Avaliação Distribuída de Centralidade em Redes Complexas

Klaus Wehmuth

Mestrado em Modelagem Computacional
Laboratório Nacional de Computação Científica (LNCC)
Av, Getúlio Vargas, 333 – 25651-075 – Petrópolis, RJ, Brasil
Email: klaus@lncc.br

Orientador: Artur Ziviani (LNCC/MCTI)

Abstract. *Among the different centrality definitions, the traditional closeness centrality ranks the nodes by how close each node is to all other nodes in the network. Nevertheless, computing closeness centrality in large complex networks is costly. Here, we present a fully distributed method capable of yielding different kinds of centrality, among them one which node ranking correlates strongly with the closeness centrality ranking, but being cheaper than the traditional algorithm and not requiring full knowledge of the network's topology.*

Resumo. *Entre os diferentes tipos de centralidade, Closeness Centrality é uma das mais tradicionais e afere a importância de cada nó pela sua proximidade com todos os demais nós da rede. Entretanto, o cálculo desse tipo de centralidade apresenta um alto custo computacional. Por disso, o seu uso se torna impraticável para redes de grande porte. Assim, este trabalho apresenta um método distribuído que pode ser utilizado para calcular vários tipos de centralidades, entre elas uma cuja ordenação dos nós tem um alto grau de correlação com a ordenação obtida pelo uso de Closeness Centrality. O método proposto funciona de maneira totalmente distribuída, baseando-se em conhecimento local dispensando o conhecimento completo da topologia da rede, e é computacionalmente menos custoso que o método tradicional.*

1. Introdução

Existem várias noções diferentes de centralidade, onde em cada uma delas os nós ou arestas de uma rede recebem valores que refletem sua importância. Entretanto, em geral, é mais relevante identificar quais são os elementos mais importantes em uma rede do que conhecer o seu valor de importância dado por uma determinada métrica. Entre as diversas formas de se determinar a importância de elementos em uma rede, algumas são mais tradicionalmente conhecidas e aplicadas, tais como *betweeness centrality* e *closeness centrality*. Essas formas de centralidade, apesar de bastante aplicadas, são computacionalmente custosas, tornando sua aplicação na análise de redes de grande porte, em muitos casos, inviável. Esse problema motiva o estudo de vários métodos para otimizar o cálculo dessas centralidades, bem como o surgimento de aproximações que sejam menos custosas.

Esta dissertação de mestrado apresentou um novo método distribuído para calcular centralidades em redes, chamado DANCE (*Distributed Assessment of Network CEntrality*). O tipo de centralidade calculado por esse algoritmo é determinado por uma função denominada classificador que atribui os valores de centralidade a cada um dos nós da

rede, fazendo com que através do uso de diferentes classificadores seja possível obter diferentes tipos de centralidade. O algoritmo DANCE também permite que estas centralidades sejam calculadas de forma distribuída com informação localizada, sem necessitar do conhecimento completo da topologia da rede. Para executar o algoritmo, basta que cada nó da rede tenha conhecimento de seus vizinhos diretos e seja capaz de trocar mensagens com eles. A implementação do algoritmo também é possível em ambientes onde a topologia da rede é conhecida e a mesma é representada como uma estrutura de grafo armazenada em memória. No caso da implementação sem uso do conhecimento da topologia da rede, o algoritmo provê ainda um mecanismo para identificação e localização dos nós de máximo de centralidade, de forma que estes possam ser identificados numa rede onde não se tem o conhecimento da identidade de todos os seus nós.

A efetividade e aplicabilidade do algoritmo DANCE é verificada pela obtenção de um algoritmo que permite que se obtenha uma aproximação da ordenação de nós obtida por *closeness centrality*, porém com um custo computacional suficientemente baixo para permitir sua utilização em redes de grande porte. Este algoritmo também é comparado com um outro algoritmo distribuído recentemente apresentado, baseado em um caminho aleatório perpétuo. Verifica-se que o algoritmo DANCE pode obter um resultado semelhante com um custo de mensagens e tempo de convergência inferiores aos obtidos com o algoritmo baseado em caminhos aleatórios.

2. Algoritmo DANCE

O conceito básico utilizado na construção do algoritmo DANCE é que uma vizinhança com um determinado raio h deve ser encontrada em torno de cada nó da rede. Uma vez que estas vizinhanças tenham sido determinadas, a centralidade de cada nó é calculada de forma localizada levando em conta apenas a vizinhança obtida em torno deste nó. Dada uma rede $G = (V, E)$, onde V é seu conjunto de nós e E o conjunto de arestas, a vizinhança de raio h em torno de um determinado nó i é definida como o objeto $H_h^i = (V_h^i, E_h^i)$ onde V_h^i é o subconjunto de V contendo todos os nós cuja distância ao nó i seja menor ou igual à h e E_h^i é o subconjunto de E contendo todas as arestas incidentes a pelo menos um nó de V_h^i. A Figura 1 mostra exemplos de vizinhanças H_h^b com raio $h = \{0, 1, 2\}$ para o nó b marcado em preto.

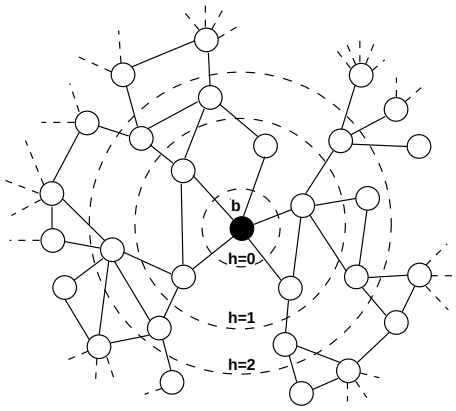


Figura 1. Vizinhanças H_0^b, H_1^b e H_2^b.

Uma vez determinadas as vizinhanças de cada um dos nós, o valor de centralidade associado a cada um deles é calculada por meio de uma função denominada classificador,

que leva do conjunto de todas as vizinhanças da rede no conjunto dos números reais. No caso da centralidade apresentada nesse trabalho, o classificador apenas soma o grau de cada um dos nós da vizinhança, obtendo assim seu volume. Este classificador tem um custo computacional bastante baixo, fazendo com que o custo total do cálculo da centralidade seja dominado pelo custo da determinação da vizinhança de cada nó, fazendo com que o custo computacional para a obtenção dessa centralidade seja baixo. Note ainda que, além de apresentar um baixo custo computacional, a centralidade de cada um dos nós da rede pode ser calculada de forma independente, fazendo com que o algoritmo proposto seja apropriado para implementações paralelas ou distribuídas.

3. Principais resultados

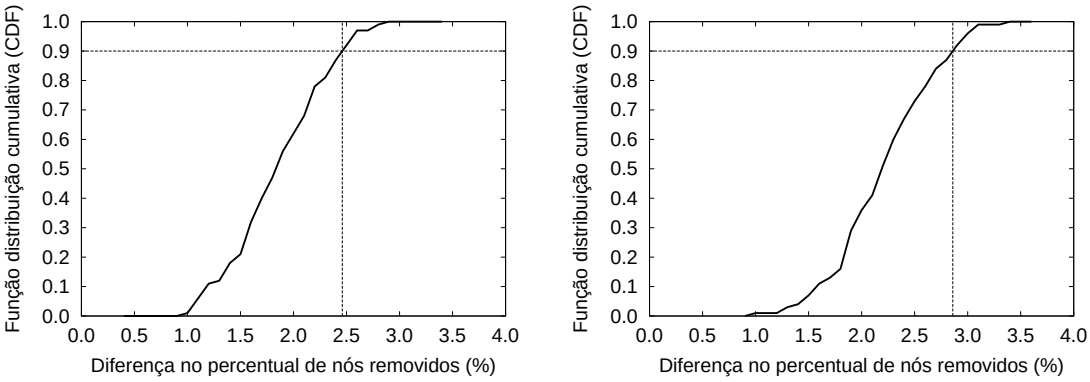
Uma análise completa dos resultados resumidos nas Seções 3.1 e 3.2 deste documento encontra-se, respectivamente, nas Seções 5.1 e 5.2 da dissertação. Resultados referentes à aplicabilidade em redes de grande porte encontram-se na Seção 5.3 da dissertação.

3.1. Comparação com o algoritmo SOC

A efetividade do algoritmo DANCE é comparada com outro algoritmo recentemente proposto chamado SOC (*Second Order Centrality*) [Kermarrec et al., 2011] para avaliação distribuída de centralidade que pode ser usado em redes onde a topologia é desconhecida. O algoritmo SOC propõem uma nova noção de centralidade, não tendo como objetivo aproximar nenhuma centralidade previamente conhecida. SOC baseia-se em um caminho aleatório perpétuo onde se mede o desvio padrão do intervalo de tempo entre visitas consecutivas do caminho aleatório a cada nó. Apesar de eficaz para avaliação distribuída de centralidade, SOC apresenta dificuldades para determinar o número de passos necessários para que se obtenha valores consistentes de centralidade para todos os nós da rede.

O critério de avaliação utilizado para a comparação foi o impacto da remoção dos nós de mais alta centralidade da rede sobre seu componente gigante. Esse método foi utilizado pelos autores do algoritmo SOC para avaliação do mesmo. Para essa avaliação foram consideradas 100 redes sintéticas de 1000 nós seguindo o modelo Barabási-Albert (BA) para redes de escala-livre e outras 100 redes de 1000 nós seguindo o modelo Erdös-Rényi (ER) para redes aleatórias. A partir de cada rede considerada, removeu-se sucessivamente o nó de maior centralidade indicado por cada algoritmo e avaliou-se o número de nós restantes no maior componente conexo resultante (componente gigante). Nesse experimento, cada rede foi considerada fragmentada quando o componente gigante ficou reduzido a menos que 20% do número inicial de nós da rede.

A Figura 2 apresenta a CDF da diferença entre o percentual necessário de nós removidos usando DANCE e SOC para fragmentação da rede (ou seja, redução do componente gigante a 20% do tamanho original). Observa-se na Figura 2(a) que em 90% dos casos de redes BA o algoritmo DANCE necessita de menos de 2,4% de nós removidos a mais que SOC para fragmentar a rede; em 99% dos casos essa diferença é inferior a 2,8%. Para redes ER (Figura 2(b)), essa diferença é inferior a 2,8% e 3,3% em 90% e 99% dos casos, respectivamente. DANCE, portanto, alcança consistentemente uma efetividade bastante similar ao SOC ao avaliar a centralidade de nós da rede distribuidamente. Porém, o custo computacional para a obtenção dos resultados com o algoritmo DANCE foi aproximadamente uma ordem de magnitude menor que o custo obtido com SOC, mostrando que o algoritmo DANCE obtém resultados similares com um custo bastante inferior.



(a) Redes BA (b) Redes ER

Figura 2. Diferença entre DANCE e SOC no percentual de nós removidos para fragmentação.

3.2. Correlação entre rankings de DANCE e *Closeness Centrality*

É avaliada a correlação obtida entre os *rankings* de nós obtidos pelo uso do algoritmo tradicional de *closeness centrality* e DANCE. São avaliadas as correlações obtidas tanto para redes sintéticas como também para registros (*traces*) de redes reais. Para avaliação em redes sintéticas, foram utilizadas 100 redes de 1000 nós construídas com o modelo BA e 100 redes de 1000 nós construídas com o modelo ER. Para cada uma dessas redes foram calculados os *rankings* de *closeness centrality* e DANCE utilizando raio $h = 2$. Em seguida, calculou-se a correlação de Pearson entre estes dois *rankings*.

Em todos os casos, observa-se uma forte correlação entre os *rankings* obtidos por meio de *closeness centrality* e DANCE redes sintéticas de diferentes tipos. Considerando-se o conjunto de todas as redes analisadas nesse experimento, a menor correlação obtida nas redes BA foi de $0,9972$ e para redes ER foi de $0,9962$, enquanto que a maior correlação obtida para redes BA foi $0,9986$ e para redes ER foi $0,9975$.

A Tabela 1 mostra as correlações obtidas utilizando DANCE com raio $h = 2$ para várias registros (*traces*) de redes de escala-livre (*scale-free*) reais. A rede *Actors* [Barabási e Albert, 1999] é uma rede social de atores de cinema, onde existe uma aresta entre atores que trabalharam em um mesmo filme. A rede *Routers-CAIDA* [CAIDA, 2003] é uma rede de roteadores da Internet onde os nós representam os roteadores e as arestas representam uma conexão entre eles. A rede *RouteViews* [Newman, 2006] é uma rede de sistemas autônomos da Internet reconstruída a partir de tabelas BGP, onde os nós representam os sistemas autônomos e as arestas representam a existência de uma rota ligando os sistemas autônomos. A rede *PGP-net* [Boguna et al., 2004] é uma rede de usuários do algoritmo *Pretty-Good-Privacy* onde os nós representam os usuários e as arestas representam as trocas de senhas entre os usuários. A rede *Yeast* [Jeong et al., 2001] é uma rede de proteínas e a rede *US Airports* [Colizza et al., 2007] é uma rede onde os nós representam os 500 aeroportos comerciais mais movimentados dos Estados Unidos e as arestas representam a existência de um voo comercial entre os aeroportos durante o ano de 2002. Como pode ser visto na Tabela 1, a correlação obtida para todos os casos foi alta, mostrando assim a efetividade do algoritmo DANCE para uso em redes reais de diversos tamanhos. É importante notar que esses resultados foram obtidos utilizando raio $h = 2$ para a determinação das vizinhanças, o que limita a complexidade de mensagens do algoritmo.

Tabela 1. Correlação de *ranking* para redes reais.

Rede	# de nós	# de arestas	raio	correlação
Actors	374.511	15.014.850	4	0,9537
Routers-CAIDA	190.914	607.610	13	0,9066
RouteViews	22.693	48.436	6	0,9954
PGP-net	10.680	24.316	12	0,8704
Yeast	1.458	1.993	11	0,9568
US Airports	500	2.980	4	0,9747

4. Conclusão

O algoritmo DANCE para avaliação de centralidades com base em vizinhanças de raio limitado proposto neste trabalho formaliza e generaliza os conceitos de ego-centralidade investigados de maneira empírica em trabalhos anteriores, tais como Marsden [2002]; Everett e Borgatti [2005]; Nanda e Kotz [2008] Além de possibilitar uma avaliação eficiente de centralidades em casos onde a topologia da rede é totalmente conhecida, DANCE também pode ser aplicado de forma distribuída visando a avaliação de centralidade em casos onde a topologia não é totalmente conhecida, bastando que cada nó da rede conheça seus vizinhos diretos. Nesse caso, DANCE ainda proporciona um método para localizar o nó de maior centralidade da rede como um todo e ainda os nós de máximo locais de centralidade com garantia de um espaçamento mínimo entre eles, determinado pelo raio das vizinhanças utilizadas. Essa possibilidade é interessante para aplicação em redes onde a topologia ou até mesmo a identidade de todos os nós não é conhecida, fazendo com que mesmo nessas condições seja possível identificar os nós mais relevantes.

Os resultados obtidos mostram que é possível utilizar o algoritmo proposto neste trabalho para obter uma forma de centralidade que tenha utilidade prática e aplicabilidade na análise de redes complexas de larga escala atualmente encontradas em várias áreas do conhecimento. A centralidade obtida utilizando DANCE com um classificador que calcula o volume de cada vizinhança permite obter uma aproximação da ordenação dos nós obtida por *closeness centrality* com um custo significativamente inferior ao do algoritmo tradicional de *closeness centrality*. Isso possibilita a aplicação dessa forma de centralidade a redes de grande porte, onde o custo computacional do cálculo de *closeness centrality* pelo algoritmo tradicional é excessivamente elevado.

A elaboração desta dissertação propiciou a publicação de alguns trabalhos que cobrem diferentes aspectos da proposta: Wehmuth e Ziviani [2011b] (premiado como melhor artigo do evento WPerformance 2011); Wehmuth e Ziviani [2011a]; Wehmuth e Ziviani [2012b]; Wehmuth e Ziviani [2012a]; e Wehmuth e Ziviani [2013].

O algoritmo proposto nesta dissertação foi estendido posteriormente à defesa para suportar outras funções classificadoras, dando origem a outras formas de centralidade distribuídas, assim como permitindo a implementação de formas já conhecidas como por exemplo *ego-betweenness* (Everett e Borgatti [2005]). Essa versão estendida foi implementada para utilização em ambientes computacionais de alto desempenho e encontra-se disponível para utilização pela comunidade científica no portal DANCE (`http://www.lncc.br/sinapad/DANCE/`).

Agradecimentos

Este trabalho foi parcialmente financiado pela FAPERJ, CNPq e MCTI.

Referências

Barabási, A. e Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512.

Boguna, M., Pastor-Satorras, R., Diaz-Guilera, A., e Arenas, A. (2004). List of edges of the giant component of the network of users of the Pretty-Good-Privacy algorithm for secure information interchange. *Physical Review E*, 70(056122).

Brandes, U. e Pich, C. (2007). Centrality Estimation in Large Networks. *International Journal of Bifurcation and Chaos*, 17(07):2303.

CAIDA (2003). Internet Router-Level Topology Measurements. `http://www.caida.org/tools/measurement/skitter/router\_topology/`

Colizza, V., Pastor-Satorras, R., e Vespignani, A. (2007). Reaction-Diffusion processes and metapopulation models in heterogeneous networks. *Nature Physics*, 3:276–282.

Everett, M. e Borgatti, S. (2005). Ego network betweenness. *Social Networks*, 27(1):31–38.

Jeong, H., Mason, S. P., Barabasi, A.-L., e Oltvai, Z. N. (2001). Lethality and centrality in protein networks. *Nature*, 411(6833):41–42.

Kermarrec, A.-M., Le Merrer, E., Sericola, B., e Trédan, G. (2011). Second order centrality: distributed assessment of nodes importance in complex networks. *Computer Communications*, 34(5):619–628.

Marsden, P. (2002). Egocentric and sociocentric measures of network centrality. *Social Networks*, 24(4):407–422.

Nanda, S. e Kotz, D. (2008). Localized Bridging Centrality for Distributed Network Analysis. *ICCCN 17th International Conference on Computer Communications and Networks*, pages 1–6.

Newman, M. (2006). Internet – a symmetrized snapshot of the structure of the internet at the level of autonomous systems. `http://www-personal.umich.edu/~mejn/netdata/`

Wehmuth, K. e Ziviani, A. (2011a). Distributed location of the critical nodes to network robustness based on spectral analysis. In *2011 7th Latin American Network Operations and Management Symposium*, pages 1–8, Quito, Ecuador. IEEE.

Wehmuth, K. e Ziviani, A. (2011b). Um Novo Algoritmo Distribuído para Avaliação e Localização de Centralidade de Rede. In *X Workshop em Desempenho de Sistemas Computacionais e de Comunicação (WPerformance 2011), XXXI Congresso Nacional da Sociedade Brasileira de Computação (CSBC)*, Natal, RN, Brasil.

Wehmuth, K. e Ziviani, A. (2012a). Distributed Assessment of Network Centralities in Complex Social Networks. In *International Workshop on Complex Social Network Analysis - CSNA 2012*, Istambul, Turkey.

Wehmuth, K. e Ziviani, A. (2012b). Distributed assessment of the closeness centrality ranking in complex networks. In *Proceedings of the Fourth Annual Workshop on Simplifying Complex Networks for Practitioners - SIMPLEX '12*, number c, page 43, Lyon, France. ACM Press.

Wehmuth, K. e Ziviani, A. (2013). DACCER: Distributed Assessment of the Closeness Centrality Ranking in Complex Networks. *Computer Networks*, Elsevier, aceito para publicação.