Game Theoretic Models for Social Network Analysis

Y. Narahari and Ramasuri Narayanam

Email ID: {hari,nrsuri}@csa.iisc.ernet.in
Electronic Commerce Laboratory
Department of Computer Science and Automation
Indian Institute of Science
Bangalore, India

28-March-2011

Outline of the Presentation

1. **Social Network Analysis: Quick Primer**
2. Foundational Concepts in Game Theory
3. Discovering Influential Individuals for Viral Marketing
4. Social Network Formation
5. Community Detection in Social Networks
6. Query Incentive Networks
7. Summary and To Probe Further

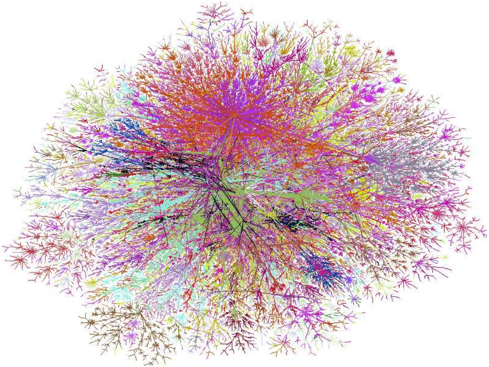
Social Networks: Introduction

Social networks are ubiquitous and have many applications:

- For targeted advertising
- Monetizing user activities on on-line communities
- Job finding through personal contacts
- Predicting future events
- E-commerce and e-business
- ...

M.S. Granovetter. The Strength of Weak Ties. American Journal of Sociology, 1973.

Example 1: Web Graph

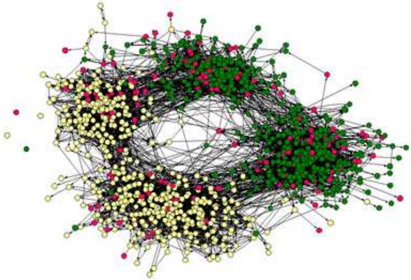


Nodes: Static web pages
Edges: Hyper-links

Reference: Prabhakar Raghavan. Graph Structure of the Web: A Survey. In Proceedings of LATIN, pages 123-125, 2000.

Example 2: Friendship Networks

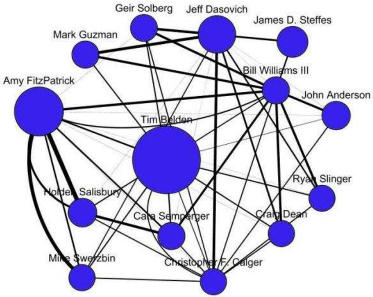
Friendship Network



Nodes: Friends
Edges: Friendship

Reference: Moody 2001

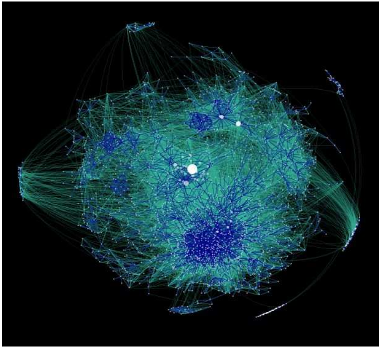
Subgraph of Email Network



Nodes: Individuals
Edges: Email Communication

Reference: Schall 2009

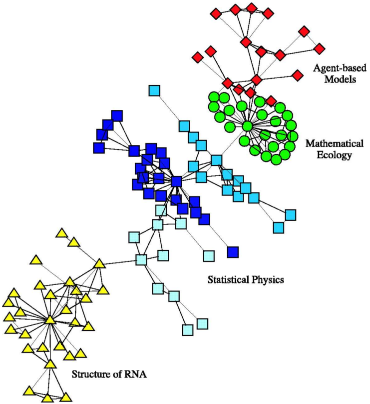
Example 3: Weblog Networks



Nodes: Blogs
Edges: Links

Reference: Hurst 2007

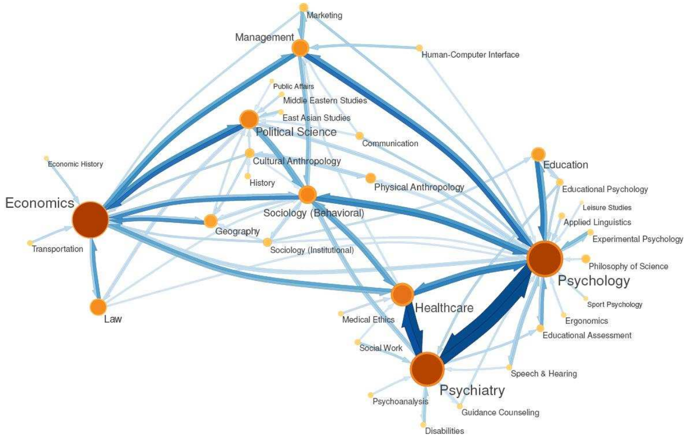
Example 4: Co-authorship Networks



Nodes: Scientists and Edges: Co-authorship

Reference: M.E.J. Newman. Coauthorship networks and patterns of scientific collaboration. PNAS, 101(1):5200-5205, 2004

Example 5: Citation Networks



Nodes: Journals and Edges: Citation

Reference: http://eigenfactor.org/

Social Networks - Definition

- *Social Network:* A social system made up of individuals and interactions among these individuals
- Represented using graphs
  - Nodes - Friends, Publications, Authors, Organizations, Blogs, etc.
  - Edges - Friendship, Citation, Co-authorship, Collaboration, Links, etc.

S.Wasserman and K. Faust. Social Network Analysis. Cambridge University Press, Cambridge, 1994

Social Network Analysis (SNA)

- Study of structural and communication patterns
  – degree distribution, density of edges, diameter of the network

- Two principal categories:
  - **Node/Edge Centric Analysis:**
    - Centrality measures such as degree, betweeneness, stress, closeness
    - Anomaly detection
    - Link prediction, etc.
  - **Network Centric Analysis:**
    - Community detection
    - Graph visualization and summarization
    - Frequent subgraph discovery
    - Generative models, etc.

U. Brandes and T. Erlebach. Network Analysis: Methodological Foundations. Springer-Verlag Berlin Heidelberg, 2005.

Why is SNA Important?

- To understand complex connectivity and communication patterns among individuals in the network
- To determine the structure of networks
- To determine influential individuals in social networks
- To understand how social network evolve
- To determine outliers in social networks
- To design effective viral marketing campaigns for targeted advertising
- ...

What are Key Issues in SNA?

- **Measures to Rank Nodes:**
  - *Degree Centrality:* It is defined as the number of links incident upon a node.
  - *Clustering Coefficient:* It measures how dense is the neighborhood of a node.
  - *Between Centrality:* It is a measure of a node and a node that occurs on many shortest paths between other pairs of nodes has higher betweenness centrality.
  - *Closeness Centrality:* It is defined as the mean shortest distance between a vertex and all other vertices reachable from it.
  - *Eigenvector Centrality:* It assigns relative scores to all nodes in the network based on the principle that connections to high-scoring nodes contribute more to the score of the node in question than equal connections to low-scoring nodes.
  - ...

What are Key Issues in SNA? (Cont.)

- **Diversity among Nodes:**
  - Nodes in the network might be having various connectivity patterns
  - Some nodes might be connected to high degree nodes, some others might be connected to bridge nodes, etc.
  - Determining diversity among the connectivity patterns of nodes is an interesting problem
  - L. Liu, F. Zhu, C. Chen, X. Yan, J. Han, P.S. Yu, and S. Yang. Mining Diversity on Networks. In DASFAA 2010.

What are Key Issues in SNA? (Cont.)

- **Link Prediction Problem:**
  - Given a snapshot of a social network, can we infer which new interactions among its members are likely to occur in the near future?
  - D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In CIKM 2003.

What are Key Issues in SNA? (Cont.)

- **Inferring Social Networks from Social Events:**
  - In the traditional link prediction problem, a snapshot of a social network is used as a starting point to predict (by means of graph-theoretic measures) the links that are likely to appear in the future.
  - Predicting the structure of a social network when the network itself is totally missing while some other information (such as interest group membership) regarding the nodes is available.
  - V. Leroy, B. Barla Cambazoglu, F. Bonchi. Cold start link prediction. In SIGKDD 2010.

What are Key Issues in SNA? (Cont.)

- **Viral Marketing:**
  - With increasing popularity of online social networks, viral Marketing - the idea of exploiting social connectivity patterns of users to propagate awareness of products - has got significant attention
  - In viral marketing, within certain budget, typically we give free samples of products (or sufficient discounts on products) to certain set of influential individuals and these individuals in turn possibly recommend the product to their friends and so on
  - It is very challenging to determine a set of influential individuals, within certain budget, to maximize the volume of information cascade over the network
  - P. Domingos and M. Richardson. Mining the network value of customers. In ACM SIGKDD, pages 5766, 2001.

What are Key Issues in SNA? (Cont.)

- **Community Detection:**
  - *Based on Link Structure in the Social Network:*
    - Determining dense subgraphs in social graphs
    - Graph partitioning
    - Determining the best subgraph with maximum number of neighbors
    - Overlapping community detection
  - Based on Activities over the Social Network
    - Determine action communities in social networks
    - Overlapping community detection
  - J. Leskovec, K.J. Lang, and M.W. Mahoney. Empirical comparison of algorithms for network community detection. In WWW 2010.
  - A.S. Maiya and T.Y. Berger-Wolf. Expansion and search in networks. In CIKM 2010.

What are Key Issues in SNA? (Cont.)

- **Query Incentive Networks:**
  - With growing number of online social communities, users pose queries to the network itself, rather than posing queries to a centralized system.
  - At present, the concept of incentive based queries is used in various question-answer networks such as Yahoo! Answers, Orkuts Ask Friends, etc.
  - In the above contexts, only the person who answers the query is rewarded, with no reward for the intermediaries. Since individuals are often rational and intelligent, they may not participate in answering the queries unless some kind of incentives are provided.
  - It is also important to consider the quality of the answer to the query, when incentives are involved.
  - J. Kleinberg and P. Raghavan. Query incentive networks. In Proceedings of 46th IEEE FOCS, 2005.
  - D. Dixit and Y. Narahari. Truthful and quality conscious query incentive networks. In WINE 2009.

What are Key Issues in SNA? (Cont.)

- **Determining the Implicit Social Hierarchy:**
  - Social stratification refers to the hierarchical classification of individuals based on power, position, and importance
  - The popularity of online social networks presents an opportunity to study social hierarchy for different types of large scale networks
  - M. Gupte, P. Shankar, J. Li, S. Muthukrishnan, and L. Iftode. Finding hierarchy in directed online social networks. In the Proceedings of World Wide Web (WWW) 2011.

How to Address Issues in SNA?

- Traditional Approaches
  - Graph theoretic techniques
  - Spectral methods
  - Optimization techniques
  - ...

How to Address Issues in SNA?

- Traditional Approaches
  - Graph theoretic techniques
  - Spectral methods
  - Optimization techniques
  - ...
- Recent Advances
  - Data mining and machine learning techniques
  - **Game theoretic techniques**

Next Part of the Talk

Contents to be Covered

- We first present a simple game theoretic model that brings out several aspects of viral marketing.
    - Noga Alon, Michal Feldman, Ariel D. Procaccia, and Moshe Tennenholtz. *A Note on Competitive Diffusion Through Social Networks.* Information Processing Letters, 110:221-225, 2010.
- We then bring out the challenges involved in viral marketing
- We discuss two standard models for propagation of influence in social networks and introduce the influence maximization problem
- Finally discuss a cooperative game theoretic model for influence maximization problem

Diffusion Game

- A game $\Gamma = (G, N)$ is induced by an undirected graph $G = (V, E)$, representing the underlying social network, and the set of agents N.
- The strategy space of each agent is the set of vertices V in the graph, that is, each agent i selects a single node and that node is colored in color i at time 1. We call them *initial trend setters*.
- Note that if two or more agents select the same vertex at time 1 then that vertex becomes gray.
- *Diffusion Process*: At time $t + 1$, each white vertex that has neighbors colored in color i, but does not have neighbors colored in color j for any $j \in N$, is colored in color i.

Diffusion Game (Cont.)

- A white vertex that has two neighbors colored by two distinct colors $i, j \in N$ is colored gray. That is, we assume that if two agents compete for a user at the same time, they cancel out and the user is removed from the game.

- The process continues until it reaches a fixed point, that is, all the remaining white vertices are unreachable due to gray vertices.

- A strategy profile is a vector $x = (x_1, x_2, \ldots, x_n)$, where $x_i \in V$ is the initial vertex selected by agent i. We also denote $x_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$

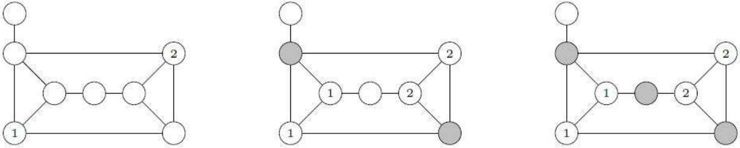
Diffusion Game (Cont.)

- Given a strategy profile $x \in V^n$, the utility of agent $i \in N$, denoted $U_i(x)$, is the number of nodes that are colored in color i when the diffusion process terminates.

- **Nash Equilibrium:** A strategy profile x is a (pure strategy) Nash equilibrium of the game Γ if an agent cannot benefit from unilaterally deviating to a diffusion strategy. That is, for every $i \in N$, and $x_i' \in V$, it holds that

$$U_i(x_i', x_{-i}) \leq U_i(x)$$

Diffusion Game - Example



(a) Time 1. (b) Time 2. (c) Time 3, the process terminates.

Diffusion Game - Nash Equilibrium

- If we can find a Nash equilibrium then we can often predict the behavior of the agents and the outcome of this competitive diffusion process.
- **Theorem:** Every game $\Gamma = (G, N)$, where $D(G) \leq 2$ admits a Nash equilibrium. Furthermore, an equilibrium can be found in polynomial time.
- **Theorem:** Let $N = \{1, 2\}$. There exists a graph G with $D(G) = 3$ such that the game $\Gamma = (G, N)$ does not admit a Nash equilibrium.

Challenges in Viral Marketing

- Propagation of influence is a stochastic process, but not a deterministic process
- The number of individuals in the social network that are getting influenced by the initial trend setters is an *expected quantity*
- Viral marketing for single or multiple products
- There can possibly exists certain types of dependencies among these products

Motivating Example 1: Viral Marketing

- Social networks play a key role for the spread of an innovation or technology
- We would like to market a new product that we hope will be adopted by a large fraction of the network
- Which set of the individuals should we target for?
- Idea is to initially target a few influential individuals in the network who will recommend the product to other friends, and so on
- A natural question is to find a target set of desired cardinality consisting of influential nodes to maximize the volume of the information cascade

Motivating Example 2: Weblogs

- In the domain of weblogs, bloggers publish posts and use hyper-links to refer to other posts and content on the web.
- Possible to observe the spread of information in the blogosphere, as each post is time stamped.
- In this setting, our goal is to select a small set of blogs (to read) which link to most of the stories that propagate over the blogosphere.

Models for Diffusion of Information

- Linear Thresholds Model
- Independent cascade model,

Models for Diffusion of Information

- **Linear threshold model**
- Independent cascade model,

Linear Thresholds Model

- Call a node active if it has adopted the information
- Initially every node is inactive
- Let us consider a node i and represent its neighbors by the set $N(i)$
- Node i is influenced by a neighbor node j according to a weight w_{ij}. These weights are normalized in such a way that

$$\sum_{j \in N(i)} w_{ij} \leq 1.$$

- Further each node i chooses a threshold, say θ_i, uniformly at random from the interval [0,1]
- This threshold represents the weighted fraction of node $i^{'}$s neighbors that must become active in order for node i to become active

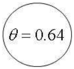
Given a random choice of thresholds and an initial set (call it S) of active nodes, the diffusion process propagates as follows:

- in time step t, all nodes that were active in step $(t-1)$ remain active
- we activate every node i for which the total weight of its active neighbors is at least θ_i
- if $A(i)$ is assumed to be the set of active neighbors of node i, then i gets activated if

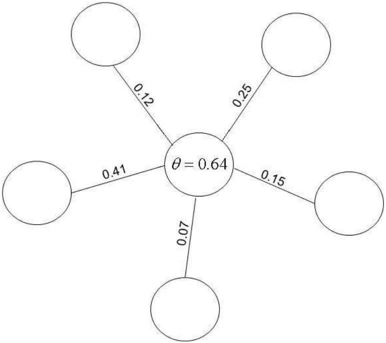
$$\sum_{j \in A(i)} w_{ij} \geq \theta_i.$$

- This process stops when there is no new active node in a particular time interval

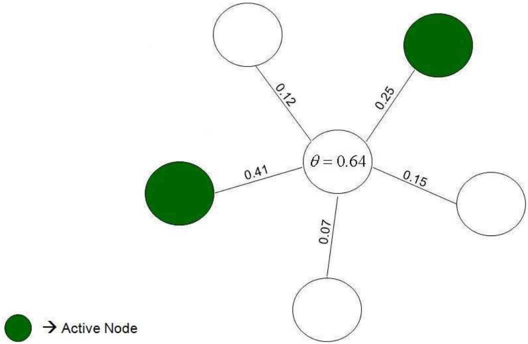
Linear Thresholds Model: An Example



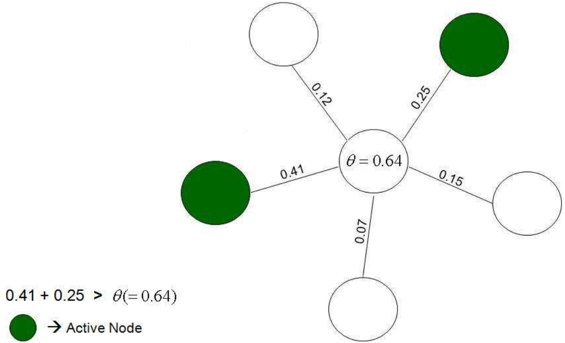
Linear Thresholds Model: An Example



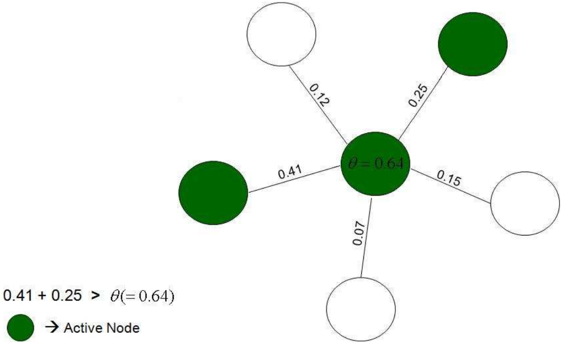
Linear Thresholds Model: An Example



Linear Thresholds Model: An Example



Linear Thresholds Model: An Example



Top-k Nodes Problem

- *Objective function ($\sigma(.)$):* Expected number of active nodes at the end of the diffusion process
- If S is the initial set of target nodes, then $\sigma(S)$ is the expected number of active nodes at the end of the diffusion process
- For economic reasons, we want to limit the size of S
- For a given constant k, the top-k nodes problem seeks to find a subset of nodes S of cardinality k that maximizes the expected value of $\sigma(S)$

Applications

- Databases
- Water Distribution Networks
- Blogspace
- Newsgroups
- Virus propagation networks

- R. Akbarinia, F.E. Pacitti, and F.P. Valduriez. Best Position Algorithms for Top-k Queries. In VLDB, 2007.
- J. Leskovec, A. Krause, and C. Guestrin. Cost-effective outbreak detection in networks. In ACM KDD, 2007.
- N. Agarwal, H. Liu, L. Tang, and P.S. Yu. Identifying influential bloggers in a community. In WSDM, 2008.

A Glimpse of State-of-the-Art

- P. Domingos and M. Richardson. Mining the network value of customers. In ACM SIGKDD, 2001.
  - Introduced this problem as an algorithmic problem
  - A model using Markov Ramom Fields
  - Show that selecting the right set of users for a marketing campaign can make a substantial difference.
- D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In SIGKDD, 2003.
  - Show that the optimization problem of selecting most influential nodes is NP-hard problem.
  - Show that this objective function is a sub-modular function.
  - Propose a greedy algorithm that achieves an approximation guarantee of $(1 - \frac{1}{e})$.

A Glimpse of State-of-the-Art (Cont.)

Greedy Algorithm - KKT (2003)

1. Set $A \leftarrow \phi$.
2. **for** $i = 1$ to k **do**
3. Choose a node $n_i \in N \setminus A$ maximizing $\sigma(A \cup \{n_i\})$
4. Set $A \leftarrow A \cup \{n_i\}$.
5. **end for**

- Running time of Greedy Algorithm: $O(knRm)$.

A Glimpse of State-of-the-Art (Cont.)

- J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In ACM SIGKDD, 2007.
  - Develop an efficient algorithm that is reportedly 700 times faster than the greedy algorithm (KKT (2003)).
  - There are two aspects to this speed up:
    1. Speeding up function evaluations using the sparsity of the underlying problem, and
    2. Reducing the number of function evaluations using the submodularity of the influence functions.

A Glimpse of State-of-the-Art (Cont.)

- W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In ACM SIGKDD, 2009.
- N. Chen. on the approximability of influence in social networks. In ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 1029-1037, 2008.
- M. Kimura and K. Saito. Tractable models for information diffusion in social networks. In PKDD, 2006.

Research Gaps

- All the existing approximation algorithms are sensitive to the value of k
- All the approximation algorithms crucially depend on the submodularity of the objective function. It is quite possible that the objective function can be non-submodular

Our Proposed Approach

- We present a cooperative game theoretic framework for the top-k nodes problem.
- We measure the influential capabilities of the nodes as provided by the Shapley value.
- ShaPley value based discovery of Influential Nodes (SPIN):
  1. Ranking the nodes,
  2. Choosing the top-k nodes from the ranking order.
- Advantages of SPIN:
  1. Quality of solution is same as that of popular benchmark approximation algorithms,
  2. Works well for both sub-modular and non-submodular objective functions,
  3. Running time is independent of the value of k.

Ranklist Construction

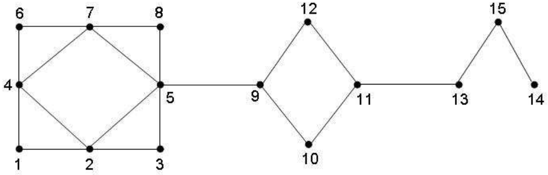
1. Let π_j be the j-th permutation in $\hat{\Omega}$ and R be repetitions.
2. Set $MC[i] \leftarrow 0$, for $i = 1, 2, \ldots, n$.
3. **for** $j = 1$ to t **do**
4.   Set $temp[i] \leftarrow 0$, for $i = 1, 2, \ldots, n$.
5.   **for** $r = 1$ to R, **do**
6.   assign random thresholds to nodes;
7.   **for** $i = 1$ to n, **do**
8. $temp[i] \leftarrow temp[i] + v(S_i(\pi_j) \cup \{i\}) - v(S_i(\pi_j))$
9.   **for** $i = 1$ to n, **do**
10. $MC[i] \leftarrow temp[i]/R$;
11. **for** $i = 1$ to n, **do**
12.   compute $\Phi[i] \leftarrow \frac{MC[i]}{t}$
13. Sort nodes based on the average marginal contributions of the nodes

Efficient Computation of Rank List

- Initially all nodes are inactive.
- Randomly assign a threshold to each node.
- Fix a permutation π and activate $\pi(1)$ to determine its influence.
- Next consider $\pi(2)$. If $\pi(2)$ is already activated, then the influence of $\pi(2)$ is 0. Otherwise, activate $\pi(2)$ to determine its influence.
- Continue up to $\pi(n)$.
- Repeat the above process R times (for example 10000 times) using the same π.
- Repeat the above process $\forall \pi \in \hat{\Omega}$.

Choosing Top-k Nodes

1. Naive approach is to choose the first k in the RankList[] as the top-k nodes.
2. *Drawback:* Nodes may be clustered.
3. RankList[]={5,4,2,7,11,15,9,13,12,10,6,14,3,1,8}.
4. Top 4 nodes, namely $\{5, 4, 2, 7\}$, are clustered.
5. Choose nodes:
    - rank order of the nodes
    - spread over the network



k value	*Greedy Algorithm*	*Shapley Value Algorithm*	*MDH based Algorithm*	*HCH*
1	4	4	4	2
2	8	7	7	4
3	10	10	8	6
4	12	12	8	7
5	13	13	10	8
6	14	14	13	8
7	15	15	13	8
8	15	15	13	8
9	15	15	13	10
10	15	15	13	11
11	15	15	13	13
12	15	15	13	13
13	15	15	14	14
14	15	15	15	15
15	15	15	15	15

Running Time of SPIN

- Overall running time of SPIN is $O(t(n+m)R + n\, log(n) + kn + kRm)$ where t is a polynomial in n.
- For all practical graphs (or real world graphs), it is reasonable to assume that $n < m$. With this, the overall running time of the SPIN is $O(tmR)$ where t is a polynomial in n.

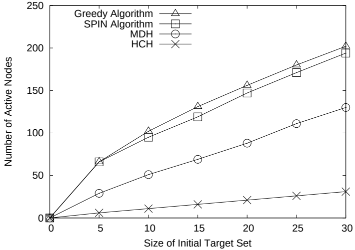
Experimental Results: Data Sets

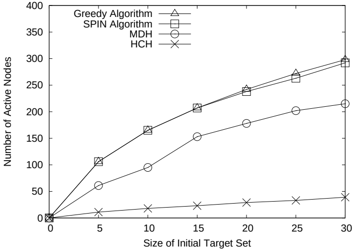
- Random Graphs
  - Sparse Random Graphs
  - Scale-free Networks (Preferential Attachment Model)
- Real World Graphs
  - Co-authorship networks,
  - Networks about co-purchasing patterns,
  - Friendship networks, etc.

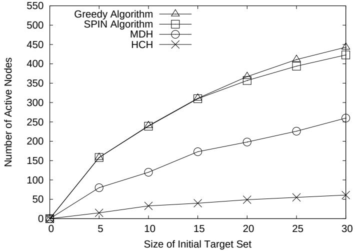
Experimental Results: Data Sets

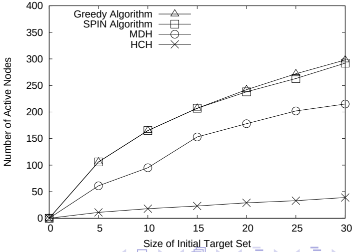
Dataset	Number of Nodes	Number of Edges
Sparse Random Graph	500	5000 (approx.)
Scale-free Graph	500	1250 (approx.)
Political Books	105	441
Jazz	198	2742
Celegans	306	2345
NIPS	1061	4160
Netscience	1589	2742
HEP	10748	52992

Experiments: Random Graphs

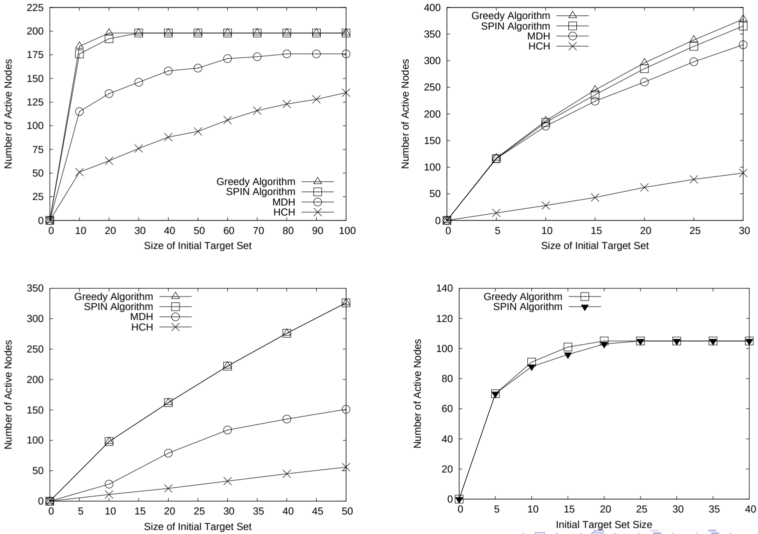




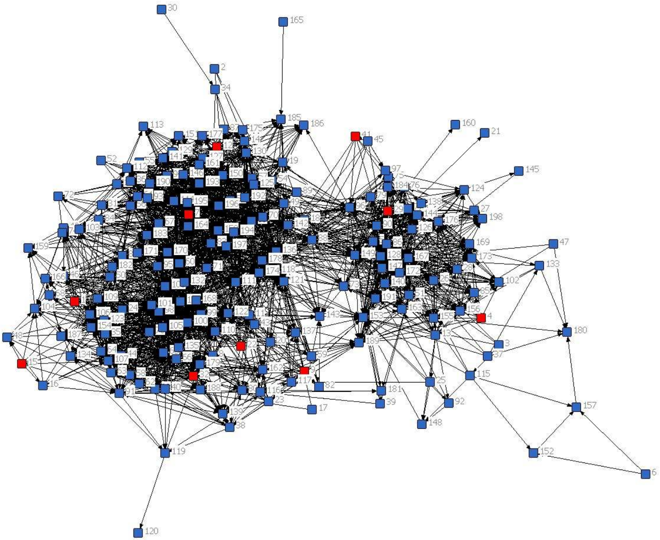




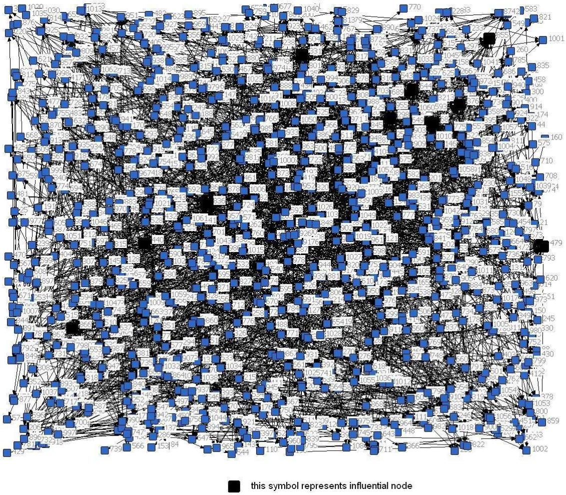
Experiments: Real World Graphs



Top-10 Nodes in Jazz Dataset



Top-10 Nodes in NIPS Co-Authorship Data Set



Running Times: SPIN vs KKT

Dataset	Nodes	SPIN (MIN)	KKT (MIN)	Speed-up
Random graph ($p = 0.005$)	500	13.9	824.93	59
Random graph ($p = 0.01$)	500	14.8	1123.16	75
Random graph ($p = 0.02$)	500	16.3	1302.46	79
Political Books	105	0.89	44.64	50
Jazz	198	1.1	366	332
Celegans	306	14.02	901	64
NIPS	1062	15.2	7201.54	473
Network-Science	1589	28.25	8539.48	302

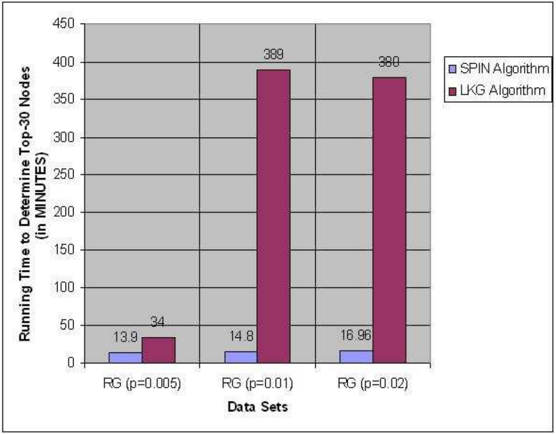
Table: Speedup of the SPIN algorithm to find Top-30 nodes on various datasets compared to that of KKT algorithm

Running Times: SPIN vs KKT

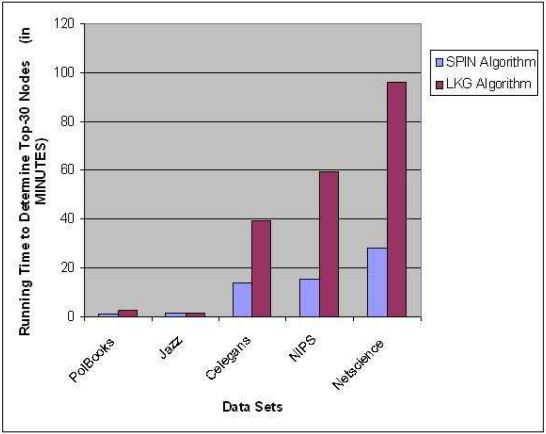
Top-k Nodes	Running Time (in MINUTES) SPIN Algorithm	KKT Algorithm	LKG Algorithm	Speed-up of SPIN over KKT
$k = 10$	28.04	1341.29	77.07	47
$k = 20$	28.09	4297.02	79.75	152
$k = 30$	28.13	8539.48	85.04	302
$k = 40$	28.18	13949.9	90.33	493
$k = 50$	28.25	20411.1	99.03	722

Table: Running times of the SPIN, KKT, and LKG algorithms on the Netscience data set ($n = 1589$) to determine top-k nodes where $k = 10, 20, 30, 40, 50$ and the speed up of the SPIN algorithm over the KKT algorithm

Running Times: SPIN vs LKG on Random Graphs



Running Times: SPIN vs LKG on Real World Graphs



Minimum Thresholds Model

- Each node i initially chooses a threshold θ_i uniformly at random from the interval $[0, 1]$,
- Node i becomes active in time step t if $f_i(S) \geq \theta_i$, where $S \subseteq N_i$ is the set of active neighbors of i in step $(t-1)$,
- We define the threshold function f_i as follows:

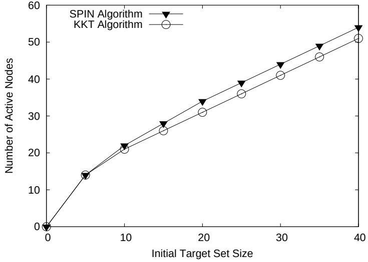
$$f_i(S) = \min_{j \in S}\{\alpha_j w_{ij}\} \tag{1}$$

where $\alpha_j \geq 0$, $\forall j \in N_i$.

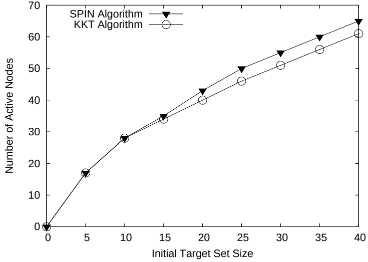
Lemma: Given the minimum linear threshold model and for any node i, the threshold function f_i is monotone decreasing and supermodular.

Experiments: Minimum Thresholds Model

(i)



(ii)



Multiplication Thresholds Model

- Each node i initially chooses a threshold θ_i uniformly at random from the interval $[0, 1]$,
- Node i becomes active in time step t if $f_i(S) \geq \theta_i$, where $S \subseteq N_i$ is the set of active neighbors of i in step $(t-1)$,
- We define the threshold function f_i as follows:

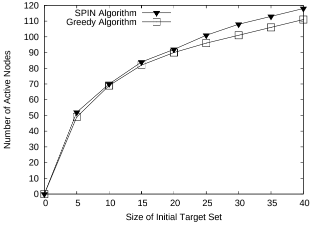
$$f_i(S) = \prod_{j \in S} w_{ij} \tag{2}$$

where w_{ij} is the normalized weight representing the level of influence of node j on node i such that $\sum_{j \in N_i} w_{ij} \leq 1$.

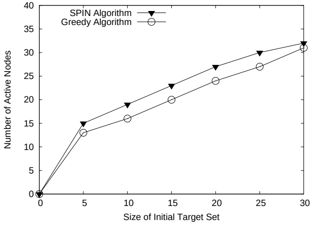
Lemma: Given the multiplication threshold model and any node $i \in N$, the threshold function f_i is monotone decreasing and supermodular.

Experiments: Multiplication Thresholds Model

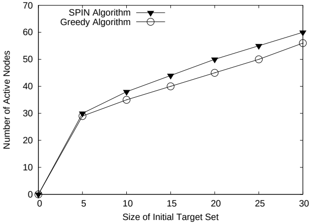
(*i*)



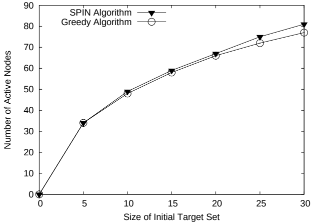
(*ii*)



(*iii*)



(*iv*)



Next Part of the Talk

1. Social Network Analysis: Quick Primer
2. Foundational Concepts in Game Theory
3. Discovering Influential Individuals for Viral Marketing
4. Social Network Formation
5. **Community Detection in Social Networks**
6. Query Incentive Networks
7. Summary and To Probe Further

Graph Partitioning

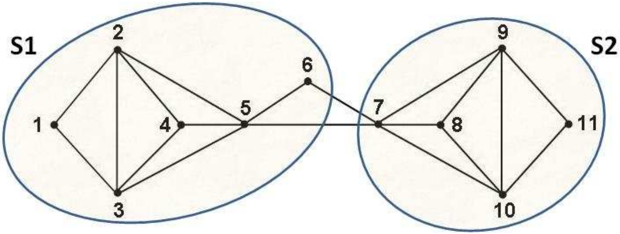
Given an undirected and unweighted graph G, partition the nodes of G into disjoint groups (or communities), such that the nodes in each group are *densely* connected themselves.

The partition of a graph is a set $\Pi = \{S_k\}_{k=1}^{K}$ that partitions the set of nodes N of the graph into K groups such that

1. $S_k \subseteq N$ for $k = 1, 2, \ldots, K$,
2. $S_x \cap S_y = \phi$ for $x, y = 1, 2, \ldots, K$,
3. $\cup_{k=1}^{K} S_k = N$.
4. Each S_k is *dense*.

We call each S_k a *group* or a *community*.

Graph Partitioning: An Example



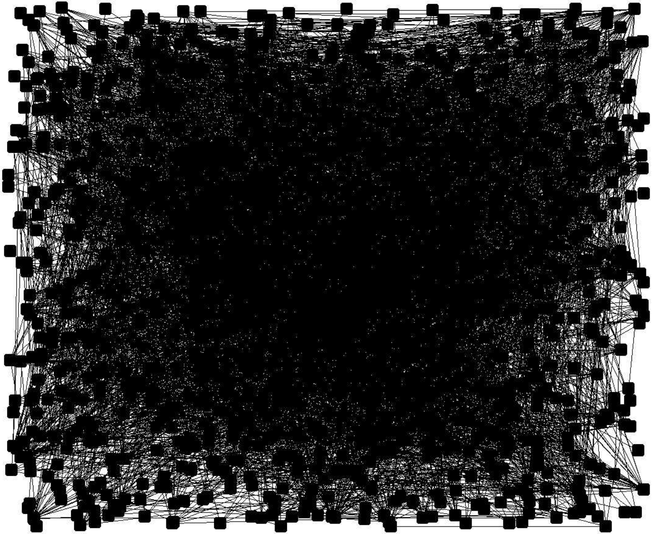
$\Pi = \{S1, S2\}$ where $S1 = \{1, 2, 3, 4, 5, 6\}$ and $S2 = \{7, 8, 9, 10, 11\}\}$

Graph Partitioning: Applications

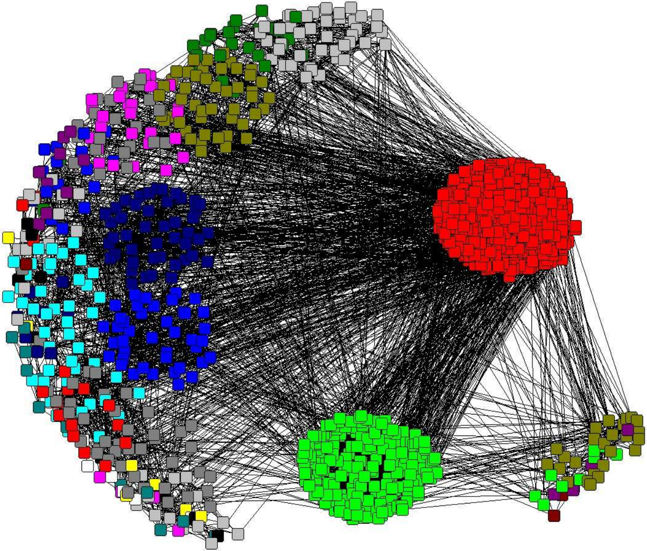
1. VLSI circuit design
2. Parallel computing
3. Social network analysis
4. Graph visualization and summarization

B. W. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. The Bell System Technical Journal, 1970.

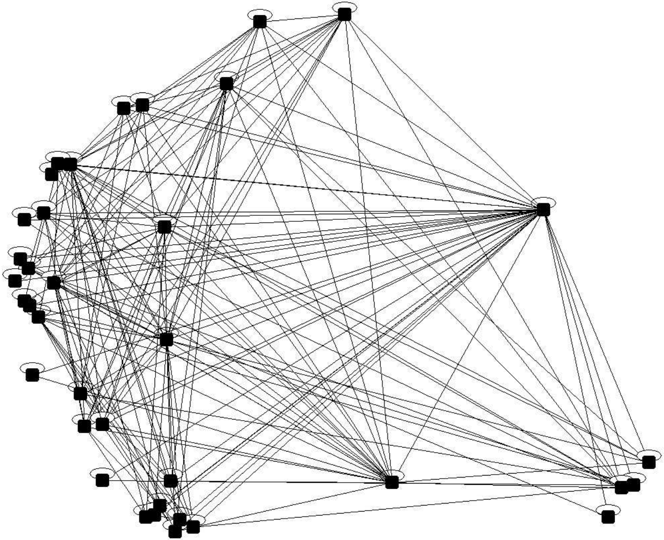
Graph Visualization and Summarization



Graph Visualization and Summarization



Graph Visualization and Summarization



Approaches for Graph Partitioning

- Spectral Approach
- Geometric Approach
- Multi-level Approach
- Social Network Analysis Approach
  - Algorithms based on Centrality Measures
  - Algorithms based on Random Walks
  - Algorithms based on Optimization

---

1. G. Karypis and V. Kumar. A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. SIAM Journal on Scientific Computing, 1997.
2. S. Arora, S. Rao, U.V. Vazirani. Geometry, flows, and graph-partitioning algorithms. Communications of ACM, 51(10):96-105, 2008.

Research Gap and Motivation

- First, many of these algorithms work with a global objective function (such as modularity, conductance) for detecting communities in networks
- Communities in social networks emerge due to the actions of autonomous individuals in the network, without regard to a central authority enforcing certain global objective function
- Thus, algorithms that work with global objective functions do not satisfactorily explain the emergence of communities in social networks
- Second, many algorithms for community detection require the number of communities to be detected as input

Community Detection Problem

- **Problem Statement:** Given an undirected and unweighted graph, we want to design a decentralized algorithm (due to autonomous actions of individuals in the network) that determines a partitioning of the graph with appropriate number of groups (or communities) such that each group in that partition is *dense*

- We propose a game theoretic approach to address the problem

Perhaps first game theoretic approach for community detection

Challenges Involved (Also Wish List)

- Algorithm should use only local information
- Existence of an equilibrium to be guaranteed
- Communities should be dense at equilibrium
- Algorithm itself should determine the number of communities

Community Detection Problem: Our Contributions

- Game theoretic framework based on Nash stable partition
- Propose a utility function based on only local information - it guarantees the existence of a Nash stable partition
- Lower bound on the coverage of any Nash stable partition
- Equivalence of NSP with another popular notion of equilibrium partition under the proposed utility function
- Leads to an efficient algorithm
  - SCoDA (Stable COmmunity Detection Algorithm)
- SCoDA does not require the number of communities as input

Graph Partitioning: Basic Definitions

- Let $G = (N, E)$ be an undirected and unweighted graph where $N = \{1, 2, \ldots, n\}$ is the finite set of nodes and E is the set of edges
- N_i is the set of all neighbors of i including i
- $A_i = \{H \mid H\ is\ a\ subgraph\ of\ G\ and\ i \in H\}$
- (N, u_i) is the graph partitioning game where N is the set of nodes in G and $u_i : A_i \longrightarrow \mathbb{R}$ is the utility of node i
- Ψ is the set of all partitions
- $\Pi(i) = \{S \mid S \in \Pi\ and\ S \cap N_i \neq 0\}$

Graph Partitioning: Basic Definitions

Nash Stable Partition (NSP): A partition $\Pi \in \Psi$ is called a Nash stable partition of the given graph if $\forall i \in N$,

$$u_i(S_\Pi(i), G) \geq u_i(S_k \cup \{i\}, G), \ \forall S_k \in \Pi(i).$$

Individually Stable Parition (ISP): A partition $\Pi \in \Psi$ is called an Individually stable partition of the given graph if there does not exist $i \in N$ and a group $S_k \in \Pi$ such that

$$u_i(S_k \cup \{i\}) > u_i(S_\Pi(i)), \ and \ u_j(S_k \cup \{i\}) > u_j(S_k), \ \forall j \in S_k.$$

A. Bogomolnaia and M.O. Jackson. The stability of hedonic coalition structures. Games and Economic Behavior, 38:201-230, 2002.

Proposed Utility Function

- Utility function should use only local information
- It should ensure the existence of NSP
- It should produce dense communities at NSP
- For each $i \in N$, the utility of node i is defined to be the number of neighbors of that node in its community plus a function of the fraction of neighbors in its community that are connected themselves

Proposed Utility Function (Cont.)

- For each $i \in N$ and $\forall S \in A_i$,

$$u_i(S) = d_i(S) + \frac{T_i(S)}{\binom{d_i(S)}{2}} f(d_i(S))$$

  where (i) $d_i(S)$ is the number of neighbors of node i in community S, and (ii) $T_i(S)$ is number of pairs of neighbors of node i in S that are connected themselves, and (iii) $f(d_i(S))$ is a weight function

- To keep matters simple, we consider linear weight function $f(.)$ such as:

$$f(d_i(S)) = \alpha d_i(S), \ \forall i \in N, \ \forall S \in A_i$$

  where $\alpha > 0$ is a constant

- To keep matters further simple, we work with $\alpha = 1$ and $\alpha = 2$

Proposed Utility Function (Cont.)

- When $\alpha = 1$, for each $i \in N$ and $\forall S \in A_i$,

$$u_i(S) = d_i(S) + \frac{2T_i(S)}{d_i(S) - 1}$$

- When $\alpha = 2$, for each $i \in N$ and $\forall S \in A_i$,

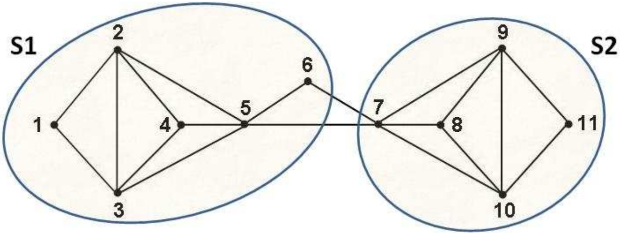
$$u_i(S) = d_i(S) + \frac{4T_i(S)}{d_i(S) - 1}$$

Proposed Utility Function (Cont.)

- **Lemma:** When $\alpha = 1$, it holds that $u_i(S) \leq u_i(\bar{S})$ for each $S \subseteq \bar{S}$.
  - This can be proved using simple algebra
  - The utility of node i does not decrease when new nodes join its community
  - Suitable for certain practical applications such as collaborative editing in Wikipedia

- **Lemma:** When $\alpha = 2$, for each $S \subseteq \bar{S}$, then $u_i(S) \leq u_i(\bar{S})$ holds only if $T_i(S) \leq \frac{d_i(S)(d_i(S)-1)}{4}$, $\forall i \in S$
  - The utility of node i need not necessarily increase when new nodes join its community
  - Suitable for certain practical settings such as collaboration networks

Nash Stable Partition: An Example



$\Pi = \{S1, S2\}$ where $S1 = \{1, 2, 3, 4, 5, 6\}$, and $S2 = \{7, 8, 9, 10, 11\}$

$u_1(S1) = 4, \quad u_1(S2) = 0;$

$u_2(S1) = 6.668, \quad u_2(S2) = 0;$

$u_3(S1) = 6.66, \quad u_3(S2) = 0;$

$u_4(S1) = 6, \quad u_4(S2) = 0;$

$u_5(S1) = 6, \quad u_5(S2) = 1;$

$u_6(S1) = 1, \quad u_6(S2) = 1;$

$u_7(S2) = 6, \quad u_7(S1) = 1;$

$u_8(S2) = 6, \quad u_8(S1) = 0;$

$u_9(S2) = 6.66, \quad u_9(S1) = 0;$

$u_{10}(S2) = 6.66, \quad u_{10}(S1) = 0;$

$u_{11}(S2) = 4, \quad u_{11}(S1) = 0;$

Important Properties of NSPs

- For each $i \in N$, let $\Psi(\Pi, i)$ be the set of all partitions where each partition is derived from Π by moving node i from $S_\Pi(i)$ to some $X \in \Pi(i)$. That is, $|\Psi(\Pi, i)| = |\Pi(i)| - 1$, (because $S_\Pi(i) \in \Pi(i)$).
- Let $INTRA(\Pi) = \{(i,j) \in E \mid \exists S \in \Pi \ni i,j \in S\}$ be the set of edges within communities in a partition Π. We also define $INTER(\Pi) = E \setminus INTRA(\Pi)$ to be the set of edges across communities in partition Π.
- Consider $i,j,k \in N$. If $(i,j) \in E$, $(j,k) \in E$, and $(k,i) \in E$, then we say that nodes i,j,k form a *triangle* in the graph.
- Finally, let $E(S)$ be the set of all edges among nodes in S only. That is, $E(S) = \{(i,j) \in E \mid i,j \in S\}$.

Important Properties of NSPs (Cont.)

$$\boxed{coverage(\Pi) = \frac{INTRA(\Pi)}{INTRA(\Pi)+INTER(\Pi)}}$$

Lemma: Assume that $\alpha = 1$. Given an undirected and unweighted graph $G = (N, E)$ and any Nash stable partition $\Pi = \{S_1, S_2\}$ with two communities of G, then $coverage(\Pi) \geq \frac{1}{3}$.

Proof Sketch:

- For each $i \in S_1$, we have that

$$d_i(S_2) + \frac{2T_i(S_2)}{d_i(S_2) - 1} \leq d_i(S_1) + \frac{2T_i(S_1)}{d_i(S_1) - 1} \tag{3}$$

- Now using fact that $|INTER(\Pi)| = \sum_{i \in S_1} d_i(S_2)$, we get that

$$|INTER(\Pi)| + \sum_{i \in S_1} \frac{2T_i(S_2)}{d_i(S_2) - 1} \leq \sum_{i \in S_1} d_i(S_1) + \sum_{i \in S_1} \frac{2T_i(S_1)}{d_i(S_1) - 1}. \tag{4}$$

Important Properties of NSPs (Cont.)

- On similar lines, repeating the above argument for S_2, we get that

$$|INTER(\Pi)| + \sum_{i \in S_2} \frac{2T_i(S_1)}{d_i(S_1) - 1} \leq \sum_{i \in S_2} d_i(S_2) + \sum_{i \in S_2} \frac{2T_i(S_2)}{d_i(S_2) - 1} \quad (5)$$

- After summing the expressions (4) and (5), we use the fact that $\sum_{i \in S_1} d_i(S_1) = 2|E(S_1)|$, $\sum_{i \in S_2} d_i(S_2) = 2|E(S_2)|$, and rearranging terms, we get that

$$\sum_{i \in S_1} \frac{T_i(S_2)}{d_i(S_2) - 1} + \sum_{i \in S_2} \frac{T_i(S_1)}{d_i(S_1) - 1} + |INTER(\Pi)| \leq$$

$$|E(S_1)| + |E(S_2)| + \sum_{i \in S_1} \frac{T_i(S_1)}{d_i(S_1) - 1} + \sum_{i \in S_2} \frac{T_i(S_2)}{d_i(S_2) - 1}$$

Important Properties of NSPs (Cont.)

- Now substituting $INTER(\Pi) = E \setminus INTRA(\Pi)$ and $E(S_1) \cup E(S_2) = INTRA(\Pi)$ in the above expression and readjusting the terms, we get that

$$\Rightarrow \ |INTRA(\Pi)| \geq \frac{1}{2}|E| - \frac{1}{2}\left[\sum_{i \in S_1} \frac{T_i(S_1)}{d_i(S_1) - 1} + \sum_{i \in S_2} \frac{T_i(S_2)}{d_i(S_2) - 1}\right]$$

- Since the maximum value for $T_i(X)$ is $\frac{d_i(X)(d_i(X)-1)}{2}$ for each $X \in \{S_1, S_2\}$; and bounding the above expression using this fact, we get that

$$|INTRA(\Pi)| \geq \frac{1}{2}|E| - \frac{1}{2}\left[\sum_{i \in S_1} \frac{d_i(S_1)}{2} + \sum_{i \in S_2} \frac{d_i(S_2)}{2}\right]$$

$$\Rightarrow \ |INTRA(\Pi)| \geq \frac{1}{2}|E| - \frac{1}{2}|INTRA(\Pi)| \ \Rightarrow coverage(\Pi) \geq \frac{1}{3}.$$

Important Properties of NSPs (Cont.)

- **Lemma:** Assume that $\alpha = 2$. Given an undirected and unweighted graph $G = (N, E)$ and a Nash stable partition $\Pi = \{S_1, S_2\}$ with 2 communities of G, then $coverage(\Pi) \geq \frac{1}{4}$.

An Algorithm for Graph Partitioning

- SCoDA: Stable COmmunity Detection Algorithm
  - Initial configuration
  - Order of nodes
  - Stopping criterion
- Running time : $O(n\, log(n) + nmd_{max}^2)$
- The resultant community structure with SCoDA is obviously a Nash stable partition

SCoDA

Let Π be the initial partition of the graph G.
Let *visit_order*[] contains nodes in non-decreasing order of the degree.
while *true*, *do*
    for $i := 1$ to n
        flag $\leftarrow 0$;
        $j \leftarrow visit\_order[i]$
        if $u_i(S_\Pi(i)) < u_i(S_k)$ for some $S_k \in \Pi(i)$, *then*
            move node i from $S_\Pi(i)$ to S_k;
            flag $\leftarrow 1$;
        end if
    end for
    if flag $= 0$, *then*
        break;
    end if
end while

Existence of NSP

Lemma: When $\alpha = 1$, Algorithm 1 always guarantees convergence to a Nash stable partition, given any undirected and unweighted graph.

- Given that $\alpha = 1$. We first define a function $\Phi : \Psi \rightarrow \mathbb{N}$ such that, for each $\Pi \in \Psi$, $\Phi(\Pi)$ represents the sum of the number of intra community edges in Π and the number of triangles within the communities in Π.
- For each $\Pi \in \Psi$, we call $\Phi(\Pi)$ as the *capacity* of the partition Π. More formally, $\forall \Pi \in \Psi$,

$$\Phi(\Pi) = \sum_{S \in \Pi} \sum_{j \in S} \frac{d_j(S)}{2} + \sum_{S \in \Pi} \sum_{i \in S} \sum_{p,q \in S \cap N_i} \frac{I(p,q)}{3} \qquad (6)$$

where $I(p,q)$ is an indicator function that takes value 1 if nodes p and q are adjacent in G and 0 otherwise.

Existence of NSP (Cont.)

- Define a partition $\Pi_x \in \Psi$ to be *maximal* if there *does not exist* any $\Pi_y \in \Psi$ such that (i) $\Pi_y \in \Psi(\Pi_x, i)$ for some $i \in N$, and (ii) the capacity of Π_y is strictly greater than the capacity of Π_x.
- Now consider a partition $\Pi_1 \in \Psi$. Let $\Pi_2 \in \Psi(\Pi_1, i)$ be a partition that is obtained from Π_1, when node i jumps from $S_{\Pi_1}(i)$ to some $X \in \Pi_1(i)$.
- Note that node i jumps from $S_{\Pi_1}(i)$ to some $X \in \Pi_1(i)$ to improve its utility, then

$$d_i(S_{\Pi_1}(i)) + \frac{2T_i(S_{\Pi_1}(i))}{d_i(S_{\Pi_1}(i)) - 1} < d_i(X) + \frac{2T_i(X)}{d_i(X) - 1}. \tag{7}$$

Existence of NSP (Cont.)

- This is possible when either $d_i(X) > d_i(S_{\Pi_1}(i))$ or $T_i(X) > T_i(S_{\Pi_1}(i))$ holds.
- A simple algebra based on this fact implies that $\Phi(\Pi_2) > \Phi(\Pi_1)$.
- This further implies that, whenever a node moves from one group to the other group, the capacity of the new partition strictly improves upon the capacity of the old partition.
- Since the number of partitions is finite, a maximal partition certainly exists.

Improved Version of SCoDA

- We further refine the Nash stable partition produced through SCoDA to improve its modularity
- *Greedy Strategy:* In each step, we determine a pair of communities to merge so that the modularity of the resultant community structure is maximized. We repeat this step until we do not find any pair of communities to merge to improve modularity
- Running time of the improved version of SCoDA is $O(n\ log(n) + nmd_{max}^2 + k^3 n)$

Benchmark Algorithms for Comparison

1. Edge Betweenness Algorithm (Girvan and Newman (2002)),
2. Fast Greedy Algorithm (Newman (2004)),
3. Spectral Algorithm (Newman (2006)).
4. A Randomized and Game Theoretic Algorithm (Chen et. al. (2010))

Performance Metric

Modularity of a partition Π of given undirected and unweighted graph $G = (N, E)$ is:

$$Q(\Pi, G) = \frac{1}{2m} \sum_{i,j \in N} \left(a_{i,j} - \frac{n_i n_j}{2m} \right) \delta(S_\Pi(i), S_\Pi(j))$$

where (i) m is number of edges in G, (ii) n_x is degree of node $x \in N$, (iii) $S_\Pi(x)$ represents ID of the community of node x, and (iv) $\delta(a, b)$ takes value 1 if $a = b$ and 0 otherwise.

M.E.J. Newman and M. Girvan. Finding and evaluating community structure in networks. Physical Review E 69, 026113, 2004.

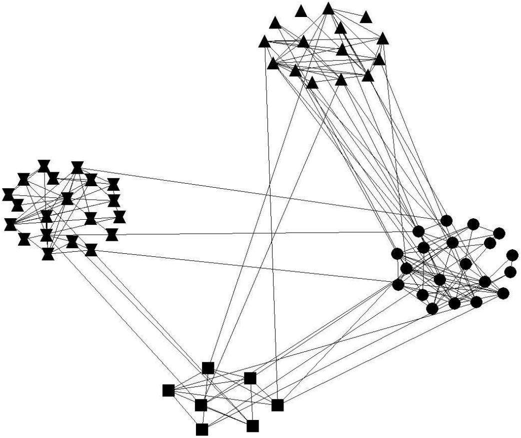
Experimental Results: Data Sets

Data Set	*Nodes*	*Edges*	*Triangles*
Karate	34	78	45
Dolphins	62	318	95
Les Miserables	77	508	467
Political Books	105	882	560
FootBall	115	1226	810
Jazz	198	2742	17899
Email	1133	5451	10687
Yeast	2361	6913	5999

Table: Description of various real world network data sets

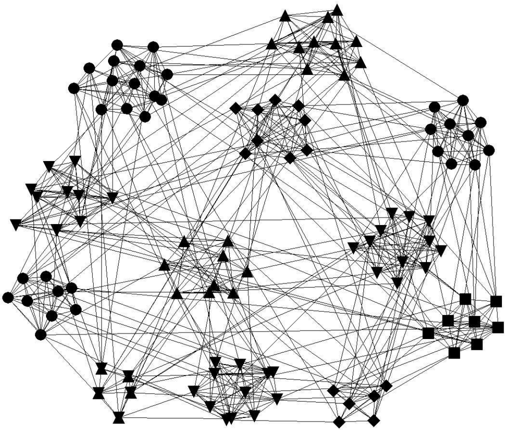
Comparison with GN Algorithm: Dolphins Data Set

- Modularity using GN Algorithm: 0.519.
- Modularity using SCoDA: 0.526.



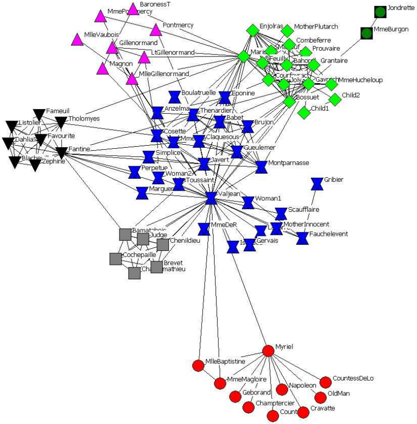
Comparison with GN Algorithm: Football Data Set

- Modularity using GN Algorithm: 0.598.
- Modularity using SCoDA: 0.6.



Comparison with GN Algorithm: Les Miserable Data Set

- Modularity using GN Algorithm: 0.538.
- Modularity using SCoDA: 0.538.



Experimental Results - Modularity

Table: Comparison of modularity due to SCoDA with that of four benchmark algorithms.

Data Set	SCoDA ($\alpha = 1$)	SCoDA ($\alpha = 2$)	GN	Greedy	Spectral	RGT
Karate	0.4	0.4	0.4	0.38	0.393	0.392
Dolphins	0.525	0.525	0.519	0.495	0.491	0.502
Les-Mis	0.545	0.545	0.538	0.5	0.532	0.54
Pol-Books	0.524	0.524	0.496	0.509	0.469	0.493
Football	0.60	0.60	0.598	0.566	0.539	0.581
Jazz	0.439	0.439	0.403	0.438	0.393	0.439
Email	0.479	0.51	0.51	0.494	0.498	0.509
Yeast	0.571	0.571	0.568	0.571	0.497	-

Experimental Results - Coverage

Table: Comparison of coverage due to SCoDA with that of four benchmark algorithms.

Data Set	SCoDA ($\alpha = 1$)	SCoDA ($\alpha = 2$)	GN	Greedy	Spectral	RGT
Karate	82.05	82.05	75.64	30.76	25.24	68.52
Dolphins	80.50	80.50	82.38	22.01	22.64	69.43
Les-Mis	75.19	75.19	73.22	35.03	25.19	72.83
Pol-Books	89.11	89.11	91.83	59.63	45.57	74.45
Football	69.00	69.00	74.22	16.15	12.39	67.92
Jazz	79.83	79.83	77.93	31.72	34.31	71.35
Email	72.46	73.18	75.67	21.61	18.33	58.12
Yeast	66.99	67.85	74.55	19.53	22.14	-

Overlapping Communities in Social Networks

- In social networks, typically individuals belong to more than one community
- It is important to determine overlapping communities
- W. Chen, Z. Liu, X. Sun, Y. Wang. *A game-theoretic framework to identify overlapping communities in social networks*. Data Mining Knowledge Discovery, 21:224-240, 2010.

Community Formation Game

- Let $G = (V, E)$ be an undirected and unweighted graph with $n = |V|$ and $m = |E|$ and it represents the underlying static acquaintance graph.
- The set of all possible communities is denoted as $[k] = \{1, 2, \ldots, k\}$, where k is polynomial in n.
- Each strategy L_i of v_i is a subset of communities that it wants to join; i.e, $L_i \subseteq [k]$.
- Let $L = (L_1, L_2, \ldots, L_n)$ be the profile of strategies of n agents.
- The utility of the i-th agent is measured by a gain function $g_i(L)$ and a loss function $l_i(L)$ and it is defined as

$$u_i(L) = g_i(L) - l_i(L)$$

Community Formation Game (Cont.)

- In general, Nash equilibria may not exist in a community formation game.
    - Consider an instance of community formation game where one node u always prefer to be with another node v in the same community while v always prefer not to be in the same community as u.

- A community formation game is a potential game if

$$\Phi(L_{-i}, L_i^{'}) - \Phi(L) = u_i(L_{-i}, L_i^{'}) - u_i(L)$$

for every strategy profile L and for every strategy $L_i^{'}$ of agent i.

Community Formation Game (Cont.)

- **Definition;** A set of functions $\{f_i(.) : 1 \leq i \leq n\}$ is locally linear with linear factor ρ if for every strategy profile L and every strategy $L_i^{'}$ of i, the following relation holds: $\forall i \in V$,
$$f_i(L_{-i}, L_i^{'}) - f_i(L) = \rho(f_i(L_{-i}, L_i^{'}) - f_i(L))$$
- **Theorem:** If $\{g_i(.) : i \in V\}$ and $\{l_i(.) : i \in V\}$ are locally linear functions with linear factor ρ_g and ρ_l, then the community formation game is a potential game.
- **Lemma:** There exists a community formation game, in which the sets of gain and loss functions are locally linear, such that both computing the best response for an individual agent and computing a Nash Equilibrium in the game are NP-hard.

Specific Gain and Loss Functions

- Now we discuss a set of gain and loss functions and they can be computed efficiently
- **Gain Function:** We use here is a generalized version of modularity.
  - Define $\delta(i,j) = 1$ if $|L_i \cap L_j| \geq 1$ and $\delta(i,j) = 0$ otherwise.
  - Let A be the adjacency matrix of graph G.
  - $g_i(L) = \frac{1}{2m} \sum_{j \in [n]} \left(A_{ij}\delta(i,j) - \frac{d_i d_j}{2m} |L_i \cap L_j| \right)$
- **Loss Function:** We use a simple loss function to model the aspect that an agent may suffer by joining new communities
  - Let $c > 0$ be a constant. Then, $l_i(L) = (|Li| - 1)c$.

Specific Gain and Loss Functions (Cont.)

- Easy to verify that the above $g_i(L)$ and $l_i(L)$ are locally linear functions, with linear factor $\frac{1}{2}$ and 1 respectively.

- **Theorem:** Given the above $g_i(L)$ and $l_i(L)$, the community formation game has a Nash equilibrium.

Specific Gain and Loss Functions (Cont.)

- Computing the best response for each agent might be hard in certain contexts from the previous Lemma
- It is not reasonable to assume that individuals always make the best response
- An agent can only locally implement the following three operations
  - *Join:* Agent i joins a new community on top of the communities she joins by adding a new label in L_i
  - *Leave:* Agent i leaves a community she is in by removing a label from L_i
  - *Switch:* Agent i switches from one community to another by replacing a label in L_i

Specific Gain and Loss Functions (Cont.)

- **Algorithm** 1: LocalEquilibrium(G)
  - initialize each node to a singleton community
  - repeat the following process until no node can improve itself
    - randomly pick a node i and perform the best operation among join, leave and switch
- Entire strategy space of agent i is $2^{[k]}$
- For each agent i with the current community label set L_i, we use $ls(L_i)$ to denote is local strategy space that is obtained by applying join, leave, or switch once on L_i

Specific Gain and Loss Functions (Cont.)

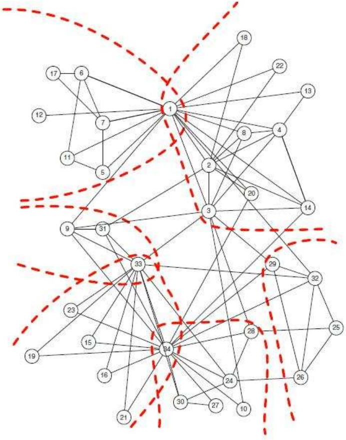
- **Local Equilibrium:** Given G, the strategy profile $L = (L_1, L_2, ..., L_n)$ forms a local equilibrium of the community formation game if all agents are playing their local optimal strategies, that is, $\forall i \in V$ and $L_i^{'} \in ls(L_i)$,

$$u_i(L_{-i}, L_i^{'}) \leq u_i(L_{-i}, L_i)$$

- **Theorem:** Let $g_i(L)$ be the personalized modularity function and $l_i(L) = c(|Li| - 1)$ be a linear loss function with constant c satisfying $4cm^2$ is an integer. Algorithm 1 takes at most $O(m^2)$ steps to reach a local equilibrium.

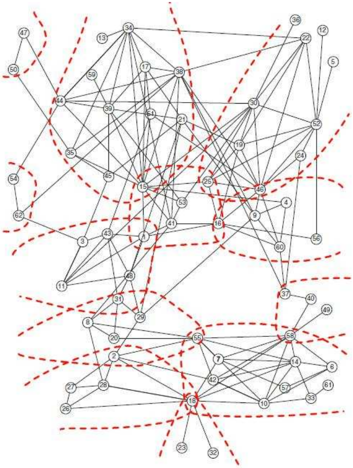
Experimental Results

Zachary's Karate Club Network:



Experimental Results (Cont.)

Dolphin Network:



Next Part of the Talk

Summary of the Tutorial

- We first presented the important fundamental concepts in social network analysis and game theory
- We then presented game theoretic models for four important problems in social network analysis
  - Determining top-k influential individuals
  - Social network formation
  - Community detection
  - Query incentive networks
- We also gave a systematic analysis of these game theoretic models to SNA

To Probe Further: Important Research Directions

- **Altruistic Game Theoretic Models:**
  - Often individuals in social networks are not only rational and intelligent but also altruistic
  - Social network formation and Bargaining on social networks
- **Time Varying Graphs:** Typically, the structure of networks change over time. Designing game theoretic models for such time varying graphs is a challenging and interesting research direction
- **Probabilistic Graphs:**
  - Complex networks often entail uncertainty and thus can be modeled as probabilistic graphs
  - M. Potamias, F. Bonchi, A. Gionis, and G. Kollios. *k*-nearest neighbors in uncertain graphs In VLDB Endowment, Vol. 3, No. 1, 2010

To Probe Further: Important Text Books

- D. Easley and J. Kleinberg. Networks, Crowds, and Markets. Cambridge University Press, 2010.
- M.E.J. Newman. Networks: An Introduction. Oxford University Press, 2010.
- M.O. Jackson. Social and Economic Networks. Princeton University Press, 2008.
- U. Brandes and T. Erlebach. Network Analysis: Methodological Foundations. Springer-Verlag Berlin Heidelberg, 2005.

To Probe Further: Important References

- Ramasuri Narayanam and Y. Narahari. A Shapley Value based Approach to Discover Influential Nodes in Social Networks. In IEEE Transactions on Automation Science and Engineering (IEEE TASE), 2011.
- Ramasuri Narayanam and Y. Narahari. Topologies of Strategically Formed Social Networks Based on a Generic Value Function - Allocation Rule Model. Social Networks, 33(1), 2011.
- Ramasuri Narayanam and Y. Narahari. Determining Top-k Nodes in Social Networks using the Shapley Value. In AAMAS, pages 1509-1512, Portugal, 2008.
- Ramasuri Narayanam and Y. Narahari. Nash Stable Partitioning of Graphs with Application to Community Detection in Social Networks. Under Review, 2010.
- D. Dikshit and Y. Narahari. Truthful and Quality Conscious Query Incentive Networks. In Workshop on Internet and Network Economics (WINE), 2009.
- Mayur Mohite and Y. Narahari. Incentive Compatible Influence Maximization in Social Networks with Application to Viral Marketing. AAMAS 2011

To Probe Further: Useful Resources

Network Data Sets:

- Jure Leskovec: http://snap.stanford.edu/data/index.html
- MEJ Newman: http://www-personal.umich.edu/m̃ejn/netdata
- Albert L. Barabasi: http://www.nd.edu/ñetworks/resources.htm
- NIST Data Sets: http://math.nist.gov/R̃Pozo/complex_datasets.html
- ...

To Probe Further: Useful Resources (Cont.)

- Y. Narahari, Dinesh Garg, Ramasuri Narayanam, Hastagiri Prakash. Game Theoretic Problems in Network Economics and Mechanism Design Solutions. In Series: Advance Information & Knowledge Processing (AIKP), Springer Verlag, London, 2009.

- Home page of Y. Narahari: http://lcm.csa.iisc.ernet.in/hari/

- Home page of Ramasuri Narayanam: http://lcm.csa.iisc.ernet.in/nrsuri/

- Blog on Social Networks: http://cs2socialnetworks.wordpress.com/

Thank You