智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

不确定图上的极大团枚举及高效验证算法

赵丹枫, 吕闫妍, 张文博, 黄冬梅, 高峰

引用本文:
赵丹枫, 吕闫妍, 张文博, 等. 不确定图上的极大团枚举及高效验证算法[J]. 智能系统学报, 2024, 19(6): 1539-1551.
ZHAO Danfeng, LYU Yanyan, ZHANG Wenbo, et al. Maximum clique enumeration and verification algorithm on α uncertain graphs[J]. *CAAI Transactions on Intelligent Systems*, 2024, 19(6): 1539-1551.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202304058

DOI: 10.11992/tis.202304058
网络出版地址: https://link.cnki.net/urlid/23.1538.TP.20240909.1130.012

不确定图上的极大团枚举及高效验证算法

赵丹枫[1], 吕闫妍[1], 张文博[1], 黄冬梅[2], 高峰[3]
（1. 上海海洋大学 信息学院, 上海 201306; 2. 上海电力大学 电子与信息工程学院, 上海 200090; 3. 广东美的制冷有限公司, 广东 佛山 528311）

摘　要: 现有的不确定图中极大团枚举方法“子图划分—枚举—验证”, 在处理大规模图时, 整体效率不高; 当挖掘出的伪极大团数量较多时, 验证速率明显下降。因此, 提出高效枚举及验证算法 (multiple inversion list enumerate uncertain maximal cliques, MILEUMC)。在子图划分和枚举前, 定义并构造概率阈值(α)不确定图, 通过缩小图的规模, 提高枚举效率; 在“验证”时, 提出基于多重倒排表的验证方法, 分为去重复和去包含关系 2 个阶段去除伪极大团, 以不同索引构建各个阶段的多重倒排表, 根据极大团的属性完成验证, 同时动态更新相应的倒排表和映射表, 以减小工作量, 提高时间效率。最后在多个真实的数据集上比较, 结果验证了 MILEUMC 算法的高效性。该算法更适用于在较为稀疏的图上寻找联系更紧密的极大团。

关键词: 不确定图; 极大团; 数据挖掘; 枚举算法; 验证算法; 子图划分; 倒排表; 映射表
中图分类号: TP311　　**文献标志码:** A　　**文章编号:** 1673−4785(2024)06−1539−13

中文引用格式: 赵丹枫, 吕闫妍, 张文博, 等. 不确定图上的极大团枚举及高效验证算法 [J]. 智能系统学报, 2024, 19(6): 1539–1551.
英文引用格式: ZHAO Danfeng, LYU Yanyan, ZHANG Wenbo, et al. Maximum clique enumeration and verification algorithm on α uncertain graphs[J]. CAAI transactions on intelligent systems, 2024, 19(6): 1539–1551.

Maximum clique enumeration and verification algorithm on α uncertain graphs

ZHAO Danfeng[1], LYU Yanyan[1], ZHANG Wenbo[1], HUANG Dongmei[2], GAO Feng[3]
(1. College of Information, Shanghai Ocean University, Shanghai 201306, China; 2. College of Electronics and Information Engineering, Shanghai University of Electric Power, Shanghai 200090, China; 3. Guangdong Midea Refrigeration Co. LTD., Foshan 528311, China)

Abstract: The existing maximal clique enumeration method for uncertain graphs, which uses “subgraph division–enumeration–verification,” is not efficient for large-scale graphs. As the number of pseudo-maximal cliques increases, the verification speed notably decreases. Therefore, the multi-inversion list enumerates uncertain maximal cliques (MILEUMC) algorithm is proposed to address the aforementioned problem. This algorithm improves efficiency by defining and constructing the α uncertain graph before subgraph division and enumeration, which reduces the size of the graph and enhances enumeration efficiency. In the “verification” phase, the algorithm introduces a novel method based on multiple inverted lists. The method involves two stages: the removal of inclusion relations (deduplication) and the removal of pseudo-maximum cliques. During each stage, multiple posting lists with different indexes are built, and verification is completed in accordance with the attributes of maximal cliques. Simultaneously, the corresponding inversion lists and mapping tables are dynamically updated, thereby reducing the workload and saving time. Compared to multiple real data sets, experimental results verify the efficiency of the MILEUMC algorithm. Furthermore, the algorithm is more suitable for identifying maximal cliques in sparser graphs, where connections between nodes are closer.

Keywords: uncertain graph; maximal clique; data mining; enumeration algorithm; verification algorithm; subgraph division; inversion list; mapping table

随着现代化数据采集技术的快速发展, 各行各业中都积累了大量用图表示的数据, 这些图数据规模不仅巨大, 而且还在不断地快速增长[1]。

收稿日期: 2023−04−28.　**网络出版日期:** 2024−09−09.
基金项目: 国家自然科学基金青年项目(42106190, 62102243).
通信作者: 高峰. E-mail: gaofeng14@midea.com.

如何在大规模的图数据中快速准确地找到重要的信息，具有十分重要的现实意义和商业价值。极大团是稠密子图的一种，极大团枚举（maximal cliques enumerate, MCE）用于从给定图中挖掘不被其他团包含的完全子图[2]。MCE 作为一个基本的图数据挖掘问题，应用场景非常多[3]，例如社交网络中发现重叠社区[4]，可进行好友推荐和广告推广[5-6]，蛋白质交互网络（protein-protein interation networks, PPI）中发现不同蛋白质之间的关系，可用于生物研究和制药等[7-9]。

在现实生活中，由于采集数据过程中的噪声[10]、测量误差[11]、预测的准确性[12] 及隐私问题[13] 导致抽象出的图具有不确定性，这种不确定性数据构建的图模型称为不确定图[14]。由于不确定图上的概率维度的增加，这也会导致确定图上的研究成果不能直接应用于不确定图[15]，而现有的MCE算法所关注的大多是精确和完备的确定图[16]，因此亟须研究不确定图上的MCE问题。

围绕不确定图数据挖掘，学者们做出了很多研究，如朱鎔等[17] 研究了不确定图上挖掘稠密子图的问题，邹兆年等[18] 研究了不确定图上挖掘频繁子图模式的问题，相关学者研究了不确定图上的MCE问题[19-24]。关于不确定图上MCE问题的研究，现有的不确定图中极大团挖掘算法通常采用裁剪策略和基于 MULE（Maximal Uncertain Clique Enumeration）算法的递归回溯思想[25]。其中，MULE算法是基于顶点编号升序处理的求解办法，总的时间复杂度是$O(n\cdot 2^n)$，n为顶点数量，但当图的规模更大时，时间效率会降低。由于MULE算法是自底向上的方法，在枚举团时会引入多余的工作，为了提高效率，Rashid等[21] 提出了一种自顶向下的枚举算法，实验证明在本身就稠密的图上该算法与具有相同理论复杂度的其他算法相比，时间快了 30%。但是在稀疏图上，时间性能有待提高。基于此算法，Li 等[20] 提出了基于核的剪枝算法，并就这些剪枝技术改进出新的算法来枚举所有的$(k,\alpha)-\text{maximal}-\text{cliques}$（$\alpha$为概率阈值；$k$为顶点个数），该算法时间复杂度为$O(n\cdot 2^l)$，$l$是顶点数量，其规模远小于原图顶点数量$n$，跟MULE相比，时间效率有所提升，但处理大规模图时仍然很耗时。

由上述分析可知，目前不确定图尤其是大规模不确定图上的MCE问题依然需要解决时间效率问题。因此，本研究提出，构造α不确定图，以减小图的规模；提出了基于该图的高效枚举算法（multiple inversion list enumerate uncertain maximal cliques, MILEUMC），该算法分为“子图划分—枚举—验证”。先将α不确定图划分为规模更小的极大团子图，在极大团子图上用经典的枚举算法MULE枚举所有的α-极大团，在验证极大团时，本研究提出了基于多重倒排表的验证方法，实现了用去重和去包含关系完成去除伪极大团的工作。在每部分使用不同的多重倒排表可以缩减验证的工作量，并且正确地找到并去除伪极大团，从而提高时间效率。

在本研究中，分别定义了不确定图模型、团相关内容和研究的问题；构造α不确定图，并说明了在α不确定图上划分子图的作用和在该图上枚举的正确性；提出了α不确定图上枚举及高效验证算法MILEUMC，以具体的实例说明枚举和验证极大团的过程，展示出所有的算法描述，并对其进行分析；实验证明，使用可表示不确定图的 7 个数据集，对所提出的算法进行 3 种实验，证明了算法的高效性；对本研究内容作出总结，并提供未来不确定图上枚举极大团可研究的相关方向。

1　基本定义

本节介绍关于不确定图模型、极大团相关研究问题的定义。

1.1　不确定图的定义

给定无向不确定图$G=(V,E,P)$，其中，$V=\{v_1,v_2,\cdots,v_n\}$，表示不确定图的顶点集合，$E=\{e_1,e_2,\cdots,e_m\}$，表示不确定图的边集合，P表示图中边的不确定性，以权值的方式给出，取值为$P\in(0,1]$。图 1 为一个不确定图模型。

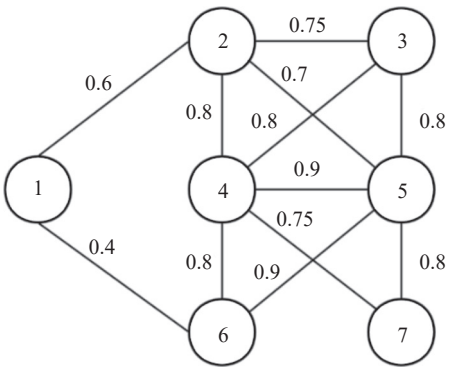


图 1　不确定图

Fig. 1　Uncertain graph

1.2　团的定义

给定无向确定图$g=(V,E)$，C为顶点集合，若$C\subseteq V$，且C中任意 2 点间均有边相连，则 C 为团。如图 2，$C_1=\{1,2,3,4\}$，$C_2=\{2,4,5\}$，C_1和C_2都是团。

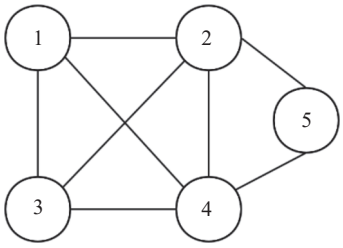


图 2　确定图

Fig. 2　Deterministic graph

1.3　团规模的定义

给定一个团C, C中包含的顶点数量即为团规模的大小。可以表示为size(C)。

1.4　成团概率的定义

给定不确定图$G=(V,E,P)$, 对于图G中的任意一个团C, 成团概率$p(\text{cliq}(C))$是指连接团C中所有顶点的边的概率之积。即

$$p(\text{cliq}(C))=\prod_{e\in C}P(e) \tag{1}$$

例如, 在图 1 上, 团$C_1=\{4,5,6\}$的成团概率$p(\text{cliq}(C_1))=0.8\times0.9\times0.9=0.648$。

1.5　α-极大团的定义

给定无向不确定图$G=(V,E,P)$和概率阈值α, 存在一个顶点集合C, $C\subseteq V$, 若C中任意 2 点之间均有边相连, 且$p(\text{cliq}(C))\geqslant\alpha$, 则$C$为$\alpha$-团。如果不存在一个顶点$v$($v\notin C$且$v\in V$), 使得$C\cup\{v\}$是一个$\alpha$-团, 即$C$不被其他任意顶点集合所包含, 则是一个$\alpha$-极大团。

例如, 在图 1 上, 给定一个α为 0.6, 虽然$C_4=\{4,5\}$是一个α-团($p(\text{cliq}(C_4))\geqslant0.6$), 但是, 其被团$C_3=\{4,5,6\}$所包含, 所以$C_4$不是一个$\alpha$-极大团; 而$C_3$既满足$p(\text{cliq}(C_3))\geqslant0.6$, 且$C_3$不被其他团所包含, 因此$C_3$是满足要求的$\alpha$-极大团。

给定一个不确定图$G=(V,E,P)$和概率阈值$\alpha(0<\alpha<1)$, 要求返回所有的α-极大团。

2　α不确定图的构造

基于子图划分思想的算法EUMC+[22], 在划分子图时过滤掉一些绝不可能成为α-极大团的顶点集合, 能一定程度上减少计算时间, 但顶点和边的数量依然很多。因此, 本研究提出构造α不确定图以缩小图的规模, 提升枚举效率。

2.1　α不确定图生成及子图划分作用

枚举极大团的基本过程是“子图划分—枚举—验证”, 在子图划分的过程中, 通过去掉彼此之间没有边相连的顶点, 可以缩小子图规模, 但在大规模图上处理起来依然很耗时。为减少枚举的工作, 本研究提出构造α不确定图, 用以在子图划分之前减小图的规模, 再进行子图划分和枚举工作。现给出α不确定图的具体定义。

α不确定图的定义　给定一个不确定图G, 去掉图中概率阈值低于α的边, 产生新的不确定图, 由于该图中所有边的概率阈值均大于等于α, 因此称这样的图为α不确定图。

关于在α不确定图上划分子图的作用, 现给出推论和相关证明。

推论 1　在α不确定图上划分子图, 会得到数量更少的极大团子图, 且划分速度会更快。

证明　由于α不确定图已经去掉很多不符合条件的边, 图的规模大大减小, 因此划分后会得到数量更少的极大团子图; 在划分子图时, Degeneracy算法时间复杂度是$O(dn\cdot3^{d/3})$, 其中, n是顶点数量, d表示图的简并度, d远远小于n, 由于α不确定图的边有所减少, 因此, d会减少, 划分的时间复杂度就会降低, 划分速度会更快。

推论 2　在α不确定图上划分子图, 与在不经过处理的不确定图上划分的结果相比, 得到的极大团子图的规模会更小, 且在这样的极大团子图上枚举α-极大团, 时间效率会提升。

证明　α不确定图与原不确定图相比, 少了很多无用的边, 划分子图后, 每个子图是极大团完全连通子图, 所以跟原来划分子图的结果相比, 减少了很多顶点和边, 极大团子图的规模会相对减小。在每个极大团子图上枚举α-极大团时, MULE算法的时间复杂度是$O(n\cdot2^n)$, 其中, n是顶点数量, 在枚举阶段, 由于极大团子图的顶点数量与原不确定图划分的极大团子图的顶点数量相比会减少, 因此枚举阶段的时间也会相应的减少, 时间效率就能够得到提升。

为了证明在α不确定图上的枚举效率更高, 本研究对此进行了实验验证。

2.2　α不确定图上枚举的正确性

在计算成团概率时, 是计算各边之积, 且每条边的概率阈值都小于等于 1, 如果在α不确定图上划分再枚举, 不仅可以减小图规模, 提高算法执行效率, 还能不损失正确性, 成功枚举出所有的α-极大团。

推论 3　在α不确定图中挖掘α-极大团时, 挖掘出来的极大团不会缺少也不会增多, 并且是完全正确的。

证明　在不确定图中, 每条边的存在概率是相互独立的。根据式(1)可知, 成团概率是各边之积。由于每条边的概率值$P\in(0,1]$, 概率阈值$\alpha\in(0,1)$, 所以当出现某条边的概率值小于α时

$(P(e)<\alpha)$，原来的团概率值乘以$P(e)$，结果一定是小于α的，因此在枚举时，低于α阈值的边不会在考虑范围之内，该边一定不会出现在α-极大团中，得到的结果和在原不确定图上的枚举结果一致。

为了验证本研究提出的处理方法可以提高算法的效率，后续将对此进行实验来证明。实验基础分别为不经过处理的不确定图和α不确定图，在不同的数据集上进行实验，检验与此思路相关的 4 种算法，分别是EUMC+、EUML、MULE和MULERPM算法在不确定图和α不确定图上的时间效率。

为了更好地说明情况，本研究将以具体的实例进行分析。

如果在图 3 中找到概率阈值为 0.6 的α-极大团，去掉概率值小于 0.6 的边，结果如图 4 所示。

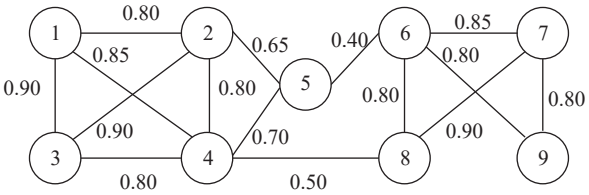


图 3　原不确定图

Fig. 3　Original uncertain graph

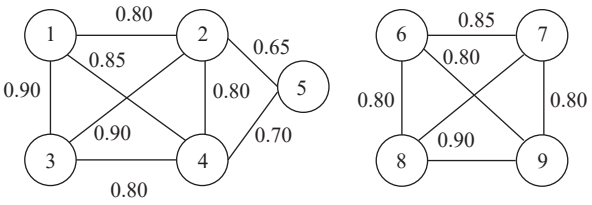


图 4　α 不确定图

Fig. 4　α uncertain graph

3　α 不确定图上的极大团枚举及高效验证

不确定图上枚举极大团现有的思想“子图划分—枚举—验证”中，子图划分阶段，在未经处理的不确定图上进行划分，可能会产生规模大且数量很多的极大团子图，导致枚举效率不高；在验证阶段，现有的使用倒排表和映射表结合的验证方法在伪极大团很多的情况下工作量会很大且耗时。针对以上问题，本研究提出了α不确定图上枚举极大团及高效验证算法 MILEUMC，在α不确定图上进行子图划分，不仅能使划分出来的子图数量减少，子图的顶点数量也能大量减少，进而提高枚举效率；此外，还提出了基于多重倒排表(multiple inversion list, MIL) 的验证方法，实现了去重和去包含关系来去除伪极大团。当伪极大团数量很多时，能够以较少的时间去伪，最终得到所有的α-极大团。具体的操作步骤和算法流程如图 5 所示。

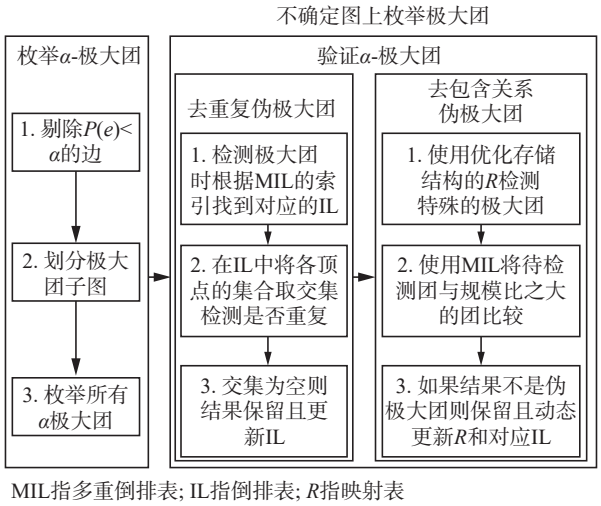


图 5　不确定图上枚举极大团算法步骤

Fig. 5　Enumeration maximal clique algorithm steps on uncertain graphs

3.1　基于子图划分的不确定图α-极大团枚举

在枚举极大团时，本研究使用了基于子图划分的算法和较为经典的MULE算法[22]。

先将α不确定图当作确定图处理，再用 Degeneracy 算法将其划分为极大团子图，然后在每个极大团子图上分别用MULE算法枚举出所有的α-极大团。通过划分可以过滤掉原图中大多数绝不可能成为α-极大团的顶点集合，以减少算法需要考虑的顶点集合的数量和验证次数，跟直接使用MULE算法相比，可以提高算法的计算效率。

针对上述的例子，用Degeneracy算法对图 4 进行划分，结果如图 6 所示。

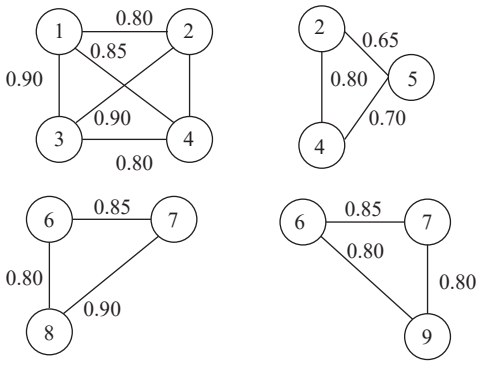


图 6　极大团子图

Fig. 6　Maximal clique subgraph

Li 等[20] 列举的诸多枚举极大团的求解办法包括：基于深度优先搜索(depth-first search, DFS)的办法、基于简并顺序的方法和基于顶点编号升序的办法等。其中基于顶点编号升序的算法(MULE)与其他算法相比性能较好，时间复杂度是$O(n\cdot 2^n)$。虽然这种算法在顶点规模较大时，时

间代价大, 但是本研究的划分思想, 将原图划分为规模更小的极大团子图, 在极大团子图上调用MULE算法, 与直接使用MULE相比, 时间性能是较优的。

在划分后的所有极大团子图上用MULE算法枚举出符合条件的α-极大团。上述例子中, 在4个子图上分别调用MULE算法, 能够得到所有的α-极大团 Aumc={{1,2,3},{1,3,4},{2,4},{2,4},{2,5},{4,5},{6,7,8},{6,7},{6,9},{7,9}}。但是这些α-极大团里面存在伪极大团。在不确定图G中, 不同的极大团子图挖掘出来许多满足条件的α-极大团, 若该团满足以下2种情况之一, 则被称为α-伪极大团。1)存在一个极大团C, 被其他极大团子图挖掘出来的更大的α-极大团所包含; 2)存在2个完全相同的α-极大团C。接下来需要去伪得到最终的结果集。

3.2　基于多重倒排表的不确定图枚举极大团验证

本研究提出基于多重倒排表的验证方法, 分为去重和去包含关系去除伪极大团。

1) 去重复极大团, 在此处, 多重倒排表MIL的索引为极大团的规模和各顶点之和, 表示为(size,sum), 每个索引后有其对应的倒排表IL, 在IL中, 索引为顶点编号, 每个索引后为极大团编号, 意为该顶点存在于哪些极大团中, 表示为(key,number)。

2) 去包含关系极大团, 在此处, MIL的索引为极大团的规模, 表示为(size), 每个索引后同样有其对应的IL, IL的索引为顶点编号, 每个顶点编号后的内容也是极大团的编号, 表示为(key,number)。

基于MIL的验证方法在处理伪极大团数量很多的问题上, 能够有效减小验证工作量, 在去重过程中, 增加团内元素之和的排序维度, 即在规模相同的情况下, 按照元素之和的大小排序, 进一步判断这些元素是否相同, 使用MIL的数据结构来去除重复伪极大团, 能够提高验证效率; 在去包含关系过程中, 使用MIL和优化后的映射表R去除伪极大团, 可以优先筛选出具有某些特征的极大团, 从而提升验证效率。

3.2.1　去重复伪极大团

去重, 去掉完全相同的α-极大团。每检测到一个α-极大团, 根据索引找到其对应的IL, 再找到极大团中包含的顶点, 将顶点对应的集合取交集, 如果交集为空, 说明该团在IL中没出现过, 是不重复的α-极大团。更新IL中该极大团出现的记录, 并将该团放入不含有重复伪极大团的结果集Aumc′中。

在验证α-极大团过程中, 对原Aumc集合按照团规模降序排序。为了能快速去除重复的伪极大团, 在此基础上, 本研究提出, 在排序时, 规模相同的情况下, 按照元素之和的大小排序, 进一步判断这些元素是否相同, 如果相同, 去掉相同的其他元素, 可以减少待验证的Aumc集合的规模; 当重复团数量很大的时候, 能提高算法执行效率。

对α-极大团进行操作:

首先建立MIL, 索引为α-极大团的规模和各顶点编号之和。根据每个团, 得出MIL的索引值有(3,6)、(3,8)、(2,6)、(2,7)、(2,9)、(3,21)、(2,13)、(2,15)、(2,16), 每个不同的(size,sum)都有对应的IL, IL的索引为顶点编号, 索引下为包含该顶点的α-团的编号的集合。规定第1个α-团编号为0。

当检测到极大团{2,4}时, 按照规定找到索引值为(2,6)的IL, 初始时, 每个顶点下的集合都为空集。在IL中找到团包含的顶点2和顶点4, 将对应的集合取交集, 发现交集为空, 说明该团中的顶点在此之前没有出现在别的团中, 更新IL的值, 将对应的团编号填入, 并将该团放入不含有重复的α-极大团结果集Aumc′中。

再次遇到极大团{2,4}时, 根据上述操作, 对顶点2和顶点4对应的集合取交集, 交集为2, 说明该团中的顶点存在于编号为2的团中, 由于此IL中存放的都是规模相同的团, 因此交集不为空, 说明该团与之前的某团重复, 不更新IL, 更新后的倒排表为表1。

表 1　更新后 (2,6) 对应的倒排表

Table 1　Inversion list corresponding to (2,6) after updating

倒排表IL索引	第1次结束后	第2次结束后
1	∅	∅
2	2	2
3	∅	∅
4	2	2
5	∅	∅
6	∅	∅
7	∅	∅
8	∅	∅
9	∅	∅

注: ∅表明空集合。

按上述流程操作完得到最后的结果集 Aumc′={{1,2,3},{1,3,4},{2,4},{2,5},{4,5},{6,7,8},{6,7},{6,9},{7,9}}。

3.2.2　去包含关系伪极大团

去包含关系的α-极大团。先将Aumc′按规模从大到小进行特殊的快速排序, 得到集合Aumc。

处理时，首先使用映射表R筛选具有特征的α-极大团。R的索引为顶点编号，每个顶点编号后是该顶点出现在α-极大团中的次数，也可以理解为出现在几个α-极大团中。

每检测到一个α-极大团，在R中寻找对应的顶点，如果某顶点下的结果$R[v]$为 0，说明该团是一个之前从未出现过的团，此时动态更新R和对应的IL。该方法可以筛选出某些顶点只出现在某个极大团中的特殊情况。使用R后，再使用MIL去伪。只有检测到的团为真正的α-极大团时，才更新R和IL。无法检测时，使用MIL验证。由于Aumc是降序排序，第 1 个极大团一定是真正的α-极大团，当检测一个极大团时，只需要与其规模更大的极大团进行比较，找到对应规模的IL进行相应操作。如果现有的MIL里面没有当前被检测的极大团规模的IL，根据当前的极大团，建立对应的IL。然后找到当前团的顶点，将顶点对应的number集合取交集。如果交集为空，说明当前团不被其他团所包含，是真正的α-极大团，动态更新R和IL；如果不为空，说明该团是伪极大团，需要使用 falseSet 记录，最后在Aumc中除去 falseSet 里存在的极大团，返回最终结果集 result。

按上述规则操作后，检验到{6,7,8}时，由于前 3 个集合中总有顶点后的value值在检测时为 0，因此能够检测出具有此特征的α-团{1,2,3}、{1,3,4}、{6,7,8}为真正的α-极大团，检测完{6,7,8}的R结果见表 2。

表 2　检测完 (6,7,8) 后的映射表

Table 2　Mapping table after testing (6,7,8)

顶点编号	1	2	3	4	5	6	7	8	9
团的个数	2	1	2	1	0	1	1	1	0

检测{2,4}时，顶点 2 和顶点 4 的value值均不为 0，所以需要借助MIL检测是否为包含关系的伪极大团。检测到{2,5}时，顶点 5 的value值为 0，所以{2,5}是一个α-极大团。同理，检测到{4,5}、{6,7}和{7,9}时，对应顶点的value值均不为 0，需要借助MIL去伪。检测{6,9}时，9 的value值为 0，因此{6,9}为α-极大团。结果见表 3。

表 3　检测完成后的映射表

Table 3　Mapping table after testing

顶点编号	1	2	3	4	5	6	7	8	9
团的个数	2	2	2	1	1	2	1	1	1

在去包含关系的伪极大团时，本研究同样使用MIL去伪，MIL的索引为团的规模，每个规模对应一张IL。对应上述实例，同步更新规模为 3 和规模为 2 的IL，使用完R后更新的最终IL结果见表 4 和表 5。

表 4　规模为 3 的倒排表

Table 4　Inversion list of size 3

倒排表IL索引	第1次更新后	第2次更新后	第3次更新后
1	0	0, 1	0, 1
2	0	0	0
3	0	1	1
4	∅	1	1
5	∅	∅	∅
6	∅	∅	2
7	∅	∅	2
8	∅	∅	2
9	∅	∅	∅

表 5　规模为 2 的倒排表

Table 5　Inversion list of size 2

倒排表IL索引	第1次更新后	第2次更新后
1	∅	∅
2	4	4
3	∅	∅
4	∅	∅
5	4	4
6	∅	7
7	∅	∅
8	∅	∅
9	∅	7

团{1,2,3}、{1,3,4}、{6,7,8}，其规模已是最大，一定不会被同规模的团包含。接下来使用MIL对{2,4}、{4,5}、{6,7}、{7,9}进行验证。由于待检验的团规模都是 2，且已去重，所以只需要将这些团与规模大于 2 的团进行比较即可。

检测{2,4}时，在规模为 3 的IL中找到顶点 2 和顶点 4，将顶点的number集合取交集，交集为空，说明该团中的顶点不存在于之前的任何团中，更新IL和R。同理，检测{4,5}，结果为α-极大团，更新 ZL 和 R。当检测到{6,7}时，由于顶点 6 和顶点 7 的number交集为 2，不为空，说明顶点 6 和 7 曾出现在团 2 中，记录 falseSet[MC] 值为 true。检测{7,9}，结果为α-极大团，更新两表。

R和IL的最终结果见表 6 和表 7。

表 6　去包含关系中映射表的最终结果

Table 6　Final results of mapping table on the eliminate containing relationship

顶点编号	1	2	3	4	5	6	7	8	9
团的个数	2	3	2	3	2	2	2	1	2

表 7　去包含关系中倒排表的最终结果

Table 7　Final results of inversion list on the eliminate containing relationship

倒排表IL索引	第1次更新后	第2次更新后	第3次更新后	最终更新结果
1	0	0, 1	0, 1	0, 1
2	0	0	0	0, 3
3	0	1	1	1
4	∅	1	1	1, 3, 5
5	∅	∅	∅	5
6	∅	∅	2	2
7	∅	∅	2	2, 8
8	∅	∅	2	2
9	∅	∅	∅	8

最后, 将不在 falseSet 中的团MC存储在 result 中, 所有的α-极大团为 result={{1,2,3},{1,3,4},{6,7,8},{2,4},{2,5},{4,5},{6,9},{7,9}}。

3.3　算法描述与分析

本研究提出了α不确定图上极大团枚举及验证算法MILEUMC, 以下给出相关算法的描述和算法分析。

算法 1　MILEUMC( )

输入　不确定图 G, 概率阈值 α

输出　α-极大团结果集 result

1) Aumc ← ∅, result ← ∅

2) //剔除 E 中概率小于 α 的边

3) **for all** $e \in E$ **do**

4) **if** $P(e) < \alpha$ **then**

5) Delete e from G

6) **end if**

7) **end for**

8) **return** G

9) Amc ← Degeneracy(G) //通过子图划分算法得到极大团子图集合

10) **for** each subgraph g in Amc **do**

11) Aumc ← MULE(g,α) //对每个极大团子图调用 MULE 算法

12) **end for**

13) Aumc = EarseDup(Aumc) //在 Aumc 中去除重复团并根据团规模进行排序

14) MIL = GenerateMIL(Aumc) //初始化多重倒排表

15) falseSet = FindAllFalseMC(Aumc,MIL)//根据多重倒排表找到所有包含关系的伪极大团

16) //去除伪极大团

17) **for** each MC in Aumc **do**

18) **if** MC is not in falseSet **then**

19) result ← MC

20) **end if**

21) **end for**

22) **return** result

算法 2　GenerateMIL( )

输入　去重后的结果集 Aumc

输出　多重倒排表 MIL

1) //将规模最大的团编号插入到多重倒排表中

2) IL ← ∅, MIL ← ∅, C ← Aumc[0]

3) **for** each v in C **do**

4) IL[v] ← 0

5) **end for**

6) IL.MCsize ← C.size

7) MIL ← IL

8) **return** MIL

算法 3　EraseDup( )

输入　调用 MULE 的结果集 Aumc

输出　不含有重复团的结果集 Aumc′

1) //此处多重倒排表 (MSIL) 索引为团规模与顶点编号之和, 加快查找重复团的速度

2) MSIL ← ∅, Aumc′ ← ∅

3) **for** each MC in Aumc **do**

4) IL ← MSIL [MC.size,MC.sum]

5) X ← IL[v_0]

6) **for** each v in MC($v \neq v_0$) **do**

7) $X \leftarrow X \cap$ IL[v]

8) **end for**

9) **if** $X = \varnothing$ **then**

10) **for** each v in MC **do**

11) IL[v] ← MC.index

12) Aumc′ ← MC

13) **end for**

14) **end if**

15) **end for**

16) **return** Aumc′

算法 4　FindAllFalseMC( )

输入　去重后的结果集 Aumc, 多重倒排表 MIL

输出　包含关系的伪极大团集合 falseSet

1) R ← ∅, falseSet ← ∅

2) //R 为映射表, falseSet 记录伪极大团

3) **for** each MC in Aumc **do**

4) //对每个团先使用映射表验证, R[v] 值为 0 说明该团是 α-极大团, 不需要再用 MIL 验证

5) **for** each v in MC **do**

6) if R[v] = 0 then

7) UpdateMILR(MC,MIL,R)

8) **end if**
9) **end for**
10) **if** 映射表检测为 α-极大团 **then**
11) **continue** //继续下一次团验证
12) **end if**
13) //在多重倒排表中按照规模大小使用倒排表验证，对每个倒排表取交集，交集均为空则该团为极大团，反之为伪极大团
14) **for** each IL in MIL **do**
15) **if** IL.MCsize ≤ MC.size **then**
16) **break**
17) **end if**
18) $X \leftarrow \mathrm{IL}[v_0]$
19) **for** each v in MC **do**
20) $X \leftarrow X \cap \mathrm{IL}[v]$
21) **end for**
22) if $X = \varnothing$ **then**
23) UpdateMILR(MC,MIL,R)
24) **else**
25) falseSet[MC] = true
26) **end if**
27) **end for**
28) **end for**
29) **return** falseSet

算法 5　UpdateMILR( )

输入　当前 α-团 MC，多重倒排表 MIL，映射表 R

输出　更新后的多重倒排表 MIL，映射表 R

1) ILL ← ∅
2) //若当前团规模与多重倒排表中最小规模倒排表规模不同，则需要新建倒排表，反之使用最小规模倒排表
3) **if** LastIL.MCsize == MC.size **then**
4) ILL = LastIL
5) **else**
6) ILL.MCsize = MC.size
7) MIL ← ILL
8) **end if**
9) //将团编号插入倒排表，并更新多重倒排表和映射表
10) **for** each v in MC **do**
11) ILL[V] ← MC.index
12) $R[v] \leftarrow R[v] + 1$
13) **end for**
14) MIL ← ILL
15) **return** MIL, R

枚举阶段中，构造α不确定图，用Degeneracy算法划分子图的最坏时间复杂度是$O(n' \cdot 3^{n'/3})$，其中，n'为子图中的顶点数目，远小于原图中的顶点数目n。在子图中调用MULE算法的最坏时间复杂度为$O(n' \cdot 2^{n'})$。

验证阶段中，Li 等[20] 的验证算法DPMC最坏时间复杂度为$O(n' \cdot 3^{n'/3})$，杜明等[23] 的改进FDPMU算法最坏时间复杂度也是$O(n' \cdot 3^{n'/3})$，最好情况的时间复杂度为$O(3^{n'/3})$，张艺等[24] 的验证算法IRPMC时间复杂度为$O(n \cdot 3^{n/3})$。而本研究提出的基于多重倒排表的验证方法去除伪极大团的最坏时间复杂度为$O(n' \cdot 3^{n'/3})$，最好情况下也是$O(3^{n'/3})$，但是在此时间维度下，由于本研究将复杂的工作简单化，工作量减小，提出的验证算法所花费的时间在大多数情况下，都要比前 2 种验证算法花费时间少。

由此可以得出，算法MILEUMC的最坏时间复杂度为$O(n' \cdot 3^{n'/3})$，与子图划分思路相同的 2 种算法EUMC+、EUML相比，执行效率更高。

4　实验分析

为验证算法的正确性和有效性，本研究进行了 3 种实验。下面将分别对实验的不同环节进行验证。

4.1　实验环境

实验所使用的硬件平台是 11th Gen Intel(R) Core(TM) i7-11700K 主频为 3.60 GHz 的 CPU，16 GB 的 RAM 内存，操作系统为 Windows10 64 位的系统；实验的运行环境是 Microsoft Visual Studio 2022；编程语言为 C++。

4.2　数据集

本研究使用 7 个数据集来评估几种算法的性能。这些数据集包括 amaze5、yago5、mtbrv5、citeseer5、anthra5、ecoo5、agrocyc5。其中 anthra5、mtbrv5、agrocyc5 和 ecoo5 描述了生物化学和基因组之间的生物学序列；amaze5描述了生物体内的化学反应网络；citeseer5描述了蛋白质反应的传递网络；yago5描述了不同语义关系之间的语义信息。这些信息各自代表了不同领域的相关知识，都蕴含有不确定信息，均可用不确定图来存储信息。具体的数据集信息见表 8。其中|V|表示不确定图中的顶点数目，|E|表示不确定图中边的数目。

4.3　性能比较与分析

本研究主要进行了 3 种实验，实验 1 为在同一数据集下，在α不确定图上和原不确定上进行极大团枚举的算法效率对比实验；实验 2 为在给定同一个α概率阈值的情况下，通过表 8 的数据

集生成不确定图来枚举极大团, 比较算法MULE[19]、EUMC+[22]、EUML[23]、IMULERPMC[25] 和本研究提出的枚举算法MILEUMC在 7 个不同的数据集上执行的时间效率; 实验 3 为在同一数据集中, 5 种算法在不同的α情况下时间性能的比较。最后通过对比, 证实MILEUMC算法的高效性。

表 8　数据集
Table 8　Datasets

数据集	$\|V\|$	$\|E\|$
amaze5	3 342	7 200
yago5	6 642	84 784
mtbrv5	9 602	20 490
citeseer5	10 720	88 516
anthra5	12 495	26 208
ecoo5	12 620	26 700
agrocyc5	12 684	26 816

4.3.1　相同数据集下的不同不确定图的算法对比实验

为更好地证明在α不确定图上枚举α-极大团算法的高效性, 选用具有代表性的 2 种数据集, 分别是$|E|/|V|$值在 2 左右的amaze5数据集和$|E|/|V|$在 12 左右的yago5数据集, 分别进行了α为 0.6 和 0.8 的 2 种实验。进行比较的算法分别是EUMC+、EUML和MULE算法, 如图 7 和图 8 所示。

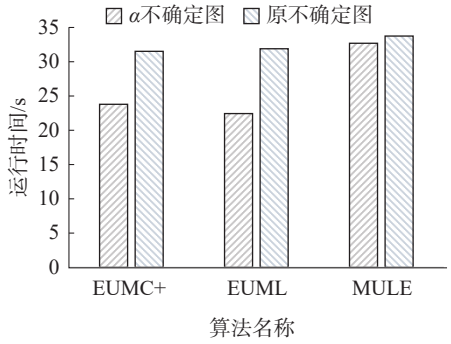


图 7　α 为 0.6 时各算法在 amaze5 数据集上的运行时间
Fig. 7　Running time of each algorithm on amaze5 data set (α is 0.6)

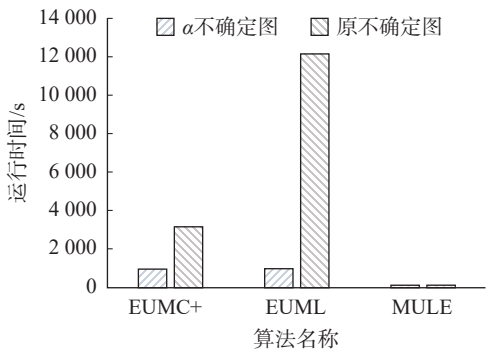


图 8　α 为 0.6 时各算法在 yago5 数据集上的运行时间
Fig. 8　Running time of each algorithm on yago5 data set (α is 0.6)

由图 7 和图 8 可以看出, 3 种算法在amaze5数据集和yago5数据集上, 在α不确定图上的运行时间效率更高, 并且经过检查对比, 3 种算法在α不确定图上均能正确枚举出所有的α-极大团。

由图 9 和图 10 可以看出, 以“划分−枚举−验证”为思路的EUMC+和EUML算法在α不确定图上比在原不确定图上运行要快很多, 在MULE算法中, 运行时间接近。由于yago5数据集的$|E|/|V|$值为 12 左右, 即在边更多的大图上, 采用划分子图的方法, 可能会产生很多的子图, 因此在每个子图上枚举反而会比较耗时, 而MULE算法按照节点编号进行回溯, 使得枚举过程不会产生伪极大团, 所以在这样的图上表现会较好, 因此, 在α不确定图和原不确定图上运行, MULE算法在原不确定图上运行时间会比在α不确定图上稍快一些, 但总体相差不大。

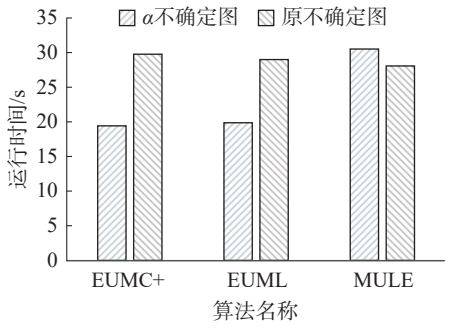


图 9　α 为 0.8 时各算法在 amaze5 数据集上的运行时间
Fig. 9　Running time of each algorithm on amaze5 data set (α is 0.8)

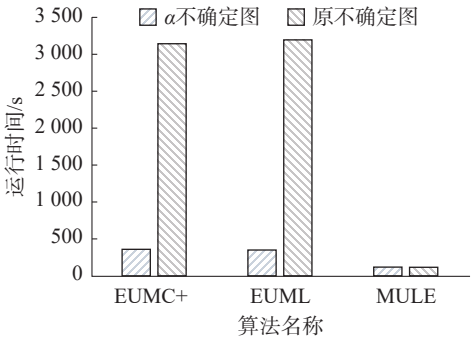


图 10　α 为 0.8 时各算法在 yago5 数据集上的运行时间
Fig. 10　Running time of each algorithm on yago5 data set (α is 0.8)

4.3.2　阈值相同下的数据集对比实验

为更好比较几种算法的执行效率, 随机确定几个α的值, 这里取α为 0.2、0.6 和 0.9。结果如图 11~13 所示。

当α设置为 0.2 时, 可以看出MILEUMC算法在不同的数据集上表现不同。在$|E|/|V|$值较大的yago5数据集和citeseer5数据集上, 该算法与其他算法的区别较大, 时间效率不高, 在这 2 个数据集

上，IMULERPMC算法是运行效率最高的。在其他数据集上，MILEUMC算法时间效率比MULE高，比IMULERPMC、EUML和EUMC+算法低。

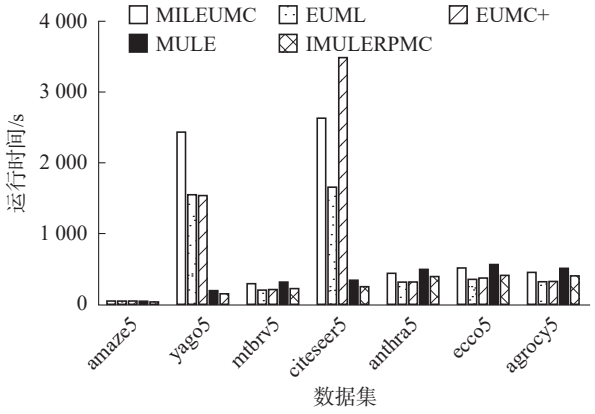


图 11　α 为 0.2 时各算法在 7 个不同的数据集上的运行时间

Fig. 11　Running time of each algorithm on 7 different data sets (α is 0.2)

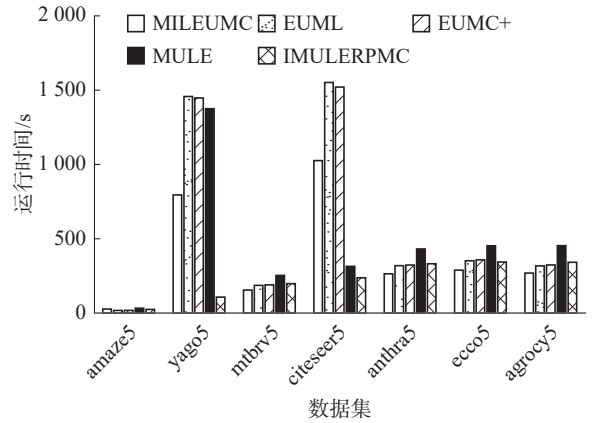


图 12　α 为 0.6 时各算法在 7 个不同的数据集上的运行时间

Fig. 12　Running time of each algorithm on 7 different data sets (α is 0.6)

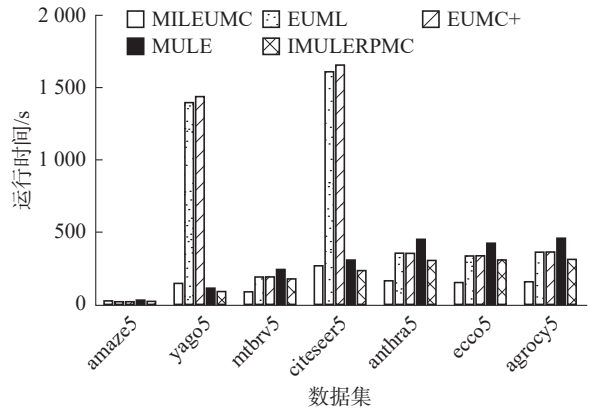


图 13　α 为 0.9 时各算法在 7 个不同的数据集上的运行时间

Fig. 13　Running time of each algorithm on 7 different data sets (α is 0.9)

在yago5数据集和citeseer5数据集中，由于此时的概率阈值较小，采用“子图划分−枚举−验证”思路的EUMC+、EUML和MILEUMC会枚举出较多的α-极大团，验证的时间也就更多；MULE算法中不需要验证阶段，按照节点编号回溯的机制避免了伪极大团的产生；而IMULERPMC算法在改进MULE算法的过程中通过计算扩展后的集合概率验证是否为α-团，减少算法递归次数，还删除已使用顶点集合X的计算，提高了算法执行效率，跟MULE算法相比，效率更高。因此，当α阈值较小时，IMULERPMC在 5 种算法中，时间效率高。

由图 12 可知，当α设置成为 0.6 时，MILEUMC算法在时间效率上总体高于其他 4 种算法，性能得到了很大的提高。只有在citeseer5数据集和yago5数据集上MILEUMC算法不是最快的，IMULERPMC 时间效率最高，但是MILEUMC算法比EUML和EUMC+算法运行效率都高。尤其在 yago5 数据集上，MILEUMC 算法跟 EUML 算法和 EUMC+算法相比，时间快了近 1 倍。

由图 13 得知，当α设置成为 0.9 时，与α为 0.6 时相比，在yago5和citeseer5数据集上表现有所提升，时间效率接近IMULERPMC，且与子图划分思路相同的EUML和EUMC+算法相比，时间效率快 7~8 倍。在其他$|E|/|V|$为 2 左右的数据集上，MILEUMC算法都比其他 4 个算法快很多，只需要花一半时间就可以枚举出所有的极大团。

结合表 9 和实验的结果发现，随着α的增大，MILEUMC算法的运行效率会越来越高，即当想要挖掘联系更加紧密的子图时，在$|E|/|V|$相对不大的图上，MILEUMC算法是最优选择，可以节约一半的时间。

表 9　各数据集顶点和边比较情况

Table 9　Comparison of vertices and edges on the each dataset

数据集	$\lvert E\rvert/\lvert V\rvert$
amaze5	2.15
yago5	12.76
mtbrv5	2.13
citeseer5	8.26
anthra5	2.10
ecoo5	2.12
agrocyc5	2.11

4.3.3　数据集相同下的阈值对比实验

本研究还进行了不同α对MILEUMC、EUML、

EUMC+、IMULERPMC和MULE算法的时间性能影响的实验。针对|*E*|/|*V*|值为 2 的amaze5数据集、|*E*|/|*V*|值为 2 的边点更多的anthra5数据集和|*E*|/|*V*|值为 8 左右的citeseer5数据集检测了α为0.1~0.9时, 5 种算法运行时间的变化。图 14 和图 15为实验的结果。

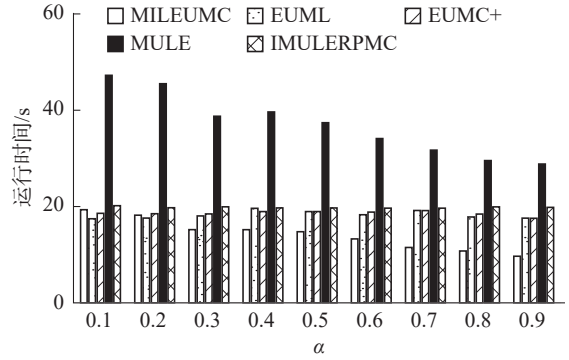


图 14　5 种算法在同一数据集 amaze5 上不同 α 的运行时间

Fig. 14　Running time of five algorithms with different α values on the same data set amaze5

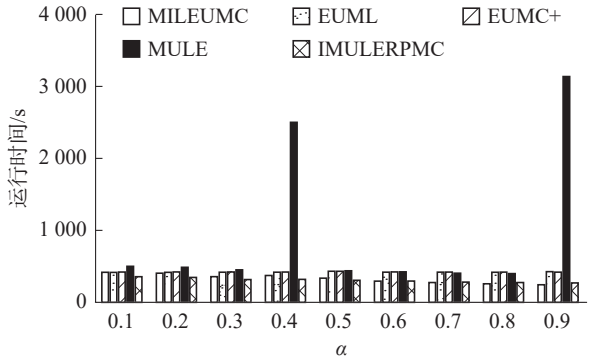


图 15　5 种算法在同一数据集 anthra5 上不同 α 的运行时间

Fig. 15　Running time of five algorithms with different α values on the same data set anthra5

由图 14 可以看出, 在amaze5数据集上, 随着α取值越来越大, EUML算法运行时间先上升后保持平稳波动, EUMC+和IMULERPMC算法时间变化不大, MULE算法在 5 种算法中始终是最耗时的, 而MILEUMC算法的运行时间越来越短, 时间效率大大提高。几乎所有的α取值都满足MILEUMC算法是最快的, 尤其是当α为 0.9 时, MILEUMC算法比MULE算法运行时间快近 2 倍, 比其他 3 种算法运行时间快 1 倍左右。

由图 15 可以看出, 当α很小, 为 0.1、0.2 时, IMULERPMC算法运行时间最短。但随着α的增长, MILEUMC算法运行时间越来越短, 当α增长到 0.6 时, 从此处开始, MILEUMC算法在 5 种算法中表现最好, 且到α为 0.9 该算法的效率都是最高的。

由图 16 可知, 在citeseer5数据集上, 随着α的不断增加, MULE、IMULERPMC 和 EUMC+算法性能较平稳, 变化不大, 其中, IMULERPMC算法时间消耗较少; EUML算法运行时间先上升后下降, 波动较大; MILEUMC算法的运行时间会越来越少, 从α变为 0.3 开始, 运行效率比 EUML 和EUMC+都好; 在 5 种算法中, 在citeseer5数据集上IMULERPMC算法表现最好。

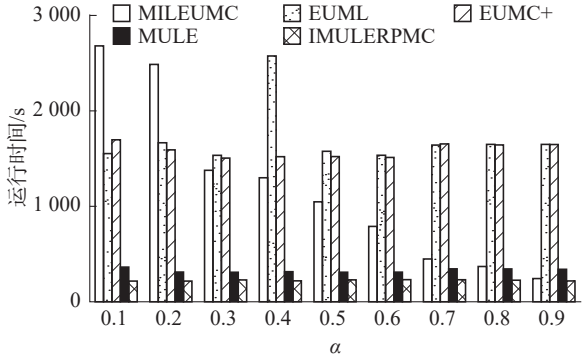


图 16　5 种算法在同一数据集 citeseer5 上不同 α 的运行时间

Fig. 16　Running time of five algorithms with different α values on the same data set citeseer5

综合表 9 和实验结果可以看出, 在|*E*|/|*V*|值为 2 的相对稀疏的数据集不管是规模小的amaze5, 还是规模较大的anthra5上, MILEUMC算法表现始终最好, 是 5 种算法中运行效率最高的; 而在|*E*|/|*V*|值较大的数据集citeseer5上, MILEUMC算法虽然没有IMULERPMC算法时间性能好, 但总体来说, 比EUML和EUMC+算法的效率更优。

5　结束语

本研究分析了在不确定图上枚举极大团的问题。针对时间效率问题, 提出了α不确定图的构建及其上的极大团枚举算法MILEUMC, 尤其在验证方面, 提出了基于多重倒排表的验证方法, 使用多重倒排表和优化过的映射表去除伪极大团。通过实验证明, 构建α不确定图用于枚举, 在大规模图上, 最快可以节约近 10 倍的时间; 基于多重倒排表的验证方法下, 枚举算法 MILEUMC在要求挖掘联系很紧密, 即阈值为 0.8、0.9 的极大团时, 7 个数据集中在|*E*|/|*V*|值为 2 左右的较稀疏的大图上, MILEUMC 算法运行效率最高, 更具有现实意义; 在|*E*|/|*V*|值约为 8 或 12 的较稠密的图上随着α值的增大, 相较 EUMC+和 EUML 算法,

该算法表现也会越来越好。

虽然大多数情况下，MILEUMC 算法与其他算法相比，运行效率较高，但是当α很小时（0.1~0.2），不管是在稀疏图还是稠密图上，该算法没有其他算法的运行效率高。未来，还需研究更适合所有情况下的效率更高的不确定图极大团枚举算法。

参考文献：

[1] 徐铭. 不确定图的独立集指标[D]. 北京: 北京交通大学, 2021: 11−12.
XU Ming. Independent set index of uncertain graph[D]. Beijing: Beijing Jiaotong University, 2021: 11−12.

[2] 王恒. 概率图上 Top-K 极大团枚举问题研究[D]. 上海: 东华大学, 2021: 5−6.
WANG Heng. Research on enumeration problem of Top-K maximal clique on probability graph[D]. Shanghai: Donghua University, 2021: 5−6.

[3] 徐兰天, 李荣华, 戴永恒, 等. 面向超图的极大团搜索算法[J]. 计算机应用, 2023, 43(8): 2319−2324.
XU Lantian, LI Ronghua, DAI Yongheng, et al. Maximal clique searching algorithm for hypergraphs[J]. Journal of computer applications, 2023, 43(8): 2319−2324.

[4] BAI Liang, CHENG Xueqi, LIANG Jiye, et al. Fast graph clustering with a new description model for community detection[J]. Information sciences, 2017, 388/389: 37−47.

[5] KUTER U, GOLBECK J. Using probabilistic confidence models for trust inference in Web-based social networks [J]. ACM transactions on Internet technology, 2010, 10(2): 1−23.

[6] MEHMOOD Y, BONCHI F, GARCÍA-SORIANO D. Spheres of influence for more effective viral marketing [C]//Proceedings of the 2016 International Conference on Management of Data. San Francisco: ACM, 2016: 711−726.

[7] ABU-KHZAM F N, BALDWIN N E, LANGSTON M A, et al. On the relative efficiency of maximal clique enumeration algorithms, with applications to high-throughput computational biology[C]//International Conference on Research Trends in Science and Technology. [S. l.]: [s. n.], 2005: 1−10.

[8] KOCH I, LENGAUER T, WANKE E. An algorithm for finding maximal common subtopologies in a set of protein structures[J]. Journal of computational biology: a journal of computational molecular cell biology, 1996, 3(2): 289−306.

[9] SAHA B, HOCH A, KHULLER S, et al. Dense subgraphs with restrictions and applications to gene annotation graphs[C]//BERGER B. Annual International Conference on Research in Computational Molecular Biology. Berlin: Springer, 2010: 456−472.

[10] AGGARWAL C C. Managing and mining uncertain data[M]. [S. l.]: Springer, 2009.

[11] ADAR E, RE C. Managing uncertainty in social networks[J]. IEEE data engineering bulletin, 2007, 30(2): 15−22.

[12] LIBEN-NOWELL D, KLEINBERG J. The link prediction problem for social networks[C]//Proceedings of the Twelfth International Conference on Information and Knowledge Management. New Orleans: ACM, 2003: 556−559.

[13] BOLDI P, BONCHI F, GIONIS A, et al. Injecting uncertainty in graphs for identity obfuscation[EB/OL]. (2012−08−21) [2021−01−01]. http://arxiv.org/abs/1208.4145.

[14] ZOU Zhaonian, LI Jianzhong, GAO Hong, et al. Mining frequent subgraph patterns from uncertain graph data[J]. IEEE transactions on knowledge and data engineering, 2010, 22(9): 1203−1218.

[15] 桂飞. 面向不确定图的社区发现与搜索算法研究[D]. 合肥: 中国科学技术大学, 2020: 5−6.
GUI Fei. Research on community discovery and search algorithm for uncertain graph[D]. Hefei: University of Science and Technology of China, 2020: 5−6.

[16] 邹兆年, 朱鎔. 大规模不确定图上的 Top-K 极大团挖掘算法[J]. 计算机学报, 2013, 36(10): 2146−2155.
ZOU Zhaonian, ZHU Rong. Mining Top-K maximal cliques from large uncertain graphs[J]. Chinese journal of computers, 2013, 36(10): 2146−2155.

[17] 朱鎔, 邹兆年, 李建中. 不确定图上的 Top-K 稠密子图挖掘算法[J]. 计算机学报, 2016, 39(8): 1570−1582.
ZHU Rong, ZOU Zhaonian, LI Jianzhong. Mining Top-K dense subgraphs from uncertain graphs[J]. Chinese journal of computers, 2016, 39(8): 1570−1582.

[18] 邹兆年, 李建中, 高宏, 等. 从不确定图中挖掘频繁子图模式[J]. 软件学报, 2009, 20(11): 2965−2976.
ZOU Zhaonian, LI Jianzhong, GAO Hong, et al. Mining frequent subgraph patterns from uncertain graphs[J]. Journal of software, 2009, 20(11): 2965−2976.

[19] MUKHERJEE A P, XU P, TIRTHAPURA S. Mining maximal cliques from an uncertain graph[C]//2015 IEEE

31st International Conference on Data Engineering. Seoul: IEEE, 2015: 243−254.

[20] LI R H, DAI Q, WANG G, et al. Improved algorithms for maximal clique search in uncertain networks[C]// 2019 IEEE 35th International Conference on Data Engineering. Macao: IEEE, 2019: 1178−1189.

[21] RASHID A, KAMRAN M, HALIM Z. A top down approach to enumerate α-maximal cliques in uncertain graphs[J]. Journal of intelligent & fuzzy systems, 2019, 36(4): 3129−3141.

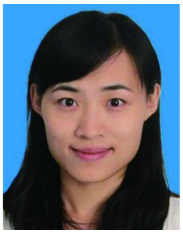
[22] 朱成名. 不确定图上极大团枚举算法研究[D]. 秦皇岛: 燕山大学, 2017: 18−42.
ZHU Chengming. Research on maximal clique enumeration algorithm on uncertain graphs[D]. Qinhuangdao: Yanshan University, 2017: 18−42.

[23] 杜明, 钟鹏, 周军锋. 一种面向不确定极大团枚举的高效验证算法[J]. 智能计算机与应用, 2020, 10(3): 14−20.
DU Ming, ZHONG Peng, ZHOU Junfeng. An efficient verification algorithm to the maximal clique enumeration [J]. Intelligent computer and applications, 2020, 10(3): 14−20.

[24] 张艺, 邹晓红. 不确定图中的极大团高效挖掘算法[J]. 燕山大学学报, 2021, 45(6): 529−536.
ZHANG Yi, ZOU Xiaohong. An efficient algorithm for mining maximal cliques in uncertain graphs[J]. Journal of Yanshan University, 2021, 45(6): 529−536.

[25] 张艺. 不确定图中极大团挖掘算法研究[D]. 秦皇岛: 燕山大学, 2022: 5−6.
ZHANG Yi. Research on maximum clique mining algorithm in uncertain graphs[D]. Qinhuangdao: Yanshan University, 2022: 5−6.

作者简介:



赵丹枫, 副教授, 博士, 主要研究方向为图计算与海洋大数据分析, 主持国家自然科学基金青年项目, 参与国家自然科学基金面上项目、省部级项目等近 10 项, 多次获上海市科学技术奖二等奖、浦东新区科学技术奖一等奖等, 发表学术论文 20 余篇。E-mail: dfzhao@shou.edu.cn。

吕闫妍, 硕士研究生, 主要研究方向为图数据挖掘。E-mail: 504096670@qq.com。

高峰, 企业导师, 博士, 主要研究方向为知识图谱、图神经网络、自然语言处理。主导及参与多项行业及团体标准, 发表数篇 CCF-A 类论文, EMNLP2022 审稿专家。E-mail: gaofeng14@midea.com。