

基于相对距离的复杂网络谱粗粒化方法

杨青林 王立夫 李欢 余牧舟

A spectral coarse graining algorithm based on relative distance

Yang Qing-Lin Wang Li-Fu Li Huan Yu Mu-Zhou

引用信息 Citation: *Acta Physica Sinica*, 68, 100501 (2019) DOI: 10.7498/aps.68.20181848

在线阅读 View online: <https://doi.org/10.7498/aps.68.20181848>

当期内容 View table of contents: <http://wulixb.iphy.ac.cn>

您可能感兴趣的其他文章

Articles you may be interested in

复杂网络谱粗粒化方法的改进算法

Improved algorithm of spectral coarse graining method of complex network

物理学报. 2017, 66(6): 060502 <https://doi.org/10.7498/aps.66.060502>

DNA超分子水凝胶的粗粒化建模与模拟

Modelling and simulation of DNA hydrogel with a coarse-grained model

物理学报. 2017, 66(15): 150201 <https://doi.org/10.7498/aps.66.150201>

非均匀外力对粗粒化DNA穿孔行为影响的模拟研究

Simulation of translocating pore of DNA in non-uniform force by coarse-grained model

物理学报. 2016, 65(14): 148701 <https://doi.org/10.7498/aps.65.148701>

硅功能化石墨烯负极材料的粗粒模型

Coarse-grain model of silicon functionalized graphene as anode material for lithium ion batteries

物理学报. 2015, 64(14): 143101 <https://doi.org/10.7498/aps.64.143101>

光学频率梳基于光谱干涉实现绝对距离测量

Spectral interferometry based absolute distance measurement using frequency comb

物理学报. 2015, 64(2): 020601 <https://doi.org/10.7498/aps.64.020601>

基于相对距离的复杂网络谱粗粒化方法*

杨青林 王立夫† 李欢 余牧舟

(东北大学秦皇岛分校 控制工程学院, 秦皇岛 066004)

(2018年10月15日收到; 2019年3月14日收到修改稿)

复杂网络的同步作为一种重要的网络动态特性, 在通信、控制、生物等领域起着重要的作用. 谱粗粒化方法是一种在保持原始网络的同步能力尽量不变情况下将大规模网络约简为小规模网络的算法. 此方法在对约简节点分类时是以每个节点对应特征向量分量间的绝对距离作为判断标准, 在实际运算中计算量大, 可执行性较差. 本文提出了一种以特征向量分量间相对距离作为分类标准的谱粗粒化改进算法, 能够使节点的合并更加合理, 从而更好地保持原始网络的同步能力. 通过经典的三种网络模型 (BA 无标度网络、ER 随机网络、NW 小世界网络) 和 27 种不同类型实际网络的数值仿真分析表明, 本文提出的算法对比原来的算法能够明显改善网络的粗粒化效果, 并发现互联网、生物、社交、合作等具有明显聚类结构的网络在采用谱粗粒化算法约简后保持同步的能力要优于电力、化学等模糊聚类结构的网络.

关键词: 复杂网络, 同步能力, 谱粗粒化, 相对距离

PACS: 05.45.Xt, 05.10.-a, 05.45.-a

DOI: 10.7498/aps.68.20181848

1 引言

从 20 世纪末开始, 复杂网络理论逐步成为社会、经济、化学和信息系统等领域的重要研究方法, 对复杂网络静态特性和动态特性的定量定性描述, 已经成为网络时代一个极其重要的挑战性课题, 被称为“网络的新科学 (new science of network)”^[1]. 复杂网络的同步作为一种重要的网络动态行为, 在激光系统、超导材料和通信等领域有重要的意义. 同步现象在自然界中也广泛存在, 例如萤火虫有规律地闪光、心脏细胞同步震荡、挂在同一横梁上的钟摆同步摆动等. 近 20 多年来, 研究人员对复杂网络的同步研究取得了丰硕的成果^[2-15]. 其中, 对于一般连续时间耦合网络最常用的方法是将网络的状态方程线性化后转化成主稳定函数来研究^[2,3]. 然而对于实际网络动辄成千上万的节点来说, 判断

他们是否同步需要求解大量耦合微分方程, 并且小规模网络得到的同步判据或定理不能直接推广到大规模网络^[16]. 因此, 如何将复杂网络进行约简并保留其同步能力不变, 对大规模网络的同步分析和仿真至关重要. Gfeller 和 Rios^[17,18] 提出的谱粗粒化 (spectral coarse graining, SCG) 算法通过分析网络的结构矩阵, 把特征向量分量相同或相近的节点合并, 从而保证了约简后网络的同步能力不变. 周建等^[19] 提出的改进谱粗粒化 (improved spectral coarse graining, ISCG) 算法采用了不同的分类方式, 改善了在特征向量分量分布极不均匀时的粗粒化效果, 并且降低了算法的复杂度. Chen 等^[20] 进一步研究了网络的聚类结构对谱粗粒化效果的影响. Wang 和 Xu^[21] 发现采用 SCG 算法对网络约简后, 其可控性也能很好地保留下来.

SCG 算法和 ISCG 算法都给出了一种简单可实现的谱粗粒化方法, 但是在确定网络中哪些节点

* 河北省自然科学基金 (批准号: F2016501023, F2017501041)、中央高校基本科研业务费 (批准号: N172304030) 和国家自然科学基金 (批准号: 61402088) 资助的课题.

† 通信作者. E-mail: wlfk@qq.com

需要合并时都是以特征向量分量间的绝对距离为判断依据, 没有考虑到特征向量分量间的相对距离, 即绝对距离占分量绝对值的比例. 本文经过理论推导证明了两个节点合并前后特征值的变化量与分量间的相对距离成正比例关系, SCG 和 ISCG 算法在分类时都忽略了分量间相对距离的影响, 从而导致节点的分类不准确. 为了克服这一缺陷, 本文提出了一种基于相对距离的谱粗粒化 (improved spectral coarse graining based on relative distance, ISCGR) 算法, 在对节点分类时以特征向量分量间的相对距离作为判断标准, 改善了网络的粗粒化效果, 提高了网络约简后保持同步的能力. 采用本文所提的算法对 27 种实际网络进行粗粒化仿真, 发现互联网、生物、社交、合作等具有明显聚类结构的网络在约简后比电力、化学等网络更容易保持同步.

2 同步能力的刻画

主稳定方程法是研究一般连续时间耦合网络同步的常用方法^[2,3]. 考虑由 N 个节点构成的复杂网络, 其中第 i 个节点的动力学方程为

$$\dot{x}_i = \mathbf{f}(x_i) - \mu \sum_{j=1}^N l_{ij} \mathbf{H}(x_j) \quad (i = 1, 2, \dots, N), \quad (1)$$

式中, $x_i \in \mathbf{R}^m$ 为节点 i 的 m 维状态变量; $\mathbf{f}(x_i)$ 为第 i 个节点自身的状态方程; 常数 $\mu > 0$ 是网络的耦合强度; $\mathbf{H}(\bullet) : \mathbf{R}^m \rightarrow \mathbf{R}^m$ 表示各个节点之间的内部耦合函数; Laplacian 矩阵 $\mathbf{L} = (l_{ij}) \in \mathbf{R}^{N \times N}$ 是外部耦合矩阵, 表示网络的拓扑结构, 满足耗散条件 $\sum_{j=1}^n l_{ij} = 0$. 特别的, 当 Laplacian 矩阵

\mathbf{L} 表示一个无向无权的网络时, 有如下定义:

$$l_{ij} = \begin{cases} -1, & \text{当节点 } i \text{ 与节点 } j \text{ 有连接时} \\ 0, & \text{当节点 } i \text{ 与节点 } j \text{ 无连接时} \end{cases} \quad (i \neq j), \quad (2)$$

对角元为

$$l_{ii} = - \sum_{j=1, j \neq i}^N l_{ij} = - \sum_{j=1, j \neq i}^N l_{ji} = k_i \quad (i = 1, 2, \dots, N), \quad (3)$$

其中 k_i 为节点 i 的度数. 此时, 外部耦合矩阵 \mathbf{L} 是一个不可约的对称阵, 具有非负特征根并且特征根满足 $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N$. 如果在 $t \rightarrow \infty$ 时有

$x_1(t) = x_2(t) = \dots = x_N(t) = s(t)$, 那么就称动态网络 (1) 式能够达到完全同步, 其中同步流形 $s(t)$ 满足 $\dot{s}(t) = \mathbf{f}(s(t))$. 对 (1) 式在同步状态 $s(t)$ 处线性化, 令 $\xi_i(t) = x_i(t) - s(t)$, 可以得到如下线性动力学方程:

$$\dot{\xi}_i = \mathbf{Df}(s) \xi_i - \mu \sum_{j=1}^N l_{ij} \mathbf{DH}(s) \xi_j, \quad (4)$$

其中 $\mathbf{Df}(s)$ 和 $\mathbf{DH}(s)$ 分别是 $\mathbf{f}(s)$ 和 $\mathbf{H}(s)$ 在同步 $s(t)$ 处的 Jacobi (雅可比) 矩阵, 令 $\boldsymbol{\xi} = [\xi_1, \xi_2, \dots, \xi_N]$, 则 (4) 式可以写为

$$\dot{\boldsymbol{\xi}} = \mathbf{Df}(s) \boldsymbol{\xi} - \mu \mathbf{DH}(s) \boldsymbol{\xi} \mathbf{L}^T, \quad (5)$$

记 $\mathbf{L}^T = \mathbf{J} \boldsymbol{\Lambda} \mathbf{J}^{-1}$ 为 Laplace 矩阵 \mathbf{L} 的 Jordan 分解, $\boldsymbol{\Lambda} = \text{diag} [\lambda_1, \lambda_2, \dots, \lambda_N]$ 为 \mathbf{L} 的特征值矩阵. 再令 $\boldsymbol{\psi} = [\psi_1, \psi_2, \dots, \psi_N] = \boldsymbol{\xi} \mathbf{J}$, 将 (5) 式对角化可得

$$\dot{\boldsymbol{\psi}} = \mathbf{Df}(s) \boldsymbol{\psi} - \mu \mathbf{DH}(s) \boldsymbol{\psi} \boldsymbol{\Lambda}. \quad (6)$$

(6) 式是一个关于 λ_k 的 N 维微分方程组, 展开后可以写成

$$\dot{\psi}_k = [\mathbf{Df}(s) - \mu \lambda_k \mathbf{DH}(s)] \psi_k, \quad (k = 2, 3, \dots, N). \quad (7)$$

将 (7) 式一般化后即可得到主稳定方程:

$$\dot{y} = [\mathbf{Df}(s) - \alpha \mathbf{DH}(s)] y, \quad (8)$$

这里, $\alpha = \mu \lambda_k$. (8) 式的最大 Lyapunov 指数 L_{\max} 是变量 α 的函数, 称为动力学网络 (1) 式的主稳定函数. 给定一耦合强度 μ , 在坐标轴上可以相应地找到一点 $\mu \lambda_k$, 若该点对应的 L_{\max} 为负数时, 则该特征模态为稳定态; 反之, 则该特征模态为不稳定态. 如果与特征值 λ_k ($k = 2, 3, \dots, N$) 对应的最大 Lyapunov 指数 L_{\max} 均为负数, 那么在该耦合强度 μ 下, 整个网络的同步流形是渐近稳定的. 我们把使得 $L_{\max} < 0$ 的 α 取值范围 S 称为动态网络 (1) 式的同步化区域, 它由孤立节点的动力学函数 $\mathbf{f}(x_i)$ 和内部耦合函数 $\mathbf{H}(\bullet)$ 决定. 如果耦合强度 μ 与 Laplacian 矩阵的每个非零特征值之积都在同步区域 S 内, 即 $\mu \lambda_k \in S$ ($k = 2, 3, \dots, N$), 那么动态网络 (1) 式的同步流形是渐近稳定的, 网络就能达到同步^[2,3]. 同步区域 S 可以分为以下 4 种类型: I. $S = (\alpha_1, +\infty)$; II. $S = (\alpha_1, \alpha_2)$; III. $S = (\alpha_1, \alpha_2) \cup (\alpha_2, \alpha_3) \cup \dots$; IV. $S = \emptyset$. 其中, 类型 I 网络的同步化能力可以用 Laplacian 矩阵 \mathbf{L} 的最小非零特征值 λ_2 来刻画, λ_2 越大, $\mu \lambda_2$ 越容易大于 α_1 , 网络的同步化能力越强; 类型 II 网络的同步能力可以

用 Laplacian 矩阵 L 的最大特征值 λ_N 与 λ_2 的比值 λ_N/λ_2 来刻画, λ_N/λ_2 越小, 网络的同步能力越强; 类型 III 网络需要所有特征值都满足一定的条件, 很难达到同步; 类型 IV 网络无论耦合强度 μ 和 Laplacian 矩阵 L 的特征值为何值, 网络都不能达到同步. 大多数网络都属于 I 或 II 两种类型, 因此在分析网络的同步能力时, 通常考虑 λ_2 和 λ_N/λ_2 两个指标^[22].

3 网络同步的 SCG 算法和 ISCG 算法

3.1 SCG 方法与 ISCG 方法的原理

由主稳定方程法可知, 当动态网络节点间的耦合强度、孤立节点的动力学方程和内部耦合函数固定后, 网络的同步能力由 Laplacian 矩阵的 λ_2 或 λ_N/λ_2 决定. 所以要想保持网络的同步能力不变, 需要尽量保持网络在粗粒化前后的 λ_2 或 λ_N/λ_2 不变. Dffller 和 Rios^[17,18] 提出的 SCG 算法较好地保留了约简前后网络的 λ_2 和 λ_N/λ_2 值, 从而保持了约简前后网络的同步能力不变. 这种算法主要解决了合并过程中的两个主要问题, 一是如何合并节点及更新边, 即如何更新约简后网络的 Laplacian 矩阵; 二是确定哪些节点可以被合并.

首先, 对于如何更新合并后的边, 我们先看一个小型的网络. 图 1(a) 是一个无向无权的网络, 如果把节点 c, d, e 合并为图 1(b) 中的节点 g , 则称节点 c, d, e 组成了团 C_g , 记 $\omega_{x \rightarrow y}$ 是节点 x 到节点 y 的权值, 我们有:

$$\omega_{a \rightarrow g} = \frac{\sum_{x \in C_g} \omega_{a \rightarrow x}}{|C_g|} = \frac{\omega_{a \rightarrow c} + \omega_{a \rightarrow d} + \omega_{a \rightarrow e}}{3} = (1 + 1 + 1)/3 = 1, \quad (9)$$

$$\omega_{g \rightarrow a} = \sum_{x \in C_g} \omega_{x \rightarrow a} = \omega_{c \rightarrow a} + \omega_{d \rightarrow a} + \omega_{e \rightarrow a} = 3. \quad (10)$$

这里, $|C_g|$ 表示团 C_g 中包含的节点个数. 同理, 我们也可以求出其他的边权. 在大型网络中, 记初始节点的标号 $i = 1, 2, \dots, N$. 约简后的节点标号为 $C = 1, 2, \dots, \tilde{N}$, \tilde{N} 为约简之后网络的节点数, C 也可以表示原始网络中要约简的节点所在的团, 同属于一个团的节点被合并成一个节点. 约简之后网络的 Laplacian 矩阵 \tilde{L} 可以用以下公式求解

$$\tilde{L} = KLR, \quad (11)$$

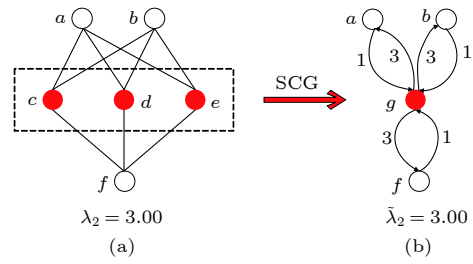


图 1 合并节点的过程

Fig. 1. The processing of merging nodes.

这里, $K \in \mathbf{R}^{\tilde{N} \times N}$ 和 $R \in \mathbf{R}^{N \times \tilde{N}}$ 分别为

$$K_{Ci} = \delta_{C,C_i}/|C|; R_{iC} = \delta_{C,C_i}, \quad (12)$$

其中, C_i 表示原始网络节点 i 所在团的标号; $|C|$ 表示团 C 中节点的个数, δ 是常用的 Kronecker 函数.

接下来考虑合并哪些边, 分别对 λ_2 和 λ_N/λ_2 两种情况进行讨论. 若要保持网络 (1) 式的 Laplacian 矩阵 L 的最小非零特征值 λ_2 不变, SCG 方法的原理是将 λ_2 所对应特征向量 P_2 中分量相同或相近的节点合并在一起, 这样得到合并后网络的 Laplacian 矩阵 \tilde{L} 的最小非零特征值 $\tilde{\lambda}_2$ 与原来网络的 λ_2 相同或相近. 通常, 我们定义特征向量 P_2 所对应的两个分量 p_{2i} 和 p_{2j} 相近是指 $\|p_{2i} - p_{2j}\| / \|p_{2\max} - p_{2\min}\| \ll 1$, 其中 $p_{2\max}$ 和 $p_{2\min}$ 分别是特征向量分量中的最大值和最小值. 具体的操作步骤为, 先把区间 $[p_{2\min}, p_{2\max}]$ 平均分成 i 个区间, 再把分在同一个区间内的特征向量分量对应的节点合并为一个节点, 最后利用 (11) 式求解约简后网络的 Laplacian 矩阵 \tilde{L} . 类似地, 当要保持网络 (1) 的 Laplacian 矩阵 L 的最大特征值与第二小特征值之比 λ_N/λ_2 时, 需要同时保持 λ_2 和 λ_N 不变. 具体的步骤为: 先把 λ_2 所对应的特征向量分量分为 i 个等分区间, 再把 λ_N 所对应的特征向量分量分为 i 个等分区间, 找出 P_2 和 P_N 中同时落入等分区间的分量, 将这些分量对应的节点合并, 最后利用 (11) 式更新合并后的 Laplacian 矩阵, 便可以得到约简后的网络.

图 1 表明, 采用谱粗粒化思想把外部连接相同的节点合并, 网络的第二小特征值 λ_2 不变, 即约简后网络的同步能力不发生改变.

ISCG 方法是谱粗粒化方法的改进算法^[19], 也是通过合并特征向量 P_2 中相同或相近的分量达到约简网络的目的. 但与 SCG 算法不同的是, ISCG 算法不是把特征向量分量等间距平分, 而是采用分

裂聚类的方法对节点进行分类. 具体操作为, 当要保持 λ_2 不变时, 先把 λ_2 所对应的特征向量分量从小到大排列, 算出两两相邻分量间的距离, 找出最大的前 $\tilde{N} - 1$ 个间距, 在这 $\tilde{N} - 1$ 个间距设置分割点, 将特征向量分量分裂为 \tilde{N} 个聚类, 把属于同一聚类的节点合并为一个节点, 最后更新 Laplacian 矩阵, 就得到了节点数为 \tilde{N} 的粗粒化网络. 同理, 当要保持 λ_N/λ_2 不变时, 先根据特征向量分量的间距分别将 \mathbf{P}_2 的分量和 \mathbf{P}_N 的分量分裂为 \tilde{N} 个聚类, 找出同时被分裂到同一聚类里的分量, 把分量对应的节点合并, 最后按照 (11) 式来更新合并后网络的 Laplacian 矩阵. ISCG 算法的优点在于可以将原始网络约简为任意指定规模 \tilde{N} 的粗粒化网络, 复杂度为 $O(N^2)$, 要低于 SCG 算法的复杂度 $O(N^3)$, 并且在特征向量分量分布不均匀时, 使用 ISCG 算法的粗粒化效果要明显优于 SCG 算法 [19].

3.2 SCG 算法和 ISCG 算法的局限

SCG 算法和 ISCG 算法在保持网络的同步能力时都有着得天独厚的优势. 现实世界中大部分特征向量分量分布非常集中, 少数特征向量分量分布特别分散, 对于这种特征向量分量分布不均匀的网络采用 SCG 算法一方面会出现距离很近的特征向量分量被分到两个聚类里面, 另一方面由于少数特征向量分量分布较为分散, 导致得到的聚类数 \tilde{N} 随 i 变

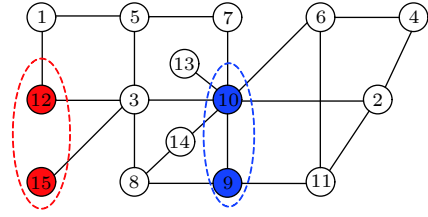


图 2 15 个节点并为 14 个节点的两方案

Fig. 2. Two schemes of merging 15 nodes into 14 nodes.

化不大, 不容易得到任意规模的粗粒化网络 [19]. 而 ISCG 方法采用分裂聚类的方式对节点分类, 因此在特征向量 \mathbf{P}_2 的分量极不均匀时, 采用 ISCG 算法约简网络更能保持其同步能力.

这两种算法分类的方式不同, 都是以特征向量分量的绝对距离作为依据对节点分类. 如果 λ_2 对应的特征向量 \mathbf{P}_2 的第 i 个分量和第 j 个分量相等, 即 $p_{2i} = p_{2j}$, 按照 SCG 算法或 ISCG 算法把这两个节点合并, λ_2 的值在网络约简前后不发生改变. 而当 \mathbf{P}_2 的两个分量有较小的距离时, 两种算法的分类方式都遵循两个分量间距离越小, 被分在一起的几率越大, 而没有考虑分量间相对距离的因素. 如果设分量 p_{2i} 与 p_{2j} 之间的间距为 ε , 则 p_{2i} 与 p_{2j} 间的相对距离就是指 ε 占 $\min\{|p_{2i}|, |p_{2j}|\}$ 的比例 $\rho = \varepsilon / \min\{|p_{2i}|, |p_{2j}|\}$. 下面我们通过一个例子说明在分类时只考虑绝对距离是不合理的.

假设原始网络如图 2 所示, 网络的 Laplacian 矩阵为

$$\mathbf{L} = \begin{bmatrix} 2 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & -1 & 0 & 0 & -1 & 0 & -1 & 0 & -1 & 0 & 0 & -1 \\ 0 & -1 & 0 & 2 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 3 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 2 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 3 & -1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & -1 & -1 & 0 & -1 & 7 & 0 & 0 & -1 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 3 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

其最小非零特征值 $\lambda_2 = 0.4130$, 对应的特征向量 $\mathbf{P}_2 = [-4.19, 2.87, -1.92, 3.61, -2.80, 2.87, -1.14, 0.11, 1.51, 1.00, 2.80, -3.85, 1.71, 0.70, -3.27]^T$, \mathbf{P}_2 的第 9 个与第 10 个分量分别是 1.51 和 1.00, 相差 0.51; 第

12 个与第 15 个分量分别是 -3.85 和 -3.27 相差 0.58. 如果以绝对距离作为合并节点的依据, 则合并节点 9 和 10 后 λ_2 的变化量会更小. 实际上, 由 (11) 式得到合并节点 9 和 10 后的 Laplacian 矩阵 $\tilde{\mathbf{L}}_1$ 为

$$\tilde{L}_1 = \begin{bmatrix} 2 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & -1 & 0 & 0 & -1 & -1 & 0 & -1 & 0 & 0 & -1 \\ 0 & -1 & 0 & 2 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 3 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 2 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 3 & -1 & 0 & 0 & 0 & -1 & 0 \\ 0 & -0.5 & -0.5 & 0 & 0 & -0.5 & -0.5 & -0.5 & 4 & -0.5 & 0 & -0.5 & -0.5 & 0 \\ 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 3 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

\tilde{L}_1 的最小非零特征值 $\tilde{\lambda}_2^1 = 0.4173$ ，比原始网络增加了 1.04%。如果合并节点 12 和 15 后网络的 Laplacian 矩阵 \tilde{L}_2 为

$$\tilde{L}_2 = \begin{bmatrix} 2 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & -1 & 0 & 0 & -1 & 0 & -1 & 0 & -2 & 0 & 0 \\ 0 & -1 & 0 & 2 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 3 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 2 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 3 & -1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 3 & -1 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & -1 & -1 & 0 & -1 & 7 & 0 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 3 & 0 & 0 & 0 \\ -0.5 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 2 \end{bmatrix},$$

\tilde{L}_2 的最小非零特征值 $\tilde{\lambda}_2^2 = 0.4141$ ，比原始网络增加了 0.27%。接下来计算分量间的相对距离，第 9 个和第 10 个分量的相对距离 $\rho_1 = 0.51/1.00 = 0.51$ ，第 12 个与第 15 个分量的相对距离 $\rho_2 = 0.58/|-3.27| = 0.1774$ 。可见，虽然第 12 个和第 15 个分量间的绝对距离比第 9 个和第 10 个大，但相对距离比较小。合并节点 12 和 15 后网络 λ_2 的变化量比合并节点 9 和 10 后的更小，更能保持原始网络的同步能力。

根据以上的讨论，我们提出约简后网络 λ_2 的变化量与所合并节点对应的特征向量分量间相对距离的关系。

定理 1: 在进行谱粗粒化过程合并两个节点时，如果同一个特征值对应的两个特征向量分量 p_{2i} 和 p_{2j} 的间距 ε 远远小于 $\min\{|p_{2i}|, |p_{2j}|\}$ ，则合并节点 i 和节点 j 后网络特征值的变化量 δ 与 p_{2i} 和 p_{2j} 间的相对距离 $\varepsilon/\min\{|p_{2i}|, |p_{2j}|\}$ 成正比。

证明: 假设 λ_2 对应的两个正的特征向量分量

分别为 p_{2i} 与 p_{2j} ($p_{2i} < p_{2j}$)，第 j 个分量与第 i 个分量之间的距离为 ε ，即 $p_{2j} = p_{2i} + \varepsilon$ ，且 $\varepsilon \ll p_{2i}$ 。合并节点 i 和节点 j 后 λ_2 的改变量设为 δ ，写出原始网络 Laplacian 矩阵的特征方程

$$\begin{pmatrix} \vdots & & \vdots \\ l_{i1} & \cdots & l_{in} \\ \vdots & & \vdots \\ l_{j1} & \cdots & l_{jn} \\ \vdots & & \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ p_{2i} \\ \vdots \\ p_{2i+\varepsilon} \\ \vdots \end{pmatrix} = \lambda_2 \begin{pmatrix} \vdots \\ p_{2i} \\ \vdots \\ p_{2i+\varepsilon} \\ \vdots \end{pmatrix}. \quad (13)$$

把 (13) 式的第 i 个等式和第 j 个等式两边相加得到

$$\sum_{k \neq i, j} (l_{ik} + l_{jk}) p_{2k} + (l_{ii} + l_{ji}) p_{2i} + (l_{ij} + l_{jj}) \times (p_{2i} + \varepsilon) = \lambda_2 \cdot (2p_{2i} + \varepsilon). \quad (14)$$

因为分量之间的距离 $\varepsilon \ll p_{2i}$ ，所以可以忽略 ε 的大小，把合并后得到新节点所对应的特征向量分

量近似为 p_{2i} . 按照 (11) 式求出合并后网络的 Laplacian 矩阵并写出特征方程

$$\begin{pmatrix} \vdots & & \dots & & \vdots \\ \frac{l_{i1} + l_{j1}}{2} & \dots & \frac{l_{ii} + l_{ji} + l_{ij} + l_{jj}}{2} & \dots & \frac{l_{in} + l_{jn}}{2} \\ \vdots & & \dots & & \vdots \end{pmatrix} \times \begin{pmatrix} \vdots \\ p_{2i} \\ \vdots \end{pmatrix} = (\lambda_2 + \delta) \begin{pmatrix} \vdots \\ p_{2i} \\ \vdots \end{pmatrix}, \quad (15)$$

(15) 式的第 i 个等式为

$$\begin{aligned} & \frac{1}{2} \cdot \sum_{k \neq i, j} (l_{ik} + l_{jk}) p_{2k} \\ & + \frac{1}{2} \cdot (l_{ii} + l_{ji} + l_{ij} + l_{jj}) p_{2i} \\ & = (\lambda_2 + \delta) \cdot p_{2i}. \end{aligned} \quad (16)$$

由 (14) 式和 (16) 式联立可以得到

$$\delta = \frac{[\lambda_2 - (l_{ij} + l_{jj})] \cdot \varepsilon}{2p_{2i}}. \quad (17)$$

(17) 式表明, 约简网络后 λ_2 的改变量 δ 不仅与特征向量分量间的绝对距离 ε 有关, 还与特征向量分量 p_{2i} 的大小有关. 在 ε 相对于 p_{2i} 足够小时, δ 与绝对距离占 $\min\{p_{2i}, p_{2j}\}$ 的比率 $\rho = \varepsilon / \min\{p_{2i}, p_{2j}\}$ 成正比例关系. 同理, 可以得到当 p_{2i} 为负值时 δ 与 $\varepsilon / \min\{|p_{2i}|, |p_{2j}|\}$ 成正比. 因此, 合并节点 i 和 j 后网络特征值的变化量 δ 与 p_{2i} 和 p_{2j} 间的相对距离 $\varepsilon / \min\{|p_{2i}|, |p_{2j}|\}$ 成正比. 证毕.

从定理 1 可知, 被合并节点对应的特性向量分量间的相对距离会直接影响合并后特征值的变化量, 从而影响到粗粒化的效果. 因此, 在对特征向量分量进行分类时, 应该以特征向量间的相对距离为依据.

4 ISCGR 算法

与 SCG 和 ISCG 算法不同, ISCGR 算法以相对距离作为分类依据, 从而更合理地对节点分类, 改善了网络的粗粒化效果, 使所得粗粒化网络保持同步的能力增强.

在实际操作中, 不能直接以 $\varepsilon / \min\{|p_{2i}|, |p_{2j}|\}$ 的大小作为设置分割点的判断标准, 因为接近 0 的分量不满足分量间的距离 ε 远远小于 $\min\{|p_{2i}|, |p_{2j}|\}$

这一条件, 从而 $\varepsilon / \min\{|p_{2i}|, |p_{2j}|\}$ 会得出很大的距离比率. 这里用 $\frac{\varepsilon}{\min\{|p_{2i}|, |p_{2j}|\} + |\bar{p}_2|}$ 来代替 $\varepsilon / \min\{|p_{2i}|, |p_{2j}|\}$ 作为设置分割点的标准, 其中 $|\bar{p}_2|$ 是 λ_2 对应的所有特征向量分量绝对值的平均值. 以 $\frac{\varepsilon}{\min\{|p_{2i}|, |p_{2j}|\} + |\bar{p}_2|}$ 代替 $\varepsilon / \min\{|p_{2i}|, |p_{2j}|\}$ 一方面避免了靠近 0 的分量算出过大的比率值, 另一方面此指标包含了相对距离的影响因素, 比采用绝对距离 ε 作为分类标准有更好的粗粒化效果. 与 ISCG 算法相比, ISCGR 算法在对特征向量分量进行分类时, 越靠近 0 的分量分得越细化, 聚类区间的长度越小, 越远离 0 的分量分得越粗糙, 聚类区间的长度越大.

ISCGR 算法的具体的步骤为: 1) 把 λ_2 对应的特征向量分量从小到大进行排序为 $\{\tilde{p}_{21}, \tilde{p}_{22} \cdots \tilde{p}_{2N}\}$, 计算两两相邻分量之间的距离 $\varepsilon_i (i = 1, 2, \dots, N - 1)$; 2) 用这个距离除以对应分量绝对值与所有分量绝对值平均值的和, 即计算出 $\eta_i = \frac{\varepsilon_i}{\min\{|\tilde{p}_{2i}|, |\tilde{p}_{2,i+1}|\} + |\bar{p}_2|}$, $(i = 1, 2, \dots, N - 1)$; 3) 根据 η_i 的大小设置分割点, 将分量分裂成若干个聚类; 4) 把同一聚类中的节点合并为一个节点, 利用 (11) 式得到粗粒化网络的 Laplacian 矩阵.

我们举例来说明 ISCGR 的操作步骤. 假设 λ_2 对应的特征向量 P_2 的分量为 $\{p_{21}, p_{22} \cdots p_{2N}\}$, 要将其分为 3 类. 首先把特征向量分量从小到大排列为 $\{\tilde{p}_{21}, \tilde{p}_{22} \cdots \tilde{p}_{2N}\}$, 相邻分量之间的间距为 $\{\varepsilon_1, \varepsilon_2 \cdots \varepsilon_{N-1}\}$, 再通过公式计算出 $\{\eta_1, \eta_2 \cdots \eta_{N-1}\}$, 选出最大的两个相对距离记为 $\eta_j, \eta_k (j < k)$, η_j 对应着 \tilde{p}_{2j} 和 $\tilde{p}_{2,j+1}$ 之间的相对距离, η_k 是 \tilde{p}_{2k} 和 $\tilde{p}_{2,k+1}$ 之间的相对距离. 接着在这两个间距处设置分割点, 把分量分裂成 3 类, 分别为 $[\tilde{p}_{21}, \tilde{p}_{2j}]$, $[\tilde{p}_{2,j+1}, \tilde{p}_{2k}]$ 和 $[\tilde{p}_{2,k+1}, \tilde{p}_{2N}]$. 最后将落入同一区间内的点合并为一个节点, 利用 (11) 式更新合并后的 Laplacian 矩阵, 就可以获得节点数为 3 的粗粒化网络. ISCGR 算法对节点分类时考虑了特征向量分量间的相对距离, 在两组分量间距相同的情况下, 由于靠近 0 的分量相对距离大, 所以会优先在靠近 0 的两个向量间设置分割点, 把相对距离小的分量所对应的节点合并. 由此可见, ISCGR 算法在区分不同节点的相似程度时更加合理, 保证了分在同一聚类的节点相似程度高于不同聚类的节点, 因此粗粒化效果较 ISCG 算法更好.

与 λ_2 类似, 对于类型 II 网络, 当考虑 λ_N/λ_2 作为同步能力衡量指标时, 先用 ISCG 算法分别对 λ_N 和 λ_2 对应的特征向量分量进行分类, 将同时分裂在同一聚类的节点合并为一个节点, 最后更新 Laplacian 矩阵, 得到约简后的粗粒化网络.

5 仿真分析

本节将对一般连续时间的动态网络模型和 27 种现实世界网络进行数值仿真, 分别应用 ISCG 方法和 ISCGR 方法对网络进行约简, 比较两种算法对网络约简后同步能力的保持效果.

5.1 模型网络仿真

首先对连续时间动态网络模型进行数值仿真, 这里选取三种典型的网络模型, 即 BA 无标度网络^[23]、ER 随机网络^[24]、和 NW 小世界网络^[25,26], 分别应用 ISCG 算法和 ISCGR 算法对其进行约简, 分析比较没有考虑相对距离和基于相对距离的算法的粗粒化效果.

对于类型 I 网络, 保持同步能力不变只需要保持 λ_2 不变即可. 图 3 展示了分别采用 ISCG 算法和 ISCGR 算法对 1000 个节点的 BA、ER、NW 网络约简后 λ_2 的变化情况. 图 3(a) 是对 1000 个节点的 BA 网络的粗粒化过程仿真, 原始网络的 λ_2 为 2.95. 采用 ISCG 方法将网络约简至 200, 102, 72 个节点时, 网络的 $\tilde{\lambda}_2$ 分别变为 2.96, 3.13, 3.26, 比原网络增加了 0.34%, 6.10%, 10.51%; 采用 ISCGR 方法将网络约简至 200, 102, 72 个节点时, 网络的 $\tilde{\lambda}_2$ 为别是 2.95, 2.95, 2.98, 变化了 0, 0, 1.02%. 由此可见, 采用 ISCGR 算法保持 BA 网络 λ_2 的能力要优于 ISCG 算法. 对于图 3(b)1000 个节点的 ER 随机网络和图 3(c)1000 个节点的 NW 小世界网络也能得出同样的结论, 即在网络约简至相同的节点时, 采用 ISCGR 算法 λ_2 值的变化量要小于 ISCG 算法, 这表明采用 ISCGR 算法保持类型 I 网络同步能力的效果要优于 ISCG 算法.

对于类型 II 网络, 同步能力用 λ_N/λ_2 衡量. 图 4 展示了分别采用 ISCG 算法和 ISCGR 算法对 1000 个节点的 BA 无标度网络、ER 随机网络、NW 小世界网络约简后保持 λ_N/λ_2 情况的对比. 从图中可以看出, 采用这两种算法对网络进行约简后, λ_N/λ_2 都是随着约简后网络节点数 \tilde{N} 的减少而

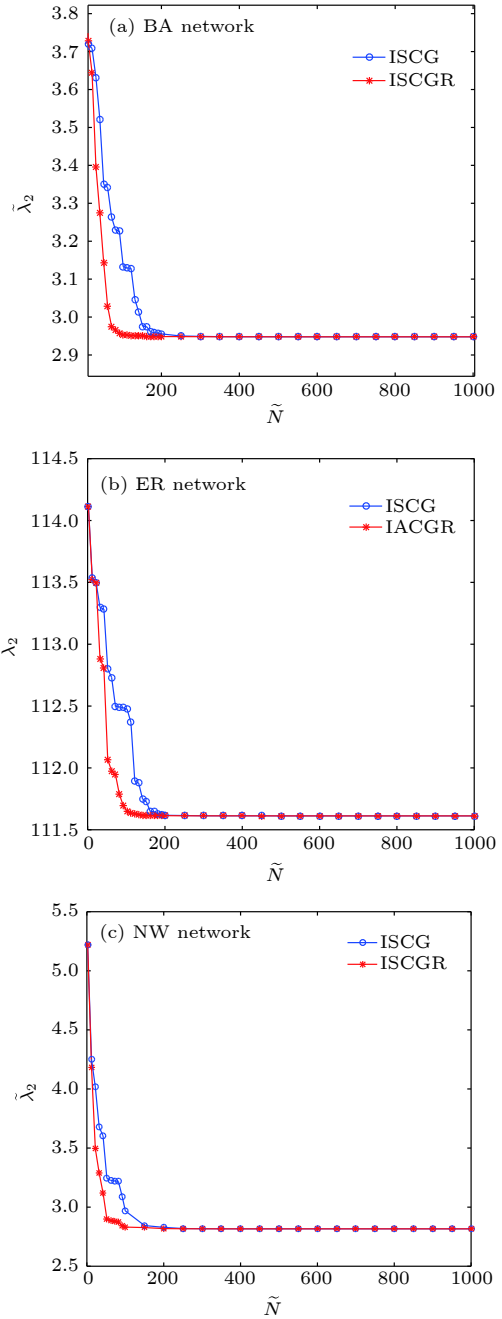


图 3 采用 ISCG 与 ISCGR 算法获得谱粗粒化网络对 λ_2 的保持情况 (a) BA 无标度网络; (b) ER 随机网络; (c) NW 小世界网络

Fig. 3. The maintaining of λ_2 obtained by using ISCG and ISCGR algorithms in coarse graining network: (a) BA network; (b) ER network; (c) NW network.

减小, 但在 \tilde{N} 相同的情况下, 用 ISCGR 算法得到粗粒化网络的 λ_N/λ_2 变化更小, 这表明了 ISCGR 算法保持类型 II 网络的同步能力比 ISCG 算法更强.

由此可见, 对于不同类型的网络和不同的网络模型, 采用 ISCGR 算法约简网络后保持同步的能力都要优于 ISCG 算法.

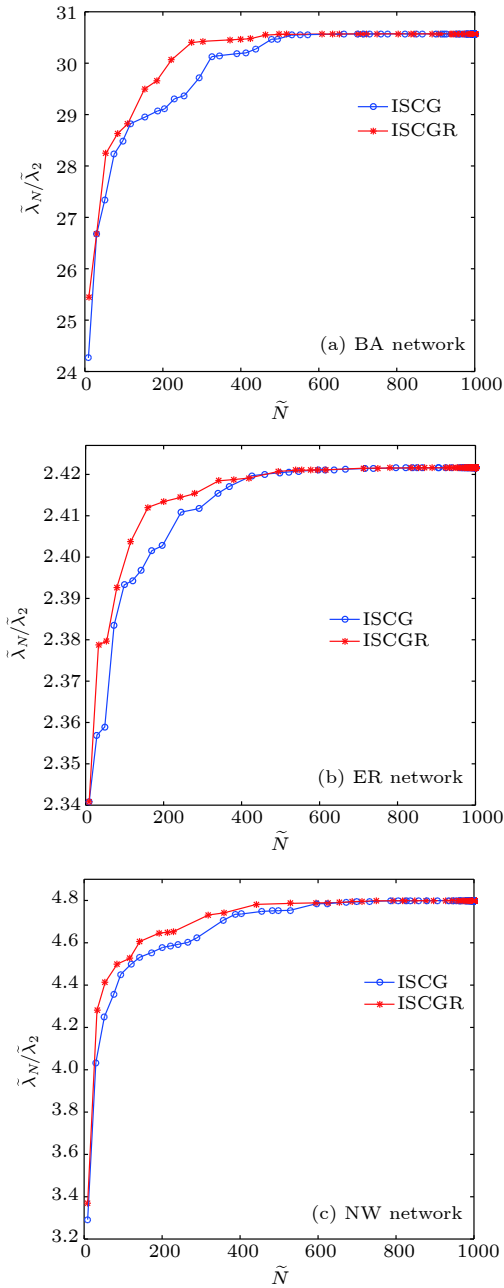


图 4 采用 ISCG 算法与 ISCGR 算法获得谱粗粒化网络对 λ_N/λ_2 的保持情况 (a) BA 无标度网络; (b) ER 随机网络; (c) NW 小世界网络

Fig. 4. The maintaining of λ_N/λ_2 obtained by using ISCG and ISCGR algorithms in coarse graining network: (a) BA network; (b) ER network; (c) NW network.

5.2 实际网络仿真

下面应用一些真实网络进行粗粒化过程仿真, 进一步比较 ISCG 算法和 ISCGR 算法保持实际网络同步能力的效果. 我们选取了协作、沟通、计算机、基础设施、社会、生物、词汇和电力等不同领域的网络 [27–29]. 这些网络的规模从几百个到几千个

节点不等, 既包括稀疏网络又包括致密网络. 表 1 给出了这些网络的基本数据及分别采用 ISCG 和 ISCGR 算法约简到原始节点数 N 的 30%, 20%, 10%, 2% 时, 网络 Laplacian 矩阵 L 的最小非零特征值 λ_2 的变化量. 图 5 展示了分别采用 ISCG 和 ISCGR 算法对这些实际网络约简后 λ_2 的变化情况, 其中图 5(a) 是从表 1 中挑选了一些随着节点数 \tilde{N} 减少, λ_2 变化比较大的网络, 图 5(b) 挑选了一些随着节点数 \tilde{N} 减少, λ_2 变化不太明显的网络. 图 5 可以看出约简后网络节点数 \tilde{N} 越小, λ_2 的变化量越大, 但是在 \tilde{N} 相同的情况下, 采用 ISCGR 方法约简网络后得到的 λ_2 误差比采用 ISCG 方法要小. 这表明对于实际网络, 采用 ISCGR 方法比 ISCG 方法更能保持网络的同步能力.

对比表 1 中平均度和约简后 λ_2 值变化量的关

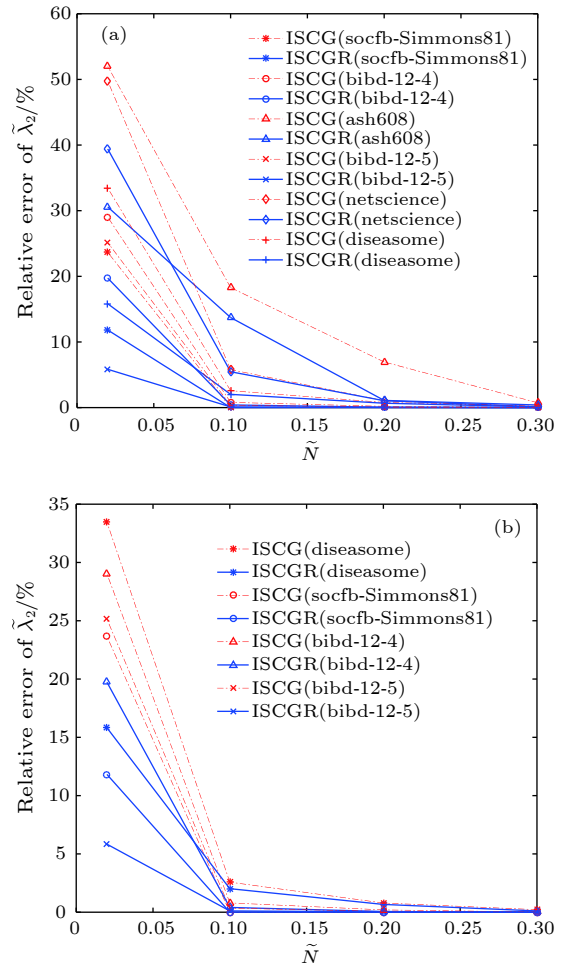


图 5 分别采用 ISCG 与 ISCGR 算法对实际网络进行粗粒化后保持 λ_2 情况的对比图

Fig. 5. The maintaining of λ_2 obtained by using ISCG and ISCGR algorithms for real-world networks in coarse graining network.

表 1 分别采用 ISCG 和 ISCGR 算法对实际网络约简后 λ_2 的保持情况统计表

Table 1. The Statistics table of maintaining λ_2 obtained by using ISCG and ISCGR algorithms for some real-world networks in coarse graining network.

种类	网络	节点	边	文献	$\Delta\lambda_2$ (30%N)		$\Delta\lambda_2$ (20%N)		$\Delta\lambda_2$ (10%N)		$\Delta\lambda_2$ (2%N)	
					ISCG	ISCGR	ISCG	ISCGR	ISCG	ISCGR	ISCG	ISCGR
化学	DD_g1	327	899	[27]	2.06%	1.77%	7.66%	7.17%	34.90%	23.46%	187%	173%
	DD_g1046	707	1646	[27]	1.40%	1.31%	7.27%	6.62%	42.92%	17.81%	191%	94%
合作	netscience	379	914	[27]	0.13%	0.13%	1.05%	1.05%	5.79%	5.46%	49.74%	39.41%
	ca-GrQc	4158	13422	[27]	0	0	0	0	0.03%	0.03%	0.34%	0.20%
社交	ia-infect-dublin	410	2765	[27]	0.91%	0.95%	3.59%	2.46%	13.47%	14.37%	104%	51%
	moreno_crime	829	1473	[28]	0.01%	0.01%	0.39%	0.24%	3.02%	2.32%	13.34%	13.05%
	socfb-Sim81	1518	32988	[28]	0	0	0	0	0	0	23.65%	11.82%
电力	ia-email-univ	1133	5451	[27]	0	0	0	0	0.07%	0.06%	0.26%	0.16%
	东北电网	58	70	[29]	8.14%	5.78%	24.36%	18.63%	54.65%	54.65%	—	—
	IEEE162	162	280	[29]	2.35%	2.10%	16.94%	8.68%	27.70%	25.65%	113%	113%
生物	IEEE145	145	422	[29]	0.85%	0.80%	4.14%	3.16%	17.85%	14.22%	240%	230%
	diseasome	516	1188	[27]	0.22%	0.11%	0.79%	0.67%	2.58%	2.02%	33.45%	15.82%
互联网	Route views	6474	12572	[28]	0	0	0	0	0.07%	0.07%	1.07%	0.57%
	Wiki-vote	889	2914	[27]	0.02%	0.01%	0.07%	0.05%	0.17%	0.17%	0.61%	0.49%
	bibd_12_4	495	2951	[27]	0.02%	0.01%	0.20%	0.06%	0.80%	0.41%	29.02%	19.73%
技术	G1	800	19176	[27]	0	0	0	0	0.27%	0.19%	0.56%	0.55%
	GD00_c	638	1020	[27]	0.02%	0.02%	0.27%	0.27%	1.50%	1.15%	4.29%	2.75%
	dwt_1005	1005	3808	[27]	2.17%	0.49%	4.25%	1.61%	23.93%	10.83%	129%	92%
	dwt_503	503	2762	[27]	4.94%	3.68%	9.33%	6.18%	44.00%	21.32%	138%	117%
	130bit	584	6058	[27]	0.01%	0	0.02%	0.01%	0.68%	0.36%	1.51%	1.15%
	ash608	608	1212	[27]	0.74%	0.40%	6.95%	1.10%	18.29%	13.73%	51.99%	30.57%
数学	bibd_12_5	792	7860	[27]	0.02%	0	0.07%	0.03%	0.35%	0.11%	25.14%	5.84%
	jagmesh3	1089	3136	[27]	0.31%	0.10%	1.44%	1.13%	16.86%	9.25%	192%	127%
	frb45-21-1	945	386854	[27]	0.03% ^{ooo}	0	0.23% ^{ooo}	0.04% ^{ooo}	0.51% ^{ooo}	0.28% ^{ooo}	1.30% ^{ooo}	1.34% ^{ooo}
随机	kneser_6_2_1	676	2017	[27]	0.03%	0.01%	0.16%	0.13%	1.83%	0.44%	81%	30%
	EX1	560	4368	[27]	0	0	0.03%	0.03%	3.08%	0.41%	123%	26%
	G43	1000	9990	[27]	0	0	0.01%	0	0.13%	0.11%	0.95%	0.50%

系发现, 在约简程度相同时, 网络的平均度越大, 约简后网络 λ_2 的变化量越小, 保持同步的能力越强. 这是因为网络的度越大, 越接近于全局耦合网络, 网络中外部连接相同或相似的节点越多, 所以合并这些节点后对网络的特征值影响不大; 而平均度越小, 网络越稀疏, 网络中外部连接相同或相似的节点越少, 约简后网络的特征值变化量就越大.

另外, 表 1 的数据表明, ISCGR 算法对互联网、生物、社交、合作等网络的同步保持能力更强一些, 对电力、化学等网络的同步保持能力则弱一些. 这是因为互联网^[30]具有模块性, 生物网络(如新陈代谢网络)中的拓扑结构是按等级组织起来的^[31], 社交网络与合作网络都具有“物以类聚, 人

以群分”的特点^[32], 即这类网络都具有高聚类的特性, 网络中包含大量外部连接相同或相似的节点, 所以 ISCGR 算法对这类网络的约简效果更好. 相反, 电力网络、化学网络中的节点大都是链状式排列, 这类网络聚类特性不明显, 外部连接相同的节点很少, 所以采用 SCG 算法对网络约简到一定程度时很难保持网络的同步能力.

6 结 论

小规模网络中的一些性质不能直接应用在大规模网络中, 所以复杂网络的约简成为了复杂网络理论中的重要课题. 通过对网络约简, 不仅可以把

小规模网络中已经研究过的性质应用到大规模网络中,且简化了分析和计算. SCG 算法是一种保持类型 I 和类型 II 复杂网络同步能力不变的约简方法. 本文在原始谱粗粒化方法基础上提出了一种基于相对距离的谱粗粒化方法,使应用特征向量分量分类时比原算法更加合理,约简后的网络保持同步的能力得到增强,为大型网络的谱粗粒化提供了一种更加精准的约简算法;并通过对 3 种经典网络模型和 27 种实际网络的仿真分析表明,ISCGR 算法比原算法粗粒化效果更好,进一步发现 SCG 算法对具有明显聚类结构的实际网络(互联网、生物、社交、合作等)比模糊聚类结构的实际网络(电力、化学等)更容易保持粗粒化网络的同步能力.

需要指出的是,SCG 算法仅考虑了保持网络的同步能力不变对网络进行约简,所得粗粒化网络的其他动态特性或拓扑结构却不一定和原始网络相同或相近. SCG 算法除了能保持同步能力之外,是否还能保持约简后网络的其他动力学特性,例如可控性、一致性、稳定性等?是否还能保持约简后网络的小世界特性、度的幂律特性等网络拓扑性质?谱粗粒化后的网络还有哪些性质保留了下来,其他性质发生了什么改变?有没有一种约简方法可以兼并保持多种性质不变?这些问题都是值得我们进一步研究和思考的.网络的约简问题还有太多的未知,这些未知也吸引着我们不断去探索.

参考文献

- [1] Watts D J 2004 *Annu. Rev. Sociol.* **30** 243
- [2] Pecora L M, Carroll Y L 1998 *Phys. Rev. Lett.* **80** 2109
- [3] Fink K S, Johnson G, Carroll T, Mar D, Pecora L 2000 *Phys. Rev. E* **61** 5080
- [4] Wang X F, Chen G R 2002 *Int. J. Bifurcat. Chaos* **12** 187
- [5] Belykh I V, Belykh V N 2004 *Physica D* **195** 159
- [6] Motter A E, Zhou C S, Kurths J 2005 *Phys. Rev. E* **71** 016116
- [7] Nishikawa T, Motter A E 2006 *Physica D* **224** 77
- [8] Arenas A, Díaz-Guilera A, Kurths J, Moreno Y, Zhou C S 2008 *Phys. Rep.* **469** 93
- [9] Zhu T X, Wu Y, Xiao J H 2012 *Acta Phys. Sin.* **61** 040502 (in Chinese) [朱廷祥, 吴晔, 肖井华 2012 物理学报 **61** 040502]
- [10] Sun J, Li X X, Zhang J H, Shen Y Z, Li Y Y 2017 *Acta Phys. Sin.* **66** 188901 (in Chinese) [孙娟, 李晓霞, 张金浩, 申玉卓, 李艳雨 2017 物理学报 **66** 188901]
- [11] Wei J, Wu X Q, Lu J A, Wei X 2017 *Europhys. Lett.* **120** 20005
- [12] Chen C, Liu S, Shi X Q, Chaté H, Wu Y L 2017 *Nature* **542** 210
- [13] Wang Y J, Xu L L, Song S, Li K Y 2018 *Acta Phys. Sin.* **67** 050504 (in Chinese) [王宇娟, 涂俐兰, 宋帅, 李宽洋 2018 物理学报 **67** 050504]
- [14] Zheng G C, Liu C X, Wang Y 2018 *Acta Phys. Sin.* **67** 050502 (in Chinese) [郑广超, 刘崇新, 王琰 2018 物理学报 **67** 050502]
- [15] Shen J, Tang L K 2018 *Chin. Phys. B* **27** 100503
- [16] Ma X J, Huang L, Lai Y C, Wang Y, Zheng Z 2008 *Chaos* **18** 043109
- [17] Gfeller D, Rios P D L 2007 *Phys. Rev. Lett.* **99** 038701
- [18] Gfeller D, Rios P D L 2008 *Phys. Rev. Lett.* **100** 174104
- [19] Zhou J, Jia Z, Li K Z 2017 *Acta Phys. Sin.* **66** 060502 (in Chinese) [周建, 贾贞, 李科赞 2017 物理学报 **66** 060502]
- [20] Chen J, Lu J A, Lu X F, Wu X Q, Chen G R 2013 *Commun. Nonlinear Sci.* **18** 3036
- [21] Wang P, Xu S 2017 *Physica A* **478** 168
- [22] Guo S Z, Lu Z M 2012 *Complex Network Basic Theory* (Beijing: Science Press) pp183-187 (in Chinese) [郭世泽, 陆哲明 2012 复杂网络基础理论(北京: 科学出版社)第183—187页]
- [23] Barabási A L, Albert R 1999 *Science* **286** 509
- [24] Erdős P, Rényi A 1960 *Publ. Math. Inst. Hung. Acad. Sci.* **5** 17
- [25] Watts D J, Strogatz S H 1998 *Nature* **393** 440
- [26] Newman M E J, Watts D J 1999 *Phys. Lett. A* **263** 341
- [27] Ahmed N, Rossi R A, Zhou R <http://networkrepository.com/index.php> [2018-9-14]
- [28] Kunegis J <http://konect.uni-koblenz.de/> [2018-9-14]
- [29] Luo X R 2012 M.S. Thesis (Chongqing: Southwest University) (in Chinese) [罗筱如 2012 硕士学位论文(重庆: 西南大学)]
- [30] Ai J, Zhao H, Kathleen M C, Su Z, Li H 2013 *Chin. Phys. B* **22** 078902
- [31] Ravasz E, Somera A L, Mongru D A, Oltvai Z N, Barabási A L 2002 *Science* **297** 1551
- [32] Xiong F, Wang X M, Cheng J J 2016 *Chin. Phys. B* **25** 108904

A spectral coarse graining algorithm based on relative distance*

Yang Qing-Lin Wang Li-Fu[†] Li Huan Yu Mu-Zhou*(School of Control Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China)*

(Received 15 October 2018; revised manuscript received 14 March 2019)

Abstract

As a key approach to understanding complex systems (e.g. biological, physical, technological and social systems), the complex networks are ubiquitous in the whole world. Synchronization in complex networks is significant for a more in-depth understanding of the dynamic characteristics of the networks, where tremendous efforts have been devoted to their mechanism and applications in the last two decades. However, many real-world networks consist of hundreds of millions of nodes. Studying the synchronization of such large-scale complex networks often requires solving a huge number of coupled differential equations, which brings great difficulties to both computation and simulation. Recently, a spectral coarse graining approach was proposed to reduce the large-scale network into a smaller one while maintaining the synchronizability of the original network. The absolute distance between the eigenvector components corresponding to the minimum non-zero eigenvalues of the Laplacian matrix is used as a criterion for classifying the nodes without considering the influence of the relative distance between eigenvector components in an original spectral coarse graining method. By analyzing the mechanism of the spectral coarse graining procedure in preserving the synchronizability of complex networks, we prove that the ability of spectral coarse graining to preserve the network synchronizability is related to the relative distance of the eigenvector components corresponding to the merged nodes. Therefore, the original spectral coarse graining algorithm is not satisfactory enough in node clustering. In this paper, we propose an improved spectral coarse graining algorithm based on the relative distance between eigenvector components, in which we consider the relative distance between the components of eigenvectors for the eigenvalues of network coupling matrix while clustering the same or similar nodes in the network, thereby improving the clustering accuracy and maintaining the better synchronizability of the original network. Finally, numerical experiments on networks of ER random, BA scale-free, WS small-world and 27 different types of real-world networks are provided to demonstrate that the proposed algorithm can significantly improve the coarse graining effect of the network compared with the original algorithm. Furthermore, it is found that the networks with obvious clustering structure such as internet, biological, social and cooperative networks have better ability to maintain synchronization after reducing scale by spectral coarse-grained algorithm than the networks of fuzzy clustering structure such as power and chemical networks.

Keywords: complex network, synchronizability, spectral coarse graining, relative distance**PACS:** 05.45.Xt, 05.10.-a, 05.45.-a**DOI:** 10.7498/aps.68.20181848

* Project supported by the Nature Science Foundation of Hebei Province, China (Grant Nos. F2016501023, F2017501041), Fundamental Research Funds for the Central Universities, China (Grant No. N172304030), and the National Natural Science Foundation of China (Grant No. 61402088).

[†] Corresponding author. E-mail: wlfk@qq.com