

Ciência das Redes

Comunidades

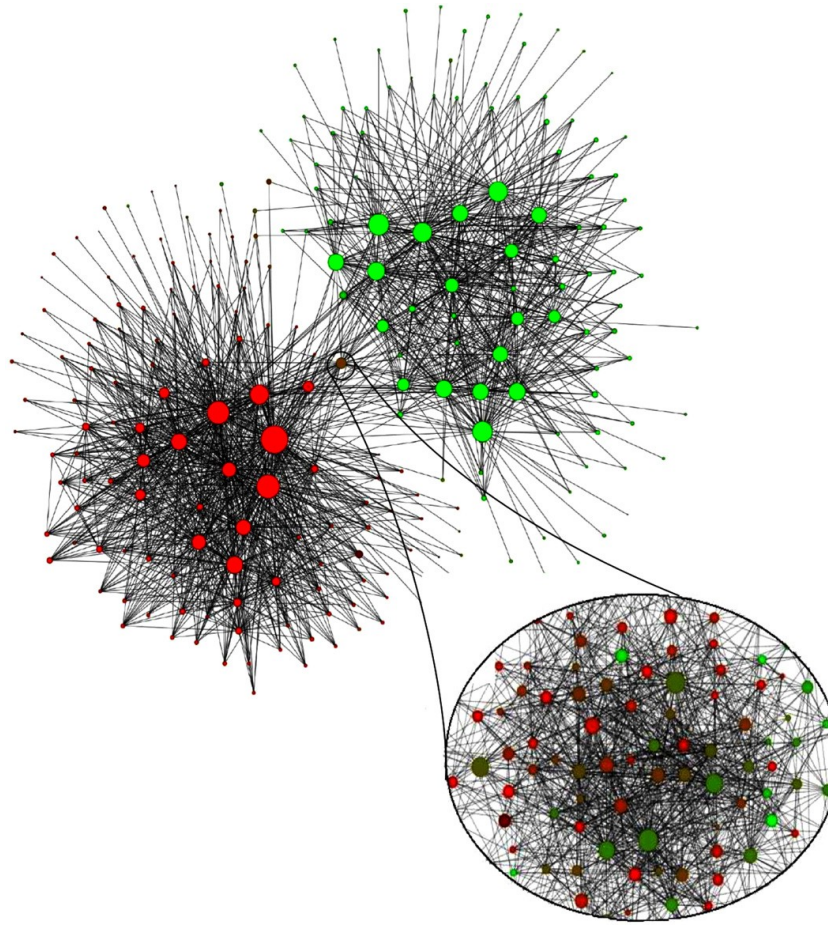
Ricardo Luders
Thiago H Silva







Área pequena
Comparável ao Massachusetts nos EUA ou o Espírito Santo no Brasil



Dados de ligações telefônicas

Nós correspondem a comunidades, cores indicam a língua falada na comunidade

Vermelho=Francês

Verde=Dutch

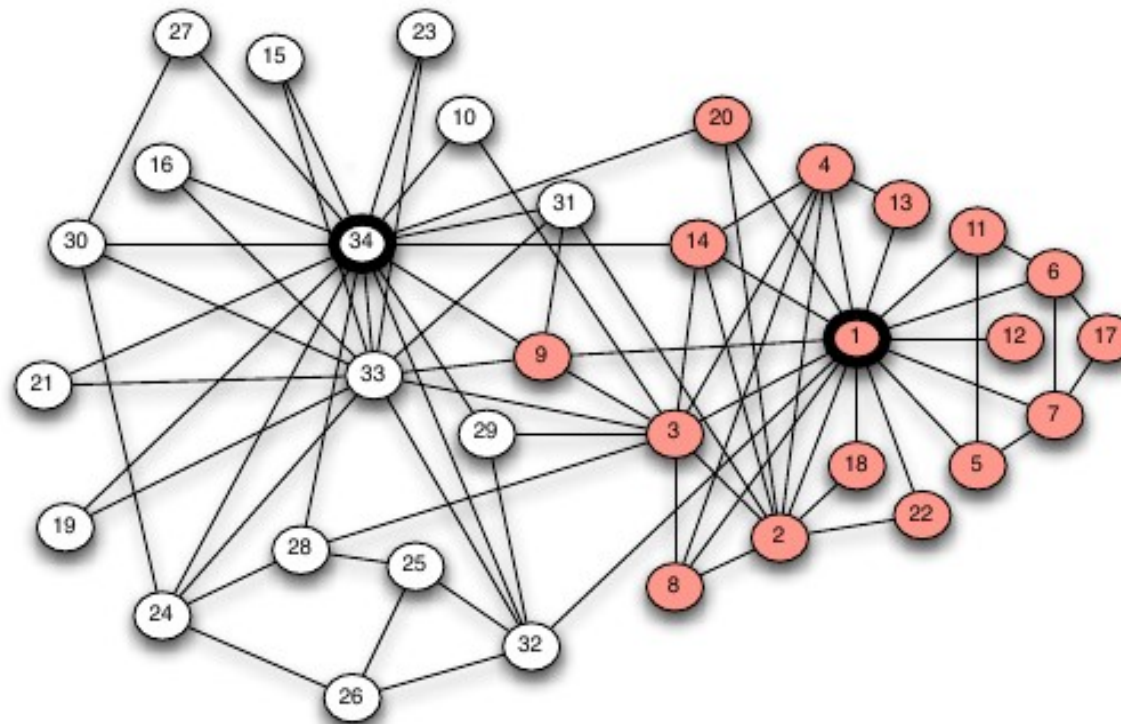
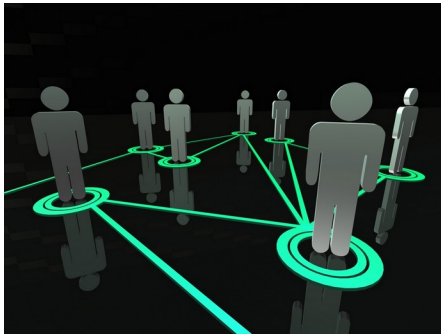


Figure 3.13: A karate club studied by Wayne Zachary [421] — a dispute during the course of the study caused it to split into two clubs. Could the boundaries of the two clubs be predicted from the network structure?



Escala micro



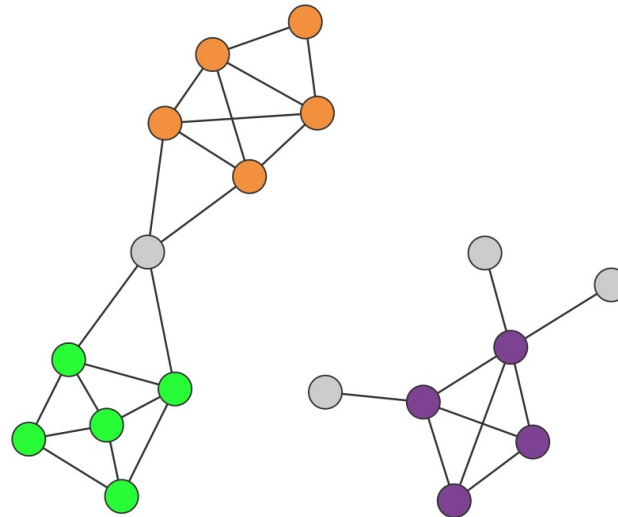
Escala macro

Foco é na
escala do meio

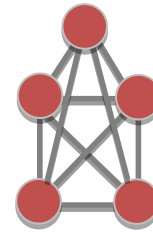
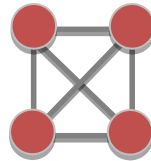
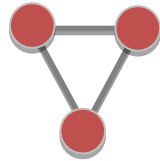
H1: A estrutura da comunidade de uma rede é codificada exclusivamente nas ligações do grafo

H2: Uma comunidade corresponde a um subgrafo conectado.

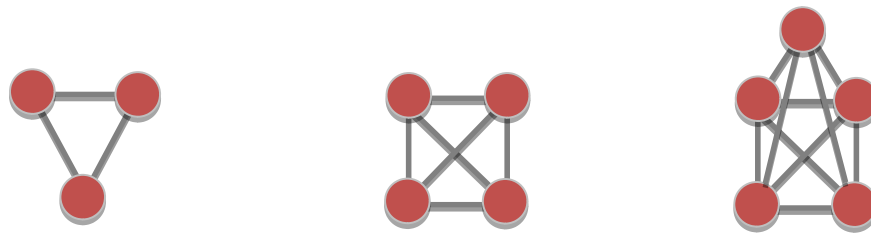
H3: Comunidades correspondem a regiões localmente densas de uma rede.



Um clique é um subgrafo completo de nós k



Uma clique é um subgrafo completo de nós k



Os triângulos são frequentes; cliques maiores são raros.

Comunidades não correspondem necessariamente a subgrafos completos, pois muitos de seus nós não se ligam diretamente uns aos outros.

Encontrar os cliques de uma rede é bastante exigente em termos computacionais, sendo um problema NP-completo.

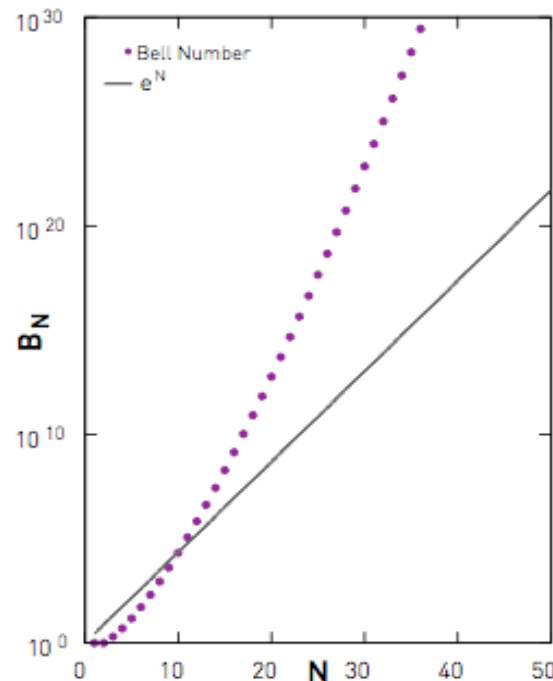
Importante diferença:

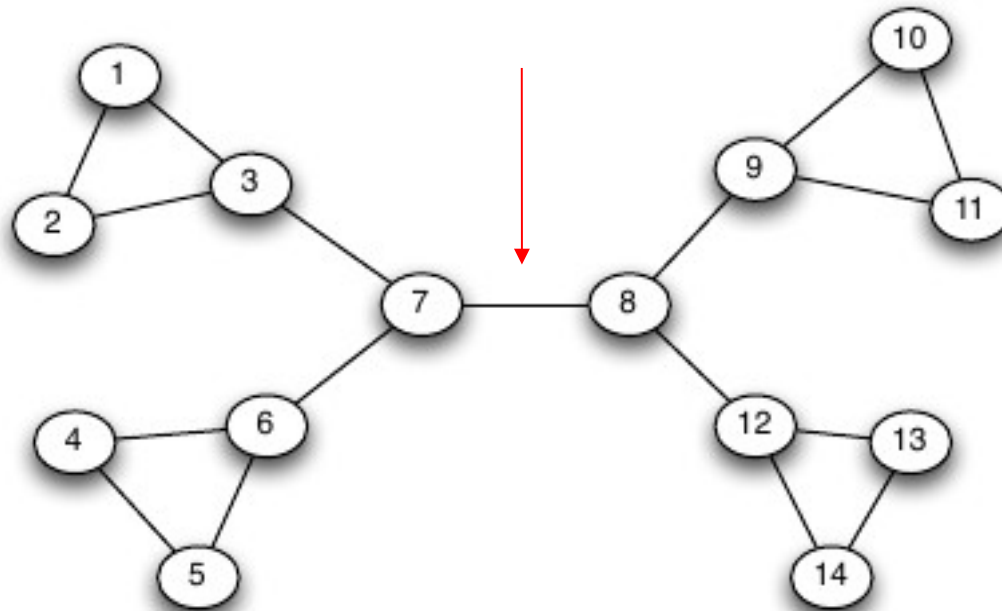
Particionamento do grafo - divide a rede em um número predefinido de subgrafos menores

Detecção de comunidade - descobre uma estrutura de comunidade presente, onde o número e o tamanho das comunidades não é definido

A noção de comunidade está relacionada com subgrafos densamente conectados.

Pode-se chegar a várias formas de identificar comunidades se inspecionarmos todas as participações, no entanto isso não é viável computacionalmente

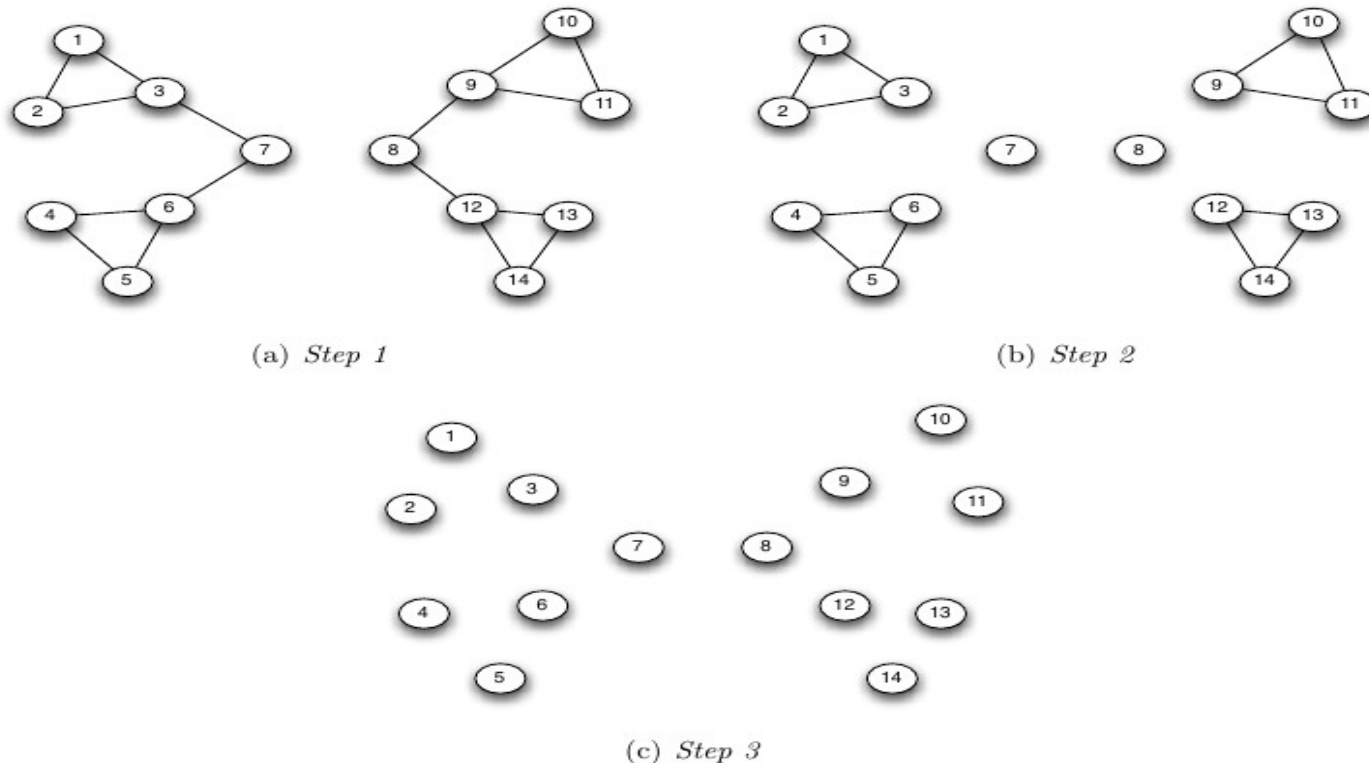




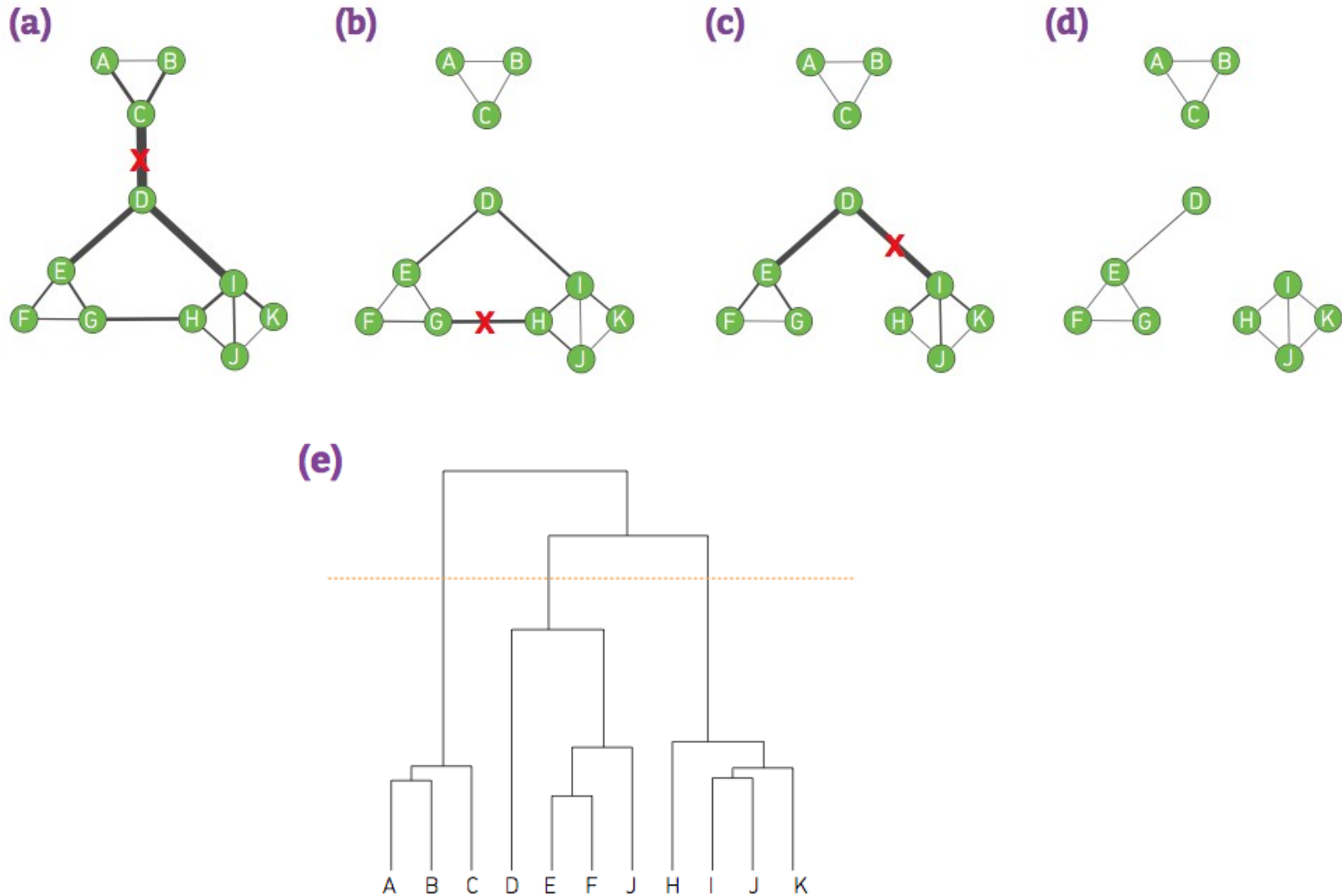
Edge betweenness: número de menores caminhos entre pares de nós que passam por pela aresta

Algoritmo de Girvan-Newmann

- Calcular o betweenness de todas as arestas
- Cortar a aresta com o maior betweenness
- Recalcular o edge betweenness
- Repetir até que todos os links sejam removidos



Algoritmo de Girvan-Newmann



Algoritmo de Girvan-Newmann

Custo $O(N^3)$

Esse tipo de estratégia também fornece outros desafios:

Por exemplo, saber onde parar de fazer cortes

H4: Não se espera que as redes formadas aleatoriamente tenham uma estrutura de comunidade.

Talvez uma boa medida para parar é quando a “coesão” de cada comunidade é maior do que seria aleatoriamente.

$$M_c = \frac{1}{2L} \sum_{(i,j) \in C_c} (A_{ij} - p_{ij})$$

Conexões
originais

Conexões
esperadas em uma
rede aleatória

N = # nodes

L = # edges

L_c = # edges comunidade c

C_c = nós no subgrafo C_c

n_c = # comunidades

K_c = grau total da
comunidade c

Após uma manipulação (ver material avançado do livro Network Science: 9.B):

$$M_c = \frac{L_c}{L} - \left(\frac{k_c}{2L} \right)^2$$

$$M_c = \frac{1}{2L} \sum_{(i,j) \in C_c} (A_{ij} - p_{ij})$$

Conexões
originais

Conexões
esperadas em uma
rede aleatória

N = # nodes

L = # edges

L_c = # edges comunidade c

C_c = nós no subgrafo C_c

n_c = # comunidades

K_c = grau total da
comunidade c

Após uma manipulação (ver material avançado do livro Network Science: 9.B):

$$M_c = \frac{L_c}{L} - \left(\frac{k_c}{2L} \right)^2$$

Generalização
(todas comunidades)

$$M = \sum_{c=1}^{n_c} \left[\frac{L_c}{L} - \left(\frac{k_c}{2L} \right)^2 \right]$$

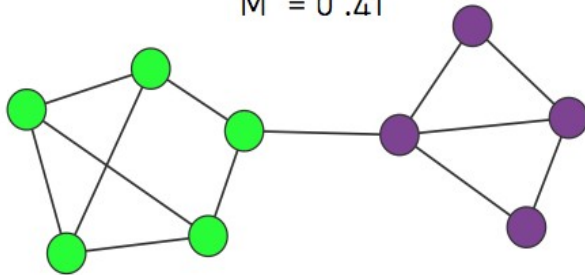
H5: Hipótese de Modularidade Máxima

A partição com a modularidade máxima M para uma determinada rede oferece a estrutura de comunidade ideal

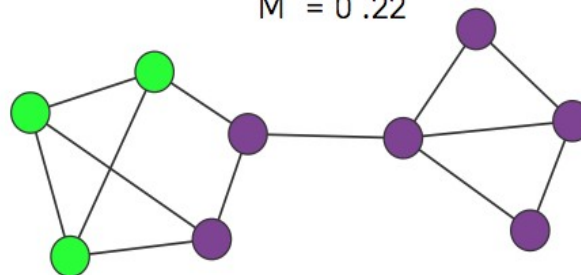
H5: Hipótese de Modularidade Máxima

A partição com a modularidade máxima M para uma determinada rede oferece a estrutura de comunidade ideal

(a) OPTIMAL PARTITION
 $M = 0.41$



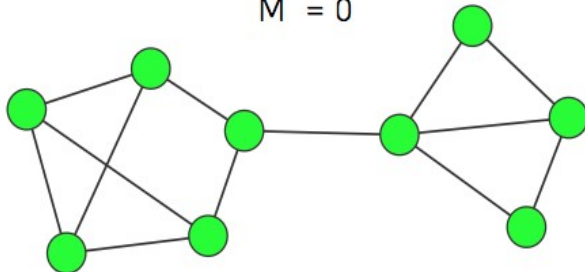
(b) SUBOPTIMAL PARTITION
 $M = 0.22$



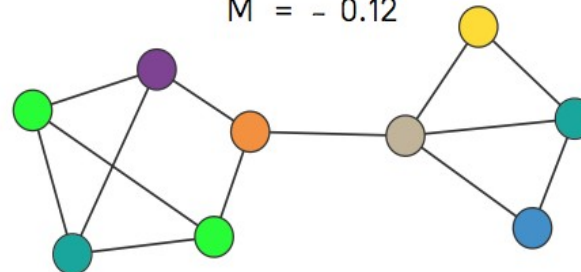
Partição ótima, que maximiza a modularidade.

Modularidade subótima, mas positiva.

(c) SINGLE COMMUNITY
 $M = 0$



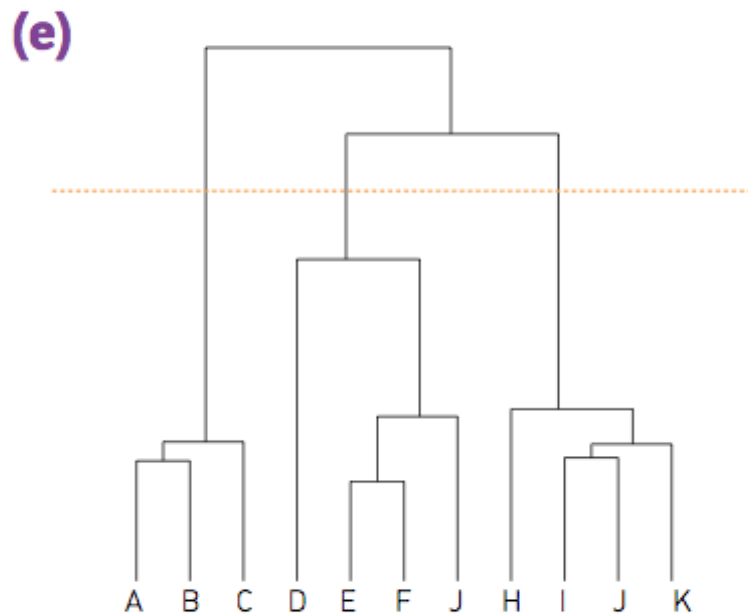
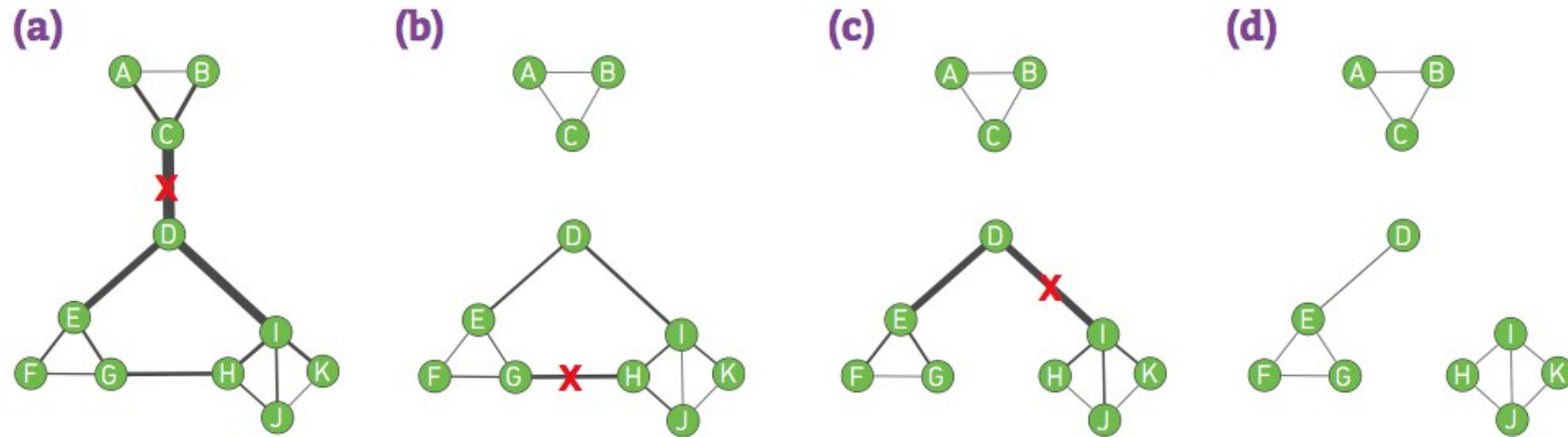
(d) NEGATIVE MODULARITY
 $M = -0.12$



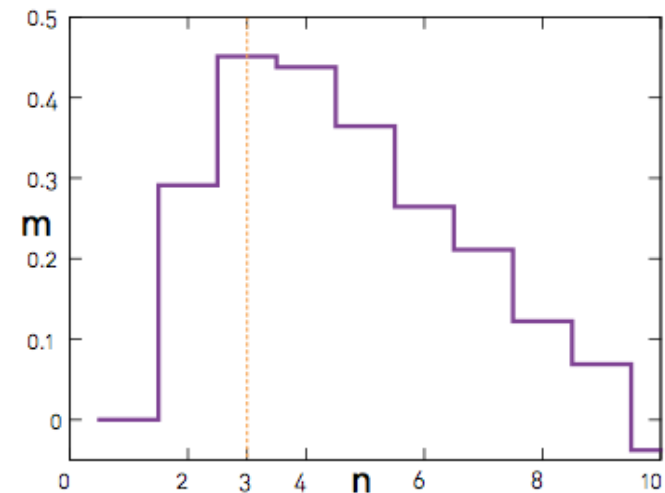
Modularidade negativa: se atribuirmos cada nó a uma comunidade diferente.

Modularidade zero: Atribuição de todos os nós à mesma comunidade, independente da estrutura da rede.

Modularidade para o Girvan-Newmann



(f) Eixo Y é a modularidade



Rede de produtos

Nós são produtos

Link entre A e B se o produto A é frequentemente comprado com B

200000 nós e 2M de arestas

Modularidade máxima resulta em 1684 comunidades

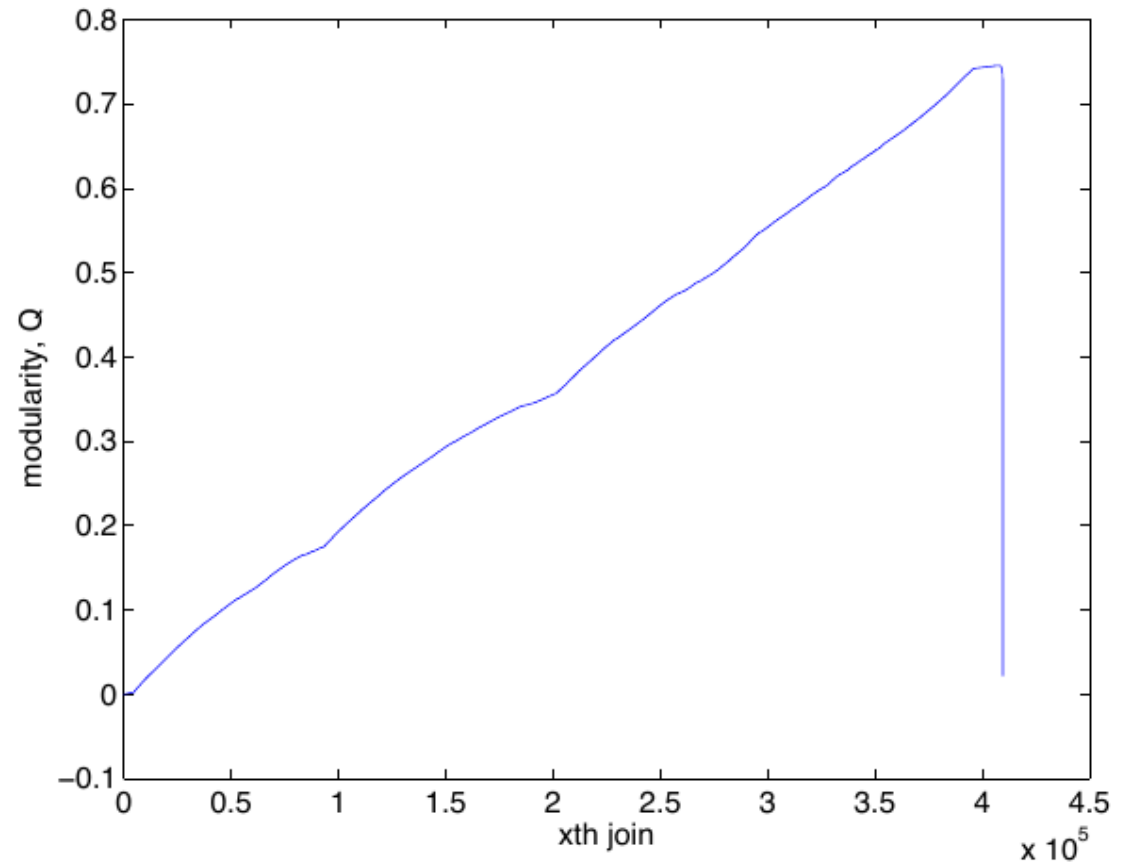


FIG. 1: The modularity Q over the course of the algorithm (the x axis shows the number of joins). Its maximum value is $Q = 0.745$, where the partition consists of 1684 communities.

Top comunidades = 87% dos nós

Rank	Size	Description
1	114538	General interest: politics; art/literature; general fiction; human nature; technical books; how things, people, computers, societies work, etc.
2	92276	The arts: videos, books, DVDs about the creative and performing arts
3	78661	Hobbies and interests I: self-help; self-education; popular science fiction, popular fantasy; leisure; etc.
4	54582	Hobbies and interests II: adventure books; video games/comics; some sports; some humor; some classic fiction; some western religious material; etc.
5	9872	classical music and related items
6	1904	children's videos, movies, music and books
7	1493	church/religious music; African-descent cultural books; homoerotic imagery
8	1101	pop horror; mystery/adventure fiction
9	1083	jazz; orchestral music; easy listening
10	947	engineering; practical fashion

TABLE I: The 10 largest communities in the Amazon.com network, which account for 87% of the vertices in the network.

Uso da modularidade máxima como critério possui alguns problemas

Não é uma medida perfeita

Força comunidades pequenas em comunidades grandes.

Nem sempre isso reflete a realidade.

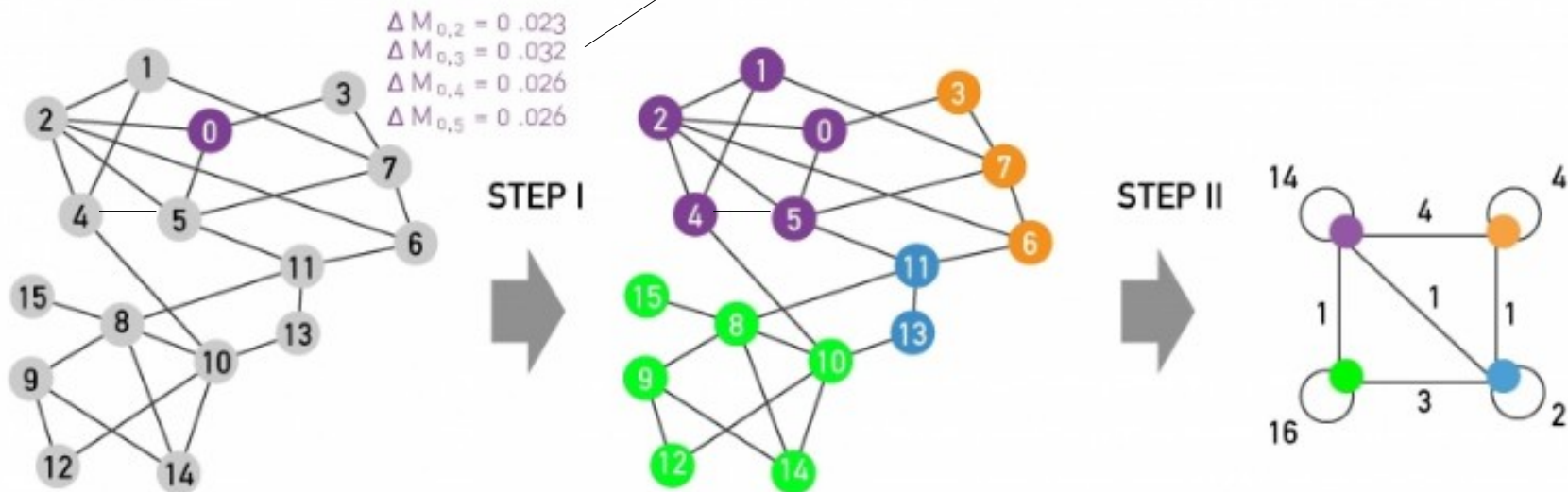
O método Louvain pode ser mais eficiente e preciso.

- **Etapa 1:** para cada nó i , considere os vizinhos (j) e avalie o ganho na modularidade da comunidade se mudar para a comunidade de j .
 - Faça isso para todos os nós.
 - Pare quando nenhuma melhoria puder ser alcançada.
- **Etapa 2:** veja cada comunidade criada como um nó e conecte-as com as links (pode ser ponderado)
 - repita a etapa 1 na rede de comunidades (que agora são os nós).
 - Pare quando nenhum aumento de modularidade for obtido.

Inicialmente cada nó está em uma comunidade – inicia repetições do step I e II

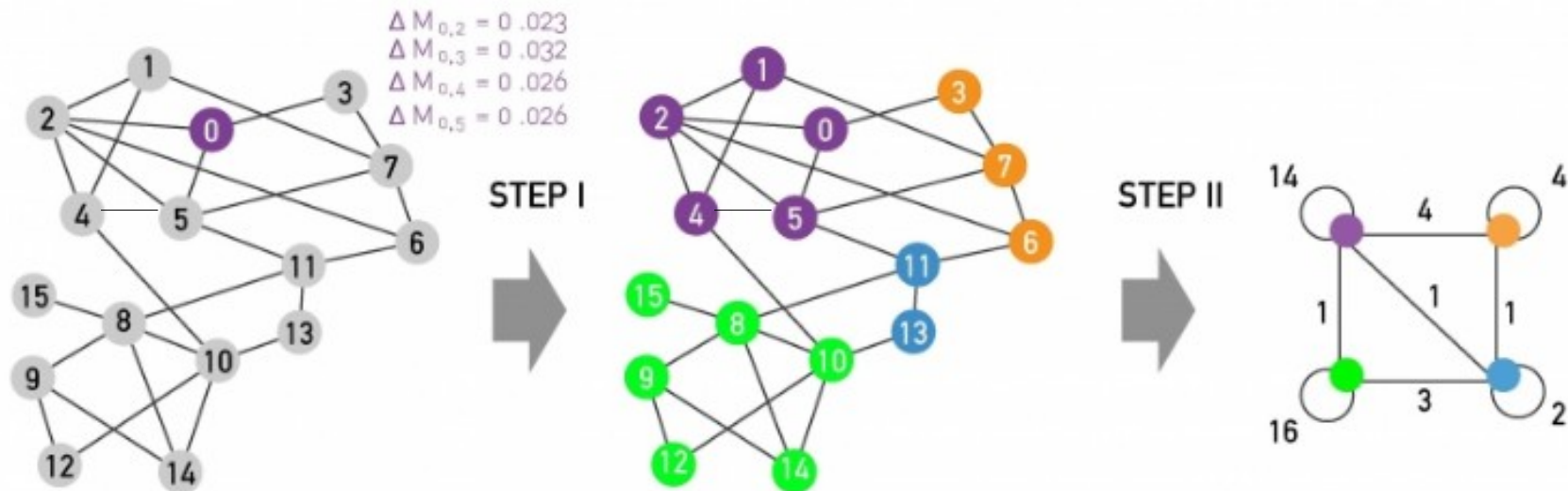
Inicialmente cada nó está em uma comunidade – inicia repetições do step I e II

1ST PASS

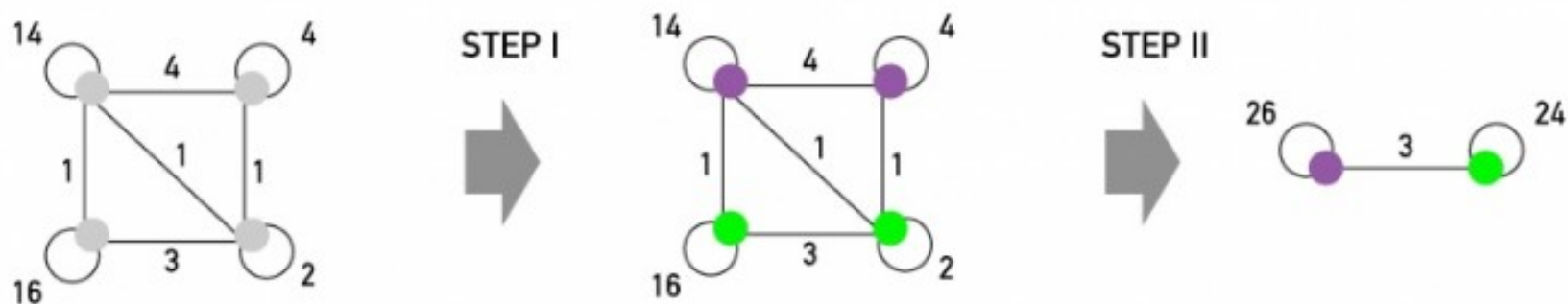


Inicialmente cada nó está em uma comunidade – inicia repetições do step I e II

1ST PASS



2ND PASS



Para quando não tiver mais ganho em termos de modularidade

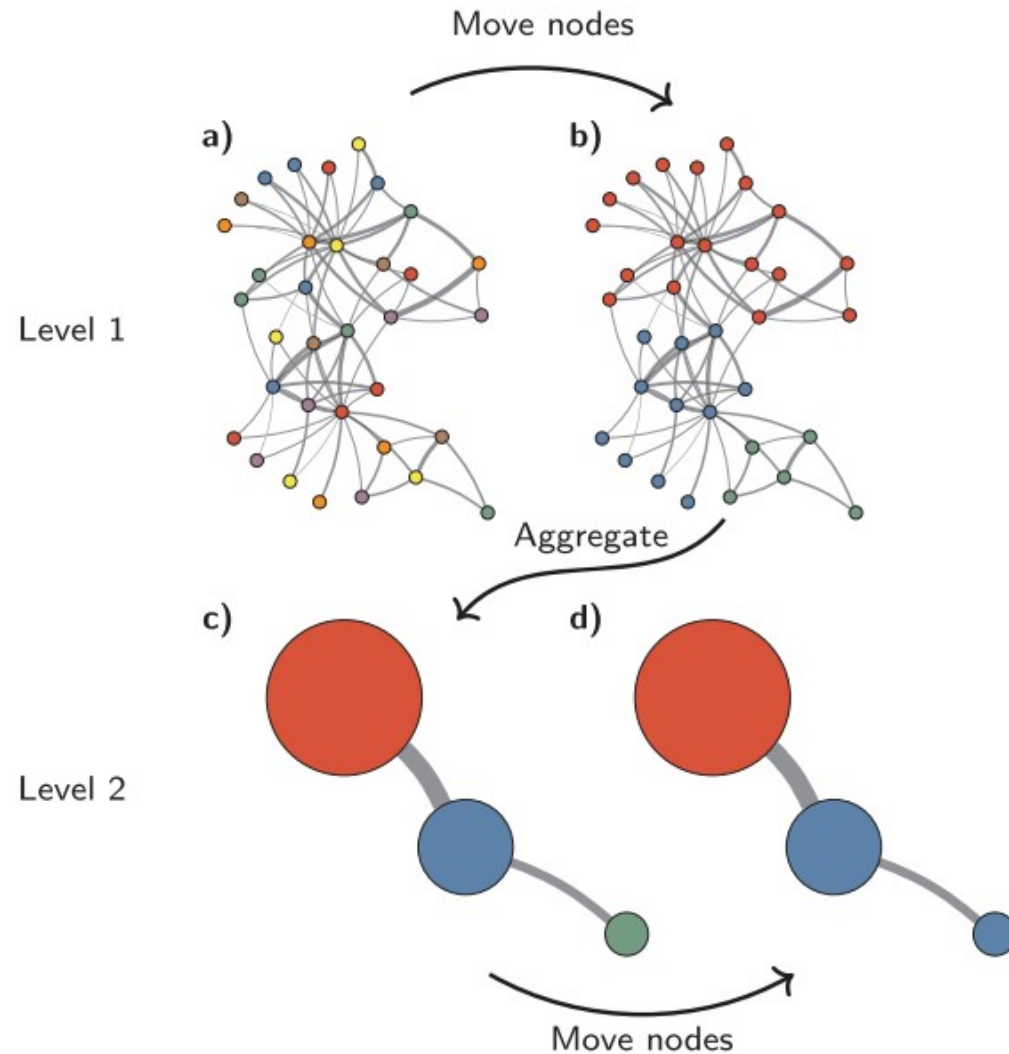


Figure 1. Louvain algorithm. The Louvain algorithm starts from a singleton partition in which each node is in its own community (a). The algorithm moves individual nodes from one community to another to find a partition (b). Based on this partition, an aggregate network is created (c). The algorithm then moves individual nodes in the aggregate network (d). These steps are repeated until the quality cannot be increased further.

Louvain pode encontrar comunidades ruins (como em 2b)

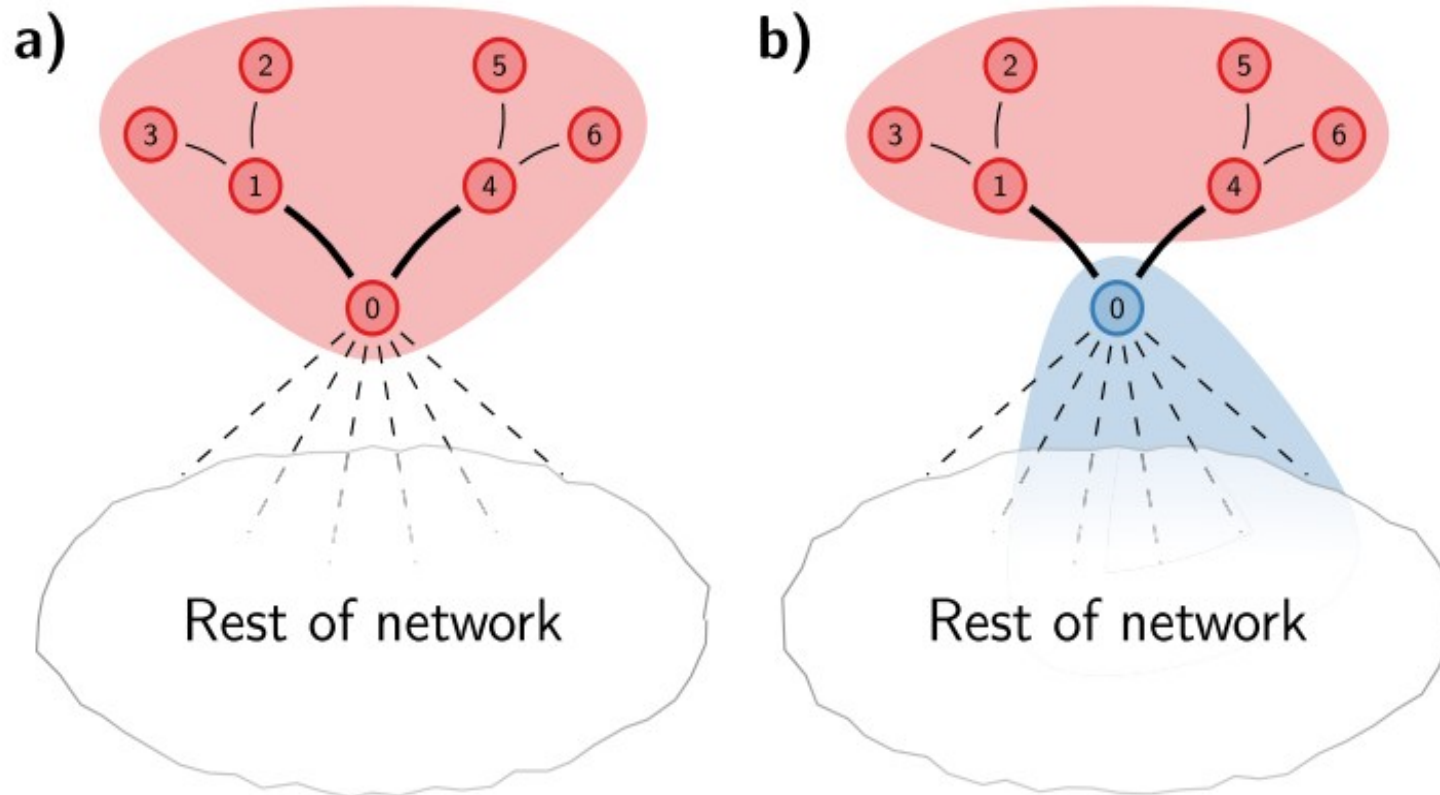


Figure 2. Disconnected community. Consider the partition shown in (a). When node 0 is moved to a different community, the red community becomes internally disconnected, as shown in (b). However, nodes 1–6 are still locally optimally assigned, and therefore these nodes will stay in the red community.

Traag, V. A., Waltman, L., & Van Eck, N. J. (2019). From Louvain to Leiden: guaranteeing well-connected communities. *Scientific reports*, 9(1), 1-12.
<https://www.nature.com/articles/s41598-019-41695-z.pdf>

Leiden visa contornar esse problema.

Refinamento:
comunidades
podem ser
divididas

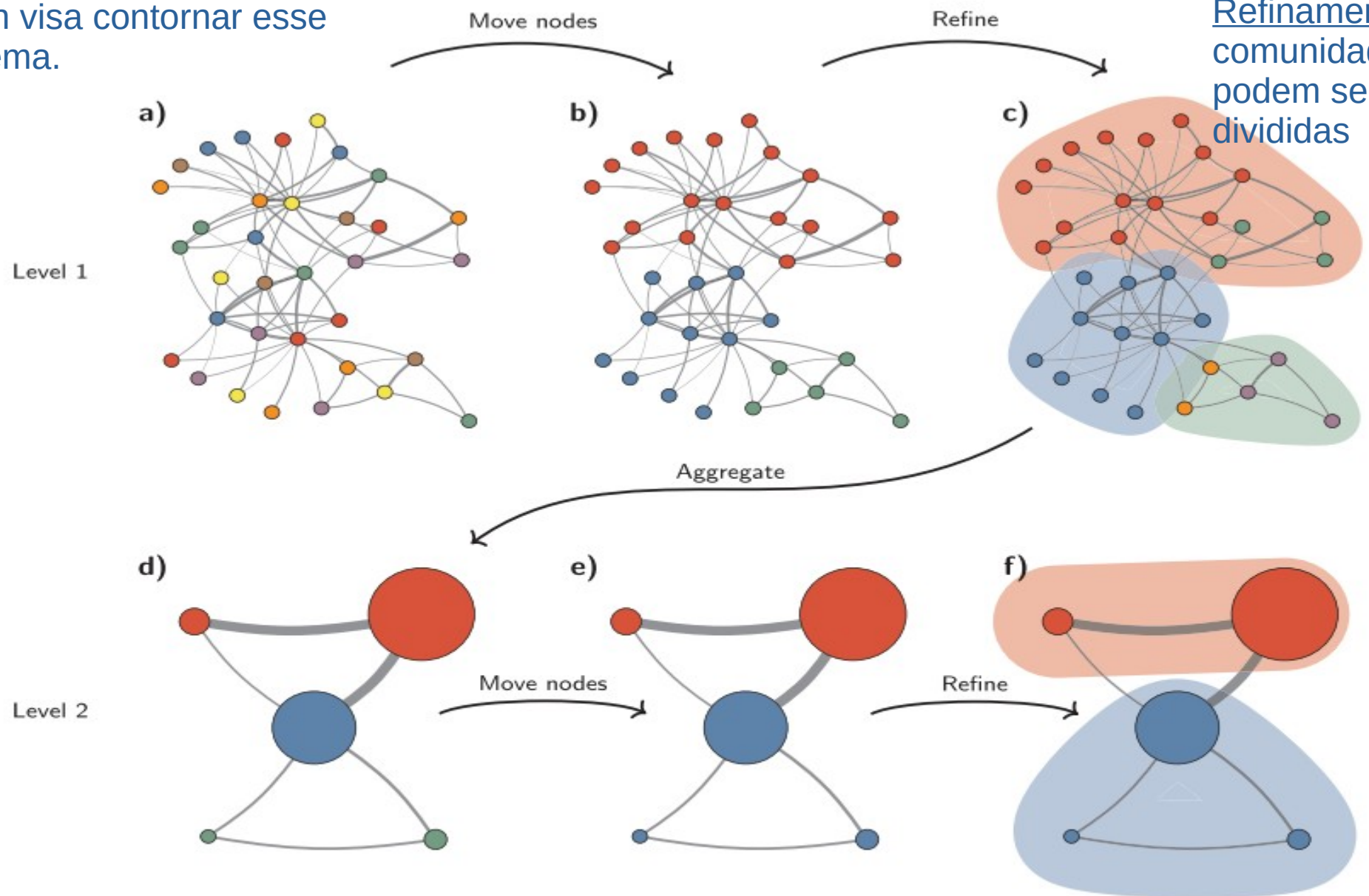
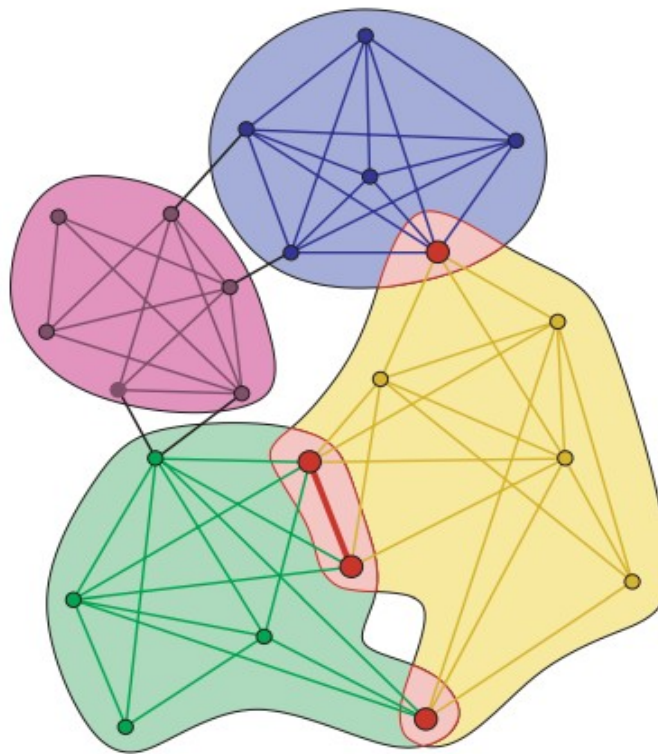


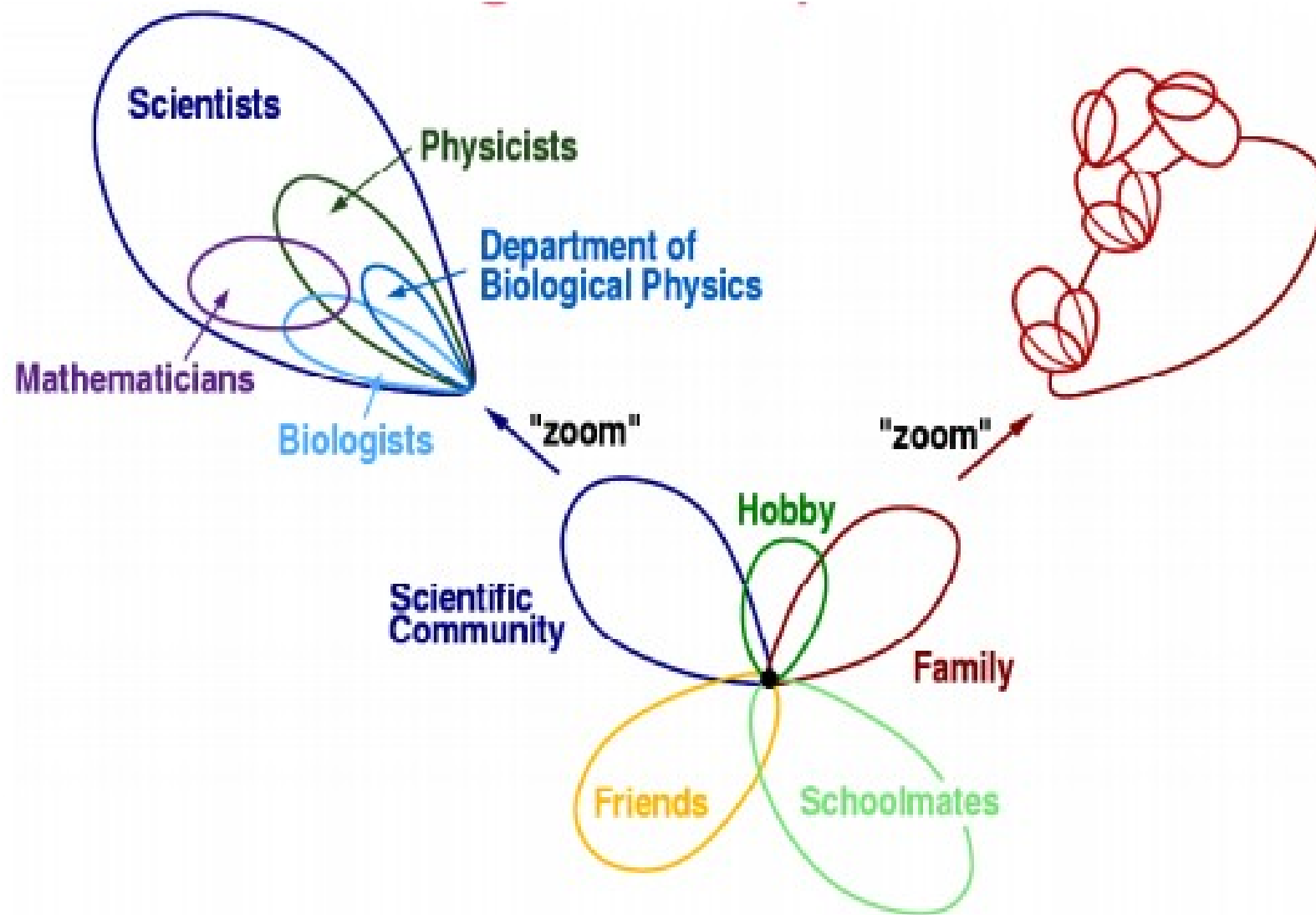
Figure 3. Leiden algorithm. The Leiden algorithm starts from a singleton partition (a). The algorithm moves individual nodes from one community to another to find a partition (b), which is then refined (c). An aggregate network (d) is created based on the refined partition, using the non-refined partition to create an initial partition for the aggregate network. For example, the red community in (b) is refined into two subcommunities in (c), which after aggregation become two separate nodes in (d), both belonging to the same community. The algorithm then moves individual nodes in the aggregate network (e). In this case, refinement does not change the partition (f). These steps are repeated until no further improvements can be made.

Pode ocorrer sobreposição entre membros comunidades

Um nó pode ser parte de mais de uma comunidade



Mais de um círculo social



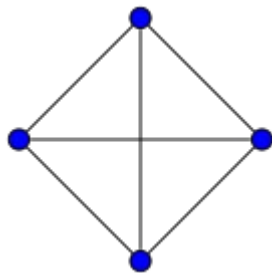
Palla 2005 – Implementado no networkx

Dois nós pertencem à mesma comunidade se eles podem ser conectados através de k -cliques adjacentes

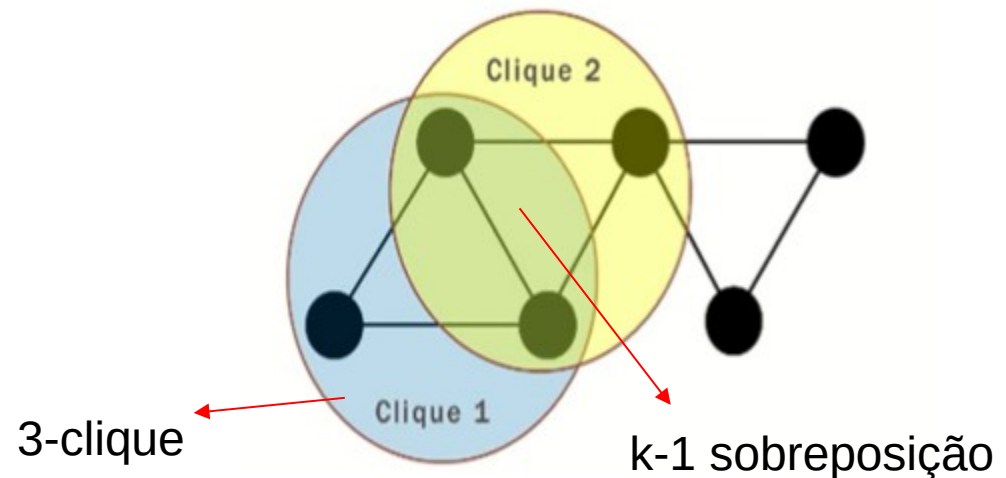
Um k -clique é um grafo completo de k nós

K -cliques são adjacentes se eles tem $k-1$ nós sobrepostos

Comunidade K -clique: nós que podem ser alcançados por uma sequência de k -cliques adjacentes



4-clique



Algoritmo:

- **Encontrar as cliques maximais**

- clique que não pode ser estendido incluindo um novo nó adjacente
- é complexo, mas redes reais são relativamente esparsas

- **Construir uma matriz de cliques sobrepostos**

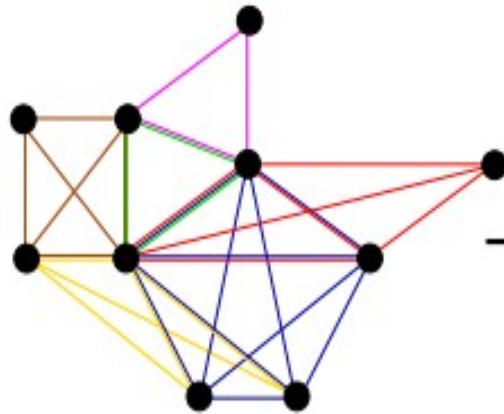
- cada clique é um nó
- conectar dois cliques se eles tem a sobreposição de pelo menos $k-1$ nós

- **Comunidades:**

- componentes conectados dessa matriz

Clique percolation method (CFinder)

Cliques
maximais



	Blue	Red	Green	Magenta	Yellow	Brown
Blue	5	3	2	1	3	1
Red	3	4	2	1	1	1
Green	2	2	3	2	1	2
Magenta	1	1	2	3	0	1
Yellow	3	1	1	0	4	2
Brown	1	1	2	1	2	4

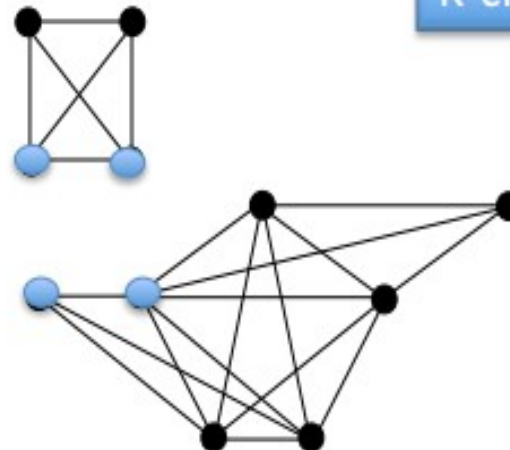
Matriz de
sobreposição:
Células são os
números de nós
sobrepostos

k=4

Remover
células < 4 na
diagonal ou <
3 em outros
locais

	Blue	Red	Green	Magenta	Yellow	Brown
Blue	1	1	0	0	1	0
Red	1	1	0	0	0	0
Green	0	0	0	0	0	0
Magenta	0	0	0	0	0	0
Yellow	1	0	0	0	1	0
Brown	0	0	0	0	0	1

K-cliques



- M. Newmann. Networks. Oxford University Press. April 2010
- V.D. Blondel et al, J. Stat. Mech. P10008 (2008).
- Kleinberg's book: Chapter 2.