doi:10.11959/j.issn.1000-436x.2017025

# 社交网络中基于模块度最大化的标签传播算法的研究

陈晶[1,2]，万云[1]

（1. 燕山大学信息科学与工程学院，河北 秦皇岛 066004；2. 河北省虚拟技术与系统集成重点实验室，河北 秦皇岛 066004）

**摘 要**：提出了一种利用模块度最大化与社区结构属性相结合的社区发现方法。首先，针对基于模块度最大化的标签传播算法中存在的时间复杂度高的问题，引入传播距离参数，依据“先传播，后合并”的原则，降低了社区合并导致整个网络需要更新带来的较高时间复杂度；其次，结合社区结构的概念提出了基于模块度最大化的标签传播算法（CDMM-LPA）；最后，基于网络数据集，验证并分析了 CDMM-LPA 算法的可行性。实验结果表明，CDMM-LPA 算法在降低了时间复杂度的同时，获得了较高的模块度值和更加稳定的强社区结构。

**关键词**：模块度；传播距离；社区结构；标签传播；社区发现

**中图分类号**：TP301.6 **文献标识码**：A

# Research on label propagation algorithm based on modularity maximization in the social network

CHEN Jing[1,2], WAN Yun[1]

(1. College of Information Science and Engineering, Yanshan University, Qinhuangdao 066004, China;
2. Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province, Qinhuangdao 066004, China)

**Abstract:** A kind of community detection method based on the combination of modularity and community structure attributes was proposed. Firstly, updating the whole network after communities merging every time could result in the high time complexity, therefore, introducing propagation distance parameter and “merger going after label propagation” was utilized to reduce time complexity. Secondly, CDMM-LPA algorithm was proposed by combing label propagation with community structure. Finally, empirical analysis on data networks verified the validity of the approaches. The experimental results show that the CDMM-LPA algorithm has a high modularity value and a more stable community structure while reducing the time complexity.

**Key words:** modularity, propagation distance, community structure, label propagation, community detection

## 1 引言

社交网络源于社会网络，是指社会个体之间由于互动而形成的相对稳定的关系体系，关注的是人们之间的互动和联系[1]。研究社会网络中社会个体成员之间存在的关系以及这些关系给社会网络整体结构或网络内部个体带来的影响，理解其结构和行为的过程，称为社会网络分析[2]。社区发现作为社会网络分析的一个基本任务，其目的是在社会网络中发现一个社区或识别一个节点的集合，使集合内节点之间的相互作用比它们与集合外节点的相互作用更强。目前，社区发现已广泛地应用在不同的领域中。在商业应用方面，为电子商务交易平台上的不同客户群体推荐与其购买能力相适应的商品，进而提高交易成功率；在公共安全方面，以犯罪成员在虚拟网络中的社区发现结果为基础，进而确定犯罪成员的核心成员集合；在疾病传播应用方面，根据患者的关系网络，对患者的通信记录以及患者的活动范围进行研究，找到可能的感染人群，并进行较早的控制；在生物学方面，社区发现技术

**收稿日期**：2016-05-11；**修回日期**：2016-12-23

**基金项目**：国家自然科学基金资助项目（No.61602401, No.61472340）；河北省自然科学基金资助项目(No.F2014203192)

**Foundation Items:** The National Natural Science Foundation of China (No.61602401, No.61472340), The Natural Science Foundation of Hebei Province (No.F2014203192)

通常应用于新陈代谢网络、神经网络、蛋白质网络以及食物链网络。因此，社区发现作为社会网络分析的基本任务，不仅研究意义重大，而且也为改善人们的生活及生产方式提供了新的途径。

虽然标签传播算法具有高随机性、低顽健性以及易形成“怪物”社区等缺点，但由于其时间复杂度较低，因此，仍被广泛地应用于社区发现。针对上述问题，许多研究者提出了改进方法，但是，在诸多方法中，对社区结构和社区稳定性方面的关注较少。因此，在相关研究成果的基础上，基于标签传播算法的思想，提出了本文的研究方法。在已有的标签传播算法中，如果社区选择了需要更新的标签，则将具有相同标签的社区合并，视为一个社区。但是每更新一次，合并一次，整个网络就需要更新一次，时间消耗多。因此，本文提出了“先传播，后合并”的原则；然后，引入传播距离参数，并结合社区结构、社区稳定性以及算法时间因素这3个方面，提出了基于模块度最大化和社区结构的标签传播算法（CDMM-LPA, community detection based on label propagation algorithm with modularity maximization)。

## 2 相关工作

早期的社交网络划分是利用图论的知识展开研究的，其代表算法是Kernighan-Lin算法[3]和谱二分算法[4]。自从模块度的概念被提出，Newman等[5]提出了基于模块度最大化的贪婪算法，该算法在迭代过程中，每次选择模块度增量最大的社区进行合并，但该算法易形成大社区。Clauset等[6]提出了新的贪婪算法，该算法在原模块度最大化算法基础上，改进了模块度增量的计算过程。Zhu等[7]于2002年提出了基于标签传播的社区发现算法，随后，以此算法为基础，各种不同的标签传播算法被相继提出。其中，为了避免标签传播过程中震荡的现象，Raghavan等[8]提出LPA算法；赵卓翔等[9]在Raghavan的基础上提出了一种新的标签传播算法，即在原标签传播算法的基础上，综合考虑邻居节点的标签个数、边的权重以及邻居节点的度，得到标签影响值的公式，利用标签影响值来更新标签，提高了社区发现的质量并且避免了原算法中的不稳定性；Lou等[10]提出了基于相干邻居亲近度的LPA改进算法，该算法在更新节点标签时，引入了节点之间的CNP来衡量任意节点对之间的亲近度；Gregory[11]在此基础上利用标签传播的思想研究重叠社区发现并提出了CORPA算法，该算法为每个节点分配了不同社区标识符，通过计算每个节点的归属系数选择社区标识符，直至形成稳定的社区；Wu等[12]改进了CORPA算法的阈值筛选方法，并通过平衡阈值的选择来提高社区生成的质量；朱牧等[13]提出一种基于链接密度聚类的重叠社区发现算法，该算法将网络的边作为研究对象，先根据网络中的边将网络划分成若干个不相连的链接社区，然后将链接社区转化为节点社区，并形成最终的社区；Barber[14]提出了一种基于模块度目标函数的标签传播算法（LPAm），该目标函数的最大值对应着社区划分；Liu等[15]在LPAm算法的基础上，结合多步贪婪凝聚方法（MSG）提出了将LPAm与MSG相结合的基于模块度最大化的标签传播算法LPAm+，该算法通过同时合并多个相似社区的方法，避免了陷入局部最大值，达到更加准确地检测网络社区的目的；为了使发现的社区结构更加合理，刘伟等[16]基于网络拓扑结构提出一种简单而新颖的算法来发现社区，该算法基于小世界效应、无标度以及社区结构3个属性为社区定义，并且将社区结构分为强结构社区和弱结构社区，通过强弱结构社区的定义发现网络中的社区结构；尚荣华等[17]基于网络中每个节点的局部信息，引入2个函数$f_{\text{MS-NN}}$与$f_{\text{MS-NC}}$，利用$f_{\text{MS-NN}}$函数找到局部潜在的完全子图，然后通过计算社区模块度进行社区合并，并利用函数$f_{\text{MS-NC}}$调整划分后的社区。

## 3 基于模块度最大化的标签传播算法

### 3.1 算法描述

给定一个网络用$G(V,E)$表示，其中，$V=(v_1,v_2,\cdots,v_m)$表示节点集合，$E=(e_1,e_2,\cdots,e_m)$表示边的集合。

**定义1** 节点属性值。CDMM-LPA算法在划分网络时考虑节点的属性。网络中每个节点所处的位置不同，其在网络中的重要性不同。不同节点是否属于同一个社区，不仅取决于节点本身，还要考虑节点的邻居以及这个节点与邻居节点之间的联系。因此，2个节点之间共同邻居越多，则这2个节点属于同一个社区的可能性越大。另外，如果这2个节点之间存在连接，那么这2个节点属于同一社区的可能性会增大。综合考虑上述2个因素，可以得到

$$W_{ij} = A_{ij} + \sum_{k \in N} \left( \frac{A_{ik}}{D_i - A_{ij}} \cdot \frac{A_{kj}}{D_k} \right) \tag{1}$$

通过式(1)为节点之间的连接分配权重，反映 2 个节点之间的紧密程度。如果节点 $i$ 和节点 $j$ 之间有连接，则 $A_{ij}$ 为 1，否则为 0。$D_i$ 表示节点 $i$ 的度，$D_k$ 表示节点 $k$ 的度。计算 2 个节点的 $W_{ij}$，为 2 个节点之间的连接分配权重。

**定义 2**　模块度。模块度作为社区发现的一个评价标准，定义为

$$Q = \sum e_{ii} - (a_i)^2 \tag{2}$$

其中，$e_{ii}$ 表示社区 $i$ 内部的边所占的比例，$a_i$ 表示连接到社区 $i$ 的边所占的比例。

如果将社区 $i$ 内部边数目定义为 $E_i$，网络中边的数目定义为 $m$，$E_{xt_i}$ 表示社区之间边的数目，则式(2)表示为

$$Q = \sum \frac{E_i}{m} - \frac{2E_i + E_{xt_i}}{2m} \tag{3}$$

**定义 3**　模块度变化值。假设 2 个子社区 $c_i$ 和 $c_j$，如果 $c_i$ 和 $c_j$ 合并后形成社区 $c_m$，$Q_i$、$Q_j$、$Q_m$ 分别表示它们的模块度，则社区 $c_i$ 和 $c_j$ 合并成社区 $c_m$ 后，其模块度变化值为

$$\Delta Q = Q_m - (Q_i + Q_j) \tag{4}$$

式(4)与式(2)结合得

$$\Delta Q = e_{mm} - a_m^2 - (e_{ii} - a_i^2 + e_{jj} - a_j^2) \tag{5}$$

结合式(3)可得

$$\Delta Q = \frac{E_{ij}}{m} - 2 \cdot \frac{2E_i + E_{xt_i}}{2m} \cdot \frac{2E_j + E_{xt_j}}{2m} \tag{6}$$

其中，$E_{ij}$ 表示社区 $i$ 与社区 $j$ 之间的连接数。

**定义 4**　社区结构。Radicchi 等[18]定义了强社区结构和弱社区结构，描述如下。

1) 强社区结构。设 $A$ 是 $G$ 网络的子社区，强社区是指 $A$ 中任意一个节点与 $A$ 中节点连接的边数大于与 $A$ 以外的节点连接的边数。

2) 弱社区结构。$A$ 中所有节点与 $A$ 内部节点的边数之和大于 $A$ 中所有节点与 $A$ 所有外部节点的边数之和。

**定义 5**　传播距离参数。传播距离参数是指在标签更新过程中，以 $\Delta Q$ 最大值为基础进行小社区与邻居社区节点间标签传播的最大次数。

若传播距离参数定义为 3，则表示以 $\Delta Q$ 为起点，社区间通过迭代计算和获取邻居社区 $\Delta Q$ 最大值并进行标签传播的次数为 3。

本文基于 Radicchi 等社区结构的描述，给出了以下 2 种形式的社区结构定义。

1) 强社区结构

对于任意社区，如果社区内部边的数目多于与该社区相连的边的数目，则将这种社区结构称为强社区结构。强结构社区中的节点都明确归属于这个社区。

2) 弱社区结构

对于任意社区，如果该社区为弱结构社区，则存在以下 2 种情况。

① 如果社区内部边的数目大于该社区与其他各个社区之间边的数目，则此时的社区称为暂时稳定的社区结构，属于弱结构社区，趋向于向强社区结构转化。

② 如果社区内部边的数目小于该社区与其他各个社区之间边的数目，则该社区称为不稳定的社区结构，属于弱结构社区，必须与强结构的邻居社区合并。

由于在强结构社区中，节点明确地归属于该社区，便于对社区进行研究。而在弱结构社区中，节点在下一刻可能不属于该社区，并且外部连接比较多的社区，通常趋向于与其他社区进行合并。从现实研究意义角度出发，稳定的社区结构更适合于进行研究和数据分析。此外，由于原始的 LPA 算法不考虑标签传播距离，为解决标签传播距离过远而形成的大社区现象，本文将社区结构和标签传播距离参数相结合，为了解决弱社区在多次标签传播的过程中，强弱社区标签传播与合并后可能造成的强社区结构模块度值下降的问题，并进一步降低算法的执行时间。

### 3.2　算法实现过程

标签传播算法主要分为标签分配和传播 2 个阶段。CDMM-LPA 算法中并未为每个节点分配标签，而是以小社区为单位，为每个小社区分配唯一的标签。采用模块度最大化的方法进行标签传播，由于模块度最大化是一个单向操作，因此能获得高质量的社区。CDMM-LPA 算法中充分考虑了社区结构的属性，以确保得到稳定的社区。算法实现步骤描述如下。

1) 网络划分。利用式(1)计算节点之间的属性

值，并将其作为节点之间边的权重，对网络进行划分。

2) 生成小社区并分配标签。依次取出每一条边，如果取出边的2个节点没有被分配社区，则为2个节点分配相同的社区号。如果至少有一个节点已被分配，则继续取出下一条边。当所有边被访问完毕，将剩余未分配社区的节点划分到与其相似度最大的社区中。

3) 设定节点传播距离参数并更新标签。在标签传播距离的设定范围内，寻找每个小社区的邻居社区集合，利用式(5)计算小社区与其邻居社区之间的模块度值，寻找 $\Delta Q$ 值最大的邻居社区，将自身标签更新为使 $\Delta Q$ 值最大的邻居社区的标签。

4) 社区结构检查。在上述过程中，社区之间只进行传播标签，不进行合并。当社区标签不再传播，则将具有相同标签的社区进行合并。合并完毕后，检查合并后的社区结构是否稳定，若存在弱社区结构，则继续传播；不存在弱社区结构，则算法结束。

5) 获得社区划分结果。当标签传播已趋于稳定，但仍存在弱结构社区，则分析该弱结构社区，将其与强结构邻居社区合并，且合并后，能够使他们自身的模块度值有增加的趋势。

### 3.3 算法伪代码

为了更好地描述本文算法，给出主要算法的伪代码，算法1描述的是标签传播过程，算法2描述的是社区结构检查过程。本文所提到的不稳定和暂时稳定社区的思想在算法中体现如下。暂时稳定的社区一般在循环多次后，若没有要合并的趋势，强制其与强邻居结构合并，即算法2；而不稳定社区通常不用强制要求与其他社区合并。所以，本文提到的强制合并的弱结构社区，都是指暂时稳定的弱结构社区。

**算法1** 标签传播算法

**输入**：节点集合 $V$，节点的传播距离 $S$

**输出**：合并后的社区

1) 依据式(3)~式(6)形成小社区 subCom；/*subCom 表示通过计算后形成的小社区*/

2) 为每个小社区 subCom 分配唯一的标签；

3) for 每个小社区 subCom do

4) 找到每个小社区 subCom 的邻居社区；

5) 依据式(3)~式(5)计算每个小社区 subCom 和其邻居社区的 $\Delta Q$ 值；

6) while (节点的传播距离 $\leqslant S$)

7) 节点传播距离 = 0;

8) if(得到了最大的 $\Delta Q$ 值) /*通过步骤5)计算并获取 $\Delta Q$ 值最大的邻居社区，并更新标签*/

9) 将其自身标签更新为邻居社区标签;

10) end if

11) 节点传播距离 = 节点传播距离 + 1;

12) end while

13)end for

14)将具有相同标签的社区进行合并；

15)得到合并后的社区;

**算法2** 社区结构检查算法

**输入**：算法1合并后的社区

**输出**：强结构社区

1) if (合并后的社区仍然存在弱结构社区) /*弱社区的2种情况，处理可得到强社区结构 */

2) if ($\Delta Q$ 值 > 0)

3) 调用算法1继续执行标签传播过程和社区合并; /* 处理暂时稳定的社区情况*/

4) else

5) 找到弱结构社区的邻居社区; /* 处理2类弱社区*/

6) 检查弱结构社区的邻居社区;

7) if (弱结构社区的邻居社区中存在多个强结构社区) /*合并后，能够使自身的模块度值有增加的趋势 */

8) 随机选择弱结构社区中的任意强结构邻居社区;

9) 将随机选择的强结构邻居社区与弱结构社区进行合并;

10) end if

11) end if

12) end if

13)得到处理后的强结构社区;

算法1中，步骤5)结束后，找到 $\Delta Q$ 值最大的邻居社区，将社区标签更新为该邻居社区的标签。步骤8)的条件表示 subCom 选择其邻居集合中模块度变化值最大的那个邻居的标签。2种算法中主要包括了寻找社区的邻居社区集合、标签传播过程以及社区结构检查过程。从算法中可以看出，当标签传播结束后进行社区之间的合并。

在 CDMM-LPA 算法实现过程中，当标签传播结束后，具有相同标签的 subCom 合并形成一个大

社区。一些小网络以及社区结构明显的网络通常在这个阶段就已经完成社区划分，不再进行标签传播。通过实验发现在这个过程中会存在这样的情况：一些 subCom 不执行模块度最大化过程，因为这些小社区执行模块度最大化时，得到的 $\Delta Q$ 值都小于 0，即与邻居社区合并后会使合并后的社区模块度值比之前不合并时模块度值小。因此，对存在的弱结构社区 subCom 与其连接紧密的强结构的邻居社区进行合并，这样得到的社区结构更加稳定，合并后社区内部结构更加紧密。此外，通过引入标签传播距离参数，也可以减缓强弱结构社区在标签更新与合并过程中，由于弱结构社区的频繁出现所带来的合并后的社区模块度值下降的趋势。因此，算法 1 和算法 2 的目的是确保所发现的社区具有稳定结构。

由上述执行过程可知，若包含 $n$ 个节点，$m$ 条边的网络 $G$，则 CDMM-LPA 算法的运行时间包括：权重计算及为生成的 $k$ 个小社区分配节点标签的时间复杂度 $O(n\bar{d}+w)$，其中，$n\bar{d}$ 为计算 $n$ 个节点的时间复杂度，$w$ 是 $k$ 个小社区内的节点总数；$k$ 个小社区中模块度增量 $\Delta Q$ 值的计算时间为 $kM_c$，更新标签的计算时间为 $kU_l$，则时间复杂度为 $O(kM_c+kU_l)$；由于算法是在标签传播完毕后进行社区合并和社区结构检查，其中，生成的 $k$ 个小社区中需要合并的社区数是 $i$ 个，时间复杂度为 $iC_m$。当模块度的变化值 $\Delta Q<0$,，而弱社区数量为 $j$ 个时，与邻居强社区合并的时间为 $jS_c$，则时间复杂度为 $O(iC_m+jS_c)$。因此，CDMM-LPA 算法的时间复杂度为 $O(n\bar{d}+w+k(M_c+U_l)+iC_m+jS_c)$。由于 CDMM- LPA 算法通常在几轮标签传播结束后，社区结构基本形成且 $j<i<k<n$，$w<n$。因此，$O(k(M_c+U_l)+mC_m+jS_c)$对于模块增量的计算、节点标签的更新和社区合并时间在几轮之后趋于固定值 $M$；则 CDMM-LPA 算法的时间复杂度可表示为 $O(Kn+M)$，保持与 LPA 算法的近似线性时间复杂度。

### 3.4　实例说明

通过 2 个实例分析 CDMM-LPA 算法，例 1 社区结构明显，通过在例 1 上运行 CDMM-LPA 算法，可以看到算法 1 分析中提到的情况。例 2 是美国跆拳道俱乐部 karate 网络数据集。例 1、例 2 分别如图 1 和图 2 所示。

1) 当CDMM-LPA算法步骤1)和步骤2)执行完毕后，图 1 中形成的小社区为{{1, 2, 3, 4},{5, 6, 7},{8, 10},{9, 11, 12}}，karate 网络中形成的小社区为{{1, 2, 22, 18}, {5,11},{6, 7, 17}, {9, 31}, {10}, {12},{15, 16, 19, 21, 23, 33, 34},{25, 26, 29, 32}, {28, 24},{27, 30},{3, 4, 14, 13, 8}}。

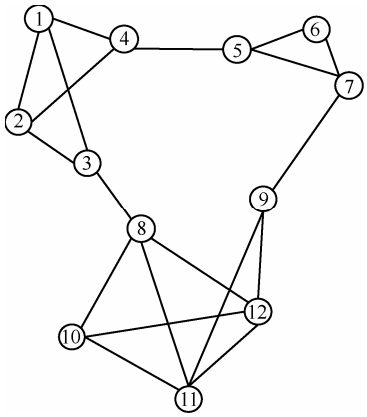


图 1　结构明显的社区实例 1

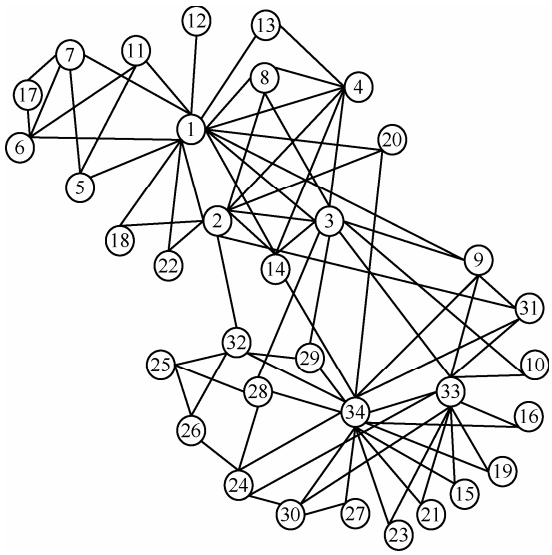


图 2　karate 网络数据集

2) 执行步骤 3)和步骤 4)之后，图 1 中的小社区以及 karate 网络所获得的小社区进行标签传播后所形成的社区如图 3 和图 4 所示。

从图 3 中可以看出最终社区基本形成，即{1,2,3,4}、{5,6,7}和{8,9,10,11,12}这 3 个社区；图 4 中 karate 网络形成了 4 个社区。

3) 通过分析可知，图 3 中的 3 个社区均为强结构社区。而在图 4 中，社区{24,25,26,28,29,32}中内部边数目为 7 条，外部边数目为 10 条，根据弱结构社区定义，该社区属于弱结构社区，需要继续执行标签传播算法。

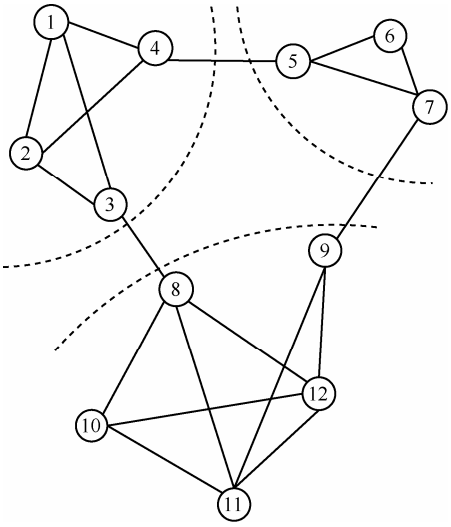


图 3　结构明显的社区实例 1 划分结果

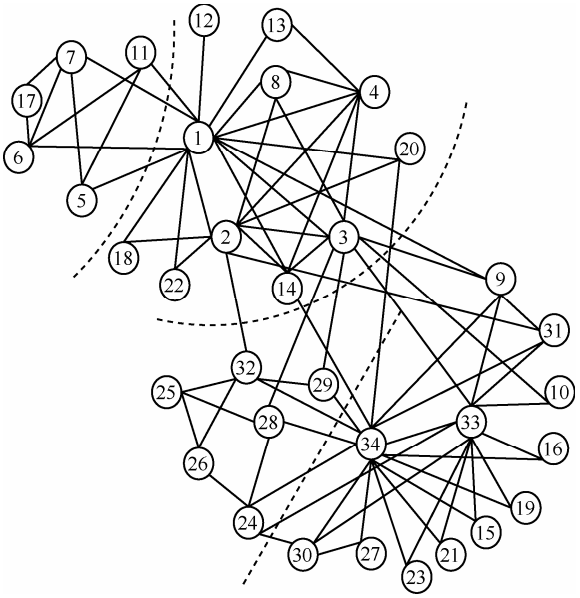


图 4　karate 网络划分结果

4) 在新一轮的标签传播过程中，社区{24,25,26,28,29,32}出现了算法 2 中所提到的情况，其原因在于该社区在模块度最大化的过程中，$\Delta Q$ 一直小于 0，不能与其他社区进行合并。但社区{24, 25, 26, 28, 29, 32}和社区{9,15,16,27,30,33,34,19,21, 23, 31}合并后能够使其自身的模块度有增大的趋势，而弱结构社区{24,25,26,28,29,32}趋向于寻找一个强结构社区的邻居并与之合并，使合并后的社区内部结构更加紧密。

因此，karate 网络的最终结果是形成了{5,6,7, 17, 11}、{9, 31, 15, 16, 21, 23, 19, 27, 33, 34,30, 29, 24, 26, 32, 25, 28}和{1, 2, 18, 3, 4, 22, 10, 8, 13, 14, 20, 12}这 3 个社区，并且在最终形成的社区中，没有弱结构社区的存在。

## 4　实验及结果分析

### 4.1　实验数据集

本部分实验选择了 5 个真实网络数据集和一个人工合成网络数据集，并设定传播距离参数值为 5。其中，真实网络数据集分别是 dolphins social network、American football、Email、Condmat2003 和 Amazon 数据集，数据集信息如表 1 所示；采用 LFR 基准网络生成人工合成数据集。由于 LFR 是社区发现算法中最常用的模拟数据集，通过设置不同的参数生成不同类型的网络。$N$ 表示节点度分布参数；$k$ 表示网络中节点的平均度；max$k$ 表示节点最大度；min$c$ 表示最小社区包含的节点数量；max$c$ 表示最大社区包含的节点数量；$mu$ 是混合参数，表示节点与社区外部节点连接的概率，通常情况下，$mu$ 越大，发现社区的难度越大。其中，前 2 组网络通过设置不同的混合参数，测试在不同网络上生成的社区质量，后 2 组网络通过改变节点个数，测试在不同大小的网络上的运行时间。人工合成网络数据集参数如表 2 所示。

**表 1　　数据集**

| 名称 | 节点数 | 边数 |
|---|---|---|
| dolphins social network | 62 | 159 |
| American football | 115 | 613 |
| Email | 1 133 | 5 451 |
| Condmat2003 | 31 163 | 120 029 |
| Amazon | 334 863 | 925 872 |

**表 2　　LFR 基准网络数据信息**

| 名称 | $mu$ | $N$ | $k$ | max$k$ | max$c$ | min$c$ |
|---|---|---|---|---|---|---|
| $nc_1$ | 0.2 | 5 000 | 15 | 50 | 50 | 20 |
| $nc_2$ | 0.4 | 5 000 | 15 | 50 | 50 | 20 |
| $nc_3$ | 0.2 | 4 000 | 20 | 100 | 50 | 20 |
| $nc_4$ | 0.2 | 5 000 | 20 | 100 | 50 | 20 |

1) dolphins social network 数据集实验

在 dolphins 网络数据集上执行标签传播过程得到的中间结果集合为{{47, 50}, {1,37,41},{54, 62 }{7, 8,10, 14,58, 20, 55, 49,42, 6, 40, 33, 57, 61 }, {2, 23, 27, 28, 18, 26, 32 }, {3, 4, 9, 11, 60, 21, 31, 48, 43, 29 },{13, 15,17, 34, 35, 38, 39,44, 45, 51, 53, 59 },{5, 19,22, 46, 12, 36, 16, 30, 52, 24, 25, 56 }}。在

执行过程的中间结果中，存在弱结构社区。因此，需要继续执行标签传播过程并检查社区结构，直至不存在弱结构社区。因此，dolphins 网络最终得到的社区生成结果如图 5 所示。

从图 5 中可知，虽然 CDMM-LPA 算法发现的最终社区与真实划分的社区存在出入，但 CDMM-LPA 算法取得的结果都是强结构社区，因此，从社区结构角度而言，CDMM-LPA 算法得到的划分结果是合理的。

2) American football 数据集网络实验

American football 数据集的真实划分结果如表 3 所示，CDMM-LPA 算法生成的结果如表 4 所示。

**表 3　American football 的真实社区**

| 社区集 | 节点集 |
|---|---|
| 1 | 2, 38, 46, 90, 104, 106, 26, 34, 110 |
| 2 | 3, 7, 48, 14, 16,33, 65, 101,40, 61, 107 |
| 3 | 4, 6, 11, 53,103,41, 75, 85, 99, 82, 73, 108 |
| 4 | 1, 5, 17, 10, 94, 42, 24, 105 |
| 5 | 8, 9, 23, 52, 79,78, 22,109, 112,69 |
| 6 | 12, 51, 60, 25, 70, 98, 64 |
| 7 | 13, 32,19, 27, 15, 35, 44, 62, 39, 55, 86, 72, 100 |
| 8 | 20,30, 56, 80, 31, 95, 36, 102 |
| 9 | 18, 96, 57, 28,63, 88,66, 21, 71, 97, 77, 114 |
| 10 | 29, 50, 47,59, 68, 89, 74, 54,84, 115 |
| 11 | 43, 37, 83, 81, 91 |
| 12 | 45, 58, 49, 93, 67, 87, 92, 76, 111, 113 |

**表 4　本文算法的社区划分结果**

| 社区集 | 节点集 |
|---|---|
| 1 | 2, 26,46, 110, 34, 90, 38,106, 104 |
| 2 | 4, 41, 11, 108, 75, 6, 85,53, 73, 82, 103,99 |
| 3 | 30, 20,31, 56, 80, 36,81, 95, 102, 83 |
| 4 | 54, 68,84, 74,89, 50, 47,115, 111 |
| 5 | 8, 112, 23,22, 79, 52,78,9, 109,69 |
| 6 | 7, 76, 45, 49, 64, 59,98, 58, 60, 87, 93, 113, 92 |
| 7 | 18, 88, 96,71, 21, 97,77, 57, 63, 66, 28,114 |
| 8 | 55, 72, 39, 19, 86, 43, 44, 35, 15, 32,100, 62, 13, 27, 37 |
| 9 | 3, 7, 14, 16, 65, 61, 107, 40, 33, 101, 48 |

将表 3 与表 4 的结果进行对比分析可知，最终结果存在差异的主要原因在于本文算法的初始阶段形成的小社区中，有几个节点划分不同导致了最终结果的不同。CDMM-LPA 算法初始阶段形成的{74, 111}、{59, 60}和{64, 98}这 3 个小社区中，节点与真实社区节点的划分结果不一致。然而，这 3 个小社区节点之间彼此的相似度最大，应归为一个社区使其形成强结构社区，因此，基于社区结构属性考虑，本文的划分结果是符合实际的。

### 4.2　不同算法之间的比较

CDMM-LPA 算法主要是基于标签传播的思想，将本文算法与原始的标签传播算法 LPA 以及在模块度最大化方面的经典标签传播算法 LPAm+进行比较。基于 LPA、LPAm+算法的主要思想，本文还原了 2 个算法的实验，而后将 CDMM-LPA 算法与

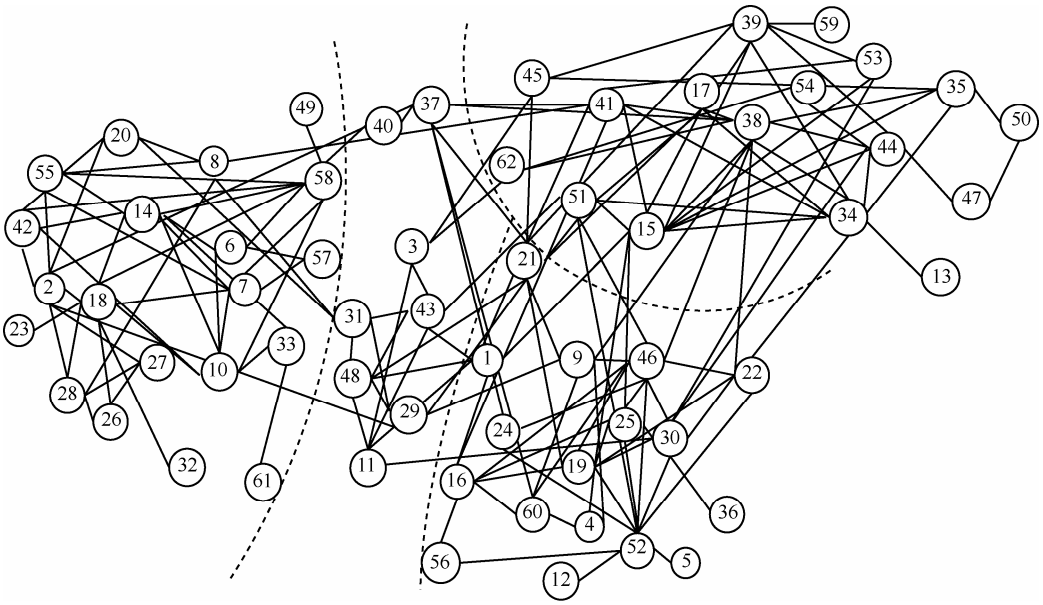


图 5　CDMM-LPA 在 dolphins social network 上的划分结果

LPA 算法、LPAm+算法分别从模块度、强社区数量的比例和运行时间 3 个方面进行比较。

1) 模块度的比较

分别将 3 种算法在所取的数据集上进行 20 次实验，观察 20 次实验中 3 种算法的比较结果。分别计算 LPA、LPAm+以及 CDMM-LPA 算法在 20 次实验中得到的模块度平均值，比较结果如表 5 所示。

**表 5 不同算法的模块度值的比较**

| 数据集 | LPA | LPAm+ | CDMM-LPA |
|---|---|---|---|
| dolphins social network | 0.504 8 | 0.481 1 | 0.512 3 |
| American football | 0.528 2 | 0.556 3 | 0.604 4 |
| Email | 0.539 1 | 0.531 3 | 0.488 5 |
| Condmat2003 | 0.604 2 | 0.667 2 | 0.736 8 |
| Amazon | 0.670 4 | 0.746 5 | 0.771 4 |
| $nc_1$ | 0.592 8 | 0.648 5 | 0.689 8 |
| $nc_2$ | 0.517 6 | 0.579 1 | 0.645 6 |

从表 5 中可以看到 CDMM-LPA 算法得到的社区的模块度值比 LPAm+和 LPA 算法获得的社区的模块度值大。由 LPAm+算法可知，该算法的整个过程是从一个较小的局部最优值跳出，达到一个较高的局部最优值，直到这个值不再增加为止。对于某些数据集而言，CDMM-LPA 算法中由于社区结构的检查，当弱结构社区过多的数据集与强结构社区进行合并，在传播距离参数的设定范围下，虽避开了局部最优值，但是弱结构社区与强结构社区合并之后，整体的社区模块度值减小，如 Email 数据集。与未设定传播距离参数相比，Email 数据集的模块度值有所增加。因此，传播距离参数，在一定程度上降低了强弱结构社区传播与合并过程中模块度变化值小于零的趋势。

2) 强社区数目的比较

为了验证算法获取稳定社区的性能，本文在随机抽取标签传播过程中，对应模块度发生变化及稳定时刻的 15 个社区划分结果，并判断获取的强社区数目，经过多次随机抽取计算强社区的比例，实验结果如表 6 所示。

**表 6 强社区数目比例**

| 数据集 | LPA | LPAm+ | CDMM-LPA |
|---|---|---|---|
| Email | 82.1% | 87.6% | 97.1% |
| Condmat2003 | 69.7% | 82.4% | 91.2% |
| $nc_3$ | 78.3% | 87.5% | 94.8% |

从表 6 中可知，LPA 算法在进行标签传播的过程中只需要依据相邻节点标签的情况更新自身节点的标签值，使模块度值很难趋向于稳定的状态，并得到了不稳定的社区划分结果；LPAm+算法在经过多次迭代后使模块度值趋于稳定，提高了社区的稳定性，但是没有针对划分后形成的弱社区进行检查，由此会导致弱社区数目过多时，社区不稳定现象的产生；而 CDMM-LPA 算法在得到较稳定的模块度值以后，有针对社区结构中的暂时稳定社区和不稳定社区进行了不同的处理方法，其结果使强社区结构的数目比例有较大程度的提高。

3) 运行时间的比较

将 3 种算法分别运行 20 次，计算平均运行时间，得到运行时间的比较结果，如表 7 所示。

**表 7 运行时间的比较**

| 数据集 | LPA/s | LPAm+/s | CDMM-LPA/s |
|---|---|---|---|
| dolphins social network | 0.047 | 0.060 | 0.046 |
| American football | 0.047 | 0.125 | 0.109 |
| Email | 0.906 | 46.953 | 13.142 |
| Condmat2003 | 48.583 | 2 157.413 | 247.257 |
| Amazon | 402.180 | 4 875.257 | 1 152.184 |
| $nc_3$ | 5.752 | 162.175 | 36.138 |
| $nc_4$ | 8.228 | 195.731 | 50.815 |

从表 7 中可以看出，由于 LPA 算法只传播标签，不进行社区合并，最后进行标签分类即可，因此运行时间最短。CDMM-LPA 算法与 LPA 算法相比运行时间较长，CDMM-LPA 算法的运行时间主要消耗在 2 个方面：1) 模块度最大化过程，需要计算社区之间的模块度变化值；2) 社区结构的检查过程。但是，由于 CDMM-LPA 算法是在标签传播距离范围内进行标签传播，而后再进行社区之间的合并，并且从 CDMM-LPA 算法的实现步骤可以看出，社区结构明显的网络通常在几轮标签传播结束后，最终社区就基本形成，因此，CDMM-LPA 算法在这类网络上的运行时间会明显减少。而 LPAm+算法运行时间主要消耗在模块度最大化过程和网络更新过程这 2 个方面。LPAm+算法也需要计算社区之间的模块度变化值，并且 LPAm+算法的社区标签更新后，社区之间要进行合并，整个网络都会发生变化，也需要进行更新，这个过程消耗时间较长。因此，LPAm+算法的运行时间比 CDMM-LPA 算法的运行

时间长，而且 LPAm+算法的运行时间不会因为网络的类型而明显减少。

综上所述，本文所提出的算法在不同类型的数据集测试中，对于运行时间、强社区数目和模块度 3 个方面的指标验证都达到了较好的实验效果。

## 5　结束语

本文提出了基于模块度最大化的标签传播算法，首先，给出了模块度和强弱社区的定义，并将弱结构社区分成了暂时稳定结构和不稳定结构 2 种情况。然后，通过实例说明了本文提出的 CDMM-LPA 算法的思想及实现过程。由于 CDMM-LPA 算法主要利用了标签传播算法的思想，并传播距离参数，本文实现 CDMM-LPA 算法的目的如下：1) 降低基于模块度最大化的这一类标签传播算法的时间复杂度；2) 保证生成的社区是强结构社区；3) 对本文提出的 CDMM-LPA 算法与 LPA 算法以及 LPAm+算法在模块度、强社区数目和运行时间方面进行实验对比验证。实验表明 CDMM-LPA 算法生成的社区模块度值比较高、运行时间也较 LPAm+算法有优势，并且可以获得稳定的社区划分结果。由于社交网络是一个动态变化的网络，现有的社区发现算法大多数是针对静态社交网络展开研究的，因此，如何有效地在大规模的动态社交网络环境下利用标签传播的思想提高重叠社区发现的效率及稳定性，以及社区演化过程对社区发现结果的影响是今后的研究工作。

## 参考文献：


[1] 刘耀庭. 社交网络结构研究[D]. 杭州: 浙江大学, 2008.
LIU Y T. Research on social network structure[D]. Hangzhou: Zhejiang University, 2008.

[2] SCOTT J. Social network analysis[M]. London: Sage Publication, 2013.

[3] KERNIGHAN B W, LIN S. An efficient heuristic procedure for partitioning graphs[J]. Bell System Technical Journal, 1970, 49(2): 291-307.

[4] POTHEN A, SIMON H D, LIOU K P. Partitioning sparse matrices with eigenvectors of graphs[J]. SIAM Journal on Matrix Analysis and Applications, 1990, 11(3): 430-452.

[5] NEWMAN M E J. Analysis of weighted networks[J]. Physical Review E, 2004, 70(5): 056131.

[6] CLAUSET A, NEWMAN M E J, MOORE C. Finding community structure in very large networks[J]. Physical Review E, 2004, 70(6): 066111.

[7] ZHU X, GHAHRAMANI Z. Learning from labeled and unlabeled data with label propagation[R]. Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002.

[8] RAGHAVAN U N, ALBERT R, KUMARA S. Near linear time algorithm to detect community structures in large-scale networks[J]. Physical Review E, 2007, 76(3): 036106.

[9] 赵卓翔, 王轶彤, 田家堂. 社会网络中基于标签传播的社区发现新算法[J]. 计算机研究与发展, 2011, 48(S3): 8-15.
ZHAO Z X, WANG Y T, TIAN J T. A novel algorithm for community discovery in social networks based on label propagation[J]. Journal of Computer Research and Development, 2011, 48(S3): 8-15.

[10] LOU H, LI S H, ZHAO Y X. Detecting community structure using label propagation with weighted coherent neighborhood propinquity[J]. Physica A, 2013, 392: 3095-3105.

[11] GREGORY S. An algorithm to find overlapping community structure in networks[C]//Knowledge discovery in databases: PKDD 2007. 2007: 91-102.

[12] WU Z H, LIN Y F, GREGORY S. Balanced multi-label propagation for overlapping community detection in social networks[J]. Journal of Computer Science and Technology, 2012, 27(3): 468-479.

[13] 朱牧, 孟凡荣, 周勇. 基于链接密度聚类的重叠社区发现算法[J]. 计算机研究与发展, 2013, 50(12): 2520-2530
ZHU M, MENG F R, ZHOU Y. Density-based link clustering algorithm for overlapping community detection[J]. Journal of Computer Research and Development, 2013, 50(12): 2520-2530.

[14] BARBER M J, CLARK J W. Detecting network communities by propagating labels under constraints[J]. Physical Review E, 2009, 80(2): 026129.

[15] LIU X, MURATA T. Advanced modularity-specialized label propagation algorithm for detecting communities in networks[J]. Physica A: Statistical Mechanics and its Applications, 2010, 389(7): 1493-1500.

[16] LIU W, PELLEGRINI M, WANG X. Detecting communities based on network topology[J]. Scientific Reports, 2014, 4(4):5739-5739.

[17] SHANG R H, LUO S, LI Y. Large-scale community detection based on node membership grade and sub-communities integration[J]. Physica A: Statistical Mechanics and its Applications, 2015, 428: 279-294.

[18] RADICCHI F, CASTELLANO C, CECCONI F. Defining and identifying communities in networks[J]. The National Academy of Sciences of the United States of America, 2004, 101(9): 2658-2663.



作者简介：

**陈晶**（1976-），女，黑龙江哈尔滨人，博士，燕山大学副教授，主要研究方向为社交网络、对等网络和 Web 服务等。

**万云**（1990-），女，山西晋城人，燕山大学硕士生，主要研究方向为社交网络和数据挖据。