

DOI: 10.7652/xjtuxb201812009

# 大规模网络图中 4 节点子图数量快速估计算法

覃遵颖<sup>1,2</sup>, 孙雨<sup>1,2</sup>, 李国栋<sup>1,2</sup>, 齐怀睿<sup>3</sup>, 陶敬<sup>1,2</sup>

(1. 西安交通大学智能网络与网络安全教育部重点实验室, 710049, 西安; 2. 西安交通大学网络信息中心, 710049, 西安; 3. 西安交通大学电子与信息工程学院, 710049, 西安)

**摘要:** 针对大规模网络图中存在的大量 4 节点子图数量难以精确统计分析的问题,提出了一种大规模网络图中 4 节点子图数量快速估计算法(SmartMoss)。该算法通过随机变量方差分析技术对比 3 路径采样算法(3PS)和中心 3 路径采样算法(C3PS)两种前沿算法的估计误差得出其各自不同的适用范围,进而通过计算被测网络图权重密度分布与误差实时选择使用 3PS 算法或 C3PS 算法对网络图中 4 节点子图进行快速采样,通过采样比例混合 3PS 算法与 C3PS 算法的估计结果实现对网络图中各 4 节点子图出现数量的快速估计。实验结果表明,在同等估计误差下提出的 SmartMoss 算法比已有 3PS 算法和 C3PS 算法快 10 倍以上。SmartMoss 算法可以实现对大规模网络图中 4 节点子图数量进行快速准确的估计,同时为网络社团演化和恶意代码检测等实际应用提供一定的理论参考。

**关键词:** 网络图;采样算法;子图数量估计

**中图分类号:** TP301.6   **文献标志码:** A   **文章编号:** 0253-987X(2018)12-0057-06

## A Fast Mining Algorithm for 4-Node Graphlets in Large Graphs

QIN Zunying<sup>1,2</sup>, SUN Yu<sup>1,2</sup>, LI Guodong<sup>1,2</sup>, QI Huairui<sup>3</sup>, TAO Jing<sup>1,2</sup>

(1. Ministry of Education Key Lab for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an 710049, China; 2. Center of Network, Xi'an Jiaotong University, Xi'an 710049, China; 3. School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

**Abstract:** A fast method (SmartMoss) for efficiently estimating the frequencies of 4-node graphlets in large-scale networks is proposed to address the challenge of exactly counting the frequencies of 4-node graphlets occurring in large-scale networks. SmartMoss compares the estimation errors of the two state-of-the-art algorithms, 3PS (3-Path Sampling) and C3PS (Center 3-Path Sampling), through random variables variance analysis to obtain their different application ranges, and then selects either 3PS algorithm or C3PS algorithm for 4-node graphlets sampling by calculating the distribution of network weight densities and real-time errors, to achieve quick estimation of the frequencies of 4-node graphlets by combing the estimation results of 3PS and C3PS based on the sampling proportions. Experimental results show that SmartMoss is more than 10 times faster than the state-of-the-art algorithms 3PS and C3PS under the same estimation error. SmartMoss efficiently and accurately estimates the frequencies of 4-node graphlets for large network graphs, and provides theoretical reference for practical applications such as

收稿日期: 2017-10-28。 作者简介: 覃遵颖(1985—),女,工程师;陶敬(通信作者),男,高级工程师,硕士生导师。 基金项目: 深圳市基础研究计划资助项目(JCYJ20160229195940462);陕西省自然科学基金基础研究计划资助项目(2016JQ6034, 2017JM6095);浙江省自然科学基金资助项目(LGG18F 020016)。  
网络出版时间: 2018-10-31      网络出版地址: <http://kns.cnki.net/kcms/detail/61.1069.T.20181029.1327.004.html>

network community evolution and malicious code detection.

**Keywords:** network graph; sampling algorithm; graphlet counting

在大型复杂网络的子图集合中,存在着大量包含3~5个节点的小型无向子图,这类子图能够反映复杂网络的一些基础结构特性,对此类子图的数量进行挖掘分析,在生物学<sup>[1-2]</sup>、社会学、社交网络<sup>[3-6]</sup>和万维网分析<sup>[7-8]</sup>等领域都有着重要作用。例如,可以将具有特定功能的氨基酸团定义为蛋白质结构网络图中的一类小型无向子图<sup>[1-2]</sup>,对这类子图进行数量统计,是认定蛋白结构、推定未知蛋白功能性质等工作的前提;类似地,将小规模用户之间的关系抽象为在线社交网络中的一类小型无向子图<sup>[9]</sup>,对这类子图的数量进行统计,可以为分析在线社交网络中社团演化、用户聚类等工作提供思路。特别地,对于恶意代码网络图<sup>[10-12]</sup>,研究人员可以将任意小规模模块间的调用关系抽象为小型无向子图,对这些子图进行数量统计,可以推定未知软件中存在恶意代码的可能性,且响应速度和准确性都优于传统的文本检测方法。

Wang 等的算法<sup>[13]</sup>对3节点子图数量的估计已经取得了比较好的性能,然而对于4节点子图数量的估计仍面临着较大的挑战,4节点子图定义结构如图1所示。现有的大部分4节点子图数量估计算法时间与空间复杂度过高,在实际使用中难以实现。为了解决这一问题,Jha 等提出了3PS和C3PS算法<sup>[14]</sup>,该算法使用节点采样估计的方法减小了计算量,然而却并未对估算结果的误差范围进行严谨地分析,未能从理论上严谨的证明C3PS算法一定能提高3PS算法的估算精度。

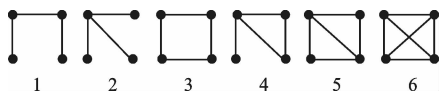


图1 4节点子图定义结构图

## 1 C3PS算法和3PS算法的误差分析

### 1.1 3PS算法及其误差分析

3PS算法采用基于带权重的采样技术。首先,对任一节点 $v, v \in D, D$ 为网络图的节点集合,记其度为 $d_v$ ,其邻居集合为 $N_v$ ,定义其权重 $\Gamma_v$ 为其所有邻居的度之和与自身度的乘积,即 $\Gamma_v = (d_v - 1) \cdot \sum_{x \in N_v} (d_x - 1)$ ,定义其节点权重密度为 $\pi_v = \Gamma_v / \Gamma$ ,其中 $\Gamma$ 为网络图的总权重,即 $\Gamma = \sum_{v \in V} \Gamma_v$ 。

3PS算法的一次采样过程主要为5个步骤。

**步骤1** 根据节点权重密度分布 $\pi = \{\pi_v : v \in D\}$ ,从节点集 $D$ 中采样出节点 $v$ ;

**步骤2** 根据节点 $v$ 的邻居集合 $N_v$ 的权重密度分布 $\sigma^{(v)} = \{\sigma_u^{(v)} : u \in D\}$ ,从 $N_v$ 中随机采样出节点 $u$ ,其中 $\sigma_u^{(v)}$ 被定义为

$$\sigma_u^{(v)} = \frac{d_u - 1}{\sum_{x \in N_v} (d_x - 1)}, u \in N_v \quad (1)$$

**步骤3** 从节点 $v$ 邻居集合剩下的元素 $N_v - \{u\}$ 中随机采样出节点 $w$ ;

**步骤4** 从节点 $u$ 邻居集合剩下的元素 $N_u - \{v\}$ 中随机采样出节点 $r$ ;

**步骤5** 判断节点 $u, v, w, r$ 能够构成的连通生成子图种类。

在3PS算法的一次采样过程中,样本中节点 $u, v$ 的度均要大于等于2,这说明一个4节点子图若要被3PS算法采样到,则其中至少要包含两个度大于等于2的点。根据图1中6种4节点子图的结构,第2种4节点子图中仅包含1个度为3与3个度为1的点,不符合采样条件,因此重复以上步骤 $K$ 次得到网络图的 $K$ 子图采样,可对第1,3,4,5,6种4节点子图的频数进行采样估计,估算结果为

$$\begin{aligned} \hat{n}_i &= \frac{m_i}{K p_i}, i \in \{1, 3, 4, 5, 6\} \\ \hat{n}_2 &= \Lambda_3 - \hat{n}_1 - 4 \hat{n}_6 \end{aligned} \quad (2)$$

式中: $m_i$ 为样本集 $K$ 中第 $i$ 种4节点子图的数量; $p_i$ 为一次采样中第 $i$ 种4节点子图能被采样到的概率, $p_i = \phi_i / \Gamma$ ,其中 $\phi_1 = 1, \phi_2 = 0, \phi_3 = 4, \phi_4 = 2, \phi_5 = 6, \phi_6 = 1, \phi_i$ 为第 $i$ 种4节点子图能被3PS算法采样到的次数; $\Lambda_3$ 为网络图中度为3的点的数量。

3PS算法的估算误差满足如下定理。

**定理1** 记 $\hat{n}_i$ 是第 $i$ 种4节点子图使用3PS算法得到的采样估计值, $n_i$ 为真实值, $1 \leq i \leq 6$ ,则 $\hat{n}_i$ 的方差为

$$V(\hat{n}_i) = \frac{n_i}{K} \left( \frac{1}{p_i} - n_i \right), i \in \{1, 3, 4, 5, 6\} \quad (3)$$

特别地, $\hat{n}_2$ 的方差满足

$$\begin{aligned} V(\hat{n}_2) &= \\ \frac{1}{K} & \left( \frac{n_4}{p_4} + 4 \frac{n_5}{p_5} + 16 \frac{n_6}{p_6} - (n_4 + 2n_5 + 6n_6)^2 \right) \end{aligned} \quad (4)$$

**证明** 记 $P(X)$ 为事件 $X$ 成立的概率, $G_{s_k}$ 为样本 $s_k$

中 4 节点所能构成的连通生成子图的种类。当事件  $Y$  为真时,记  $A(Y)=1$ ,当事件  $Y$  为假时,记  $A(Y)=0$ ,则对于  $i \in \{1,3,4,5,6\}$ ,且  $k \leq K$ ,有

$$P(G_{s_k} = i) = \sum_{s \in C} P(s_k = s) A(G_s = i) = p_i n_i \quad (5)$$

由于样本  $s_k$  由随机采样而得,因此  $m_i$  服从二项分布

$$P(m_i = x) = \binom{K}{x} (p_i n_i)^x (1 - p_i n_i)^{K-x} \quad x = 0, \dots, K \quad (6)$$

$m_i$  的期望与方差分别为

$$\left. \begin{aligned} E(m_i) &= K p_i n_i \\ V(m_i) &= K p_i n_i (1 - p_i n_i) \end{aligned} \right\} \quad (7)$$

$\hat{n}_i$  的期望与方差分别为

$$\left. \begin{aligned} E(\hat{n}_i) &= E\left(\frac{m_i}{K}\right) = \frac{E(m_i)}{K p_i} = n_i \\ V(\hat{n}_i) &= V\left(\frac{m_i}{K p_i}\right) = \frac{V(m_i)}{K^2 p_i^2} = \frac{n_i}{K} \left(\frac{1}{p_i} - n_i\right) \end{aligned} \right\} \quad (8)$$

特别地,可计算  $\hat{n}_2$  的期望为

$$\begin{aligned} E(\hat{n}_2) &= E(\Lambda_3 - \hat{n}_4 - 2\hat{n}_5 - 4\hat{n}_6) = \\ &\Lambda_3 - E(\hat{n}_4) - 2E(\hat{n}_5) - 4E(\hat{n}_6) = \\ &\Lambda_3 - n_4 - 2n_5 - 6n_6 = n_2 \end{aligned} \quad (9)$$

为了计算  $\hat{n}_2$  的方差,首先需要计算  $\hat{n}_i, \hat{n}_j$  的协方差,  $i \neq j, i, j \in \{1,3,4,5,6\}$

$$\begin{aligned} C(\hat{n}_i, \hat{n}_j) &= C\left(\frac{m_i}{K p_i}, \frac{m_j}{K p_j}\right) = \\ &\frac{C\left(\sum_{k=1}^K A(G_{s_k} = i), \sum_{l=1}^K A(G_{s_l} = j)\right)}{K^2 p_i p_j} = \\ &\frac{\sum_{k=1}^K \sum_{l=1}^K C(A(G_{s_k} = i), A(G_{s_l} = j))}{K^2 p_i p_j} = \\ &\frac{\sum_{k=l}^K C(A(G_{s_k} = i), A(G_{s_l} = j))}{K^2 p_i p_j} = -\frac{n_i n_j}{K} \end{aligned} \quad (10)$$

其中

$$\begin{aligned} C(A(G_{s_k} = i), A(G_{s_l} = j)) &= \\ E(A(G_{s_k} = i) A(G_{s_l} = j)) - \\ E(A(G_{s_k} = i)) E(A(G_{s_l} = j)) &= \\ 0 - p_i n_i p_j n_j &= -p_i n_i p_j n_j \end{aligned} \quad (11)$$

且当  $k \neq l$  时,有  $C(A(G_{s_k} = i), A(G_{s_l} = j)) = 0$ ,因此,  $\hat{n}_2$  的方差为

$$V(\hat{n}_2) = V(\Lambda_3 - \hat{n}_4 - 2\hat{n}_5 - 4\hat{n}_6) =$$

$$\begin{aligned} V(\hat{n}_4 + 2\hat{n}_5 + 4\hat{n}_6) &= V(\hat{n}_4) + 4V(\hat{n}_5) + \\ 16V(\hat{n}_6) &+ 4C(\hat{n}_4, \hat{n}_5) + 8C(\hat{n}_4, \hat{n}_6) + 16C(\hat{n}_5, \hat{n}_6) \end{aligned} \quad (12)$$

$$V(\hat{n}_2) = \frac{1}{K} \left( \frac{n_4}{p_4} + 4 \frac{n_5}{p_5} + 16 \frac{n_6}{p_6} - (n_4 + 2n_5 + 6n_6)^2 \right) \quad (13)$$

## 1.2 C3PS 算法及其误差分析

C3PS 算法是专门针对第 3,5,6 种 4 节点子图提出的数量估计算法。对于任一节点  $v$ ,记  $N_{u,v}$  为比节点  $v$  拥有更多邻居的节点  $u$  的邻居的集合,即  $N_{u,v} = \{x \in N_u \mid d_x > d_v\}$ ,记  $L_{u,v}$  为  $N_{u,v}$  集合中节点的数量,即  $L_{u,v} = |N_{u,v}|$ 。记  $\tilde{r}_v$  为节点  $v$  的权重,则  $\tilde{r}_v = \sum_{x \in N_v} L_{v,x} L_{x,v}$ 。对于整个网络图,有总权重  $\tilde{r} = \sum_{v \in N_v} \tilde{r}_v$ ,密度  $\tilde{\pi}_v = \tilde{r}_v / \tilde{r}$ 。

C3PS 算法的一次采样过程同样为 5 个步骤。

**步骤 1** 根据网络图节点权重密度分布  $\tilde{\pi} = \{\tilde{\pi}_v : v \in D\}$ ,从节点集  $D$  中采样出节点  $v$ ;

**步骤 2** 根据节点  $v$  的邻居集合  $N_v$  的节点权重密度分布  $\tilde{\sigma}_u^{(v)} = \{\tilde{\sigma}_u^{(v)} : u \in N_v\}$ ,从  $N_v$  中随机地采样出节点  $u$ ,其中  $\tilde{\sigma}_u^{(v)}$  分布被定义为

$$\tilde{\sigma}_u^{(v)} = \frac{L_{u,v} L_{v,u}}{\tilde{r}_v}, \quad u \in N_v \quad (14)$$

**步骤 3** 从  $N_{v,u}$  中随机采样出节点  $w$ ;

**步骤 4** 从  $N_{u,w}$  中随机采样出节点  $r$ ;

**步骤 5** 获取节点  $u, v, w, r$  构成连通的生成子图种类。

在 C3PS 算法的一次采样过程中若要采样到一个 4 节点子图,则其中至少要有两个点满足其  $N_{u,v}$  集合不为空。根据图 1 中 4 节点子图的结构,仅有第 3,5,6 种 4 节点子图符合采样条件。因此,C3PS 算法通过重复以上步骤得到  $K$  子图采样,对第 3,5,6 种 4 节点子图数量估计如下

$$\tilde{n}_i = \sum_{k=1}^K \frac{\tilde{m}_i}{K \tilde{p}_i}, \quad i \in (3,5,6) \quad (15)$$

式中: $\tilde{m}_i$  为样本集  $\tilde{K}$  中第  $i$  种 4 节点子图的数量; $\tilde{p}_i$  为一次采样中第  $i$  种 4 节点子图能被采样到的概率, $\tilde{p}_i = \tilde{\phi}_i / \tilde{r}$ ,其中  $\tilde{\phi}_3 = 2, \tilde{\phi}_5 = 2, \tilde{\phi}_6 = 6$ 。

C3PS 算法的估算误差满足如下定理。

**定理 2** 记  $\tilde{n}_i$  为 C3PS 算法对第  $i$  种 4 节点子图数量的估计值, $n_i$  为真实值,  $1 \leq i \leq 6, \tilde{n}_i$  的方差为

$$V(\tilde{n}_i) = \frac{n_i}{\tilde{K}} \left( \frac{1}{\tilde{p}_i} - n_i \right), \quad i \in (3,5,6) \quad (16)$$

其证明与定理 1 类似,不再赘述。

### 1.3 复杂度分析

3PS 算法的复杂度主要来自于采样算法的前 4 个步骤。步骤 1 中,为了计算出所有节点的采样权重,其复杂度为  $O(|L|)$ ,  $L$  为网络图中边的集合。步骤 2 中,为了能够根据节点的采样权重采样出节点  $v$ ,其复杂度为  $O(\lg|D|)$ ,  $D$  为网络图中节点的集合。步骤 3 中,从  $v$  的邻居集中采样出节点  $u$ ,所需要的计算复杂度是  $O(\lg|d_v|)$ 。最后,对节点  $w$  进行采样的计算复杂度为  $O(1)$ 。因此,3PS 算法的计算复杂度为  $O(|L| + K\lg|D|)$ ,  $K$  为样本集的大小。

C3PS 算法可由 3PS 算法扩展而得,扩展过程中只需改变节点权重分布的计算方法即可,采样过程中其他部分所需要的复杂度不变。对于计算  $\tilde{\Gamma}$  和  $\pi_v$ ,有  $O(\sum_{v \in V} d_v^2) \geq O(2|L|^2)$ ,而 3PS 算法一次采样的复杂度为  $O(\lg|L|)$ ,因此 C3PS 算法初始化计算节点权重的复杂度为 3PS 算法一次采样的  $2|L|^2/\lg|L|$  倍。

## 2 C3PS 和 3PS 算法的准确度比较

文献[14]中指出,在相同的条件下,C3PS 算法对第 3,5,6 种 4 节点子图的数量估计准确率显著高于 3PS 算法,可以使用 C3PS 算法来提高 3PS 算法估算结果的准确度。然而,根据误差分析可以看出,C3PS 算法能否提高 3PS 算法的准确率与网络图的性质有关。根据定理 1 和定理 2 可以看出,当样本集大小相同,即  $\tilde{K}=K$  时,有

$$\frac{V(\hat{n}_i)}{V(\tilde{n}_i)} = \frac{1/p_i - n_i}{1/\tilde{p}_i - n_i} \approx \frac{\tilde{p}_i}{p_i}, i \in \{3, 5, 6\} \quad (17)$$

式中:  $\tilde{p}_3/p_3 = \Gamma/(4\tilde{\Gamma})$ ,  $\tilde{p}_5/p_5 = \Gamma/(6\tilde{\Gamma})$ ,  $\tilde{p}_6/p_6 = \Gamma/(4\tilde{\Gamma})$ 。因此,可以通过计算两种算法的网络图权重密度比值  $\Gamma/(4\tilde{\Gamma})$  来确定是否需要使用 C3PS 算法来降低 3PS 算法对于  $n_3$ 、 $n_5$ 、 $n_6$  估算的误差。

对于任一网络图,若其  $\Gamma/\tilde{\Gamma} \geq 4$ ,则使用 C3PS 算法来减小 3PS 算法对  $n_3$ 、 $n_6$  的估算误差,若其  $\Gamma/\tilde{\Gamma} \geq 6$ ,则使用 C3PS 算法来减小 3PS 算法对  $n_5$  的估算误差。否则,使用 C3PS 算法不能减小 3PS 算法的估算误差。例如,对于网络图 Ca-GrQc<sup>1</sup>,  $\Gamma/\tilde{\Gamma} = 5.5$ ,则理论上使用 3PS 算法对  $n_3$ 、 $n_5$ 、 $n_6$  的估算误差分别是使用 C3PS 算法的 1.38 倍、0.91 倍和 1.38 倍。在实际实验中可以看出,使用 C3PS 算法后  $n_3$ 、 $n_6$  的误差有了明显减小,约为使用 3PS 算法误差的 72%,然而  $n_5$  的误差相比使用 3PS 算法的

情况却偏大,约为使用 3PS 算法的 109%。这与我们的理论推断也是一致的,验证了 C3PS 算法能否提高 3PS 算法的准确率是与网络图的权重性质密切相关的这个结论。

## 3 SmartMoss 优化算法

本文设计了一种性能更好的 4 节点子图数量估计 SmartMoss 算法。

本文算法包含了 3PS 和 C3PS 两个采样器,能够根据网络图的权重密度分布与当前估算误差实时调整采样策略,最后通过对两个采样器的采样结果  $\hat{n}_i$  和  $\tilde{n}_i$  进行融合,进而估计网络图中所有 4 节点子图的数量  $n_i$ 。

首先介绍本文算法对  $i=3, 5, 6$  这 3 种 4 节点子图数量的估计算法。由于 3PS 和 C3PS 两个采样器均可以对这 3 种 4 节点子图数量进行估计,定义如下

$$\hat{\lambda}_i = \frac{V(\tilde{n}_i)}{V(\tilde{n}_i) + V(\hat{n}_i)}; \quad \check{\lambda}_i = \frac{V(\hat{n}_i)}{V(\tilde{n}_i) + V(\hat{n}_i)} \quad (18)$$

记  $\tilde{n}_i$  为 SmartMoss 算法对  $i=3, 5, 6$  的 3 种 4 节点子图数量的估计,  $V(\tilde{n}_i)$  为  $\tilde{n}_i$  的方差,则

$$\tilde{n}_i = \hat{\lambda}_i \hat{n}_i + \check{\lambda}_i \tilde{n}_i, \quad V(\tilde{n}_i) = \frac{V(\hat{n}_i)V(\tilde{n}_i)}{V(\tilde{n}_i) + V(\hat{n}_i)} \quad (19)$$

SmartMoss 算法共包含以下步骤:首先使用 3PS 采样器采样并估算采样误差,然后根据复杂度分析,判断使用 C3PS 采样器是否会在不过度降低采样效率的情况下提高采样准确率,若满足条件,则使用 C3PS 采样器继续对第 3,5,6 种 4 节点子图进行采样,输出采样结果  $\tilde{n}_i$  和误差  $S(\tilde{n}_i)$ 。否则,继续使用 3PS 采样器采样,并输出采样结果  $\hat{n}_i$  和误差  $S(\hat{n}_i)$ 。具体算法如下。

**步骤 1** 读入网络图  $G$ ;

**步骤 2** 设定最大误差参数  $S^{\max}$ ,最大样本集参数  $K^{\max}$ ;

**步骤 3** 初始化样本集  $K=0$ ,  $\tilde{K}=0$ ;

**步骤 4** 若  $K \leq K^{\max}$  且  $S(\hat{n}_i) \geq S^{\max}$ ,则采用 3PS 算法对图  $G$  进行采样,跳转至步骤 8;

**步骤 5** 计算  $K_i = \left( \frac{\max(S(\hat{n}_i))}{S^{\max}} \right)^2 - K$ ;

**步骤 6** 若  $2|L|^2/\lg|L| \geq K_p$ ,则采用 3PS 算法对图  $G$  进行采样,跳转至步骤 8;

**步骤 7** 若  $\Gamma/\tilde{\Gamma} \geq 6$ ,则采用 C3PS 算法对图  $G$  进行采样,否则采用 3PS 算法对图  $G$  进行采样;

**步骤 8** 若对图  $G$  的采样未完成,跳转至步骤 4;

**步骤 9** 根据 3PS 算法和 C3PS 算法的估计结果,混合估计网络图中各 4 节点子图数量,并计算相应估计误差。

4 实验设计

4.1 数据集

本文在多个真实数据集上对理论分析结果以及本文算法的性能进行了测试,实验数据集来自 Standard 大学网络数据分析平台(SNAP)。表 1 详细给出了实验所用数据集的性质和特点。

表 1 实验数据集参数

| 无向图            | 节点数     | 边数      | 最大度数  |
|----------------|---------|---------|-------|
| SOC-Epinion    | 75 897  | 405 740 | 3 044 |
| SOC-Slashdot08 | 77 360  | 469 180 | 2 539 |
| Ca-GrQc        | 5 241   | 14 484  | 81    |
| COM-Amazon     | 334 863 | 925 872 | 549   |

4.2 性能指标

本文定义了平均方差指标(设为  $R$ )和标准方差指标(设为  $S$ ),用于评估本文算法的准确度,实验结果如图 2 所示。对于任意一种 4 节点子图,记  $n_i$  是真实数量, $\hat{n}_i$  是估算数量,定义估算平均方差为

$$R(\hat{n}_i) = \frac{\sqrt{M(\hat{n}_i)}}{n_i}, i = 1, \cdots, 6 \tag{20}$$

其中  $M(\hat{n}_i)$ 被定义为

$M(\hat{n}_i) = E[(\hat{n}_i - n_i)^2] = V(\hat{n}_i) + (E[\hat{n}_i] - n_i)^2$   
此外,本文根据定理 1 和 2 分析的理论结果,计算估计值的标准方差为

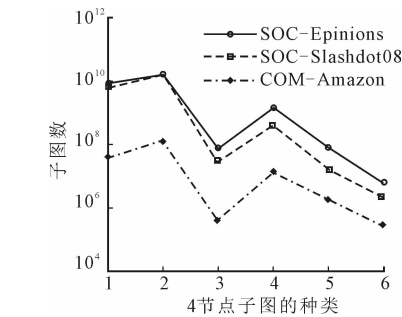
$$S(\hat{n}_i) = \frac{\sqrt{V(\hat{n}_i)}}{\hat{n}_i}, i = 1, \cdots, 6 \tag{21}$$

当 $\hat{n}_i$ 是 $n_i$ 的无偏估计值时, $R(\hat{n}_i)$ 等于 $\hat{n}_i$ 的标准差。

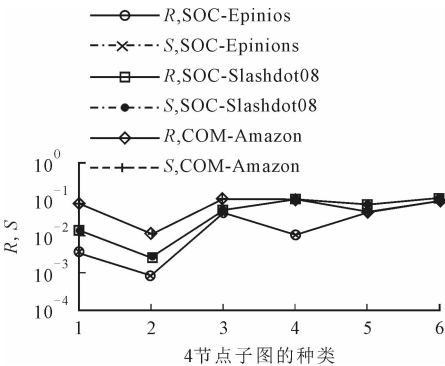
4.3 本文算法应用及准确度评估

为了验证本文算法的性能,在 SOC-Epinions、SOC-Slashdot08 和 COM-Amazon 这 3 个网络图上对本文算法进行了检验并对其准确度进行了评估。这 3 个网络图中分别包含有  $2.58 \times 10^{10}$ 、 $2.17 \times 10^{10}$  和  $1.78 \times 10^8$  个 4 节点子图,且第 3,5,6 种 4 节点子图的数量明显小于其他几种 4 节点子图。

图 2a 中分别给出了实验所用网络图中 6 种 4 节点子图数量的真实值,图 2b 给出了使用本文算法对 6 种 4 节点子图数量估算时的  $R$  和  $S$ 。可以发现,出现频率较高的 4 节点子图拥有更小的  $S$ 。在本实验中,参数设定为  $S^{\max} = 0.1$ , $K^{\max} = 10^5$ ,且实



(a) 6 种 4 节点子图数量真实值



(b) 本文算法测量误差

图 2 本文算法的估算准确度评估

验重复 1 000 次,计算平均结果和误差。实验结果表明,本文算法具有较高的准确性,而且误差分析的理论结果  $S$  可以准确地描述实际的估计误差。

4.4 本文算法与 3PS 和 C3PS 算法对比

本文算法可以根据中心极限定理及定理 1 和 2 给出的方差分析结论,估计达到预期的估计误差所需要的最小采样数。对比 3PS 算法和 C3PS 算法中最小采样数的估计,实验证明,本文算法更为准确。

本文在 SOC-Epinions、SOC-Slashdot08 和 COM-Amazon 这 3 个网络图上对两种算法进行了对比。当本文算法满足  $S \leq 0.01$  时,记预估所需的最小样本集大小为  $K_s$ ,记满足相同条件时 C3PS 和 3PS 算法所需的最小样本集大小为  $K_p$ 。对于 SOC-Epinions、SOC-Slashdot08 和 COM-Amazon 这 3 个网络图, $K_p/K_s$  的值分别为 38.6、20.9 和 17.0,这说明相对本文算法,C3PS 和 3PS 算法的采样成本高出数十倍。本文算法性能优于 3PS 算法和 C3PS 算法。

5 结 论

本文研究了大型网络图中 4 节点子图的采样估计问题,提出了一种新的 4 节点子图采样估计 SmartMoss 算法,并通过实验对该算法性能进行了验

证。通过理论分析与实验验证得出:首先,前沿算法C3PS能否提升3PS算法估算精度以及能提升多少取决于被测网络图的结构特征,C3PS算法并不一定能够提升3PS算法的估算精度;其次,本文算法通过被测网络的结构特性判断是否有必要使用C3PS算法提升子图数量估计的准确度,进而基于3PS和C3PS两种算法估算结果混合估计网络图中4节点子图的数量。对比实验同时证明,本文算法的准确率显著高于C3PS算法和3PS算法。

## 参考文献:

- [1] SHAI S, RON M, SHMOOLIK M, et al. Network motifs in the transcriptional regulation network of *escherichia coli* [J]. *Nature Genetics*, 2002, 31(1): 64-68.
- [2] 谢莹, 吴建国, 李炜, 等. 基于gSpan算法的未知化合物毒性预测 [J]. *合肥工业大学学报(自然科学版)*, 2007, 30(10): 1278-1280.  
XIE Ying, WU Jianguo, LI Wei, et al. Predicting the toxicity of unknown chemicals based on gSpan algorithm [J]. *Journal of Hefei University of Technology (Natural Science)*, 2007, 30(10): 1278-1280.
- [3] CHUN H, KWAK H, EOM Y H, et al. Comparison of online social relations in terms of volume vs interaction: a case study of Cyworld [C]//*Proceedings of the ACM SIGCOMM Conference on Internet Measurement*. New York, USA: ACM, 2008: 57-70.
- [4] KUNEGIS J, LOMMATZSCH A, BAUCKHAGE C. The slashdot zoo: mining a social network with negative edges [C]//*Proceedings of the 18th International Conference on World Wide Web*. New York, USA: ACM, 2009: 741-750.
- [5] ZHAO Junzhou, LUI J C S, TOWSLEY D, et al. Empirical analysis of the evolution of follower network: a case study on Douban [C]//*Proceedings of the 2011 IEEE Conference on Computer Communications Workshops*. Piscataway, NJ, USA: IEEE, 2011: 941-946.
- [6] UGANDER J, BACKSTROM L, KLEINBERG J. Subgraph frequencies: mapping the empirical and extremal geography of large graph collections [C]//*Proceedings of the 22nd International Conference on World Wide Web*. New York, USA: ACM, 2013: 1307-1318.
- [7] 严玉良, 董一鸿, 何贤芒, 等. Fsmbus: 一种基于spark的大规模频繁子图挖掘算法 [J]. *计算机研究与发展*, 2015, 52(8): 1768-1783.  
YAN Yuliang, DONG Yihong, HE Xianmang, et al. FSMBUS: a frequent subgraph mining algorithm in single large-scale graph using spark [J]. *Journal of Computer Research and Development*, 2015, 52(8): 1768-1783.
- [8] 栾华, 周明全, 付艳. 多核处理器上的频繁图挖掘方法 [J]. *计算机研究与发展*, 2015, 52(12): 2844-2856.  
LUAN Hua, ZHOU Mingquan, FU Yan. Frequent graph mining on multi-core processor [J]. *Journal of Computer Research and Development*, 2015, 52(12): 2844-2856.
- [9] PINAR Y, VISHWANATHAN S V N. Deep graph kernels [C]//*Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, USA: ACM, 2015: 1365-1374.
- [10] CHEN K, WANG P, LEE Y, et al. Finding unknown malice in 10 seconds: mass vetting for new threats at the Google-play scale [C]//*Proceedings of the Usenix Conference on Security Symposium*. New York, USA: USENIX Association, 2015: 659-674.
- [11] GASCON H, YAMAGUCHI F, ARP D, et al. Structural detection of android malware using embedded call graphs [C]//*Proceedings of the 2013 ACM Workshop on Artificial Intelligence and Security*. New York, USA: ACM, 2013: 45-54.
- [12] SHEN Tong, ZHONGYANG Yibing, XIN Zhi, et al. Detect android malware variants using component based topology graph [C]//*Proceedings of the 2014 13th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*. New York, USA: ACM, 2014: 406-413.
- [13] WANG Pinghui, LUI J C S, RIBEIRO B, et al. Efficiently estimating motif statistics of large networks [J]. *ACM Transactions on Knowledge Discovery from Data*, 2014, 9(2): 1-27.
- [14] JHA M, SESHADHRI C, PINAR A. Path sampling: a fast and provable method for estimating 4-vertex subgraph counts [C]//*Proceedings of the 24th International Conference on World Wide Web*. New York, USA: ACM, 2015: 495-505.

(编辑 武红江)