

一种高效的大规模网络 k 团挖掘算法

柴旭清 董永亮

(河南师范大学 新乡 453000)

摘要 网络结构中的 k 团挖掘是各种基于网络的应用的基础问题之一。针对大规模网络 k 团挖掘效率低的问题,提出了一种高效的大规模网络 k 团挖掘算法。首先,将寻找最大密度的 k 团问题进一步转化为寻找超过给定密度值 k 团的问题。然后,以网络中的顶点和 k-1 团顶点为两类顶点构建二部图,并证明应用二部图可以在多项式时间内求解 k 团问题。在稀疏网络中,提出的算法的时间和空间复杂度分别为 $O(c_k^2)$ 和 $O(c_k)$ 。实验表明,提出的算法与目前最优的算法相比能更准确地挖掘大规模网络中的 k 团,并且具有更高的运行效率。此外,提出的算法可应用于不完全网络中的 k 团挖掘。

关键词 社团挖掘, 社会网络, k 团, 不完全网络

中图分类号 TP391

文献标识码 A

DOI 10.11896/j.issn.1002-137X.2016.5.050

Efficient k-Clique Mining Algorithm in Large-scale Networks

CHAI Xu-qing DONG Yong-liang

(Henan Normal University, Xinxiang 453000, China)

Abstract k-clique mining in networks is a basis for most network-based applications. Concerning the problem of low efficiency for mining k-cliques, this paper proposed an efficient k-clique mining algorithm for large-scale networks. Firstly, the problem of finding k-clique with maximal density is transformed to the problem of finding a k-clique whose density is larger than a given density value. Next, a bipartite network, whose left nodes are nodes in the original network and right nodes are k-1 cliques, is constructed, and the k-cliques problem can be solved within polynomial time applying it. In sparse networks, the time and space complexities of the proposed algorithm are $O(c_k^2)$ and $O(c_k)$ respectively. The experiments show that, the proposed algorithm is more accurate and efficient than the best algorithm at present. Moreover, the proposed algorithm can be also applied to mine k-cliques in incomplete networks.

Keywords Community mining, Social networks, k-clique, Incomplete network

互联网的快速发展催生了大量复杂的基于网络结构的数据^[1],如用户在社交网络中的连接关系或者交互行为、用户在内容分享网站中对内容的评价、电子商务中用户对商品的购买历史记录以及网页的超链接结构等。通过对这些网络结构数据(简称网络)进行分析,可以发现隐藏在网络背后的用户行为信息,从而为用户提供更优质的服务^[2]。例如,通过分析社交网络中用户的交互行为可以发现网络中具有相似行为的用户团体,从而指导用户未来的交互行为^[3,4];通过分析电子商务中用户的交易记录可以进一步分析用户的长期和短期偏好,从而进行商品的个性化推荐^[5,6]。

网络中的社团挖掘又称为聚类分析,是一种无监督的机器学习方法。研究人员根据具体的应用需求对社团进行定义,给出社团挖掘的目标函数,并试图找到最优的挖掘方法^[7]。在具体的社团分析应用中,社团的定义与目标函数往往是联系在一起的。当给定某社团的定义时,社团挖掘的目标是发现最符合该定义的那些社团。由于网络结构可以用网络的邻接矩阵表示,因此传统的基于高维向量的聚类方法都可以应用于网络的社团挖掘^[8-10]。此外,在网络结构中,还可

以通过密集子图来定义节点的社团结构,如 k 团(k-Clique)^[11-13]、k 核(k-Core)^[14]等。

在网络中,k 团密集子图是一种重要的社团挖掘方法。当 $k=2$ 时,该问题简化为经典的密集子图问题^[15,16],即找到图中顶点集合 V 的子集 $S \subseteq V$,使得经 S 诱导的子图的平均度最大。当 $k>2$ 时,k 团密集子图问题可以描述为^[11]:对于每个顶点子集 $S \subseteq V$,将 S 诱导为 $c_k(S)$ 个 k 团,并且最大化 $c_k(S)/|S|$ 。当 k 为常数时,k 团密集子图问题可以在多项式时间内解决;当 $k=3$ 时,文献^[11]中算法的空间复杂度是线性的,时间复杂度约为 $O((n^5 m^{1.4} + n^6) \log(n))$,其中 n 为网络中顶点的个数, m 为边的个数。

文献^[11]中提出的算法虽然可以在多项式时间内解决 k 团密集子图挖掘问题,但是算法的效率不高,很难扩展到大规模的网络应用中。为了提高大规模网络中 k 团挖掘的性能,本文提出了一种高效的 k 团挖掘算法。在将寻找最大密度的 k 团问题进一步转化为寻找超过给定密度值的 k 团的问题后,构建了一个二部图网络,并基于该二部图网络进行最大割的计算。证明了算法的正确性,并分析了算法的时间和空间复杂性。

到稿日期:2015-11-27 返修日期:2015-12-21 本文受河南省科技厅资助性项目(9412012Y0004,9412012Y0005),河南省教育厅项目(13A510520,2013-gh-12,14A520053,SKL-2014-795)资助。

柴旭清(1982—),女,硕士,工程师,主要研究方向为计算机网络、社会网络以及云计算等,E-mail:cxq@htu.cn;董永亮(1978—),男,硕士,讲师,主要研究方向为计算机网络。

1 k 团挖掘算法

1.1 问题描述

在无向的网络中, k 团是一个具有 k 个顶点和 $k(k-1)/2$ 条边的无向网络。在无向的网络 $G=(V, E)$ 中, V 和 E 分别表示顶点和边的集合, $|V|$ 和 $|E|$ 分别表示顶点和边的个数。令 $f_k(v)$ 为 G 中顶点 $v \in V$ 所属的 k 团的个数, C_k 为 G 中所有 k 团的集合, 那么 $|C_k| = \sum_{v \in V} f_k(v)/k$ 。

定义 1 (k 团密度) 给定无向网络 $G=(V, E)$, 对于任何顶点子集 $S \subseteq V$, 它的 k 团密度为

$$\rho_k(S) = c_k(S)/|S|, k \geq 2 \quad (1)$$

其中, $c_k(S)$ 为 S 在 G 中诱导得到的 k 团的个数。

对于给定的无向网络, k 团挖掘问题是在网络中寻找密度最大的 k 团。该问题可描述为:

问题 1 给定无向网络 $G=(V, E)$, 寻找顶点子集 $S^* \subseteq V$, 使得 $\forall S \subseteq V$, 满足 $\rho_k^* = \rho_k(S^*) \geq \rho_k(S)$, 即

$$S^* = \arg \max_{S \subseteq V} \rho_k(S) \quad (2)$$

通过对问题 1 进行分析, 可以将该问题进一步分解为如下的子问题。

问题 2 给定无向网络 $G=(V, E)$ 及常数 D , 存在顶点子集 $S \subseteq V$ 满足 $\rho_k(S) \geq D$ 吗?

于是, 当 k 为固定的常数时, 通过对问题 2 进行 $O(\log n)$ 次迭代就可以找到问题 1 所示的密度最大的 k 团结构。

1.2 二部图构建与 k 团挖掘

给定无向网络 $G=(V, E)$ 和密度阈值 D , 以及源节点 s 和目标节点 t , 通过如下方法构建一个二部图。令二部图的左侧节点集合为 V , 右侧节点集合为 $k-1$ 团集合 C_{k-1} , 二部图的边通过如下方式添加:

- 1) 对于 $\forall v \in V$, 添加一条从 s 到 v 的边, 并令其权重为 $f_k(v)$;
- 2) 对于 $\forall v \in V$, 以及 $\forall c \in C_{k-1}$, 如果 v 和 c 可以构成一个 k 团, 那么在它们之间加一条权重为 1 的边;
- 3) 对于 $\forall v \in V$, 以及 $\forall c \in C_{k-1}$, 如果 v 是 c 中的一个顶点, 那么在 v 和 c 之间加一条权重为 ∞ 的边;
- 4) 对于 $\forall c \in C_{k-1}$, 添加一条从 c 到 t 的边, 并且令权重为 kD 。

按照上述步骤, 可以得到如图 1 所示的二部图示意图。

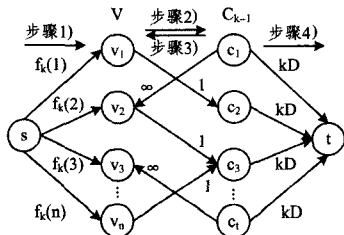


图 1 二部图构建示意图

在图 1 中, $k-1$ 团的个数为 $|C_{k-1}| = t$, s 和 t 分别为源节点和目标节点。为了便于下文描述, 令 $A=V, B=C_{k-1}$ 。在二部图的构建过程中, 步骤 3) 加入了权重为 ∞ 的边, 这可以保证: 当一个 $k-1$ 团在图的割的左侧时, $k-1$ 团所包含的顶点也在割的左侧。根据上述步骤得到二部图后, 可以得到如下定理。

定理 1 当 k 为常数 ($k=\theta(1)$) 时, 应用上述方法得到的网络可以在多项式时间内求解 k 团问题。

证明: 首先按照上述方法构建一个网络 H_D , 然后计算从 $s(s \in S)$ 到 $t(t \in T)$ 的最小割。对于 $k=\theta(1)$, 易知上述 4 个步骤都可以在多项式时间内完成。由于 $S=\{s\}, T=V(H_D) \setminus S$ (符号 $\setminus$ 表示集合的差) 是有效的割, 因此 $s-t$ 的割的上界为 kc_k 。令 $A_1=S \cap A, B_1=S \cap B, A_2=T \cap A, B_2=T \cap B$ 。步骤 3) 产生的边使得 k 团可以包含 A_1 中的所有顶点, 然而包含这些边会增大割的权重, 那么可以得出 A_1 诱导的 $k-1$ 团的集合就是 B_1 。在计算 $s-t$ 的最小割时, 主要包括 3 种情况: 第一种情况是步骤 1) 中的边, 即 $\sum_{v \in A_2} f_k(v)$; 第二种情况是步骤 4) 中的边, 即 $kD|A_1|$; 第三种情况是在 k 团下的边。

在第三种情况下, 定义 $c_k^{(j)} (1 \leq j \leq k)$ 为原图 G 中包含 A_1 中 j 个顶点的 k 团的个数, 于是有 $c_k^{(k)} = c_k(A_1)$ 。对于每个包含 A_1 中 $j (1 \leq j \leq k-1)$ 个顶点的 k 团 ∇ , 由于从 $v \in A_1$ 到 $c \in C_{k-1}$ 共有 j 条边 (由步骤 2) 生成), 因此 $s-t$ 的割的代价为 j 。所以, $s-t$ 的最小割的值为

$$val_{st} = \sum_{v \in A_2} f_k(v) + kD|A_1| + \sum_{j=1}^{k-1} j c_k^{(j)} \quad (3)$$

由于 $\sum_{j=1}^k j c_k^{(j)} = \sum_{v \in A_1} f_k(v) = kc_k - \sum_{v \in A_2} f_k(v)$, 可以得到 $val_{st} = kc_k + kD|A_1| - kc_k(A_1)$ 。于是可以得到如下结论: $\exists U \subseteq V$, 使得 $\rho_k(U) > D$ 当且仅当存在一个最小割, 并且该最小割的值小于 kc_k 。当最小割存在时, A_1 就是其对应的顶点集合。

1.3 算法复杂性分析

在上述构建的网络 H_D 中, 仅仅那些能构成 k 团的 $k-1$ 团对算法的时间和空间复杂度有影响。由于每个 k 团当中包含 k 个 $k-1$ 团, 因此 $|B| = O(kc_k)$, 于是 H_D 的顶点和边的个数分别为:

$$|V(H_D)| = O(n + kc_k) \quad (4)$$

$$|E(H_D)| = O(n + \sum_{v \in V} f_k(v) + (k-1)kc_k) = O(n + k^2 c_k) \quad (5)$$

式 (5) 表明算法的空间复杂度为 $O(n + k^2 c_k)$ 。算法的时间复杂度取决于 k 团搜索算法和最大流算法。在 k 团搜索算法中, 文献 [17] 所述的算法列出 C_k 的时间复杂度为 $O(km\alpha(G)^{k-2})$, 其中 $\alpha(G)$ 为图的荫度。在最大流算法中, 本文采用文献 [18] 所述的算法, 该算法的时间复杂度为 $O(m\sqrt{n} \log^2 U)$, 其中 U 为与最小割相对应的最大流值。所以当 k 为固定值时, 本文采用的算法的时间复杂度为 $O(km\alpha(G)^{k-2} + (n + c_k)^2)$ 。

在实际的应用中, 当 k 取值很小 ($k \ll n < m$) 时, 网络中包含的 k 团个数 c_k 远远大于 m 和 n , 即 $n < m \ll c_k$ 。此外, 网络往往是稀疏的, 网络的荫度也很小, 那么 k 团搜索算法的效率也很高。所以, 在稀疏的网络中, k 团挖掘问题的时间和空间复杂度近似为 $O(c_k^2)$ 和 $O(c_k)$ 。

1.4 近似采样算法

在准确的算法中, k 团挖掘问题的时间和空间复杂度近似为 $O(c_k^2)$ 和 $O(c_k)$, 当网络中的 c_k 很大时, 计算量仍然非常大。为此, 本文对原网络进行采样, 从而得到一个较小的网络, 使得以适当的概率进行采样后得到的 k 团仍然保持原网络的最大密度。

在图 1 所示的二部网络中, 采用随机游走的方式对网络进行采样, 然后应用得到的结果对网络中的 k 团进行近似计算。随机游走采样的基本思想如下:

- 以节点 s 为初始点开始随机游走;

- 以 $1/n$ 的概率随机选择一条出边,并走向该边的另一个定点 v ;
- 以 $1/k$ 的概率随机选择一个 $k-1$ 团 c ;
- 从 c 点走向终点 t ,完成一次采样;
- 重复上述过程若干次,得到多个从 s 到 t 的路径。

在每一次采样结束后,依据 $f_k(v)+D|A_1|+jc_k^{(j)}$ 计算 $s-t$ 的值,重复若干次后,用总和来近似估计式(3)的 $s-t$ 的最小割的值。

2 实验结果与分析

2.1 实验环境与数据集

实验平台采用单服务器节点,该节点包含一个 Intel Xeon 2.83GHz CPU、6144kB 高速缓存和 50GB 内存。运行的操作系统为 Ubuntu 10.04,网络流计算工具采用 HIPR-3.7¹⁾。实验采用的数据集为公开的数据集 SNAP²⁾,本文选取了其中 soc-Epinions1 和 Web-Google 两个网络数据集。soc-Epinions1 为有向的用户与用户之间的信任关系网络,Web-Google 为有向的网页之间的超链接网络。本文将这两个网络无向化,即用一条无向边代替任意两个节点间的一条或者两条有向边。表 1 列出这两个数据集的基本统计信息,其中 n 和 m 分别为网络节点和边的个数, $k=3$ 和 $k=4$ 分别表示网络中 3 团和 4 团的个数。

表 1 数据集统计信息

数据集	n	m	$k=3$	$k=4$
soc-Epinions1	75879	508837	1624494	5803405
Web-Google	875713	5105039	11385529	32473410

实验采用如下准确率计算方法对算法的性能进行评估:

$$\text{准确率} = \frac{\rho_k(S)}{\rho_k^*} \tag{6}$$

其中, $\rho_k(S)$ 表示顶点集合 S 在给定图中的 k 团密度(见定义 1), ρ_k^* 为给定图中所有 k 团的最大密度值(见问题 1)。

2.2 实验结果分析

在实验中,本文将提出的算法与文献[11]中的算法进行了对比,主要对比了算法的准确率和运行时间。

首先,实验对比了两种算法的准确率,分别选取了 $k=3$ 和 $k=4$ 两种情况,实验结果见表 2。对比两个数据集下的实验结果发现, soc-Epinions1 数据集下算法的准确率要高于 Web-Google 数据集下算法的准确率,这是由于 soc-Epinions1 数据集的网络密度要大于 Web-Google 数据集的网络密度。对比 $k=3$ 和 $k=4$ 两种 k 团下的结果发现,当 k 增大时,算法的准确率降低,这是因为同一网络中 $k=4$ 时的 4 团个数要少于 $k=3$ 时的 3 团个数,因而准确率降低了。对比本文提出的方法与文献[11]中的方法可知,本文方法的准确率要高于文献[11]的方法。

表 2 准确率对比

数据集	$k=3$		$k=4$	
	文献[11]	本文方法	文献[11]	本文方法
soc-Epinions1	0.925	0.973	0.896	0.945
Web-Google	0.893	0.965	0.877	0.934

接下来将通过实验对比两种方法的运行时间,仍然选取

$k=3$ 和 $k=4$ 两种情况,实验结果见表 3。由于 soc-Epinions1 数据集的网络规模远远小于 Web-Google 数据集的网络规模,因此算法的运行时间也较短。当 k 从 $k=3$ 变化为 $k=4$ 时,文献[11]的方法在 soc-Epinions1 和 web-Google 数据集下的时间分别增加到原来的 5.8 倍和 3.5 倍,而本文方法的时间则分别增加到原来的 4.6 倍和 2.7 倍,这说明本文方法在挖掘较大的 k 团时具有更好的可扩展性。此外,在同一实验设置下,本文方法所需的时间要远远少于文献[11]中算法所需的运行时间。

表 3 运行时间对比(s)

数据集	$k=3$		$k=4$	
	文献[11]	本文方法	文献[11]	本文方法
soc-Epinions1	28.3	16.4	164.9	75.3
Web-Google	243.8	133.6	845.2	362.7

最后,通过实验进一步评估本文方法在不完全网络下的 k 团挖掘性能。通过固定的采样概率 p 对 soc-Epinions1 数据集的边进行采样,这样每次可以得到原网络比例为 p 的不完全网络。在不完全网络中,应用本文提出的方法分析 k 团挖掘问题,观察算法的准确率和运行效率。由于每次采样会得到不同的不完全网络,因此每组实验重复 10 次并取结果的平均值。在准确率评估中,应用原网络的最优值作为准确值。在运行时间的评估中,用原网络所需的运行时间作为基准值,观察算法在不同采样概率下的加速比。图 2—图 4 分别为 $k=3$ 、 $k=4$ 和 $k=5$ 这 3 种情况下的实验结果。当网络的采样概率 p 不断增大时,算法的平均准确率不断提高,加速比不断减小,这与预想的结果是一样的。在图 2 中($k=3$ 时),当 p 增大到 0.3 后,算法的平均准确率和加速比都变化得比较平缓,这说明随着网络不完全性的提高(p 的减小),虽然不能获得较好的加速比,但是算法的准确率却保持得很好。在图 4 中($k=5$ 时),随着网络不完全性的提高(p 的减小),加速比近似呈线性提高,并且算法在 $p>0.3$ 以上时都保持着很好的准确率。当 $k=4$ 时,算法的加速比变化和准确率变化都在 $k=3$ 和 $k=5$ 之间。通过这 3 幅图可以得出,当 k 的值较大时,算法对不完全网络不仅尽可能地保持着准确率,还有着较高的运算效率。

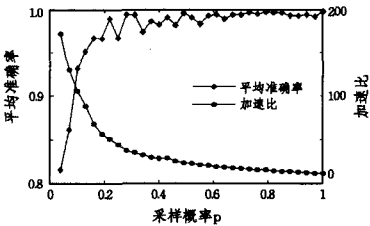


图 2 不完全网络测试结果(soc-Epinions1, $k=3$)

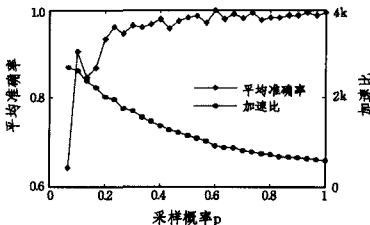


图 3 不完全网络测试结果(soc-Epinions1, $k=4$)

¹⁾ <http://www.avglab.com/andrew/soft.html>

²⁾ <http://snap.stanford.edu/data/index.html>

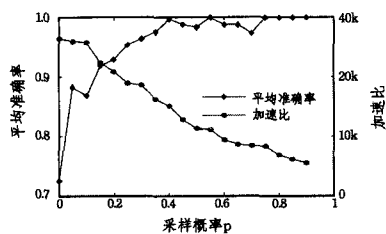


图4 不完全网络测试结果(soc-Epinions1, $k=5$)

结束语 本文研究了大规模网络中的 k 团挖掘问题, k 团挖掘是各种基于网络应用的基础问题之一。将寻找最大密度的 k 团问题进一步转化为寻找超过给定密度值的 k 团的问题。接下来, 以网络中的顶点和 $k-1$ 团顶点为两类顶点构建二部图, 并在该二部图中寻找网络的最大割。在稀疏网络中, 本文提出的算法的时间和空间复杂度分别为 $O(c_k^2)$ 和 $O(c_k)$, 因而可应用于大规模稀疏网络的 k 团挖掘。通过与文献[11]中的算法进行对比发现, 提出的算法能更准确地挖掘大规模网络中的 k 团, 并且具有更高的运行效率。此外通过实验发现, 本文提出的算法可应用于不完全网络中的 k 团挖掘, 并且在 k 较大时具有较高的准确率和性能加速比。在未来的研究工作中, 将进一步研究 k 团挖掘问题的并行性, 并在主流的分布式和并行系统中进行测试。

参考文献

- [1] Xu Ke, Zhang Sai, Chen Hao, et al. Measurement and analysis of online social network [J]. Chinese Journal of Computers, 2014, 37(1): 165-188 (in Chinese)
徐恪, 张赛, 陈昊, 等. 在线社会网络的测量与分析[J]. 计算机学报, 2014, 37(1): 165-188
- [2] Wang Li, Cheng Su-qi, Shen Hua-wei, et al. Structure inference and prediction in the co-evolution of social network [J]. Chinese Journal of Computer Research and Development, 2015, 50(12): 2492-2503 (in Chinese)
王莉, 程苏琦, 沈华伟, 等. 在线社会网络共演化的结构推断与预测[J]. 计算机研究与发展, 2015, 50(12): 2492-2503
- [3] Lin You-fang, Wang Tian-yu, Tang Rui, et al. An effective model and algorithm community detection in social networks [J]. Chinese Journal of Computer Research and Development, 2015, 49(2): 337-345 (in Chinese)
林友芳, 王天宇, 唐锐, 等. 一种有效的社会网络社区发现模型和算法[J]. 计算机研究与发展, 2015, 49(2): 337-345
- [4] Wang Li, Cheng Xue-qi. Dynamic community in online social networks [J]. Chinese Journal of Computers, 2015, 38(2): 219-237 (in Chinese)
王莉, 程学旗. 在线社会网络的动态社区发现及演化[J]. 计算机学报, 2015, 38(2): 219-237
- [5] Gao Quan-li, Gao Ling, Yang Jian-feng, et al. A preference Elicitation method based on users' cognitive behavior for Context-

aware recommender system [J]. Chinese Journal of Computers, 2015, 38(9): 1767-1776 (in Chinese)

高全力, 高岭, 杨建峰, 等. 上下文感知推荐系统中基于用户认知行为的偏好获取方法[J]. 计算机学报, 2015, 38(9): 1767-1776

- [6] Wang Li-cai, Meng Xiang-wu, Zhang Yu-jie. Context-aware recommender systems [J]. Journal of Software, 2012, 23(1): 1-20 (in Chinese)
王立才, 孟祥武, 张玉洁. 上下文感知推荐系统[J]. 软件学报, 2012, 23(1): 1-20
- [7] Sun Ji-gui, Liu Jie, Zhao Lian-yu. Research on clustering algorithm [J]. Journal of Software, 2008, 19(1): 48-61 (in Chinese)
孙吉贵, 刘杰, 赵连宇. 聚类算法研究[J]. 软件学报, 2008, 19(1): 48-61
- [8] Yang Bo, Liu Da-you, Jin Di, et al. Complex network clustering method [J]. Journal of Software, 2009, 20(1): 54-66 (in Chinese)
杨博, 刘大有, 金弟, 等. 复杂网络聚类方法[J]. 软件学报, 2009, 20(1): 54-66
- [9] Chen Ke-han, Han Pan-pan, Wu Jian. User clustering based social network recommendation [J]. Chinese Journal of Computers, 2013, 36(2): 349-359 (in Chinese)
陈克寒, 韩盼盼, 吴健. 基于用户聚类的异构社交网络推荐算法[J]. 计算机学报, 2013, 36(2): 349-359
- [10] Chiang K Y, Hsieh C J, Natarajan N, et al. Prediction and clustering in signed networks: a local to global perspective [J]. The Journal of Machine Learning Research, 2014, 15(1): 1177-1213
- [11] Tsourakakis C. The K-clique Densest Subgraph Problem [C] // Proceedings of the 24th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2015: 1122-1132
- [12] Rossman B. The monotone complexity of k-clique on random graphs [J]. SIAM Journal on Computing, 2014, 43(1): 256-279
- [13] Bahmani B, Kumar R, Vassilvitskii S. Densest subgraph in streaming and mapreduce [J]. Proceedings of the VLDB Endowment, 2012, 5(5): 454-465
- [14] Bhawalkar K, Kleinberg J, Lewi K, et al. Preventing unraveling in social networks: the anchored k-core problem [M] // Automata, Languages, and Programming. Springer Berlin Heidelberg, 2012: 440-451
- [15] Khuller S, Saha B. On finding dense subgraphs [M] // Automata, Languages and Programming. Springer Berlin Heidelberg, 2009: 597-608
- [16] Andersen R, Chellapilla K. Finding dense subgraphs with size bounds [M] // Algorithms and Models for the Web-Graph. Springer Berlin Heidelberg, 2009: 25-37
- [17] Chiba N, Nishizeki T. Arboricity and subgraph listing algorithms [J]. SIAM Journal on Computing, 1985, 14(1): 210-223
- [18] Lee Y T, Sidford A. Path finding methods for linear programming: Solving linear programs in $o(v \text{rank})$ iterations and faster algorithms for maximum flow [C] // FOCS, 2014: 424-433

(上接第 251 页)

- 王万良, 蔡竞. 稀疏约束下非负矩阵分解的增量学习算法 [J]. 计算机科学, 2014, 41(8): 241-244
- [6] Li Qian, Jing Li-ping, Yu Jian. Multi-kernel Projective Nonnegative Matrix Factorization Algorithm [J]. Computer Science, 2014, 41(2): 64-67 (in Chinese)
李谦, 景丽萍, 于剑. 基于多核学习的投影非负矩阵分解算法[J]. 计算机科学, 2014, 41(2): 64-67
- [7] Lazebnik F. On Systems of Linear Diophantine Equations [J]. Mathematics Magazine, 1996, 69(4): 261-266
- [8] Hanson R. Integer Matrices Whose Inverse Contain Only Inte-

gers [J]. The Two-Year College Mathematics Journal, 1992, 13(1): 18-21

- [9] Penrose R. A generalized inverse for matrix [J]. Proc. Cambridge Philos. Soc., 1955, 51: 406-413
- [10] Penrose R. On best approximate solution of linear matrix equations [J]. Proceedings of the Cambridge Philosophical Society, 1956, 52: 17-19
- [11] Stanimirovic P. Determinants of Rectangular Matrices and Moore Penrose Inverse [J]. Novi. Sad. J. Math., 1997, 27(1): 53-69
- [12] Gregory. Inverse of an integer matrix [EB/OL]. http://www.openproblemgarden.org/op/inverse_of_an_integer_matrix