利用Betweenness Centrality计算网络流量矩阵的新算法

欧 鹏[1]，李志蜀[1]，胡 建[1]，林 珣[2]

(1. 四川大学计算机学院 成都 610064; 2. 西南财经大学经济信息工程学院 成都 610074)

【摘要】 引入Betweenness Centrality中间度核心性作为候选快照的选择指标，特别是以其中的GBC群组中间度核心性作为考量多链路权重改变时各链路的选取问题，实验结果表明，BC的引入加快了秩的提高，而GBC可以衡量群组大小不同时会对原系统产生影响的程度；同时指出将GBC作为唯一指标在实际操作层面存在问题，需要综合考虑其他因素。最后提出将来结合序列RBC与GBC进行计算的研究方向。

关 键 词 中间度核心性； 复杂网络； 贪婪算法； 流量矩阵

中图分类号 TP393 **文献标识码** A doi:10.3969/j.issn.1001-0548.2012.01.029

Novel Algorithm of Introducing Betweenness Centrality into Traffic Matrix Computing

OU Peng[1], LI Zhi-shu[1], HU Jian[1], and LIN Xun[2]

(1. School of Computer Science, Sichuan University Chengdu 610064;
2. School of Information Engineering, Southwestern University of Finance and Economy Chengdu 610074)

Abstract Traffic matrix estimation problem remains one of the research focus for network designer and administrator for many years, especially the traffic estimation of back-bone networks for ISPs. In this article, we introduce betweenness centrality as the measure index of candidate snapshots and group betweenness centrality (GBC) particularly for choosing multiple link weight changes. Our experiments show that the introduction of BC actually accelerates the increase of ranks, and GBC reflects the influences of different group sizes. Some considerations are suggested for further research.

Key words betweenness centrality; complex networks; greedy algorithm; traffic matrix

计算机网络的高速发展，不仅提供了更多样化的服务与更强的可靠性，同时也给网络管理带来了空前的挑战。良好的网络管理能够优化整个网络运行，减轻服务供应商ISP的负担，对用户来说也意味着更好的QoS服务质量。要实现良好的网络管理，拓扑层面上信息的获取和执行很重要，有了拓扑信息，在不同的路由策略下都可以推算出各自的相关数据[1]。这方面的技术已经比较成熟，而本文对流量计算的研究也是基于拓扑结构不变条件，对链路的路由策略进行调整，通过观察前后的数据来估算整个或者大部分网络的流量矩阵。

流量矩阵对于网络管理者进行网络拓扑规划、流量工程等实际工作具有很大的意义[2]，如网络管理员如何使整个网络达到最优化的使用效率，即每一条链路得到充分的使用，不致长时间空闲或者长时间超过负荷，是一个很实际的首要问题。流量矩阵的计算问题自从被学者提出以后就受到相当关注。有了流量矩阵，就能对针对当前的网络拓扑结构，制定出相适应的链路路由结构，使网络的整个流量趋于平衡。但用户的个别数据流可能变化性比较大，如果在一定的时间内是稳定的，则可以固定地用流量矩阵进行监测，因此本文的算法必须避免过多的复杂步骤，或者算法产生的中间变量必须限定在一定的数量内，后面将详细分析。

1 相关工作

最早的计算流量矩阵的方法是在 ISP 层面上直接的链路流量测量 [1]，由于缺乏效率，学者们开始转向数学统计领域的推断方法 [2-5]，该算法在非常有限的信息内计算流量矩阵。

第一代估算流量计算模型参数的数学推断方法

收稿日期： 2011－08－04; 修回日期： 2011－10－25
基金项目：教育部留学回国启动基金(20091341-11-03)
作者简介：欧鹏(1980－)，男，博士生，主要从事计算机网络管理、网络测量方面的研究.

包括：瞬时参数产生方法[2]；贝叶斯方法[3]；最大似然估计[4]。所有这些方法对付流量模型的高度不可控性的思路是，引进施加于OD对(origin-distination pair)的二阶矩上的其他一些限制条件和假设。文献[2-3]假设OD对之间的流量符合泊松分布；文献[4]则假设是符合高斯分布；文献[5]对它们进行了对比，认为以上3个方法都对先验信息过分依赖，因此，产生采用智能先验信息的第二代估算方法[5-7]。

文献[8-15]提出的链路权重改变算法大大降低了推断的基础不可靠性。但该算法由于系首次离开数学统计意义上的推测在实体层面上做改进，因此在具有新颖性的同时不可避免地具有很多不完善，本文针对该系列算法进行改进。

2　流量矩阵计算模型与链路权重改变算法

2.1　基本模型介绍

设链路流量为矢量$\boldsymbol{Y}$，$\boldsymbol{A}$为路由矩阵，$\boldsymbol{X}$为流量矩阵(此时组织为矢量形式)，则$\boldsymbol{Y}=\boldsymbol{AX}$。在场景$k$下，称此时的路由矩阵$\boldsymbol{A}(k)$为一个快照$k$。数学统计推断类算法就是基于某一个快照进行计算，多快照算法将$\boldsymbol{A}(k_i)$集成为$\boldsymbol{A}=[\boldsymbol{A}^{\mathrm{T}}(0),\boldsymbol{A}^{\mathrm{T}}(1),\cdots,\boldsymbol{A}^{\mathrm{T}}(K)]^{\mathrm{T}}$，使之接近满秩，从而避开了统计推断，可以直接计算。

设一个网络的节点为$V=\{v_1,v_2,\cdots,v_N\}$，链路为$L=\{l_1,l_2,\cdots,l_m\}$，相应的权重为$W=\{w_1,w_2,\cdots,w_m\}$，因为$\boldsymbol{A}_0$的列数远远大于行数，则在初始场景快照k_0下利用$\boldsymbol{Y}=\boldsymbol{A}_0\boldsymbol{X}$计算$\boldsymbol{X}$是不实际的。接下来改变链路$l_i$的权重$w_i$为$w_i'$，这有可能导致某OD对的数据路线发生变化，如由$(\cdots\to l_{i-1}\to l_i\to l_{i+1}\to\cdots)$变为$(\cdots\to l_{i-1}\to l_{i+k}\to\cdots)$，则在该快照$k_1$下的$\boldsymbol{A}_1$必然存在不同于$\boldsymbol{A}_0$的行，集成两个快照后，有$\boldsymbol{A}=[\boldsymbol{A}_0{}^{\mathrm{T}},\boldsymbol{A}_1{}^{\mathrm{T}}]^{\mathrm{T}}$的秩必然增加。根据该算法的思想，如果以穷举的方式不断变化链路权重，使$\boldsymbol{A}$达到满秩，则由$\boldsymbol{Y}=\boldsymbol{AX}$可以计算出流量矩阵$\boldsymbol{X}=\boldsymbol{A}^{-1}\boldsymbol{Y}$。

2.2　链路权重改变算法

针对在一个固定场景下计算流量数据的不精确性，文献[8-9]提出在多场景环境下通过变化链路的路由权重，从而动态地计算流量数据的算法。其步骤为：

1) 对每一条l_h确定其候选快照集合$\Delta_h'=\{\delta w_h(p_1),\delta w_h(p_2),\cdots,\delta w_h(p_P)\}$，得到所有候选快照集合累加$\Delta=\sum\limits_{h=1,2,\cdots,m}\Delta_h'$。

2) 将不符合网络运营条件的快照删除，只保留可行的在Δ中。运营条件是指ISP为保证服务层协议SLA而施加于网络上的限制，主要有两个限制：① 延迟限制，保证数据包能在一定的时间内到达目的地；② 流量限制，保证网络的数据链路不会涌入过量的数据流而导致拥塞。因此，以这两个条件来筛选候选快照集合能够使算法实际适用于工作，而不只是停留在理论层面。

3) 在得到可行的候选快照集合Δ后，链路权重改变算法的思想是以网络路由拓扑改变代价最小且对路由矩阵贡献最大的快照先入手。这是因为在实际的网络条件下，运营商ISP不愿意对已经处于良好运行的网络进行变动，而且该变动实施起来也比较复杂。因此算法使用一种启发式搜索算法发现具有链路最大分离度的快照。设当前链路为l_h，通过变化原始权重w_h为w_{h_i}，使原本不经过l_h的OD对p_i改变路径经过l_h，则$w_{h_i}=w_h+\Delta w_h(p_i)$。然后考察该变化的链路分离度大小，设$p_i$重新经过$l_h$后，其余的$p$用$\beta_{h_i}(p)$计算，若$p$也被重新变化到$l_h$上，则$\beta_{h_i}(p)=1$；反之，$\beta_{h_i}(p)=0$，由此设$\Gamma_{h_i}$为所有受到$l_h$改变权重为$w_{h_i}$影响的OD对集合，则$\sum\limits_{p\in\Gamma_{h_i}/p_t}\beta_{hi}(p)$为除了$p_t$以外同时随$p_t$转移到$l_h$上的OD对的数量。$\sum\limits_{p\in\Gamma_{h_i}/p_t}\beta_{h_i}(p)=0$表示只有一个$p_t$转移到$l_h$上，通过将现在$l_h$流量减去前一个快照的$l_h$流量，得到准确的$p_t$值，这也是算法期望看到的结果，即以最小的变化代价得到了最大的分离度(完全分离)。

除了完全分离，为衡量权重改变以后的链路分离度，算法定义了一个分离度指标：

$$m_{h_i}(p_t)=\min_{l_h\in R_{h_i}(p_t)}\sum_{p\in\Gamma_{h_i}/p_t}\beta_{h_i}(p)$$

$$\forall w_{h_i}:(l_h,w_{h_i})\in S$$

其中，$R_{h_i}(p_t)$为p_t在转移到l_h后，其路径上新出现但没包括在转移前路径上的链路集合。可见，$m_{h_i}(p_t)\leqslant\sum\limits_{p\in\Gamma_{h_i}/p_t}\beta_{h_i}(p)$，因为其具有更宽的搜索域。例如，当$l_h$的权重$w_h\to w_{h_i}$后，$p_t$转移至$l_h$，同时$p_x$、$p_y$也一起转移至$l_h$，则$\sum\limits_{p\in\Gamma_{h_i}/p_t}\beta_{h_i}(p)=2$；但存在一条链路$l_{h_i}(p_t)\in R_{h_i}$，其$\sum\limits_{p\in\Gamma_{h_i}/p_t}\beta_{h_i}(p)=1$，则$m_{h_i}(p_t)=1$，小于$\sum\limits_{p\in\Gamma_{h_i}/p_t}\beta_{h_i}(p)$，$m_{h_i}(p_t)$的分离度明

显更好，故被算法使用。将特定的 p_t 扩大范围至 $\varGamma_{h_i}$，得到在 l_h 改变权重为 w_{h_i} 时，对于所有OD对的最大分离度为：

$$M_{h_i}=\sum_{p_t\in\varGamma_{h_i}}(1/(Bm_{h_i}(p_t)+1))$$
$$\forall w_{h_i}:(l_h,w_{h_i})\in S$$

式中，B 为一足够大的整数，使 $P/(B+1)<1$。假若快照 (l_h,w_{h_i}) 引起 x 对PoP数据流改变路径，而每个OD对 p 的 $m_{h_i}(p)=0$，则 M_{h_i} 为最终衡量每一个快照在候选快照集合中排列顺序的指标。因为每一个快照对应着一个 (l_h,w_{h_i}) 变化，而 M_{h_i} 刚好是任何变化的最大分离度指标。每个快照按照 M_{h_i} 的大小顺序排列好，则形成有序的候选快照集合，步骤3)结束。

4) 考察加入新快照后对原有路由矩阵的贡献度。快照被采用的唯一标准是其加入能增加矩阵的秩，否则该快照被丢弃。

5) 在遍历完所有单链路，但权重改变快照仍不满秩的情况下，进行多链路权重改变。多链路权重改变对网络路由的影响比单链路更大，会影响实际的使用效果。因此，对于两条链路的情况，算法更加严格限定在相邻接的两条链路上。不仅如此，权重的改变也只限于都取最小值 $W_{\min}$ 或者都取最大值 $W_{\max}$ 两种情况，$W_{\min}$ 和 $W_{\max}$ 分别为网络运营商所指定的每条链路权重的下限和上限。若仍然不满秩，则以特定方式取3条链路进行考察。该特定方式是以形成“三角形”的3条链路为一个对象，如 $\{(a,b),(b,c),(c,a)\}$ 3条链路首尾相连为一个三角形，同样将权重变化范围规定为 $W_{\min}$ 和 $W_{\max}$。

如果步骤5)完成以后仍然没有达到满秩，则放松对权重的延迟和负载限制条件，重新考虑在步骤2)中被舍弃的快照，放入候选快照集合处理，直到满秩。

3 改进的新GBC算法

3.1 原算法在指标与链路数量选取上的不足与本文的出发点

在排列候选快照顺序的问题上，原算法在每个快照下采用的 $m_{h_i}(p_t)$ 和 M_{h_i} 考量方法不一定具有最佳效率，因为该方法最终搜寻的是所有在当前快照下发生变化的所有链路中具有最大分离度的一条，而与当前快照所基于的 l_h 和 w_{h_i} 关系并不大，l_h 和 w_{h_i} 只是一个实现全搜索的遍历功能的辅助性角色，所以不同快照下出现最大分离度的链路为同一条链路的情况就容易出现，会浪费计算时间，影响算法的效率。本文采用以GBC值进行排列，大大提高了效率。

3.2 新GBC算法对中间度核心性BC的引入

本文算法首先引入一个考量某节点在整个网络结构图中的重要性的指标——路由中间度核心性(routing betweenness centrality，RBC)，采用该指标来间接衡量链路的重要程度，然后按照重要程度排序组成候选快照集合[10]。RBC是文献[11]首次提出来的，对BC(betweenness centrality)已有了很多的研究。BC一开始是出现在社会网的研究领域，认为一条路径的中间成员对两端的成员具有“更大的人际关系影响”，即如果一个成员位于其他成员的多条最短路径上，该成员就是核心成员，就具有较大的中间度核心性[13]。而文献[11]使用RBC将SPBC(shortest path betweenness centrality)、LC(load centrality)和FBC(flow betweenness centrality)等3种主要路由方式下的betweenness centrality进行概括，通过四元组 $R(s,u,v,t)=p$ 表达各种路由方式，其中s为源节点，t为目标节点，则OD对 (s,t) 到达u以后再通过v的概率为p。若 ϕ 表示任何值，则 $R(\phi,u,v,\phi)=0$ 表示u没有一条到v的链路，而 $R(\phi,v,v,\phi)$ 的值就是1。既然RBC能够提供网络各节点和链路的数据流负载状况信息，则可考虑如何使用RBC优化流量矩阵的计算步骤。RBC的计算、分析和应用都是建立在具有power-law幂律属性的scale-free网络中[12]，在计算群RBC时，都认为当前研究的网络具有scale-free特性，按照普遍的研究结果，表明互联网是具有scale-free特征的[12]，所以后面将不探讨该前提而是直接认为在当前的条件下成立，可以进行各种RBC计算。

文献[14]能够很好地反映当前网络中的节点的负荷状态，对于节点 k，$b_k=\sum r(s,k,d)$ 表示k所在的网络中所有经过k的OD对的数量。如果 b_k 的值比较高，说明节点k位于大量OD对的最短路径上，因此其权重值的改变很容易引起整个网络的路由改变。在文献[13]的基础上，定义BC为 $C_k=\sum\limits_{s\in V}\sum\limits_{d\neq s\in V}r_{sd}(k)/r_{sd}$，其中$V$为网络中所有节点的集合。可见BC比betweenness的值更精确，因为后者只是考虑了链路的数量，对于具有多条最短路径的OD对误差较大。如果一个节点的BC很高，则任何关于它的变动都会大面积影响整个网络。如果由一些节点组成的逻辑上的组节点的BC明显偏高，则可以认为这几个节点是整个网络的敏感部分，也是进

行链路权重改变的很好的对象。

3.3　新GBC算法中3类RBC的计算

本文将RBC的计算对象归为节点、序列和集合3种。若一个数据包的起点为s，终点为t，则其经过节点v的概率$\delta_{s,t}(v)$称为(s, t)节点对中间节点v的成对依赖性，明显有$\delta_{s,t}(s)=\delta_{s,t}(t)=1$。设该OD对$(s,t)$在$v$节点前经过的节点为$\mathrm{Pred}_{s,t}(v)$，由当前的路由策略$R(s,u,v,t)$可知：

$$\delta_{s,t}(s)=1$$

$$\delta_{s,t}(v)=\sum_{u\in \mathrm{Pred}_{s,t}(v)}\delta_{s,t}(u)R(s,u,v,t)$$

将v的计算转化为对u的计算，如此反复不断地回溯计算直到s节点，考虑走遍所有边的最差情况，故算法的复杂度为$O(m)$，m为网络中所有边的数量。则节点v的RBC为网络中所有数据流经过v的数据量：

$$\delta_{\bullet,\bullet}(v)=\sum_{s,t\in V}\delta_{s,t}(v)T(s,t)$$

式中，$T(s,t)$为s和t之间的流量。

序列RBC计算网络中所有数据流经过某序列的数据量，设该序列为$S=(s_1,s_2,\cdots,s_k)$，$\tilde{\delta}_{s,t}(S)$为s到t的过程中经过S内所有节点(即经过s_1，再经过s_2，最终经过s_k)的概率，则$\tilde{\delta}_{s,t}(S)T(s,t)$为所有由$s$开始，最后经过$t$的数据流经过$S$的数量。可以反复回溯计算序列RBC：

$$\tilde{\delta}_{s,t}((s))=1$$

$v_{k-1}=v_k$时，$\tilde{\delta}_{s,t}((\cdots,v_{k-1},v_k))=\tilde{\delta}_{s,t}((\cdots,v_{k-1}))$

$v_{k-1}\neq v_k$时，

$$\tilde{\delta}_{s,t}((\cdots,v_{k-1},v_k))=\sum_{u\in \mathrm{Pr}ed_{s,t}(v_k)}\tilde{\delta}_{s,t}((\cdots,v_{k-1},u))R(s,u,v,t)$$

算法复杂度同样为$O(m)$。所以一个序列的RBC为所有数据流中按照序列内部前后顺序通过序列的数量：

$$\tilde{\delta}_{\bullet,\bullet}(S)=\sum_{s,t\in V}\tilde{\delta}_{s,t}(S)T(s,t)$$

采用序列RBC估算监视节点的冗余度是最近几年新提出的一个研究课题，但传统的序列RBC计算算法在冗余度的估算上仍然不够优化，这主要是因为传统RBC计算算法使用的是有序序列，但网络管理者更多是关心序列内哪几个监测节点能监测多少流量，以及其重复监测的程度等。

3.4　新GBC算法中GBC的计算

集合RBC是本文研究的重点，因为比起序列RBC来，它能反映出一部分节点在网络中的关键程度。集合RBC的定义是，如果一个数据包至少经过集合内任何一个节点，则被计算入该集合RBC的值。核心性(centrality)最初被应用是在网络的节点集合研究中(Everett and Borgatti[1999])。设一个节点集合为$M=\{v_0,v_1,\cdots,v_k\}$，由s至t的数据包至少经过M中一个节点的概率为$\ddot{\delta}_{s,t}(M)$，则$\ddot{\delta}_{s,t}(M)T(s,t)$为经过$M$的数据流数量，从而集合$M$的RBC为：

$$\ddot{\delta}_{\bullet,\bullet}(M)=\sum_{s,t\in V}\ddot{\delta}_{s,t}(M)T(s,t)$$

本文以GBC表示群RBC(group RBC)，GBC的概念比集合RBC更广泛、更复杂，GBC的群不一定只由节点组成，而可以由链路、路由器等组成，或者既包含节点又有链路、路由器，因此GBC在复杂网络的分析设计中有很大作用[3,15]。文献[12]指出网络服务提供商(NSP)的骨干网基础结构是一种复杂网络，并且呈现scale-free特征[3,12]。可见，若一组链路、路由器组成的群拥有最高的GBC值，则可以认为该组链路、路由器在NSP骨干网络中具有最大的影响。文献[14]将寻找一个scale-free网络中的核心节点群问题称为KPP-Com问题，并给出了贪婪算法和启发式搜索算法。

设网络为$G=(V,E)$，其中V为n个节点，E为m条边，一组节点$M\subseteq V$，则GBC(M)为：

$$\mathrm{GBC}(M)=\sum_{s,t\in V|s\neq t}\left(\frac{\ddot{\delta}_{s,t}(M)}{\delta_{s,t}}\right)$$

在网络中寻找一个具有最高GBC的组群已经在最小节点覆盖问题中被证明是一个NP困难问题[2]，必须采用某种启发式算法以亚多项式时间复杂度来搜索。

3.5　新算法实际操作步骤

文本以多链路权重改变为排列候选快照顺序的第一步，所以需要找到影响最大的多链路组成的组群。良好的启发式搜索算法能够大大减少需要搜索的群组数量，缩短整体搜索时间。本文先尝试贪婪算法，再使用改进的启发式算法进行比较。

采用贪婪算法的步骤为：

1) 对所有节点计算其RBC值并排序；

2) 以设定的群组大小k，选取前k个节点作为最大GBC的群组；

3) 以GBC值递减组成大小仍为k的群组序列，形成候选快照集合。

改进的贪婪算法的步骤为：

1) 对所有节点计算其RBC值并排序；

2) 确定第一个RBC最大值的节点，以后继的节

点以和已经确定的节点组成的群组RBC值最大作为选择的标准，一次加入一个节点直到k个节点；

3) 以GBC值递减组成大小仍为k的群组序列，形成候选快照集合。

贪婪算法的不足在于所计算出来的最大GBC不一定是真正的最大GBC。因此，本文考虑采用决策树算法来搜索。首先按照各自的RBC值对节点进行排序，值高的排在前位，反映在决策树上就是最上面的节点。要添加该节点入候选，则往右子树前进，右子树的候选中就被占用一个位置；考虑到RBC值最高的节点不一定形成最高的GBC值，在左子树去掉它，此时左子树候选仍然为空，形成一个决策树。若设定的GBC群组大小为k，则当前子树候选集已经到k时不再向右子树前进。考虑到遍历某些子树不会找出比当前最高GBC更高的值，继续前进只会浪费时间、空间资源，因此定义一个界限值，该界限值为当前GBC(可能节点数量小于k)加上候选中可能最大的GBC值。设当前子树根节点上已经加入了m个节点，则GBC(m)已经确定，剩下的n个节点中需要再加入($k-m$)个节点，判断当前可能的最大GBC(k)有4种方法：

1) 将剩下的n个节点按照各自的 RBC 值排序$\{x_1,x_2,\cdots,x_n\}$，$\mathrm{GBC}(k)=\mathrm{GBC}(m)+\mathrm{BC}(x_1)(k-m)$；

2) 逐步累加($k-m$)个节点，$\mathrm{GBC}(k)=\mathrm{GBC}(m)+\sum_{i=1}^{n}BC(x_i)$；

3) 将剩下的n个节点按照各自对当前已经确定的 m 个节点组成的群组的 GBC 贡献值排序：$\{y_1,y_2,\cdots,y_n\}$，$\mathrm{GBC}(k)=\mathrm{GBC}(m)+\mathrm{GBC}^{m}(y_1)(k-m)$；

4) 逐步累加($k-m$)个节点，$\mathrm{GBC}(k)=\mathrm{GBC}(m)+\sum_{i=1}^{n}\mathrm{GBC}^{m\cup y_{i-1}}(y_i)$。

很明显，方法2)比方法1)更精确，方法4)比方法3)更精确。因为GBC贡献值更能反映加入节点后的状况，故方法4)为最佳判断标准，被采用为启发式算法的界限值。

但同时值得提出的是，贪婪算法虽然搜索的精确性不是最好，但其计算具有多项式时间复杂度，而用决策树进行搜索是级数时间复杂度，高于多项式时间复杂度。

4 结果分析

本文在NS2模拟环境下进行仿真，实验网络由20个节点、45条链路组成，可见有$20\times(20-1)=380$个OD对。以$k=3$作为GBC群组大小。各算法对提高路由矩阵的秩的性能比较如图1所示。可以看出，采用了GBC作为选择和排列候选快照的指标后，在一定的快照数量下，秩的提高更快，达到一定比例的满秩所需要的快照数量也少于原算法。群组大小对路由矩阵的秩的影响如图2所示。可以看出，在k的选择上，其值越大则整体的秩提高得越快。但是k越大对网络路由结构造成的改变越大。虽然能大幅度提高秩，但实现难度很大，k每增加1都会造成大量链路超出负荷，所以候选快照集合反而大大缩小，还有可能达不到满秩或者可能一直停留在低秩层面。

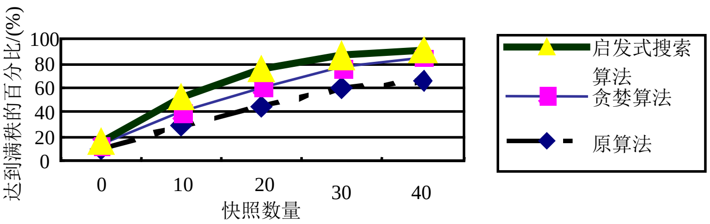

图1 各算法对提高路由矩阵的秩的性能比较

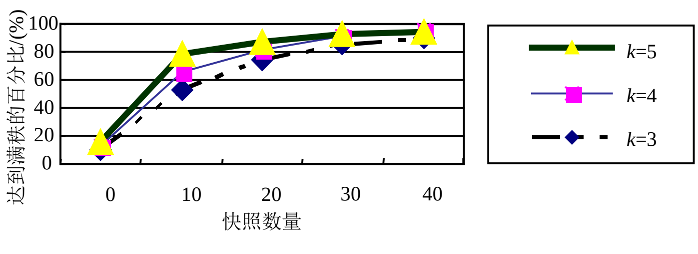

图2 群组大小对路由矩阵的秩的影响

5 结 论

本文以流量矩阵的计算为背景，通过分析链路权重改变算法，以加速提高路由矩阵的秩为目标。认为原算法在排列候选快照方面，其指标选择和同时改变链路数量均存在不足。本文通过引入中间度核心性RBC考量在网络中扮演关键角色的节点，特别是将群组中间度核心性GBC和多链路权重改变相结合，从而能够衡量多链路快照对提高秩的贡献率的大小。以GBC对多链路快照进行排序，对原算法只是简单将多链路权重同时设为最低值和最高值相比，不仅更加精确，而且更加有效，因为原算法不能预测加入该快照后整体秩的变化情况，而GBC本身就是体现现有快照对原有快照共享程度的指标。实验结果表明，GBC的引入比原算法对秩的提高帮助较大，GBC群组数量越大则效果越明显，但受实际情况影响群组数量不易过大。

序列RBC可以用作考量几个节点流量的相似性的指标，在排除冗余候选快照方面比较有效[16]，引入RBC必然进一步缩短候选快照的搜寻时间和候选集合的大小，提高链路权重改变算法的整体效能。

未来的研究方向可以结合序列RBC和GBC同时对复杂网络[17-18]进行分析。

参 考 文 献

[1] PAPAGIANNAKI K, TAFT N, LAKHINA A. A distributed approach to measure IP traffic matrices[C]//Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement. [S.l.]: ACM, 2004:161-174.

[2] VARDI Y. Estimating source-destination traffic intensities from link data[J]. Journal of the American Statistical Association, 1996, 91(433): 385-397.

[3] TEBALDI C, WEST M. Bayesian inference of network traffic using link count data[J]. Journal of the American Statistical Association, 1998, 93(442): 557-576.

[4] CAO J, DAVIS D, WEIL S V, et al. Time-varying network tomography: router link data[J]. Journal of the American Statistical Association, 2000, 95(452): 1053-1068.

[5] MEDINA A, TAFT N, SALAMATIAN K, et al. Traffic matrix estimation: Existing techniques and new directions [C]//Proceedings of ACM SIGCOMM. Pitsburgh, PA: ACM, 2002.

[6] ZHANG Y, ROUGHAN M, DUFFIELD N, et al. Fast accurate computation of large-scale IP traffic matrices from link loads[C]//Proceedings of ACM Sigmetrics. [S.l.]: ACM, 2003.

[7] ZHANG Y, ROUGHAN M, LUND C, et al. An information theoretic approach to traffic matrix estimation[C]// Proceedings of the ACM SIGCOMM. Karlsruhe, Germany: ACM, 2003.

[8] NUCCI A, CRUZ R, TAFT N, et al. Design of IGP link weight changes for estimation of traffic matrices[C]// Proceedings of IEEE INFOCOM. [S.l.]: IEEE, 2004.

[9] YOOK S H, JEONG H, BARABASI A L. Modeling the internet’s large-scale topology[C]//Proceedings of the National Academy of Sciences of the United States of America. [S.l.]: [s.n.], 2002.

[10] ZHOU T, LIU J G, WANG B H. Notes on the algorithm for calculating betweenness[J]. Chinese Phys Lett, 2006, 23: 2327-2329.

[11] DOLEV S, ELOVICI Y, PUZIS R. Routing betweenness centrality[J]. Journal of the ACM, 2010, 57(4): 1-27.

[12] BARABASI A L, ALBERT R. Emergence of scaling in random networks[J]. Science, 1999, 286: 509-512.

[13] FREEMAN L C. A set of measures of Centrality based upon betweenness[J]. Sociometry, 1977, 40: 35-41.

[14] BORGATTI S P. Identifying sets of key players in a social network[J]. Computational and Mathematical Organization Theory, 2006, 12(1): 21-34.

[15] PUZIS R, ELOVICI Y, DOLEV S. Finding the most prominent group in complex networks[J]. AI Communications, 2007, 20(4): 20-30.

[16] OU Peng, LI Zhi-shu; HU Jian, et al. An improved link weight change algorithm for traffic matrix estimation[C]//Proceedings of IFITA’2009. Chengdu: IEEE, 2009.

[17] SOULE A, NUCCI A, CRUZ R L, et al. Estimating dynamic traffic matrices by using viable routing changes[J]. IEEE/ACM Transactions on Networking, 2007, (3): 485-498.

[18] BARABASI A L, ALBERT R, JEONG H. Scale-free characteristics of random networks: the topology of the world-wide web[J]. Physica A: Statistical Mechanics and its Applications, 2000, 281(1-4): 69-77.

编 辑 税 红