专业获得工学博士学位。现任北京理工大学计算机学院副教授, 网络空间安全硕士生导师。研究领域为网络空间安全、软件工程, 研究兴趣包括复杂信息系统及安全工程、软件工程、计算机应用。Email: yhzhi@bit.edu.cn