一种基于邻居信息的最大派系过滤算法

陈端兵[1] 周玉林[2] 傅 彦[1]
(电子科技大学计算机科学与工程学院 成都 611731)[1] (陕西工程勘察研究院 西安 710068)[2]

摘 要 最大派系问题(Maximal Clique Problem,MCP)是组合优化中经典而重要的问题之一,在信息抽取、信号传输、计算机视觉、社会网络及生物信息学等众多领域有着重要的应用。学者们根据不同的思想策略,提出了许多方法求解最大派系问题,如分支定界、遗传算法、模拟退火、交叉熵及DNA方法等。现根据派系的邻居信息提出一种基于派系邻接顶点和邻接边的派系过滤算法。算法从一个已知派系(初始为一个单独顶点)出发,每次考察派系的邻接顶点,并以派系的邻接边为基础,扩展已有派系而得到更大的派系。用两个大规模的科学家合作网络对提出的算法进行了分析,并讨论了大规模社会网络中的派系分布情况。实验表明,提出的算法可有效地抽取网络中的最大派系。

关键词 最大派系问题,社会网络,派系过滤算法,邻接顶点,邻接边

Maximal Clique Percolation Algorithm Based on Neighboring Information

CHEN Duan-bing[1] ZHOU Yu-lin[2] FU Yan[1]
(School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China)[1]
(Shanxi Engineering Investigation Institute, Xi'an 710068, China)[2]

Abstract Maximal clique problem(MCP) is a classical and important combinational optimization problem with many prominent applications, for example, information retrieval, signal transmission, computer vision, social network and bioinformatics, etc. Researchers presented many algorithms to solve it by using various strategies, such as branch-and-bound, genetic algorithm, simulation annealing, cross entropy and DNA method. In this paper, a new clique percolation algorithm was presented based on neighboring vertices and edges of clique. From a given clique(it's a vertex at initial) at each step, investigated its all neighboring vertices and expanded it to a larger clique through a neighboring edge of clique. Two large scale author collaboration networks were used to test the performance of proposed algorithm and the clique distribution in large scale social network was also discussed. Experimental results demonstrate that the presented algorithm is efficient to percolation the maximal clique in network.

Keywords Maximal clique problem, Social network, Clique percolation, Neighboring vertex, Neighboring edge

1 引言

最大派系问题(Maximal Clique Problem,MCP)是组合优化中的一个重要问题,它有着多方面的应用,如信息抽取、信号传输、计算机视觉等[1]。最近,最大派系问题在生物信息学中也得到了越来越重要的应用[2,3]。另外一个重要的应用就是社会网络中的社区发现,社会网络通常是一个大规模的稀疏图,而且随着网络中的派系派系的增大,其数量呈指数衰减,找出其中的大派系有利于发现社会网络中的社区结构[4-6]。

已经证明,最大派系问题是一个NP问题[7],学者们根据不同的思想策略提出了多种派系过滤算法。1973年,Bron和Kerbpsch提出了一种采用回溯和分支定界技术查找无向图中所有的派系的算法[8]。2007年,Tomita和Kameda[9]提出了一种更为有效的分支定界方法求解最大派系问题。Derényi等人[4]提出了一种E-R随机图中派系过滤算法,论文作者讨论了提出的派系过滤算法对于实际网络中重叠社区检测的有效性。Palla,Adamcsek等人[3,4]设计了一种快速算法CFinder,以发现生物网络中的所有k-派系($k\geqslant 3$)。作者提出的CFinder可用于预测蛋白质的功能、发现新的蛋白质单元,而且CFinder能有效地找出大规模稀疏图中的派系。Pullan等学者[10,11]提出了派系过滤的局部搜索算法,其基本思想是逐步将节点加入当前派系从而扩展成更大的派系。吕强等人[12]提出了一种领导者-跟随者协作求解的交叉熵并行算法,领导者活跃在并行处理器之间采集数据,并根据当前获得的信息对跟随者做出决策,跟随者则根据领导者的决策信息自适应地调整搜索空间。近年来,不少学者还提出了应用DNA计算技术求解最大派系问题[13-15]。李肯立等人[14]采用DNA计算技术提出了一种新的最大派系问题求解算法,算法由顶点度数搜索器、团生成器、稀疏图与稠密图并行搜索器以

到稿日期:2010-02-05 返修日期:2010-05-07 本文受国家自然科学基金(编号:60973069,90924011,60903073,60973120),中国博士后科学基金项目(编号:20080431273)资助。

陈端兵 博士,副教授,硕士生导师,主要研究方向为数据挖掘、社会计算、NP难问题高效求解等;周玉林 工程师,主要研究方向为水文地质、复杂网络;傅 彦 教授,博士生导师,主要研究方向为数据挖掘、信息安全、模式识别等。

及最大团搜索器组成，算法时间复杂度有较大幅度的改善。Cui 等人[15]提出了一种应用 DNA 自组装方法求解最大派系问题。最近几年，其它一些启发式算法，如遗传算法、模拟退火算法、神经网络方法等也相继提出用于求解最大派系问题[16-18]。

本文在已有研究方法基础上，提出一种基于派系邻接顶点和邻接边的派系搜索算法。算法从一个初始派系（单独顶点）出发，每次考察派系的邻接顶点，并从派系的邻接边出发，扩展已有派系而得到更大的派系。论文用两个大规模的科学家合作网络对提出的算法进行了分析，并讨论了大规模网络中的派系分布情况。

2 问题描述

给定一个具有 n 个顶点，m 条边的网络 $G(V,E)$，其中 $|V|=n$，$|E|=m$，需要找出网络 G 中所有的最大派系。本文主要讨论无向无权网络的最大派系查找问题，对有向或有权网络，最大派系的定义和无向无权网络中派系的定义有一些差异。只要给出在有向有权网络中派系的定义形式，将本文算法进行简单修正即可实现有向有权网络中最大派系的查找。

3 基本概念

3.1 最大派系

在网络 G 中，一个 k 派系是一个具有 k 个顶点的完全子图。而最大 k 派系是指对于一个 k 派系，若再向此 k 派系中加入一个顶点，形成的子图不再是网络 G 的完全子图，这时此 k 派系就是一个最大 k 派系。例如，图 1 中，{1,2,3,4,8}，{1,2,7}，{1,6,7}和{1,4,5,6}都是最大派系，而{1,4,5,6,7}不是派系，更不是最大派系，{1,2,3,4}是派系但不是最大派系。

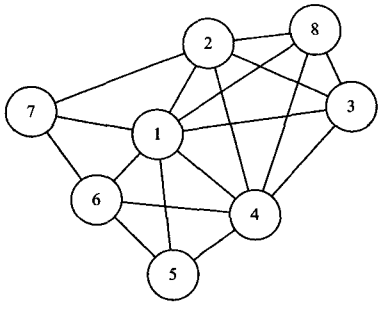

图 1 最大派系的例子

3.2 派系邻接顶点

若一个顶点 v 和派系 K 中所有顶点都邻接，则称顶点 v 为派系 K 的邻接顶点。例如，图 1 中，顶点 4 是派系{1,2,3}的邻接顶点；顶点 5 是派系{1,4,6}的邻接顶点。特别地，若派系的邻接顶点与此派系的其他邻接点不再邻接，则称此邻接顶点为第一类邻接顶点；若派系的邻接顶点与派系的其他邻接顶点有邻接，则称此邻接顶点为第二类邻接顶点。例如，图 1 中，{1,2,4}是一个派系，顶点 3,8 是此派系的第二类邻接顶点，而顶点 6 是派系{1,4,5}的第一类邻接顶点。

3.3 派系邻接边

若一条边 e 所连接的两个顶点都是派系 K 的邻接顶点，那么边 e 称为派系 K 的邻接边。例如，图 1 中，{1,2,4}是一个派系，边(3,8)是派系{1,2,4}的邻接边。

3.4 顶点的派系度

和网络中顶点度的定义类似，可以定义网络中顶点的派系度。由于派系有大小之分，因此，在定义顶点的派系度时，必须明确是几派系度，如 3-派系度、4-派系度、5-派系度等等。比如顶点的 3-派系度定义为包含此顶点的最大 3-派系的数量，一般地，顶点的 k-派系度定义为包含此顶点的最大 k-派系的数量[1]。例如，在图 1 中，顶点 1 的 3-派系度为 2，4-派系度和 5-派系度分别为 1；顶点 6 的 3-派系度为 1，4-派系度为 1。

4 算法描述

4.1 基本思想

本文提出的派系过滤算法的基本思想是基于邻接边不断递归地扩展派系而得到更大的派系。具体地，派系的扩展过程为：从已经找到的派系出发，若派系的邻接点本身形成了一个完全子图，则将邻接顶点全部加入原有派系而形成最大派系；若存在第一类邻接点则将第一类邻接点中每个顶点分别加入原有派系中，形成最大派系；对第二类邻接点，将邻接边的两个顶点同时加入原有派系，形成一个更大的派系，同时更新此更大派系的邻接顶点。这个过程一直进行下去，直至找到最大派系。初始时，一个单独的顶点形成一个派系。例如，在图 1 中，如果从顶点 1 出发，它有 7 个邻接点：2,3,4,5,6,7 和 8，这 7 个邻接点都属于第二类邻接点，从其中的一条边出发，比如，从邻接边(2,4)出发，将顶点 2,4 加入原有派系{1}形成更大派系{1,2,4}，与{1,2,4}邻接的顶点为 3,8，而 3,8 两个顶点形成了一个完全子图，将它们加入{1,2,4}形成最大派系{1,2,3,4,8}。如果从邻接边(4,5)出发，将顶点 4,5 加入派系{1}形成更大的派系{1,4,5}，与{1,4,5}邻接的顶点 6 为第一类邻接点，将它加入{1,4,5}形成最大派系{1,4,5,6}。若从邻接边(6,7)出发，这时，派系{1,6,7}没有邻接顶点，因此，{1,6,7}也就是找到的最大派系。

4.2 基于邻接边的派系过滤算法

本文提出的派系过滤算法包括两个主要过程：RecursiveClique 和 MaxCliqueFinding。第一个过程为递归调用扩展派系的过程，第二个主要过程为从网络中每一个顶点出发，根据此顶点的邻接点的不同情况进行处理。在派系扩展过程中，由于每次都必须在当前派系的邻接点中考察，因此，当加入一条派系的邻接边之后，只需在派系邻接点中考察与此邻接边的两个顶点相邻接的顶点。下面给出过程 RecursiveClique 和 MaxCliqueFinding 的伪代码描述。

```
Procedure RecursiveClique(u1,u2,W,K)
    此过程中，K 为已找到的派系，W 为派系 K 的邻接点集合，u1，
      u2 组成的边(u1,u2)是派系 K 的邻接边。此过程是以派系
      K 为基础，从邻接边(u1,u2)出发，对派系 K 进行扩展。
  从 W 中找出所有与顶点 u1,u2 都相邻的顶点，记为集合 V；
  找出 V 中各顶点之间形成的边，记为集合 E；
  从 V 中找出派系 K 的第一类邻接顶点，记为集合 V'；
  If |E|=|V|*(|V|-1)/2 then
    // V 中顶点形成一个完全子图
    若派系 K+V 不在派系集合中，将 K+V 加入派系集合；
    return;
```

[1] 顶点的 k-派系度有时也定义为包含此顶点的 k-派系数量，而不仅仅是最大 k-派系。

```
    End if
    For V′中每一个点 v do
      //第一类邻接点的处理
      若 K+{v}不在派系集合中,将 K+{v}加入派系集合;
    End for
    For E 中每一条边(u,v) do
      //第二类邻接点的处理,即处理邻接边
      RecursiveClique(u,v,V−V′,K+{u1,u2});
    End for
End procedure

Procedure MaxCliqueFinding()
    输入具有 n 个顶点,m 条边的网络 G(V,E);
    // 从网络 G 出发,找出网络的所有最大派系,这里不考虑 1-派系和 2-派系的情形。
    For V 中每一个顶点 v do
        // 顶点 v 形成派系{v}
        找出{v}的所有邻接顶点,记为集合 Vn;
        If Vn=Φ then continue;//{v}为孤立点,形成一个 1-派系,不考虑
        找出 Vn 中顶点之间形成的边,记为集合 En;
        从 Vn 中找出派系{v}的第一类邻接顶点,记为集合 Vn′;
        If |En|=|Vn|*(|Vn|−1)/2 then
            // Vn 中顶点形成一个完全子图
            若派系{v}+Vn 不在派系集合中,将{v}+Vn 加入派系集合;
                    continue;
        End if
        For En 中每一条边(u1,u2) do
            //Vn′中的顶点和{v}形成 2-派系,属于平凡派系,不考虑
        RecursiveClique(u1,u2,Vn−Vn′,{u1,u2,v});
        End for
    End for
End procedure
```

5 实验结果分析

对本文提出的算法,用 C# 编程语言实现,所有程序运行于主频 2.0GHz,内存 512MB 的奔腾 4 微型计算机。本文选取了两个大规模复杂网络来验证本文的算法。一个是科学家合作网络 sciencenet[19],此网络包括 1589 个复杂网络理论与实验研究领域的研究人员,他们之间的连边表示两个研究者之间合作发表过论文,共有 2742 条连边。另一个更大规模的网络 MathSciNet(Mathematical review collection of the American Mathematical Society)[20],共包括来自 6499 个不同研究领域的 391529 个数学家以及这些数学家之间的 873775 条连边(合作关系)。

利用本文提出的算法,找出了这两个大规模网络中的所有派系,计算时间在 15s 以内,sciencenet 和 MathSciNet 网络中最大的最大派系分别为 20-派系和 25-派系。两个网络的派系大小分布如图 2 和图 4 所示,从图中可以看出,派系大小分布为近似的幂律分布,sciencenet 的幂指数为−3.09,MathSciNet 的幂指数为−5.81。两个网络的顶点派系度分布如图 3 和图 5 所示,从图 3 和图 5 可看出,顶点的派系度分布服从幂指数近似为−3.0 的幂律分布。图 3 和图 5 只列出了 3-派系、4-派系、5-派系和 6-派系的分布,事实上,7-派系乃至更大的派系分布也满足幂律分布的规律。从图 2−图 5 还可看出,对于各领域的科学家,小规模紧密合作的科学家团体很多,而大规模的合作团体很少;而对于某一个科学家,可能参与很多个小规模的团体合作,而参与多个大规模团体合作的情况极为少见,这与实际的科研合作情况是吻合的。

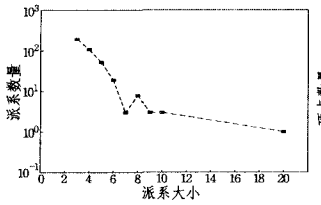

图 2 sciencenet 网络中派系大小分布

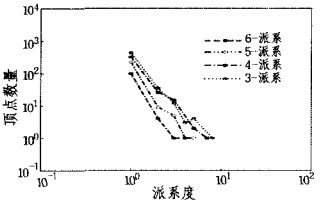

图 3 sciencenet 网络中顶点的派系度分布

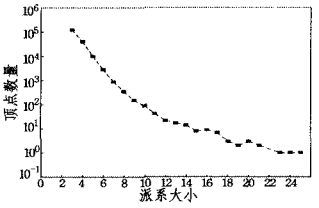

图 4 MathSciNet 网络中派系大小分布

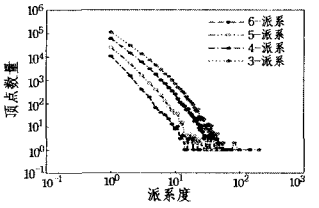

图 5 MathSciNet 网络中顶点的派系度分布

结束语 找出网络的派系,对于分析网络的社区结构,发现网络中紧密合作的团体或机构有极其重要的作用。本文利用派系的邻接顶点和邻接边信息,提出了基于邻接边的递归扩展派系过滤算法。利用本文提出的算法,对 sciencenet 和 MathSciNet 两个大规模复杂网络的派系大小分布和派系度分布进行了分析,实验结果表明,派系大小分布和派系度分布均服从幂律分布。对本文提出的算法,稍加修改即可适合于有向有权网络中派系的查找。

参 考 文 献

[1] Balaus E, Yu C. Finding a maximum clique in an arbitrary graph [J]. SIAM Journal on Computing, 1986, 15(4): 1054-1068

[2] Ji Y, Xu X, Stormo G D. A graph theoretical approach for predicting common RNA secondary structure motifs including pseudoknots in unaligned sequences[J]. Bioinformatics, 2004, 20(10): 1591-1602

[3] Adamcsek B, Palla G, Farkas I J, et al. CFinder: locating cliques and overlapping modules in biological networks[J]. Bioinformatics, 2006, 22(8): 1021-1023

[4] Palla G, Derényi I, Farkas I, et al. Uncovering the overlapping community structure of complex networks in nature and society [J]. Nature, 2005, 435: 814-818

[5] Derényi I, Palla G, Vicsek T. Clique percolation in random networks[J]. Physical Review Letters, 2005, 94: 160202

[6] Du N, Wu B, Pei X, et al. Community detection in large-scale social networks[C]// Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network. , San Jose, California, Aug. 2007: 16-25

[7] Garey M R, Johnson D S. Computers and intractability: A Guide to the theory of NP-completeness[M]. Freeman, San Francisco, CA, USA, 1979

[8] Bron C, Kerbosch J. Algorithm 457: finding all cliques of an undirected graph[J]. Communications of the ACM, 1973, 16(9): 575-577

[9] Tomita E, Kameda T. An efficient branch-and-bound algorithm for finding a maximum clique with computational experiments [J]. Journal of Global Optimization, 2007, 37: 95-111

[10] Pullan W, Hoos H H. Dynamic local search for the maximum clique problem[J]. Journal of Artificial Intelligence Research, 2006, 25: 159-185

[11] Pullan W. Phased local search for the maximum clique problem [J]. Journal of Combinatorial Optimization, 2006, 12: 303-323

[12] 吕强,柏战华,夏晓燕. 一种求解最大团问题的并行交叉熵算法[J]. 软件学报, 2008, 19(11): 2899-2907

[13] Ouyang Q, Kaplan P D, Liu S, et al. DNA solution of the maximal clique problem[J]. Science, 1997, 278: 446-449

[14] 李肯立,周旭,周舒婷. 一种改进的最大团问题 DNA 计算机算法[J]. 计算机学报, 2008, 31(12): 2173-2181

[15] Cui G, Li C, Li H, et al. Application of DNA self-assembly on maximum clique problem[C]// Yu W, Sanchez E N, eds. Advances in Computational Intell., AISC 61. Springer, 2009: 359-368

[16] Singh A, Gupta A K. A hybrid heuristic for the maximum clique problem[J]. Journal of Heuristics, 2006, 12: 5-22

[17] Geng X, Xua J, Xiao J, et al. A simple simulated annealing algorithm for the maximum clique problem[J]. Information Sciences, 2007, 177(22): 5064-5071

[18] Yang G, Yi J, Zhang Z, et al. A TCNN filter algorithm to maximum clique problem[J]. Neuro Computing, 2009, 72(4-6): 1312-1318

[19] Newman M E J. Finding community structure in networks using the eigenvectors of matrices[J]. Physical Review E, 2006, 74: 036104

[20] Palla G, Farkas I J, Pollner P, et al. Fundamental statistical features and self-similar properties of tagged networks[J]. New Journal of Physics, 2008, 10: 23-26

(上接第 180 页)

7 应用分析举例

我们利用上述方法对第 2 节中示例文本进行评估。首先提取文本中一些与信任模式相关的信息,包括主题、各个内容块中的定义、概念和术语、标题以及总结,再利用第 5.2.2 中的论述性 Web 文本阅读自动机构建算法 GARA(HTML),构建文本阅读自动机,生成的阅读自动机如图 4 所示。

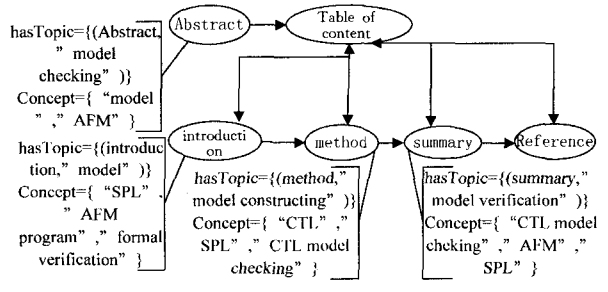

图 4 示例文本的阅读自动机

然后利用第 6.1 节中改进的标记算法 Mark(M, ϕ)对该文本验证模型进行检测,检测结果如表 1 所列。

表 1 示例文本的模型检测结果

序号	ALCCTL 描述的信任模式	文本是否满足
1	AG(→termUdefition)	仅在 Method 中满足
2	Keywords⊑AF∃ addressedBy. Abstract	否
3	majortopic⊑AF∃ addressedBy. narrativeunit	否
4	AG(Definition⊑∀ defines. EX ∃ illustratedBy. Example)	仅在 method 中满足
5	majorparagraph. definition⊑EF∃ addressedBy. narrativeunit	否
6	AG(titleBnarrativeunit)	仅在 summary 中不满足
7	textUmajorparagraph⊑EF ∃ followed. summary	是
8	AG(summary⊑→∃ definition)	是

该检测结果和实际情况是相符的。最后,利用式(1)计算该论述性 Web 文本的可信度。计算结果为文本可信度是 0.25,表明该文本由于行文不规范致使可信度低下,这与实际情况也是相符的。

结束语 本文提出了基于信任模式验证的 Web 文本内容信任判断这一新方法,其主要通过定义 Web 文本的信任模式并用形式化方法描述,然后对 Web 文本阅读自动机,最后利用模型检测算法对 Web 文本内容进行信任模式的验证,从而判断 Web 文本的可信性。这种方法可以在已有的模型检测技术基础上通过设计实现对 Web 文本的信任模式的验证,进而实现 Web 文本可信性的判定。实验证明这种方法是有效的。

在今后的工作中,我们将研究 Web 文本的元数据的自动提取及向描述逻辑的自动转化,并努力挖掘 Web 文本的其它信任模式,再结合这些信任模式对 Web 文本进行建模,然后进行模型检测,判断文本可信性,从而提高 Web 文本可信判断的准确性。

参 考 文 献

[1] Golbeck J, Hendler J. Inferring reputation on the semantic Web [C]// Proceedings of the 13th International World Wide Web Confe-rence. May 2004: 265-275

[2] Gil Y, Artz D. Towards content trust of Web resources[C]// Proceedings of the 15th International World Wide Web Conference. May 2006: 345-357

[3] wiki[EB/OL]. http://en.wikipedia.org/wiki

[4] Weitl F, Jaksic M, Freitag B. Towards the automated verification of semi-structured documents[J]. February 2009: 292-317

[5] 张东启,曾国荪,王伟. 基于信任事实的信任文本信任度评估方法[J]. 计算机科学, 2008, 35(8A): 202-205, 240

[6] Liu P, Chetal A. Trust-based Secure Information Sharing Between Federal Government Agencies[J]. J. of the American for Information Science and Technology, 2005, 56(3): 283-298

[7] 刘群,张华平,俞鸿魁,等. 基于层叠隐马模型的汉语词法分析[J]. 计算机研究与发展, 2004, 41(8): 1421-1429

[8] Weitl F. Document Verification with Temporal Description Logics[D]. Fakultat fur Informatik and Mathematic, University Passau, 2007: 114-145

[9] Schonberg C, Jaksic M, Weitl F, et al. Veirfication of Web-Content: A Case Study on Technical Documentation[C]// Proceedings of the 5th International Worskshop on Automated Specification and Verification of Web System(WWV09). Linz, Austria, February 2009