Social network analysis: Community detection

Rushed Kanawati
LIPN, CNRS UMR 7030; USPC
http://lipn.fr/∼kanawati
rushed.kanawati@lipn.univ-paris13.fr

PLAN

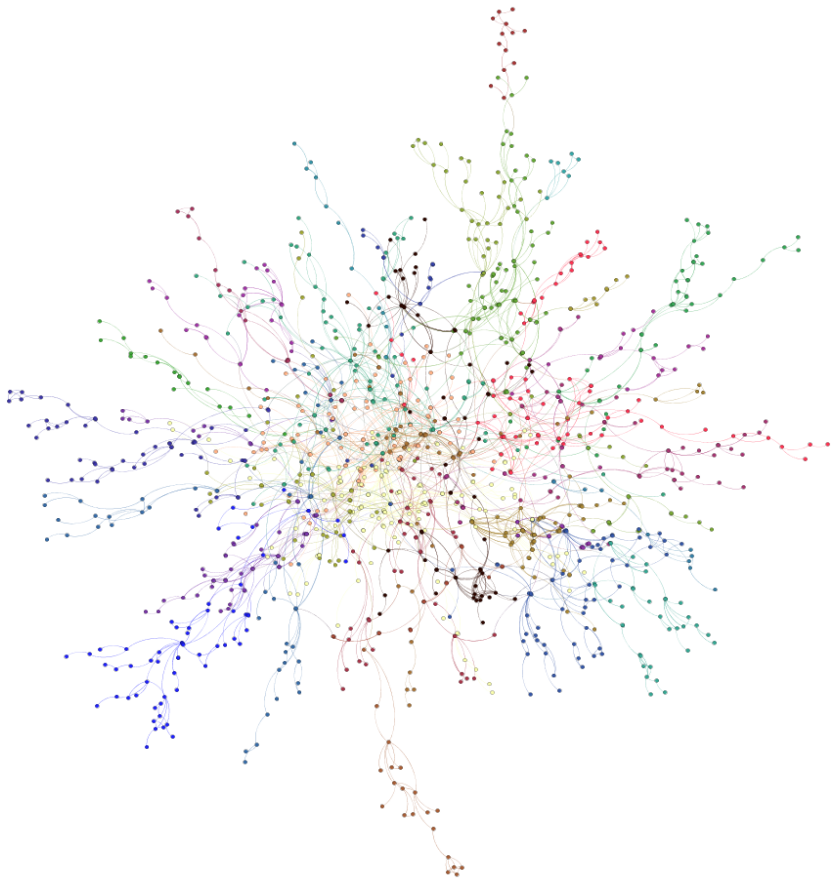
COMPLEX NETWORK

Definition

Graphs modeling (direct/indirect) interactions among actors.

Basic topological features

- Low Density
- Small Diameter
- Heterogeneous degree distribution.
- High Clustering coefficient
- Community structure



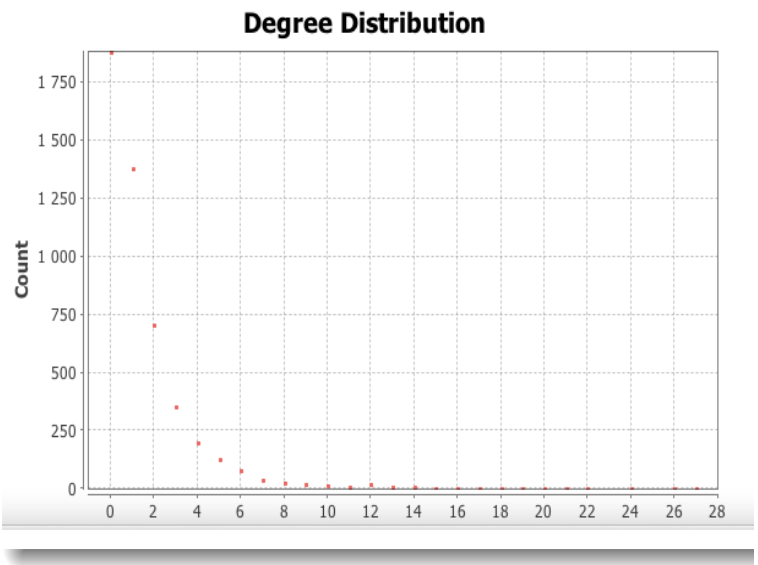
EXAMPLE I : SCIENTIFIC COLLABORATION NETWORK

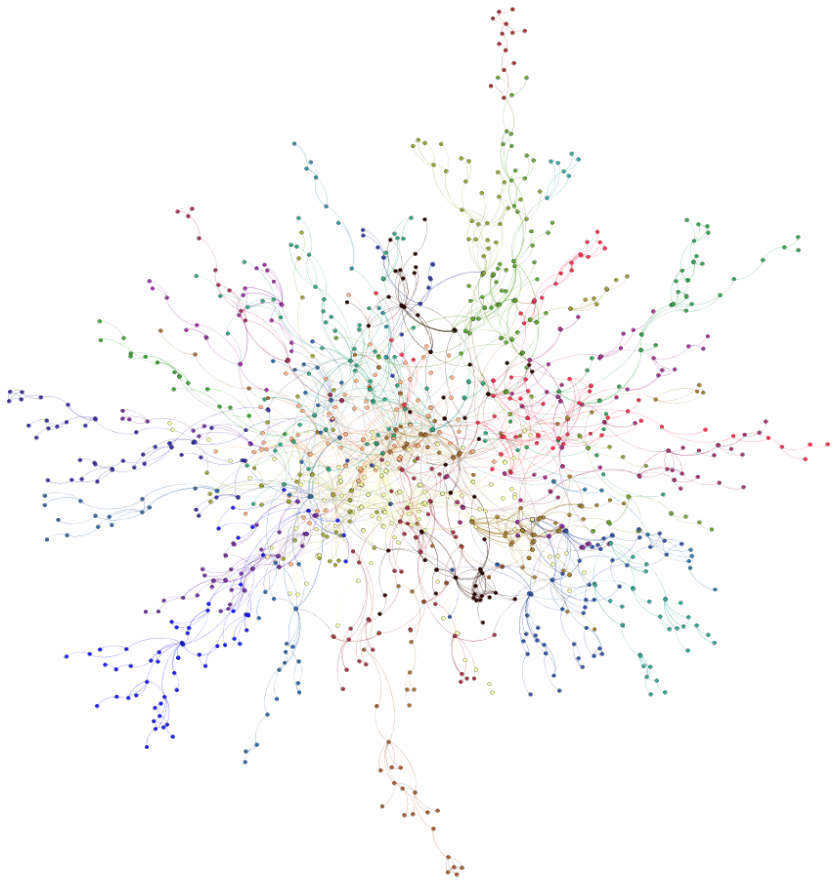
Co-authorship network - DBLP 1980-1984 (for authors active for more than 10 years)

Density : 10^{-4}

Diameter : 24

Clustering coeff.: 0.67





EXAMPLE II: SPATIAL NETWORK

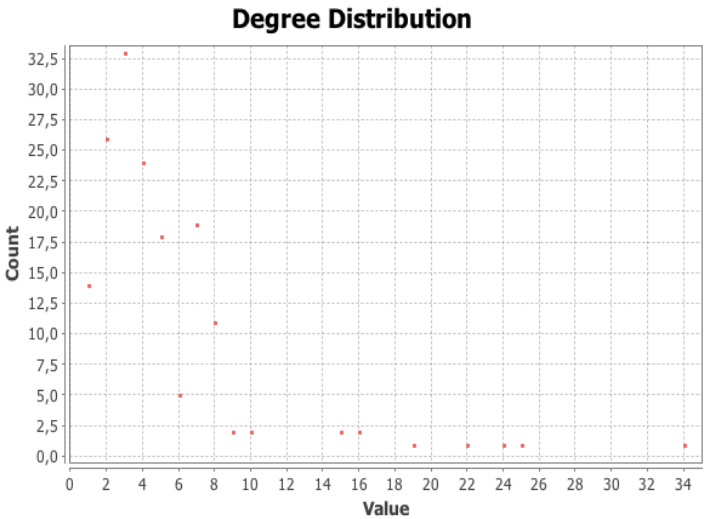
Public sites accessibility in the Bourget district

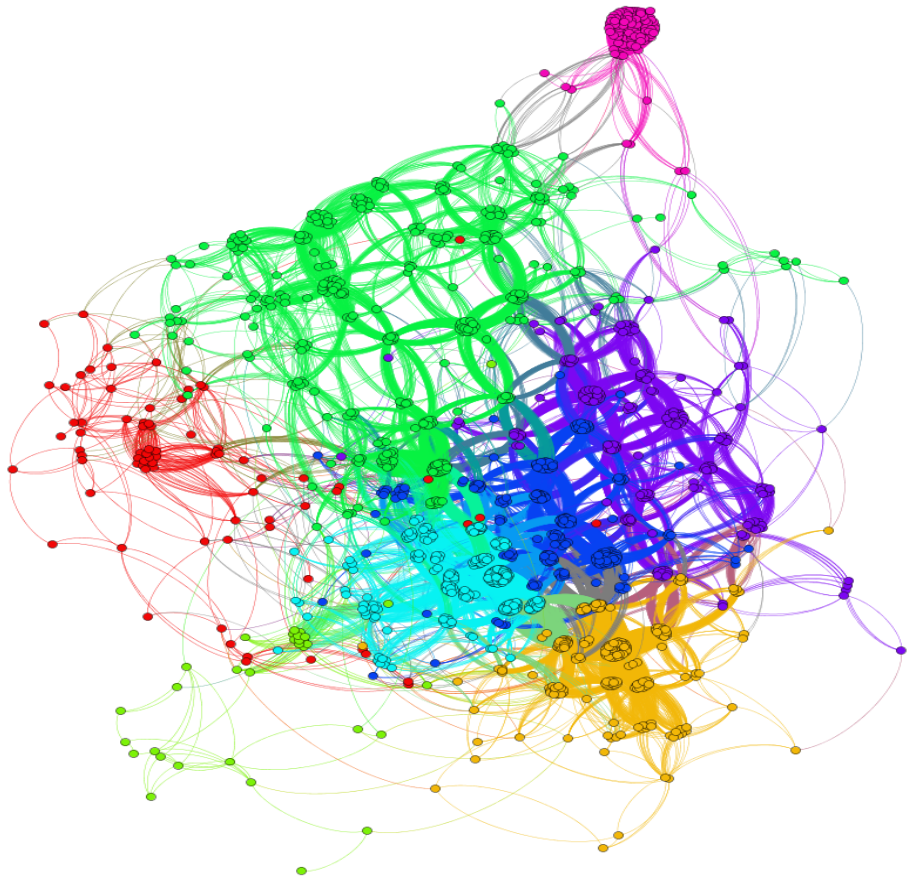
projet FUI UrbanD 2009-2012

Densiy : 0.052

Diameter : 7

clustering Coef.: 0.87





A relative-neighbourhood graph

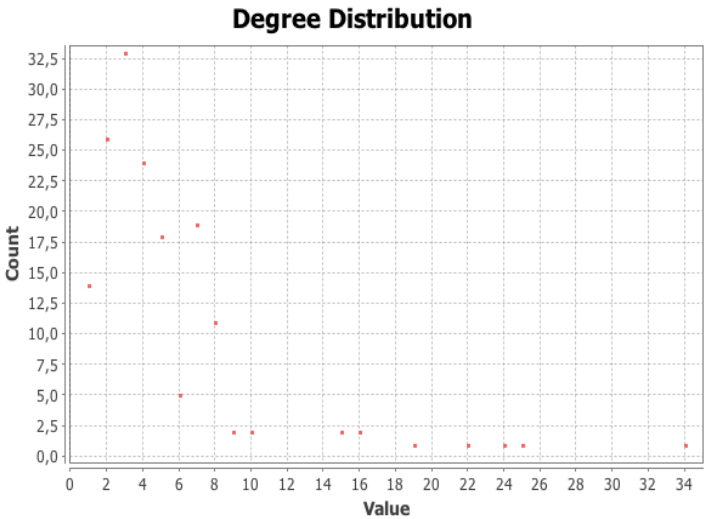
EXAMPLE III: PRODUCT PURCHASE NETWORK

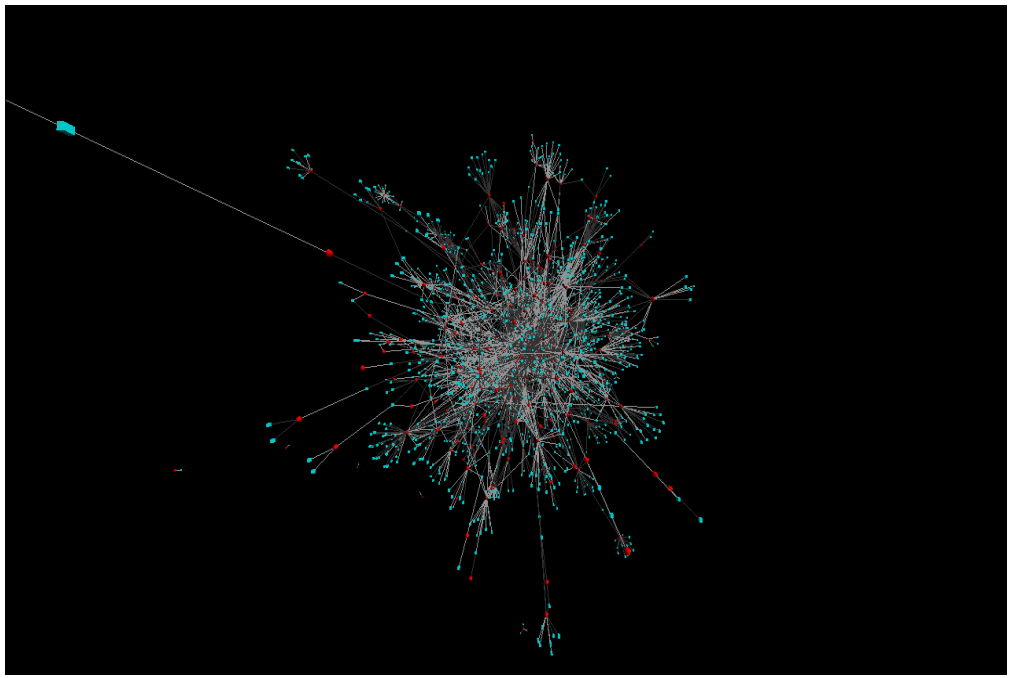
projet ANR CADI 2008-2010

Density : 0.02

Diameter : 8

Clustering coef. : 0.84





Music purchase on Mondomix site during April-June 2004

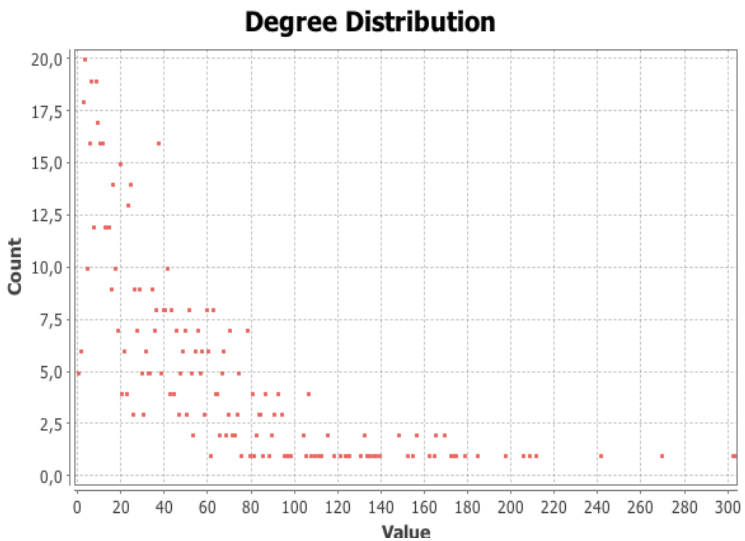
EXAMPLE IV: CO-RATING NETWORK

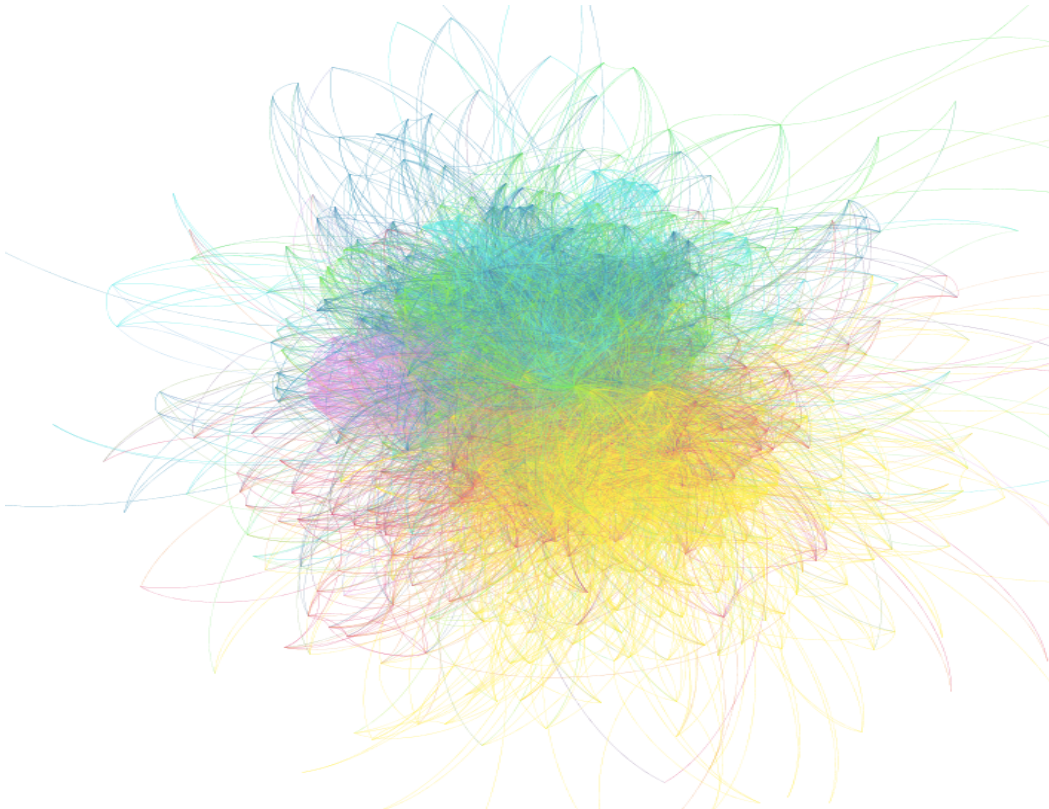
Movie Lens

Density : 0.061

Diameter : 4

Clustering coef. : 0.626





Users that co-rate by 1 at least one movie

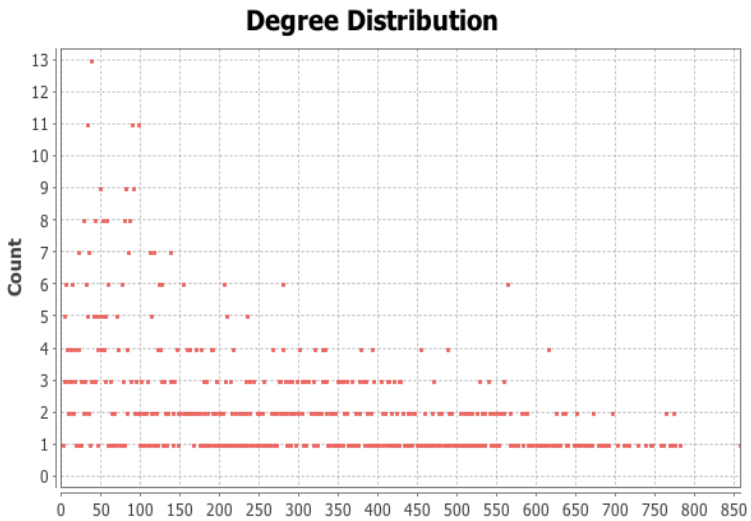
EXAMPLE V: PRODUCT NETWORK

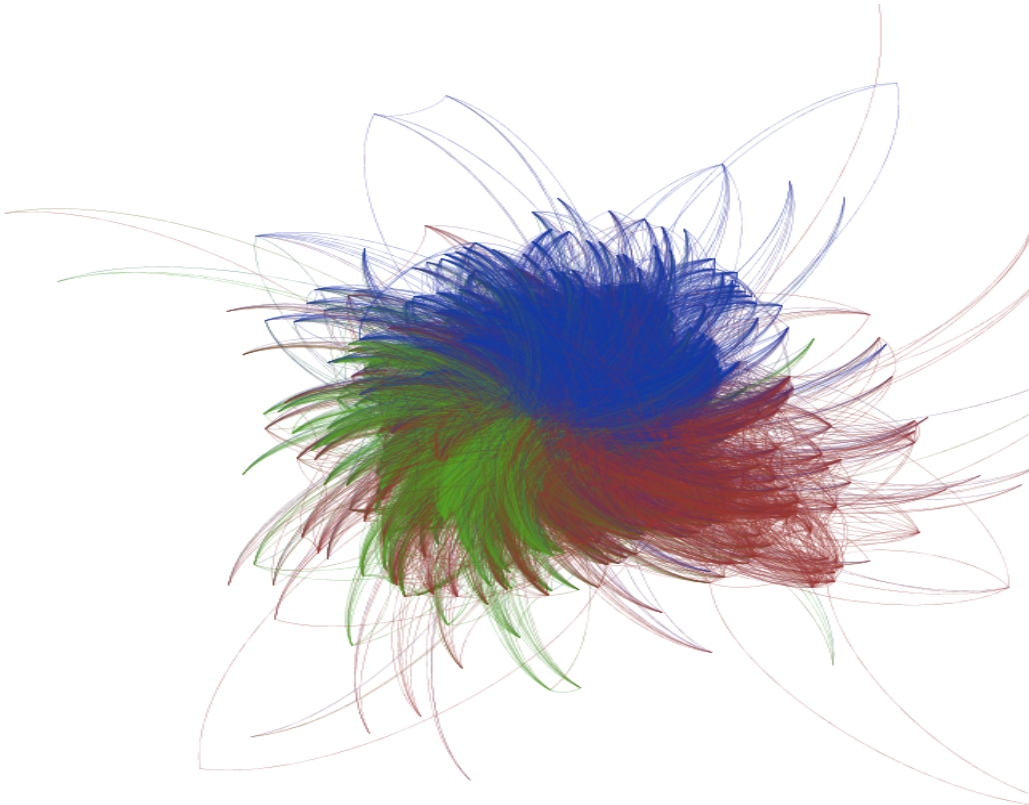
MovieLens

Density : 0.22

Diameter : 4

Clustering coef. : 0.746

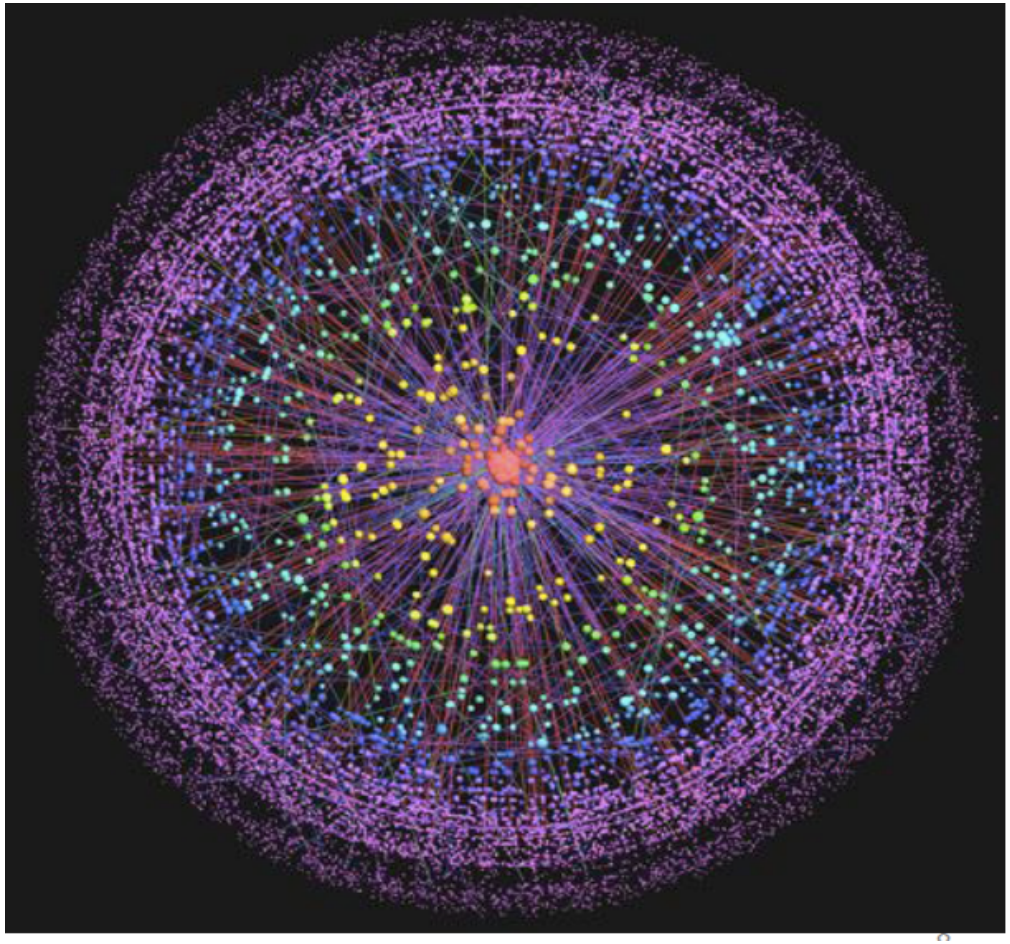




Movies co-rated 5 by at least 1 user

ANOTHER EXAMPLE

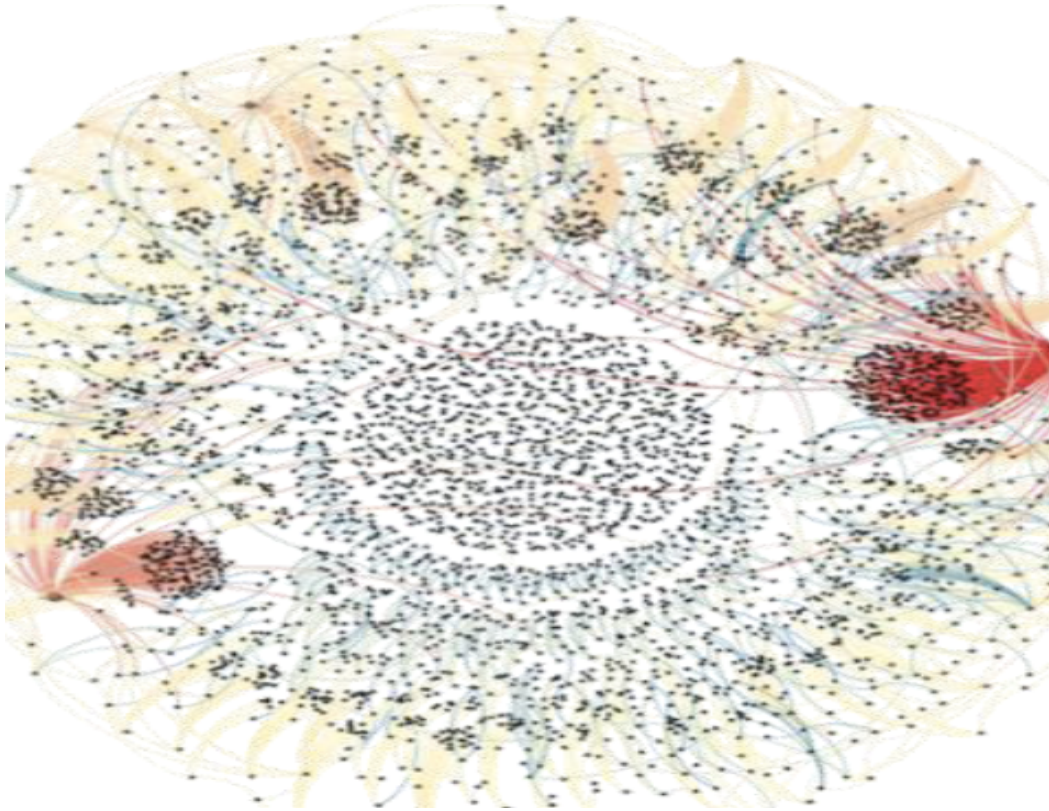
- Graph of the Internet
- Nodes = service providers
- Edges : Connections



[CHK+07]

ANOTHER EXAMPLE

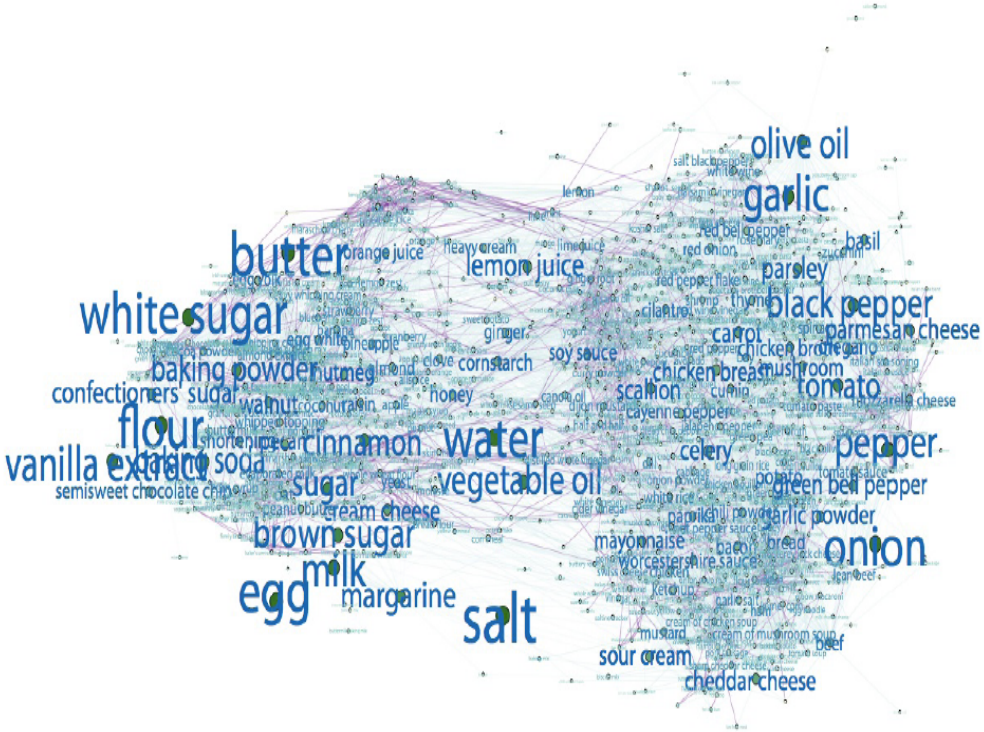
- Mubarak's resignation announcement.
- Nodes = Twitter user
- Edges : retweet on #jan25 hashtag



http://gephi.org/2011/the-egyptian-revolution-on-twitter/

LAST EXAMPLE !

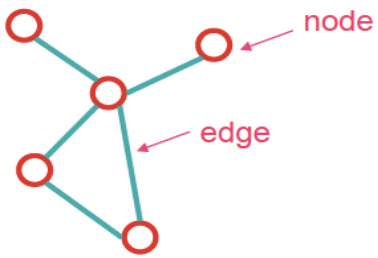
- Ingredients network extracted fom cooking receipts
- Nodes = Ingredients
- Edges : usage in the same receipt.



[Lada Adamic Course on Network Analysis: Coursera]

NOTATIONS

A graph $G =< V, E \subseteq V \times V >$



- V is the set of nodes (a.k.a. vertices, actors, sites)
- E is the set of edges (a.k.a. ties, links, bonds)

Notations

- A_G Adjacency Matrix d : $a_{ij} \neq 0$ si les nœuds $(v_i, v_j) \in E$, 0 otherwise.
- $n = |V|$
- $m = |E|$. We have often $m \sim n$
- $\Gamma(v)$: neighbor's of node v. $\Gamma(v) = \{x \in V : (x, v) \in E\}$.
- Node degree : $d(v) =\| \Gamma(v) \|$

COMPLEX NETWORK ANALYSIS: TASKS

Node oriented

- Nodes ranking in function of their importance, influence, . . .
- Applying *centrality* functions
- Applications: Ranking (ex. researchers !), Viral Marekting, Cyper-attacks (prevention); . . .

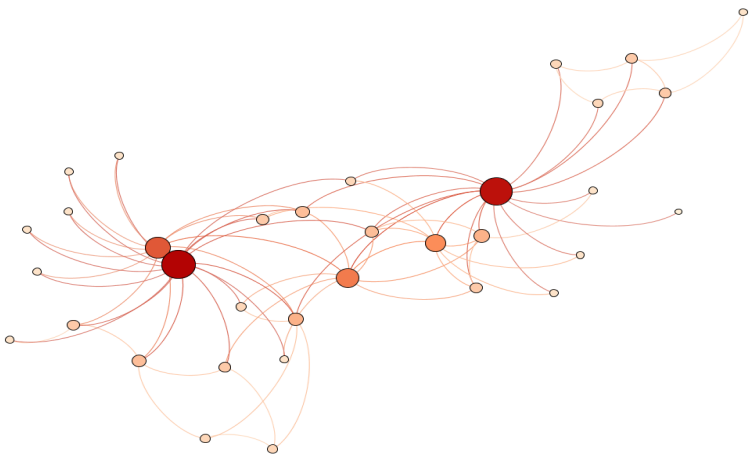
Network oriented

- Network formation & evolution models
- Link prediction, Diffusion models
- Applications: Explanation, Recommendation, Anomalies detection, infirmation diffusion, . . .

Community oriented

CENTRALITY: EXAMPLES

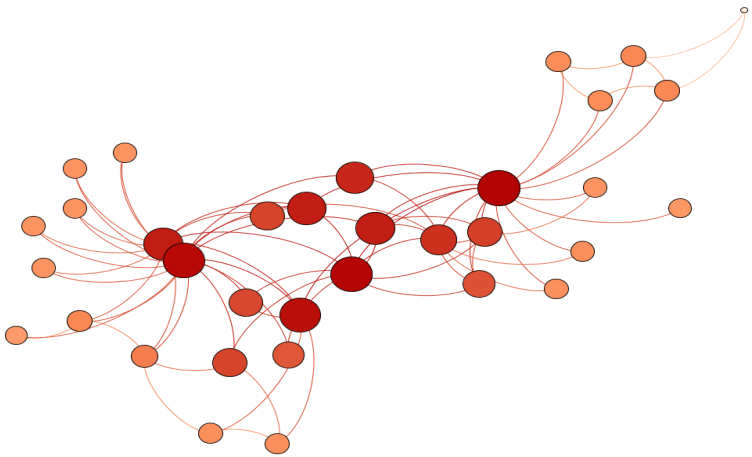
Degree centrality



- $C_d(v) = \frac{\|(\Gamma(v)\|}{max_{u \in V} \|\Gamma(u)\|}$
- Complexity : $\mathcal{O}(m)$

CENTRALITY: SOME EXAMPLES

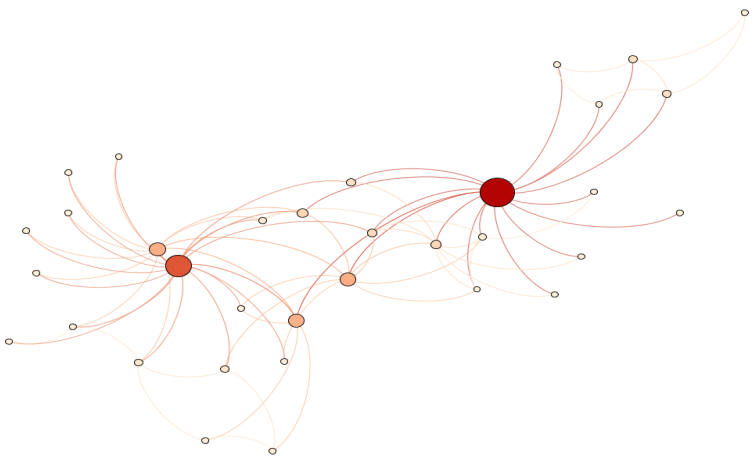
Closeness centrality



- $C_c(v) = \frac{1}{\sum_{u \in V} sp(v,u)}$
- $sp(u,v)$: shortest path length between u, v.
- Complexity = $\mathcal{O}(n \times m + n^2 logn)$

CENTRALITY: SOME EXAMPLES

Betweenness centrality

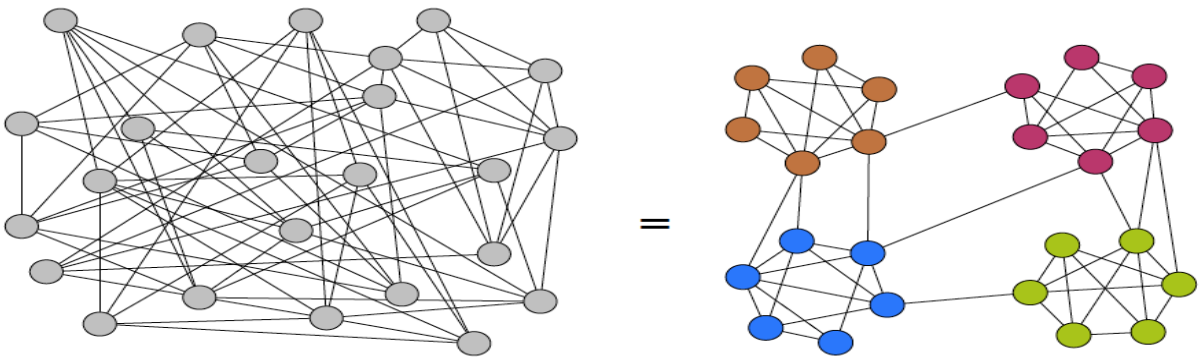


- $C_i(v) = \sum_{s,t \in V, stv} \frac{\sigma_{s,t}(v)}{\sigma_{s,t}}$
- $\sigma_{s,t}(v)$: number of shortest paths linking s to t that include v
- $\sigma_{s,t}$: total number of shortest paths linking s to t
- Complexity = $\mathcal{O}(n^3)$

COMMUNITY ?

Definitions

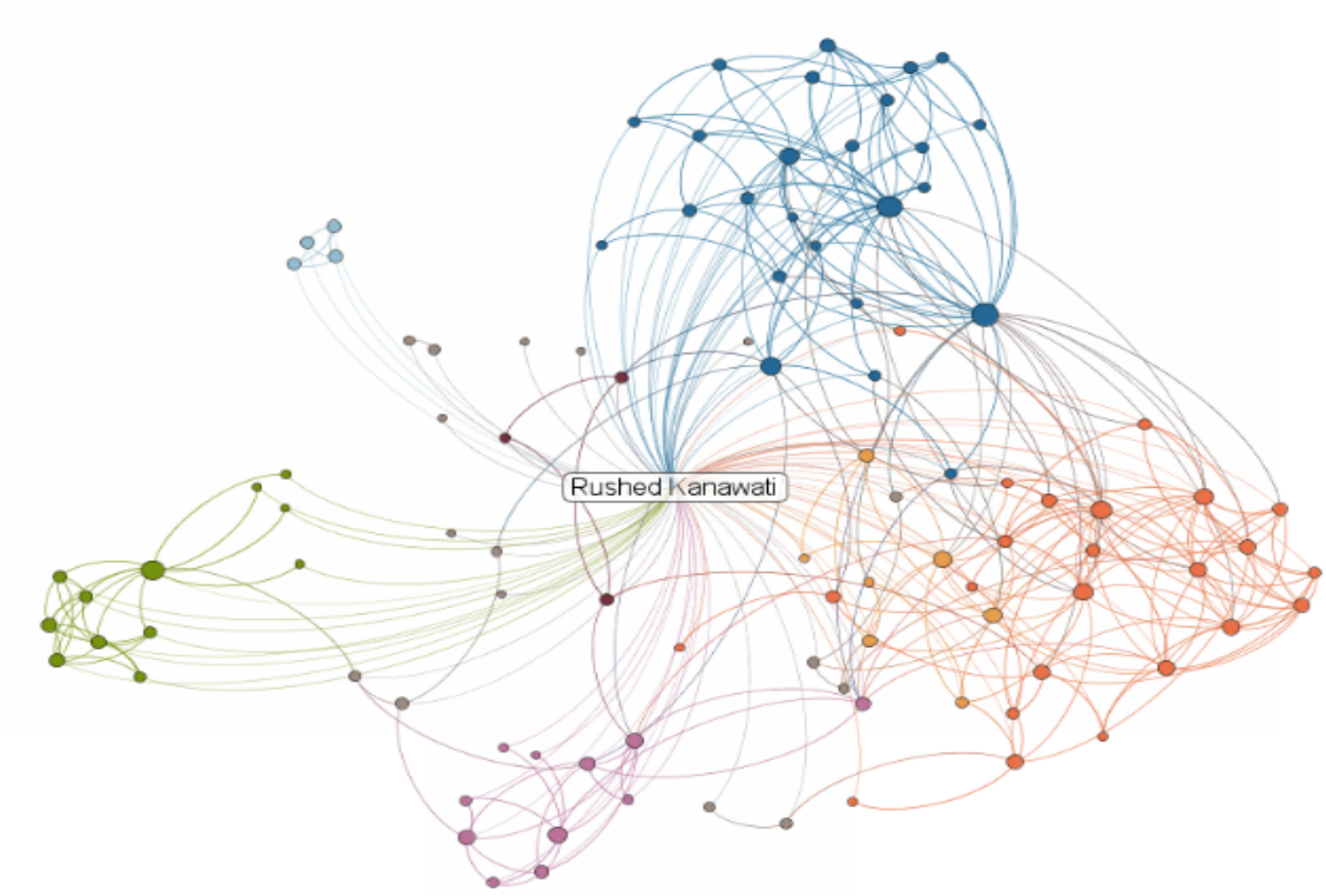
- A dense subgraph loosely coupled to other modules in the network
- A community is a set of nodes seen as one by nodes outside the community
- A subgraph where almost all nodes are linked to other nodes in the community.
- ...



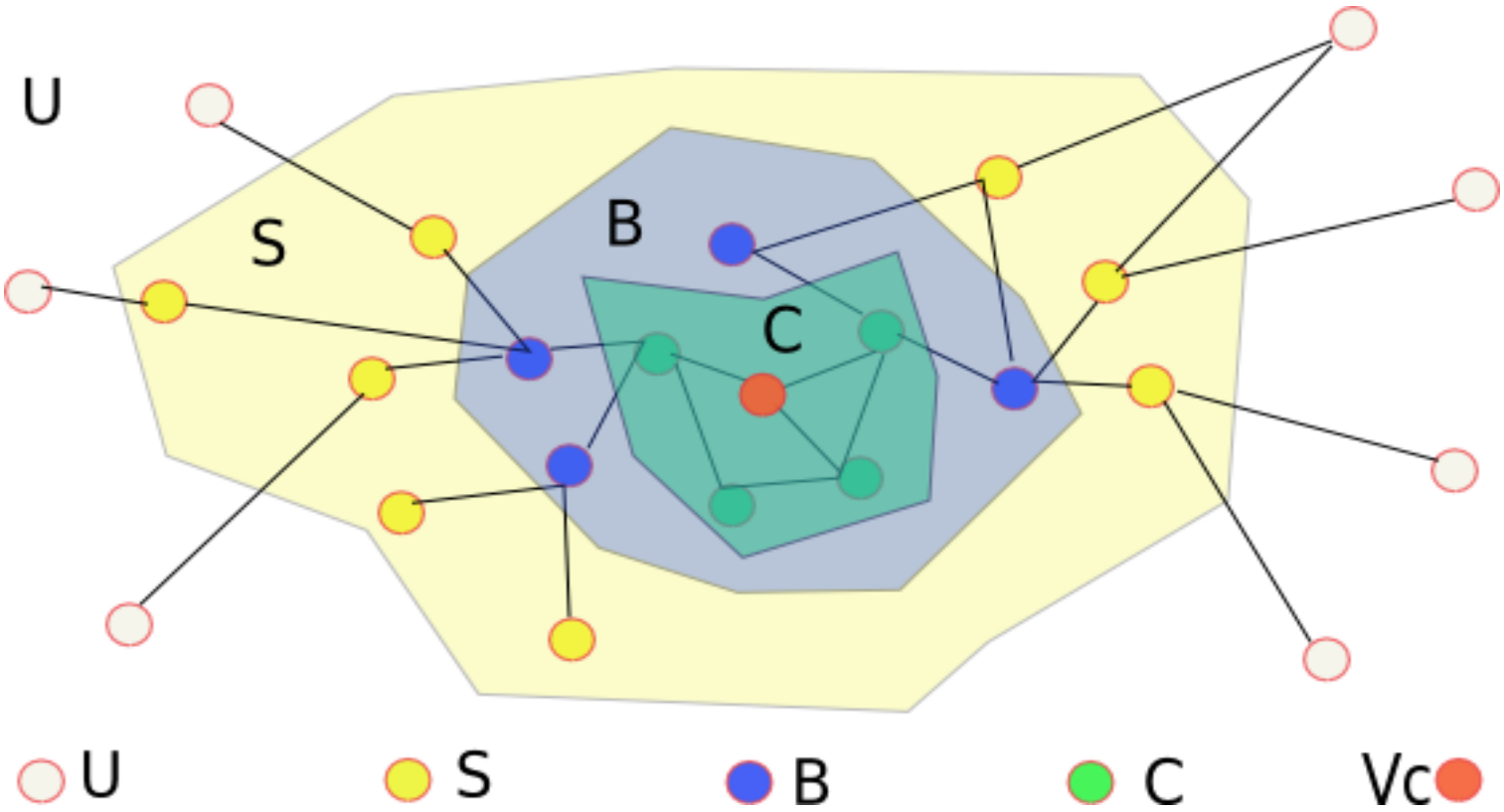
COMMUNITY DETECTION PROBLEM

- Local community identification (ego-centred).
- Network partition computing
- *Overlapping community detection*

LOCAL COMMUNITY



LOCAL COMMUNITY



LOCAL COMMUNITY

1. $C \leftarrow \{\phi\}, B \leftarrow \{n_0\}\ S \leftarrow \Gamma(n_0)$
2. $Q \leftarrow 0$ /* a community **quality function** */
3. While Q can be enhanced Do
    1. $n \leftarrow argmax_{n \in S} Q$
    2. $S \leftarrow S - \{n\}$
    3. $D \leftarrow D + \{n\}$
    4. update B, S, C
4. Return D

QUALITY FUNCTIONS : EXEMPLES I

Local modularity R [Cla05]

$$R = \frac{B_{in}}{B_{in}+B_{out}}$$

Local modularity M [LWP08]

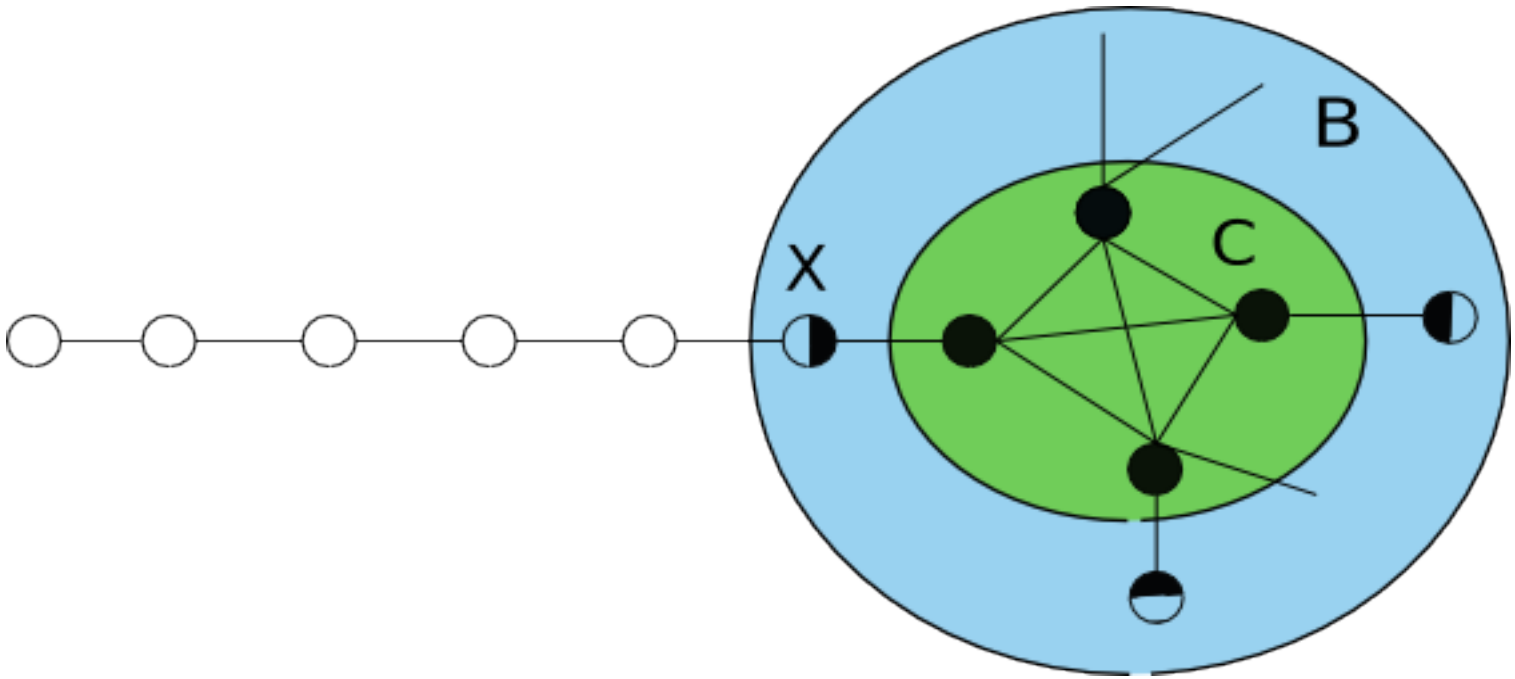
$$M = \frac{D_{in}}{D_{out}}$$

Local modularity L [CZG09]

$$L = \frac{L_{in}}{L_{ex}} \text{ where : } L_{in} = \frac{\sum_{i \in D} \|\Gamma(i) \cap D\|}{\|D\|} \text{ , } L_{ex} = \frac{\sum_{i \in B} \|\Gamma(i) \cap S\|}{\|B\|}$$

And many many others . . . [YL12]

LOCAL MODULARITY LIMITATIONS: AN EXEMPLE



- Blank nodes enhance B_{in} and D_{in} without affecting B_{out} and D_{out}
- Bank nodes will be added if M or R modularities are used
- Low precision computed communities
- **Proposed solution: Ensemble approaches**

MULTI-OBJECTIVE LOCAL COMMUNITY IDENTIFICATION [?]

Three main approaches

- Combine then Rank
- **Ensemble ranking**
- Ensemble clustering

COMBINE THEN RANK

Principle

Let $Q_i(s)$ be the local modularity value induced by node $s \in S$

$$\widetilde{Q_i(s)} = \begin{cases} \frac{Q_i(s) - \min\limits_{u \in S} Q_i(u)}{\max\limits_{u \in S} Q_i(u) - \min\limits_{u \in S} Q_i(u)} & \text{if } \max\limits_{u \in S} Q_i(u) \neq \min\limits_{u \in S} Q_i(u) \\ 1 & \text{otherwise} \end{cases}$$

$$Q_{com}(s) = \frac{1}{k} \sum_{i=1}^{k} \widetilde{Q_i(s)}$$

ENSEMBLE RANKING APPROACHES

Principle

- Rank S in function of each local modularity Q_i
- Select the winner after applying **ensemble ranking** approach
- What stopping criteria to apply ?

Stopping criteria

- *Strict policy* : All modularities should be enhanced
- *Majority policy* : Majority of local modularities are enhanced
- *Least gain policy* : At least one local modularity is enhanced.

ENSEMBLE RANKING

Problem

- Let S be a set of elements to rank by n rankers
- Let σ_i be the rank provided by ranker i
- **Goal: Compute a consensus rank of** S.

ENSEMBLE RANKING

Problem

- Let S be a set of elements to rank by n rankers
- Let σ_i be the rank provided by ranker i
- **Goal: Compute a consensus rank of** S.

Déjà Vu: Social choice algorithms, but . . .

- Small number of voters and big number of candidates
- Algorithmic efficiency is required
- Output could be a complete rank

ENSEMBLE RANKING : APPROACHES

Jean-Charles de Borda [1733-1799]

Borda

- Borda's score of $i \in \sigma_k$:
  $B_{\sigma_k}(i) = \{count(j) | \sigma_k(j) < \sigma_k(i) \; ; j \in \sigma_k\}$.
- Rank elements in function of
  $B(i) = \sum_{t=1}^{k} w_t \times B_{\sigma_t}(i)$.

ENSEMBLE RANKING : APPROACHES

Marquis de Condorcet [1743-1794]

Condorcet

- The winner is the candidate who defeats every other candidate in pairwise majority-rule election
- The winner may not exists

ENSEMBLE RANKING : APPROACHES

Marquis de Condorcet [1743-1794]

Condorcet

- The winner is the candidate who defeats every other candidate in pairwise majority-rule election
- The winner may not exists

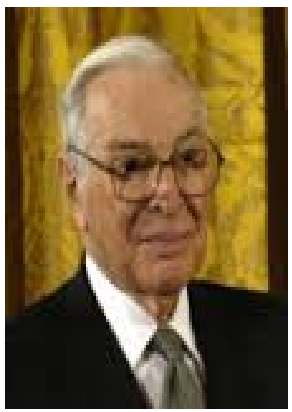
Condorcet $\neq$ Borda

- Votes : $6 \times A \succ B \succ C, 4 \times B \succ C \succ A$
- Borda winner : B
- Condorcet winner : A

ENSEMBLE RANKING : APPROACHES

Extended Condorcet criterion

If for every $a \in A$ and $b \in B$, majority prefers a to b the all elements in A should be ranked before any element in B.



Kenneth Arrow, 1921-

Arrow's Theorem

No vote rule can have the following desired proprieties :

- Every result must be achievable somehow.
- Monotonicity.
- Independence of irrelevant attributes.
- Non-dictatorship.

ENSEMBLE RANKING : APPROACHES

John Kemeny 1926-1992

Optimal Kemeny rank aggregation

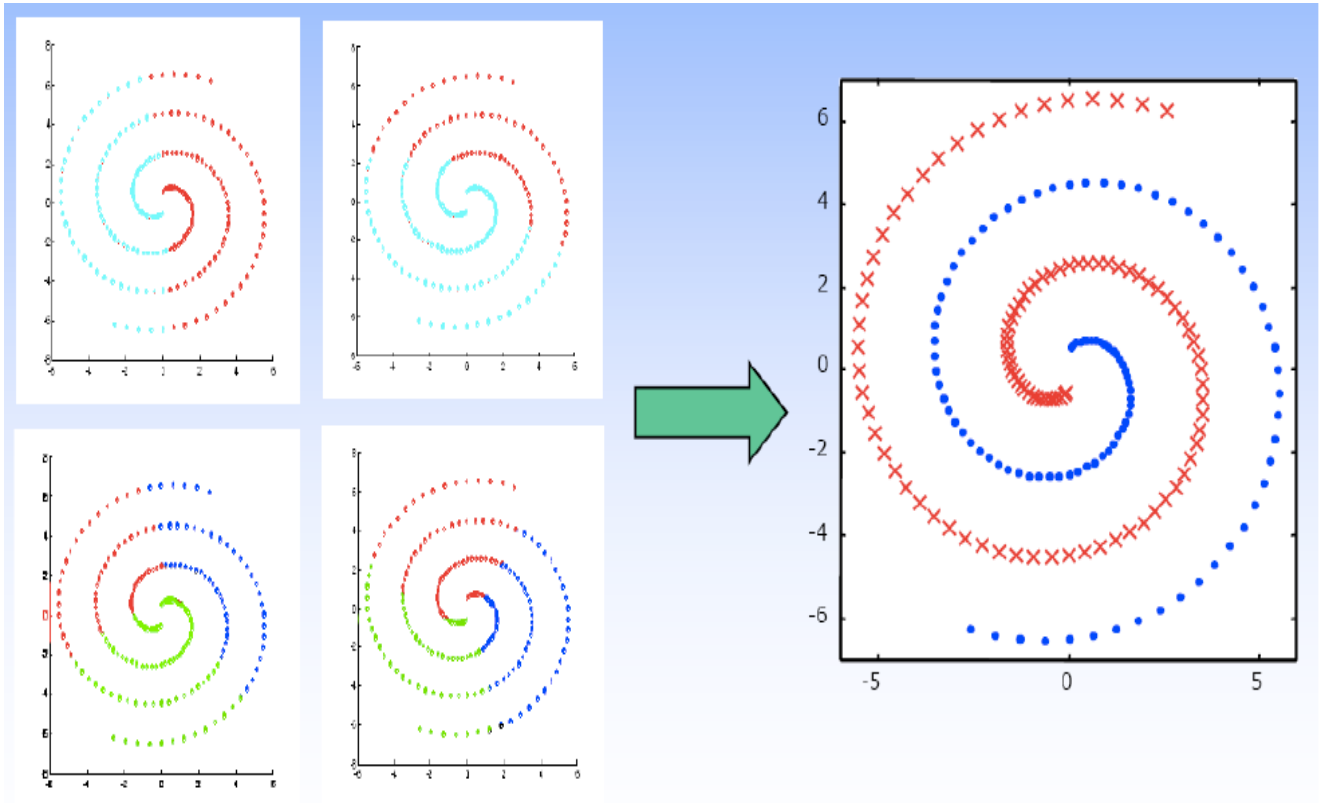
- Let $d()$ be distance over rankings σ_i (ex. Kendall τ, Spearman's footrule)
- **Find π that minimise** $\sum_i d(\pi, \sigma_i)$
- NP-Hard problem
- Approximation : Local Kemeny : two adjacent candidats are in the good order.
- **Local Kemeny** : Apply Bubble sort using the *majority preference partial order relationship*
- **Approximate Kemeny** : Apply QuickSort

ENSEMBLE CLUSTERING APPROACHES

Principle

- Let $C_{v_q}^{Q_i}$ be the the local community of v_q applying Q_i.
- We have a natural partition : $P_{Q_i} = \{C_{v_q}^{Q_i}, \overline{C_{v_q}^{Q_i}}\}$
- **Apply an ensemble clustering approach.**

ENSEMBLE CLUSTERING: PRINCIPLE



from A. Topchy et. al. Clustering Ensembles: Models of Consensus and Weak Partitions. PAMI, 2005

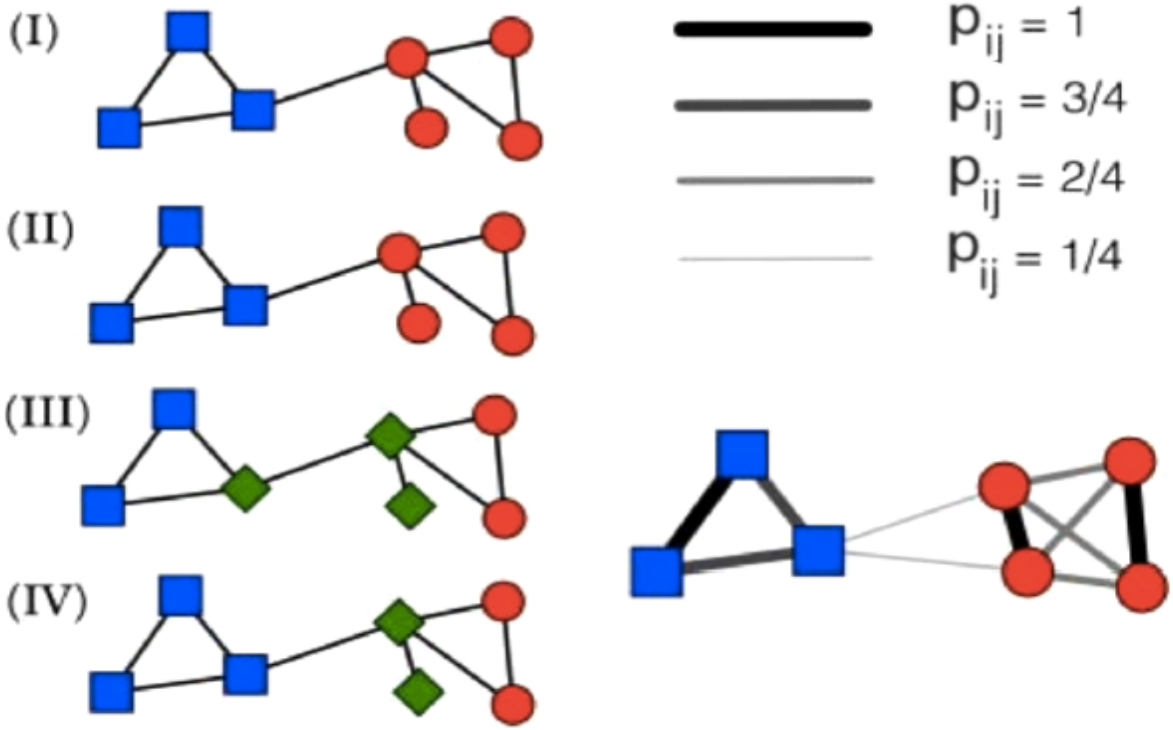
ENSEMBLE CLUSTERING: APPROACHES

CSPA: Cluster-based Similarity Partitioning Algorithm

- Let K be the number of basic models, $C_i(x)$ be the cluster in model i to which x belongs.
- Define a similarity graph on objects : $sim(v,u) = \frac{\sum_{i=1}^{K} \delta(C_i(v),C_i(u))}{K}$
- Cluster the obtained graph :

  Isolate connected components after prunning edges
  Apply community detection approach
- Complexity : $\mathcal{O}(n^2kr)$: n # objects, k # of clusters, r# of clustering solutions

CSPA : EXEMPLE



from Seifi, M. Cœurs stables de communautés dans les graphes de terrain. Thèse de l'université Paris 6, 2012

ENSEMBLE CLUSTERING: APPROACHES

HGPA: HyperGraph-Partitioning Algorithm

- Construct a hypergraph where nodes are objects and hyperedges are clusters.
- Partition the hypergraph by minimizing the number of cut hyperedges
- Each component forms a meta cluster
- Complexity : $\mathcal{O}(nkr)$

ENSEMBLE CLUSTERING: APPROACHES

MCLA: Meta-Clustering Algorithm

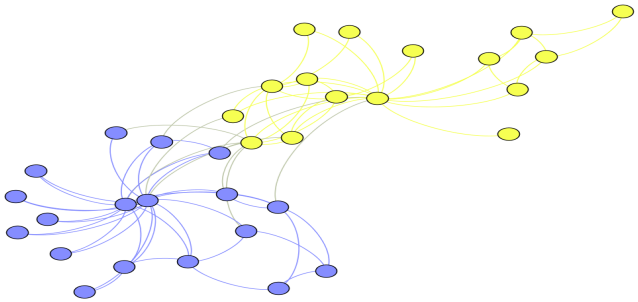
- Each cluster from a base model is an item
- Similarity is defined as the percentage of shared common objects
- Conduct meta-clustering on these clusters
- Assign an object to its most associated meta-cluster
- Complexity : $\mathcal{O}(nk^2r^2)$

EXPERIMENTS

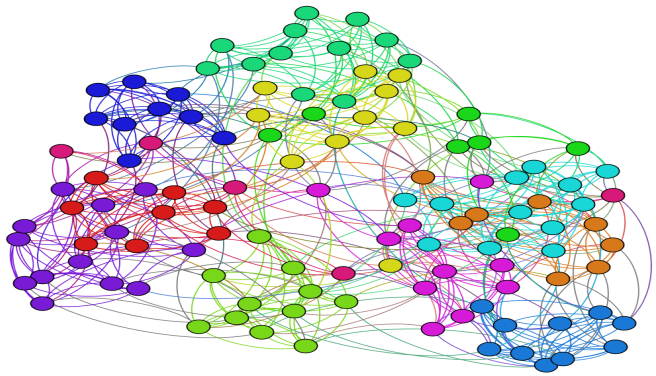
Protocol ([Bag08])

1. Apply the different algorithms on nodes in networks for which a ground-truth community partition is known.
2. For each node compute the distance between the real-partition and the computed one (Ex. NMI [Mei03])
3. Compute average and standard deviation for the network.

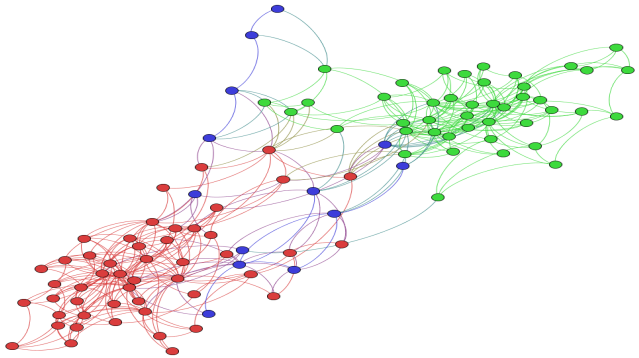
DATASETS



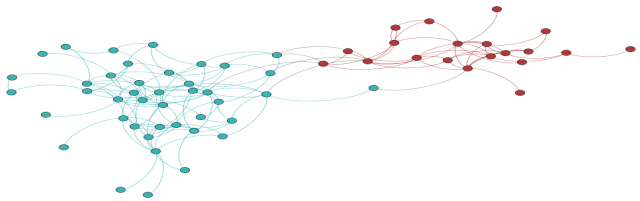
Zachary's Karate Club



Football network

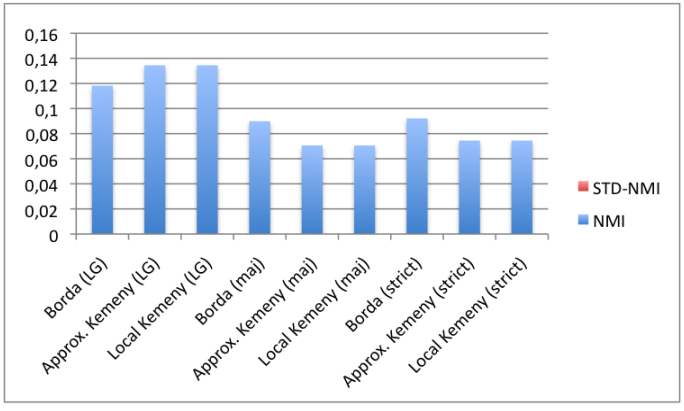


US Political books network

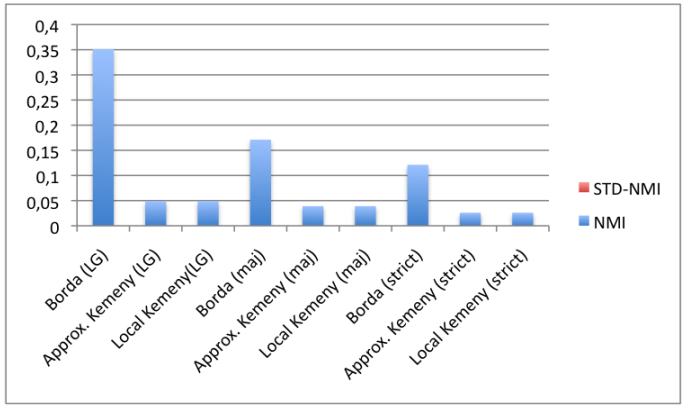


Dolphins social network

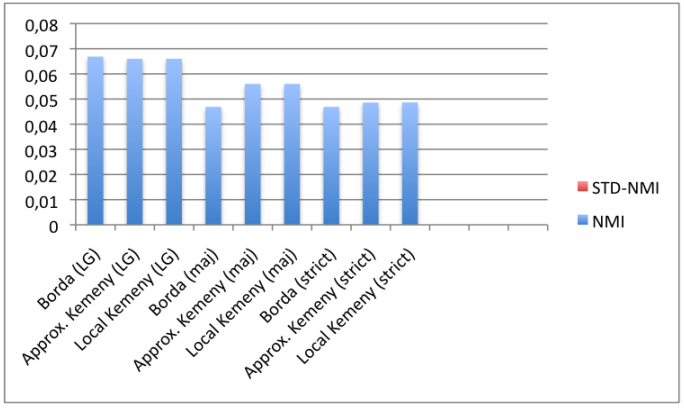
RESULTS : EVALUATING STOPPING CRITERIA (NMI)



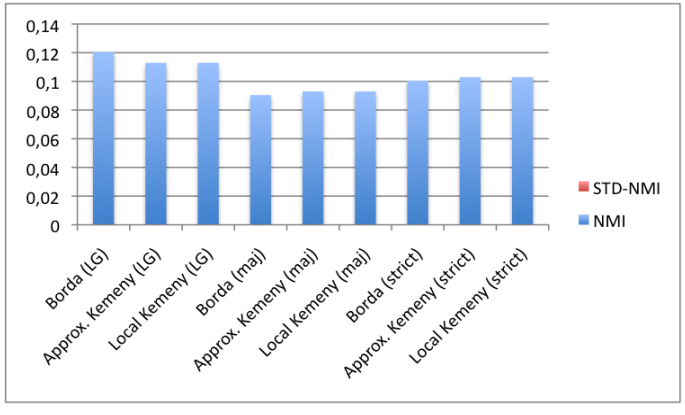
Zachary's Karate Club



Football network

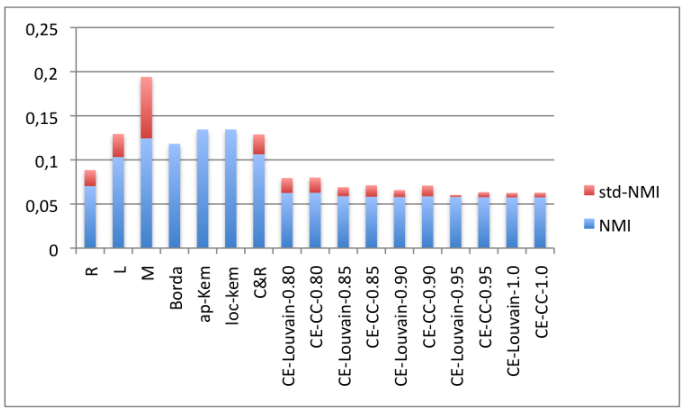


US Political books network

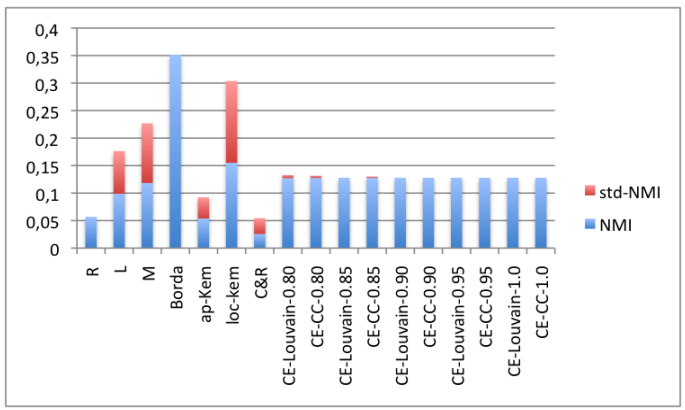


Dolphins social network

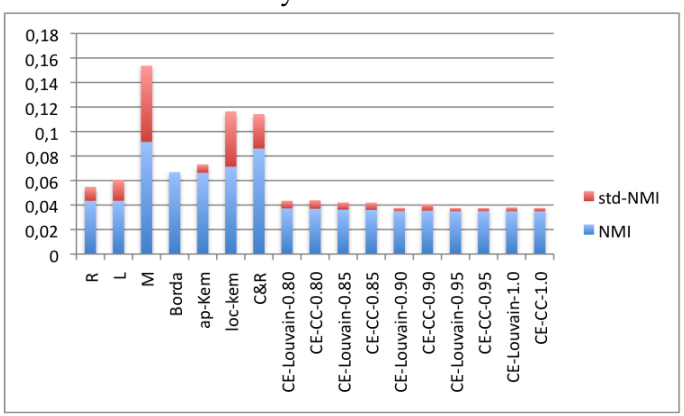
RESULTS : COMPARATIVE RESULTS (NMI)



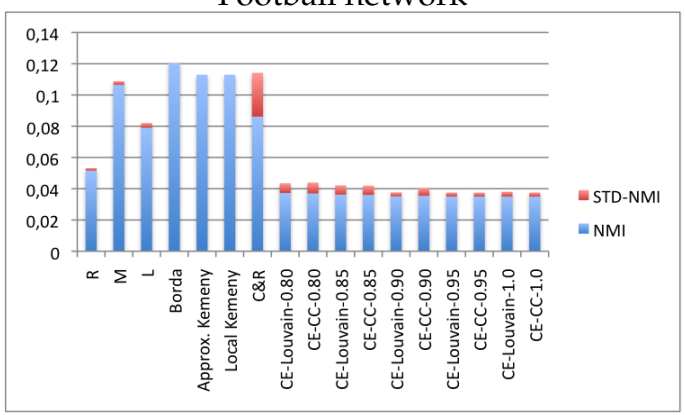
Zachary's Karate Club



Football network

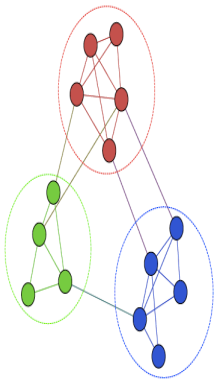


US Political books network



Dolphins social network

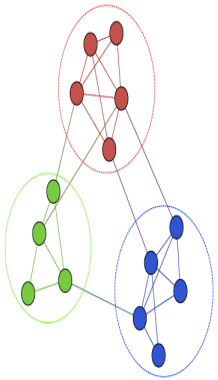
GLOBAL COMMUNITIES DETECTION



Problem

- Divid the set of nodes in a number of (overlapping) subsets such that induced subgraphs are dense and loosley coupled.

GLOBAL COMMUNITIES DETECTION



Problem

- Divid the set of nodes in a number of (overlapping) subsets such that induced subgraphs are dense and loosley coupled.

Recommended readings

- **S. Fortunato**. *Community detection in graphs*. Physics Reports, 2010, 486, 75-174
- **L. Tang, H. Liu**. *Community Detection and Mining in Social Media*, Morgan Claypool Publishers, 2010
- **R. Kanawati**, *Détection de communautés dans les grands graphes d'interactions multiplexes : état de l'art*, (RNTI 2014), hal-00881668

COMMUNITIES DETECTION: METHODS

Classification

- **Group-based**
  Nodes are grouped in function of shared topological features (ex. clique)
- **Network-based**
  Clustering, Graph-cut, block-models, ***modularity optimization***
- **Propagation-based**
  Unstable approaches, good when used with ensemble clustering
- **Seed-centred**
  from local community identification to global community detection

GROUP-BASED APPROCHES

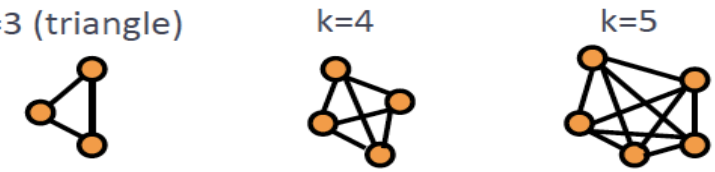
Principle

Search for special (dense) subgraphs:

- k-clique
- n-clique
- γ-dense clique
- K-core

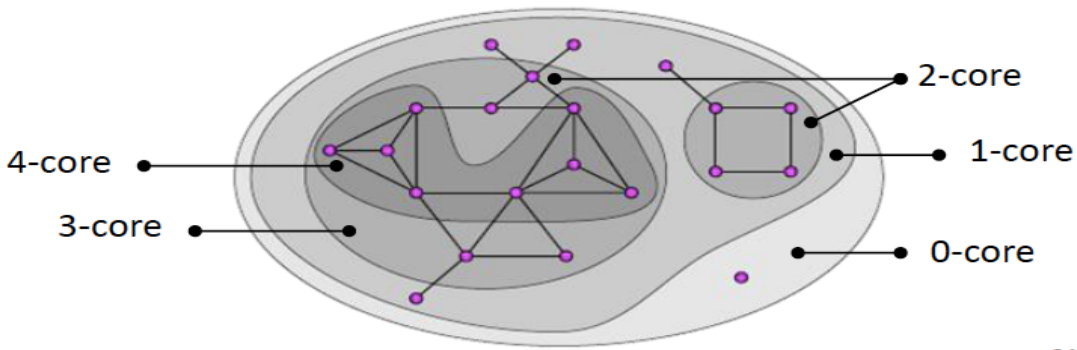
GROUP-BASED APPROCHES

- k-clique



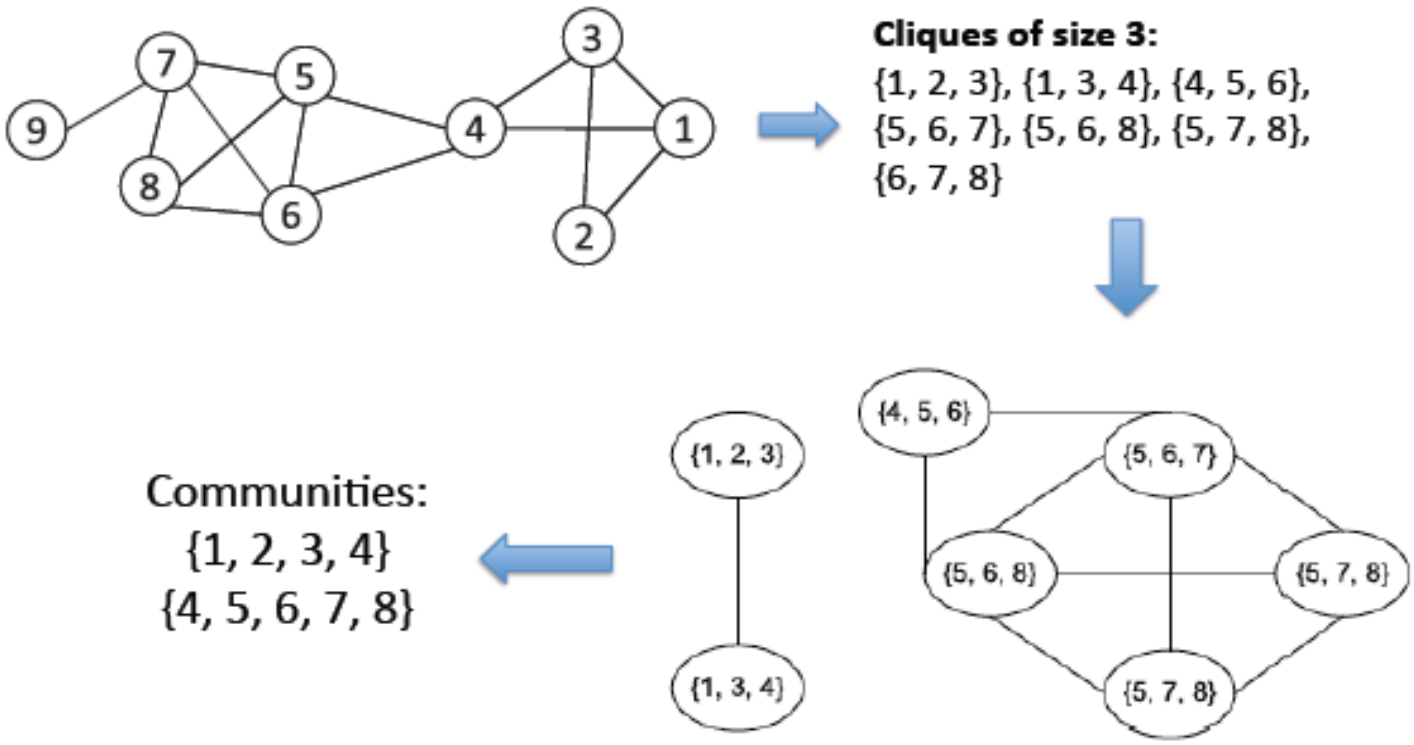
- N-clique

- k-core



from Symeon Papadopoulos, Community Detection in Social Media, CERTH-ITI, 22 June 2011

EXAMPLE: CLIQUE PERCOLATION



Suits fairly dense graph.

NETWORK-BASED APPROCHES

Clustering approaches

- Apply classical clustering approaches using *graph-based* diastase function
- Different types of Graph-based distances: neighborhood-based, path-based (Random-walk)
- **Usually requires the number of clusters to discover**

MODULARITY OPTIMIZATION APPROACHES

Modularity: a partition quality criteria

$$Q(\mathcal{P}) = \frac{1}{2m} \sum_{c \in \mathcal{P}} \sum_{i,j \in c} (A_{ij} - \frac{d_i d_j}{2m}) \quad (1)$$

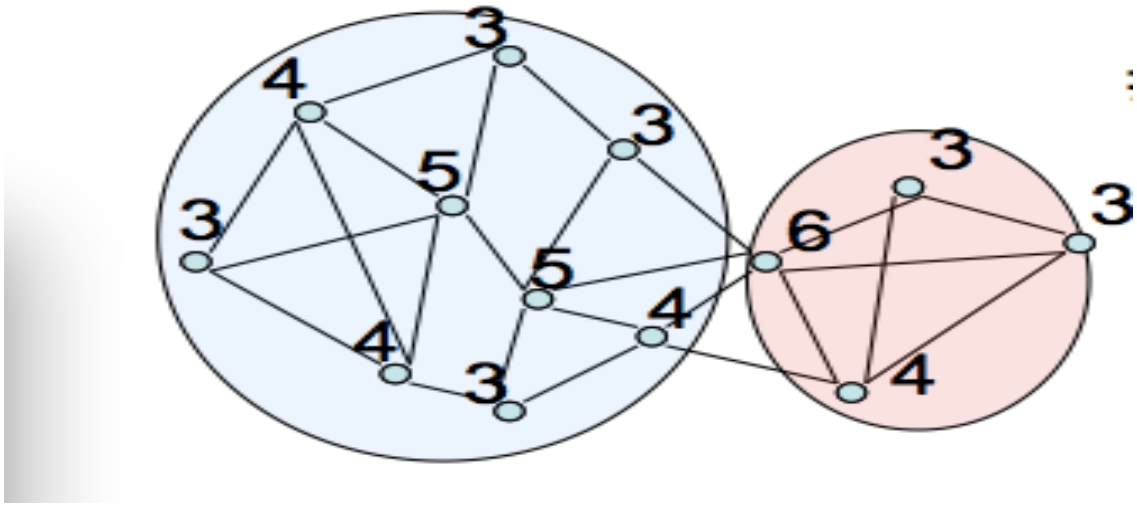


Figure: Example : $Q = \frac{(15+6)-(11.25+2.56)}{25} = 0.275$

MODULARITY OPTIMIZATION APPROACHES

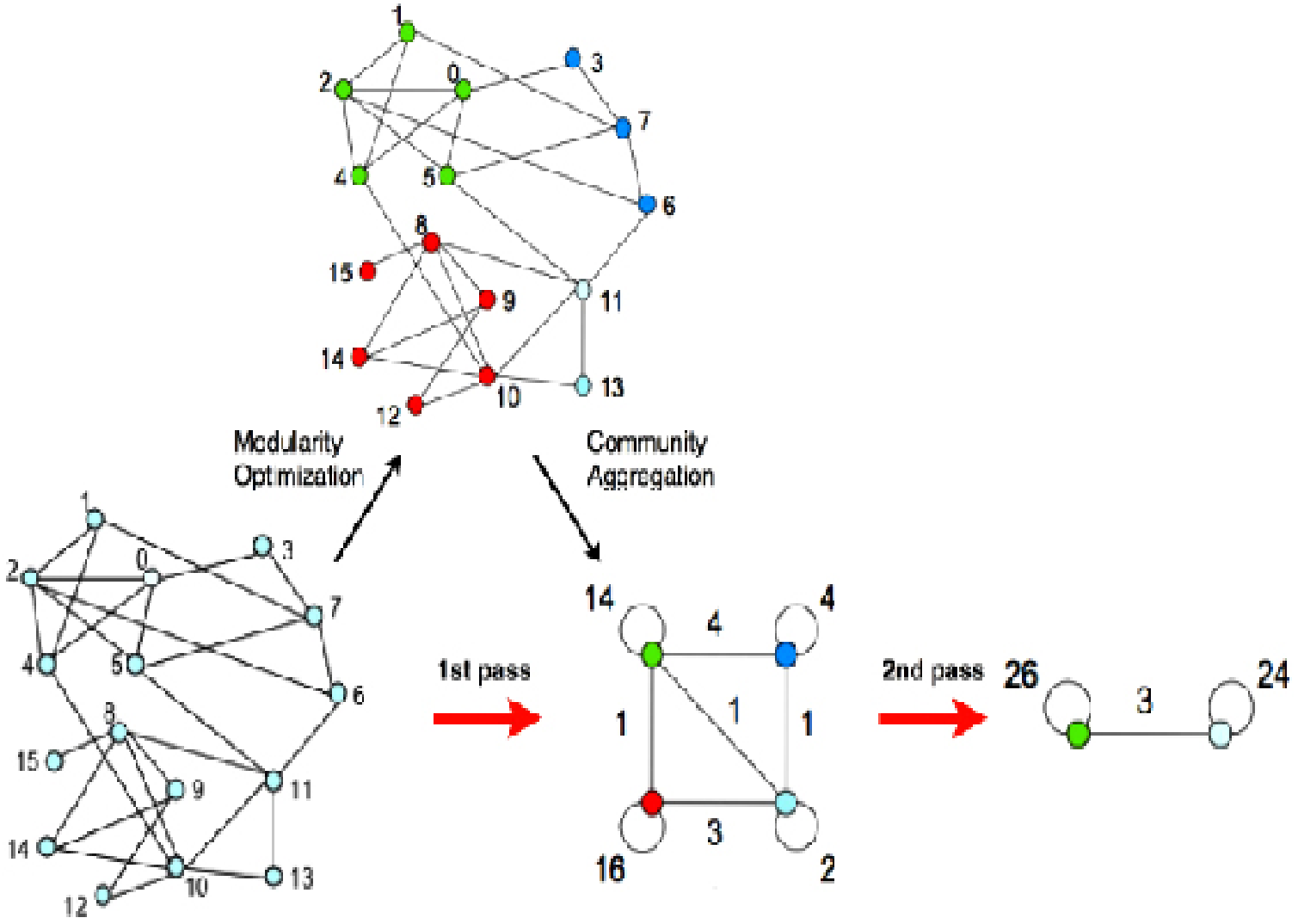
- Applying classical optimization algorithms (ex. Genetic algorithms [Piz12]).
- Applying hierarchical clustering and select the level with Q_{max} (ex. Walktrap [PL06])
- Divisive approach : Girvan-Newman algorithm [GN02]
- Greedy optimization : Louvain algorithm [BGL08]
- ...

EXAMPLE: GIRVAN-NEWMAN ALGORITHM

1. Compute betweenness centrality for each edge.
2. Remove edge with highest score.
3. Re-compute all scores.
4. Repeat 2nd step.

Complexity : $\mathcal{O}(n^3)$

EXAMPLE: LOUVAIN APPROACH



MODULARITY OPTIMIZATION LIMITATIONS

Hypothesis

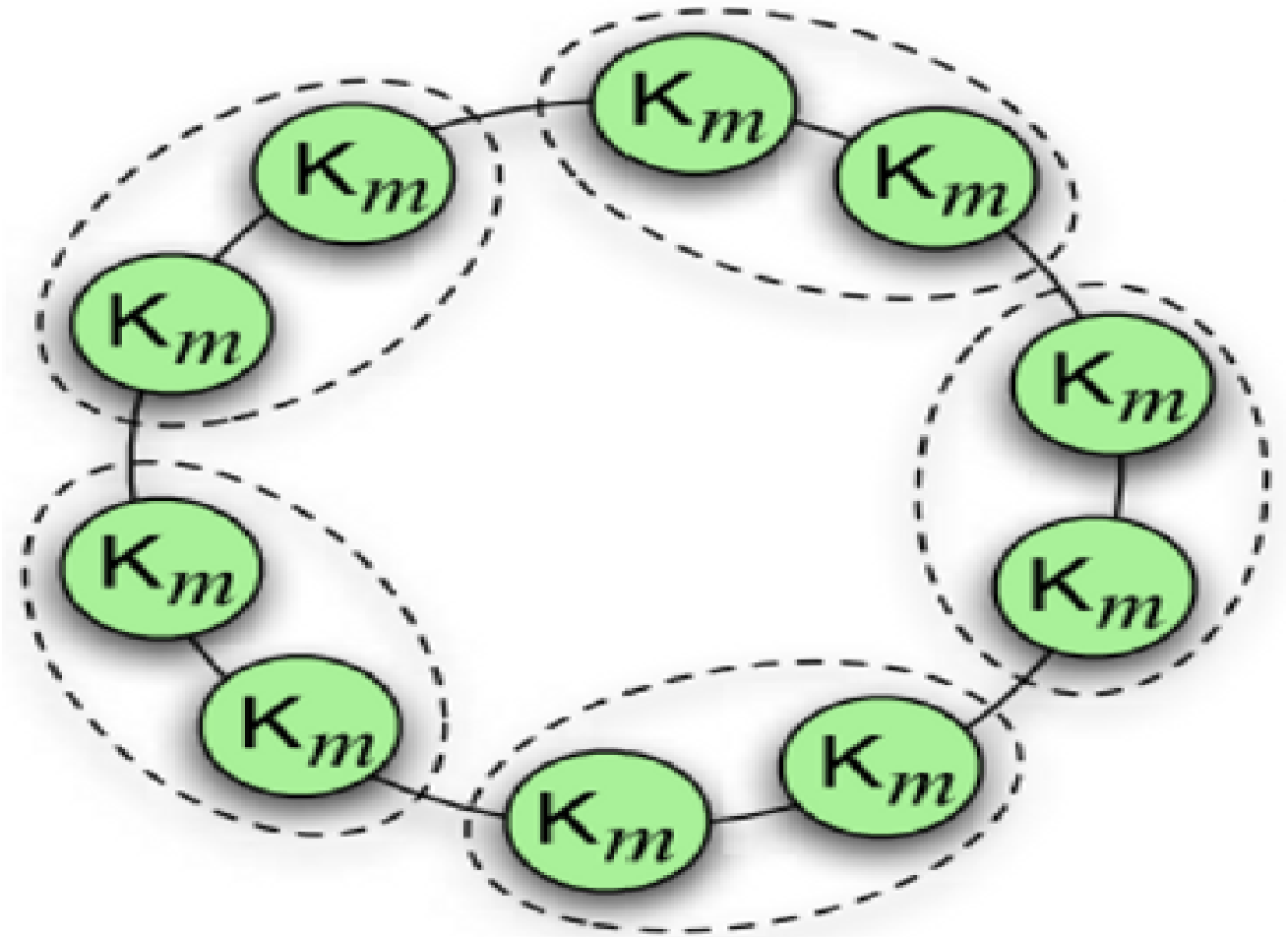
The best partition of a graph is the one that maximize the modularity.

If a network has a community structure, then it is popsicle to find a precise partition with maximal modularity

If a network has a community structure, then partitions inducing high modularity values are structurally similar.

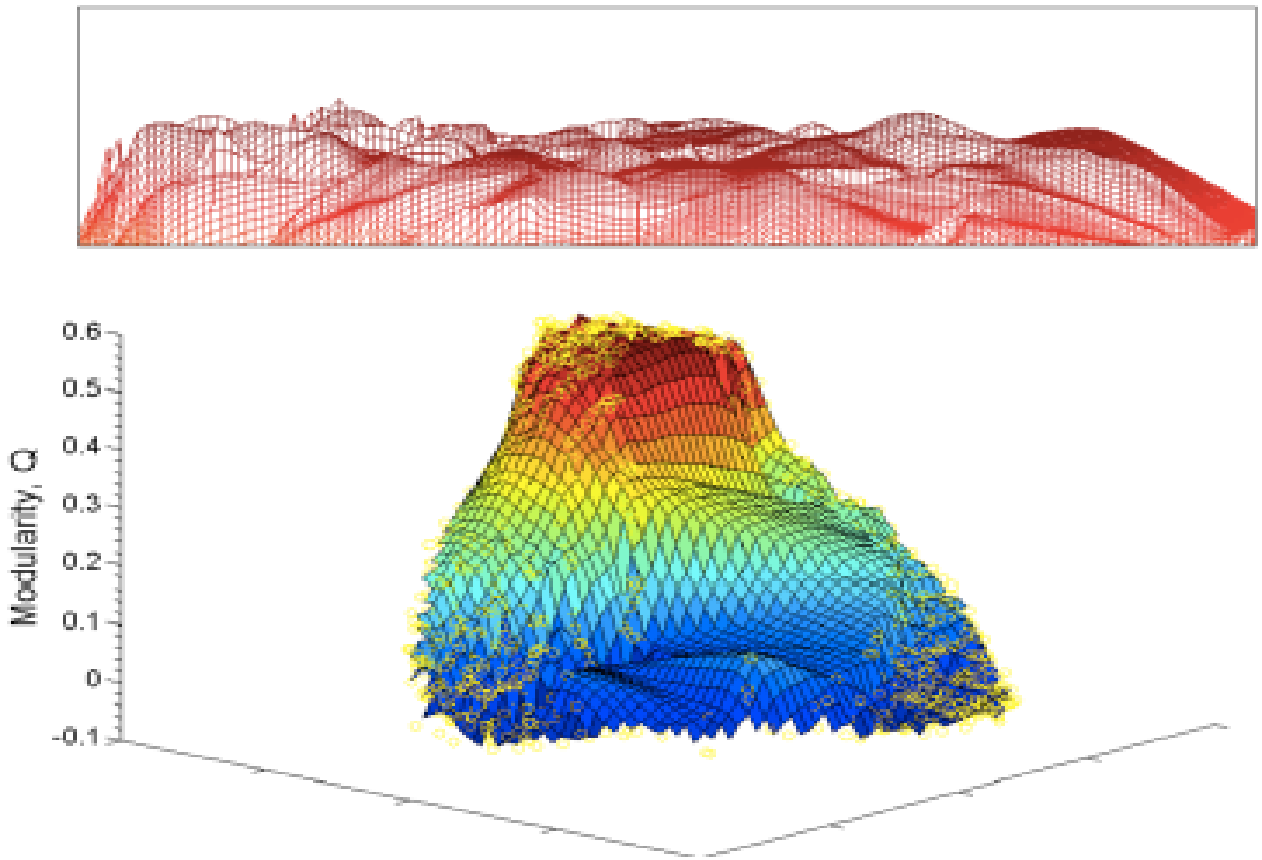
All three hypothesis do not hold [GdMC10, LF11].

MODULARITY RESOLUTION LIMITE



For $m = 3$ $Max_Q = 0.675$ while natural partition has $Q = 0.65$

Modularity distribution



PROPAGATION-BASED APPROACHES

Algorithm 1 Label propagation

Require: $G =< V, E >$ a connected graph,

1: Initialize each node with unique label l_v
2: **while** Labels are not stable **do**
3: **for** $v \in V$ **do**
$l_v =_l |\Gamma^l(v)|$ /* random tie-breaking */
4: **end for**
5: **end while**
6: **return** communities from labels

$\Gamma^l(v)$: set of neighbors having label l

LABEL PROPAGATION

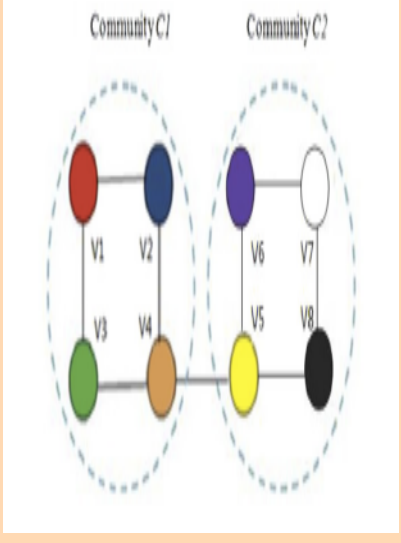
Advantages

- Complexity : $\mathcal{O}(m)$
- Highly parallel

Disadvantages

- No convergence guarantee, oscillation phenomena
- Low robustness

  Different runs yields very different community structure due to randomness



LABEL PROPAGATION: CONVERGENCE ENHANCEMENT

- Asynchronous label update, [RAK07]
- Semi-synchronous label update (graph coloring + propagation by color) [CG12].

- Making stability even worse !
- Harden parallel implementations.

ROBUST LABEL PROPAGATION

- *Label hop attenuation* [LHLC09]
- *Balanced label propagation* [SB11].

- Multiplex approach !
  adding new neighborhood similarity based relationships between adjacent nodes $\rightarrow$ *multiplex network*

- **Ensemble clustering** $\rightarrow$ Communities core [OGS10, SG12, LF12].

PARALLEL LP

Algorithm 1: PLP: Parallel Label Propagation

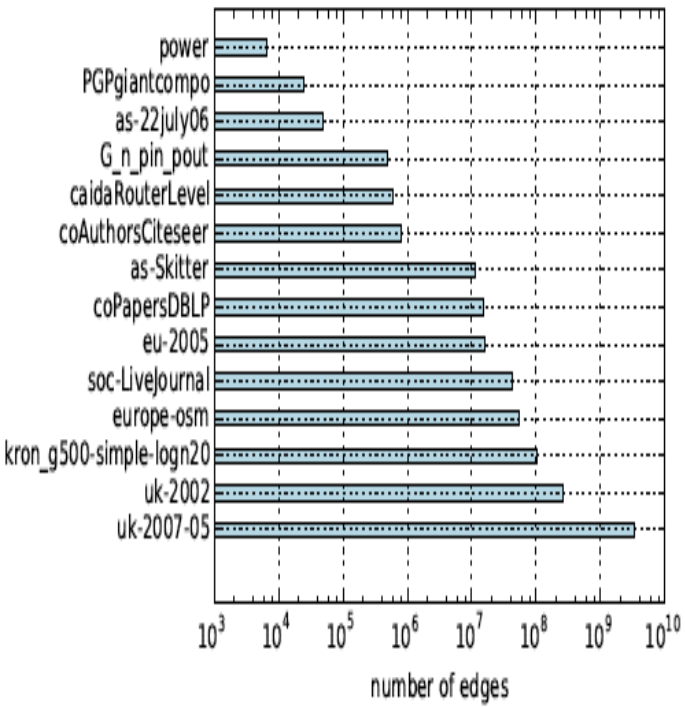
Input: graph $G = (V, E)$
Result: communities $\zeta : V \to \mathbb{N}$

```
1  parallel for v ∈ V
2      ζ(v) ← id(v)
3  updated ← n
4  V_active ← V
5  while updated > θ do
6      updated ← 0
7      parallel for v ∈ {u ∈ V_active : deg(u) > 0}
8          l* ← arg max_l {Σ_{u∈N(v), ζ(u)=l} ω(v, u)}
9          if ζ(v) ≠ l* then
10             ζ(v) ← l*
11             updated ← updated + 1
12             V_active ← V_active ∪ N(v)
13         else
14             V_active ← V_active \ {v}
15 return ζ
```

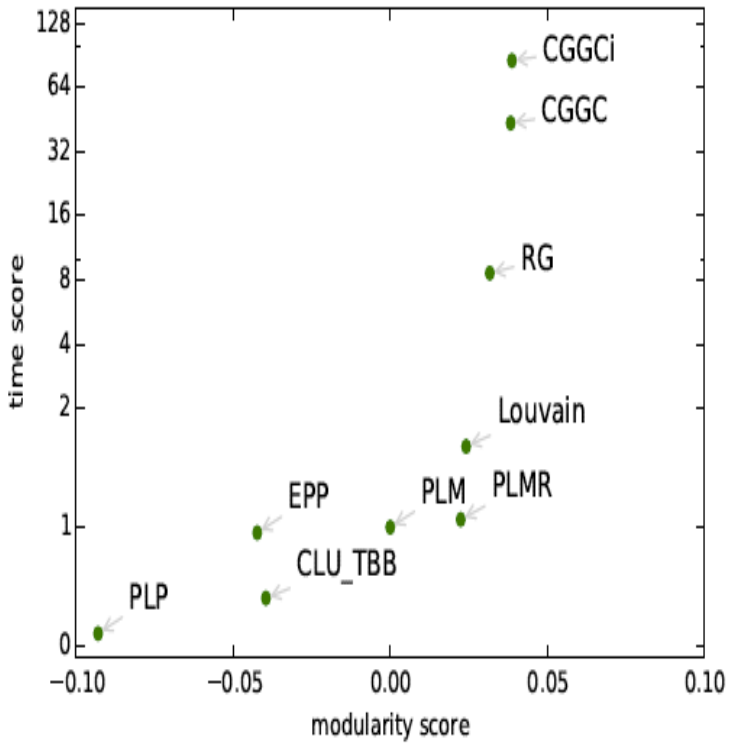
LABEL PROPAGATION PREPROCESSING (EPP)

1. Apply an ensemble clustering on results of basic Parallel LP
2. Coarse the graph according to obtained communities
3. Apply a high quality community detection algorithm on coarsed graph.
4. Expand obtained results to the initial graph.

EVALUATIONS



benchmark networks



Pareto front

SEED-CENTRIC ALGORITHMS (KANAWATI, SCSM'2014)

Algorithm 2 General seed-centric community detection algorithm

Require: $G =< V, E >$ a connected graph,

1: $\mathcal{C} \leftarrow \emptyset$
2: $S \leftarrow$ **compute_seeds(G)**
3: **for** $s \in S$ **do**
4: $\quad C_s \leftarrow$ **compute_local_com(s,G)**
5: $\quad \mathcal{C} \leftarrow \mathcal{C} + C_s$
6: **end for**
7: **return compute_community($\mathcal{C}$)**

SEED-CENTRIC APPROACHES

Table: Characteristics of major seed-centric algorithms

Algorithm	Seed Nature	Seed Number	Seed selection	Local Com.	Com. computation
Leaders-Followers [SZ10]	Single	Computed	informed	Agglomerative	-
Top-Leaders [KCZ10]	Single	Input	Random	Expansion	-
PapadopoulosKVS12	Subgraph	Computed	Informed	Expansion	-
WhangGD13	Single	Computed	informed	Expansion	-
BollobasR09	Subgraph	Computed	informed	expansion	-
Licod [Kan11]	Set	Computed	Informed	Agglomerative	-
Yasca [**?**]	Single	Computed	Informed	Expansion	Ensemble clustering

EXEMPLE: LICOD (KANAWATI, SOCILACOMP'2011)

Licod : the idea !

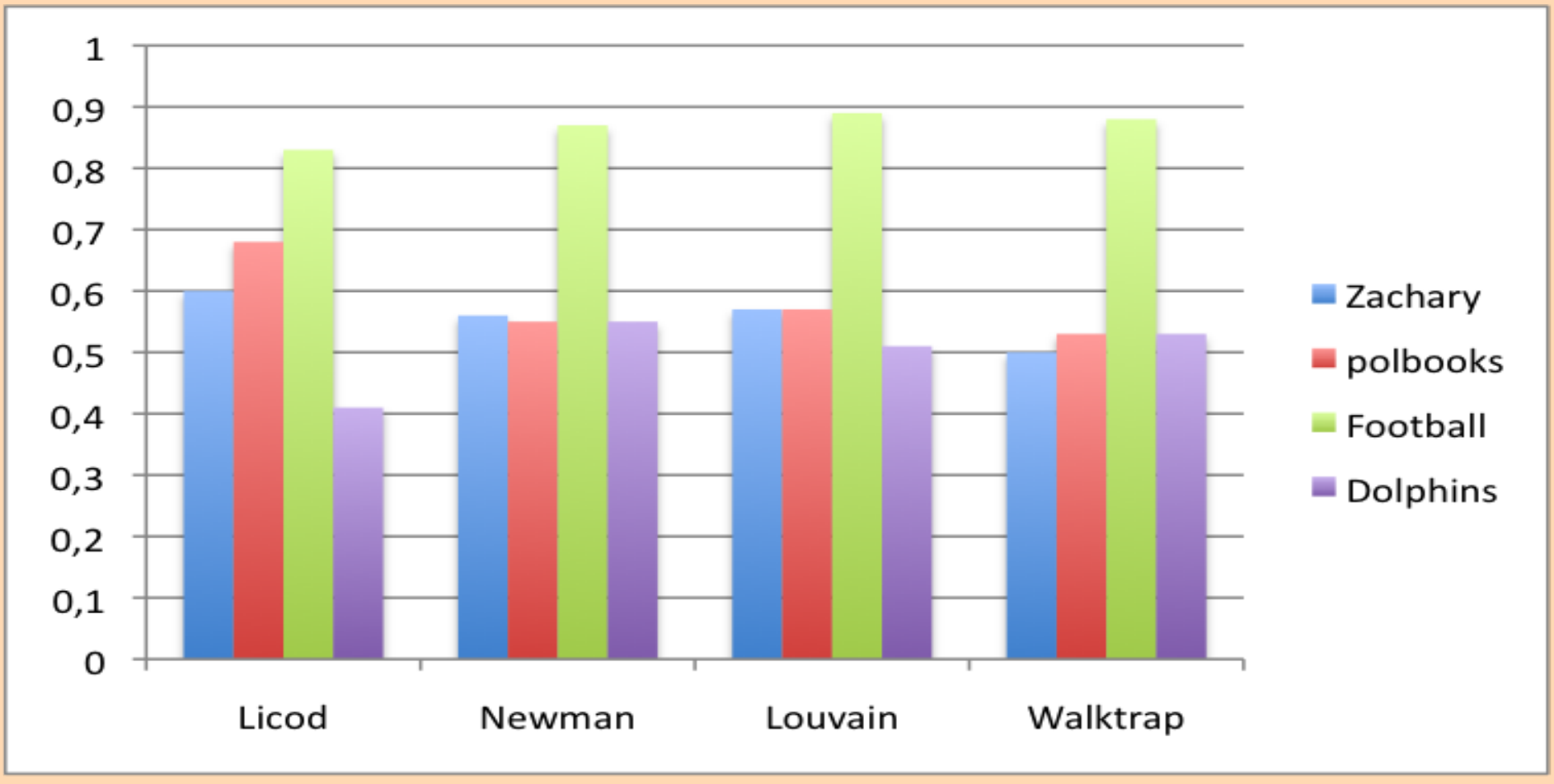
1. Compute a set of seeds that are likely to be leaders in their communities

   Heuristic : nodes having higher centralities than their neighbors

2. Each node in the graph ranks seeds in function of its own preference
3. Each node modify its preference vector in function of neighbor's preferences
4. iterate max times or till convergence.

LICOD: SOME RESULTS

Comparative results on benchmark networks : NMI



More results in ([YK14])

THE YASCA ALGORITHM (KANAWATI, COCOON'2014)

The algorithm

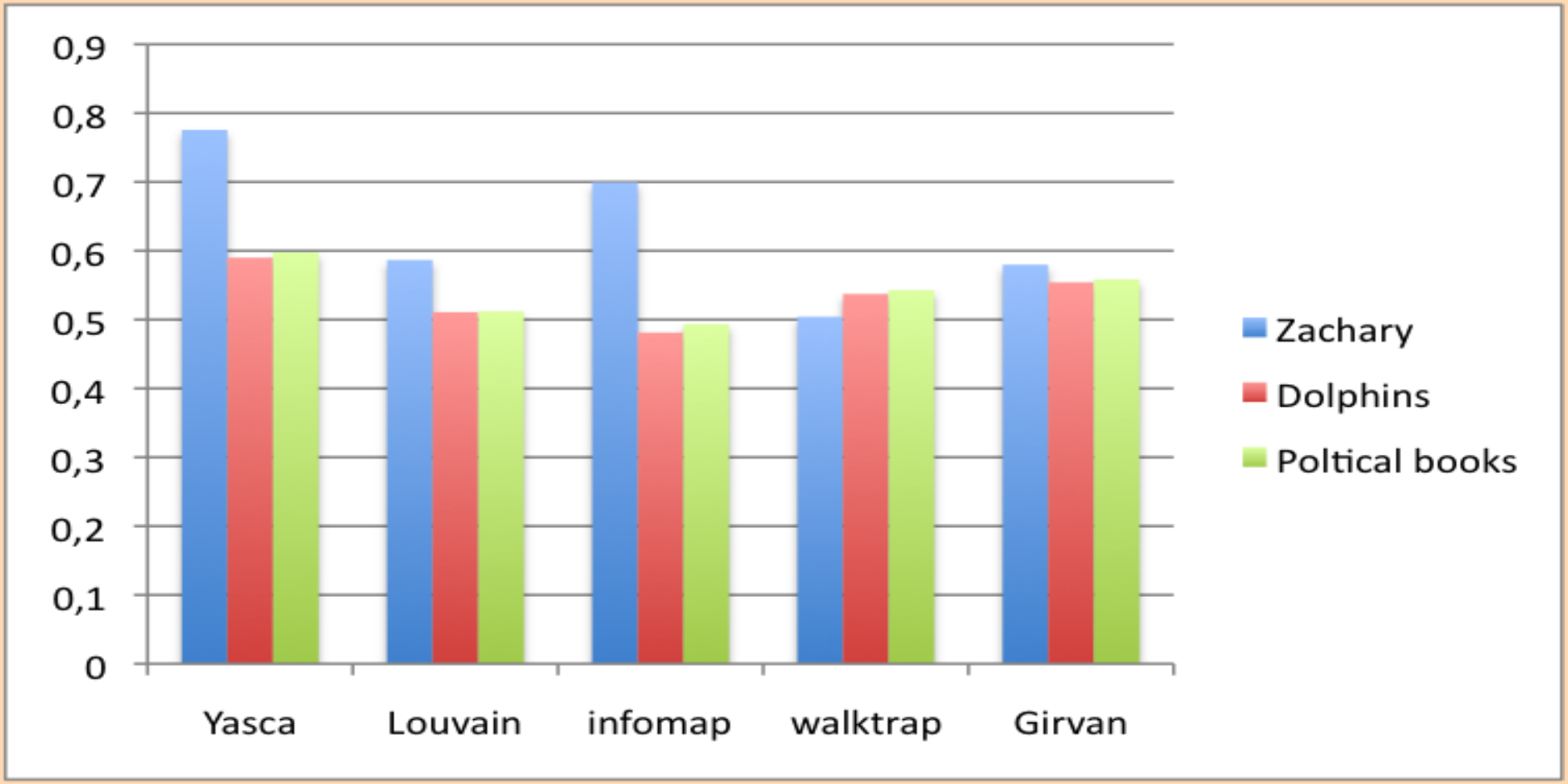
Yet Another Seed-centric Community detection Algorithm

1. Compute a set of **diverse** seed nodes
2. For each node compute a bi-partition of the whole graphe based on local community identification.
3. Apply ensemble clustering to obtain a graph partition.

YASCA: SOME RESULTS

Comparative Results on benchmark networks : NMI

Select 15% of high central nodes and 15% of low central nodes



EVALUATION METHODS

- Topological criteria : is it useful for applications ?
- Ground-truth comparaison : hard to find on large-scale
- Task-oriented approaches : Clustering , Recommendation, link prediction, . . .

TOPOLOGICAL EVALUATION CRITERIA

Quality of $\mathcal{C} = \{S_1, \ldots, S_i\}$:

$$Q(\mathcal{C}) = \frac{\sum_i f(S_i)}{|\mathcal{C}|} \tag{2}$$

$f()$ is a single- community quality metric

4 types of quality functions

- Internal connectivity
- Externeal connectivity
- Hybrid functions
- Model-based functions: *the modularity*

INTERNAL CONNECTIVITY FUNCTIONS

Internal density : $f(S) = \frac{2 \times m_S}{n_S \times (n_S - 1)}$

Average Degree : $f(S) = \frac{2 \times m_S}{n_S}$

FOMD: Fraction over Median Degree : $f(s) = \frac{|\{u : u \in S, |(u,v), v \in S| > d_m\}|}{n_S}$,
d_m est le médian de degrés des nœuds dans V

TPR: Triangle Participation Ratio : $\frac{|\{u \in S\} : \exists v, w \in S : (u,v), (w,v), (u,w) \in E|}{n_S}$

EXTERNAL CONNECTIVITY FUNCTIONS

$$\text{Expansion} : f(S) = \frac{C_S}{n_S}$$

$$\text{Cut} : f(S) = \frac{C_s}{n_S \times (N - n_S)}$$

HYBRID FUNCTIONS

- Conductance : $f(S) = \frac{C_S}{2m_S + C_S}$
- MAX-ODF : Out Degree Fraction : $f(S) = max_{u \in S} \frac{|\{(u,v) \in E, v \notin S\}|}{d(u)}$
- AVG-ODF : $f(S) = \frac{1}{n_S} \times \sum_{u \in S} \frac{|\{(u,v) \in E, v \notin S\}|}{d(u)}$

GROUND-TRUTH BASED EVALUATION

- Principle : Computing a *similarity* between obtained partition and a ground-truth one.
- Ground-truth :

  Expert defined.
  Model-generated
- Two types of metrics :

  Agreement-based metrics: purity, rand, ARI
  Information theory metrics: MI, NMI, . . . , etc.

PURITY

- $\mathcal{P} = \{p_1, p_2, \ldots, p_k\}$: a computed partition
- $\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$: ground-truth partition
- $purity(\mathcal{P}, \mathcal{R}) = \frac{1}{|V|} \sum_{j=1}^{k} max_i(|p_k \cap r_i|)$

RAND INDEX

- $\mathcal{P} = \{p_1, p_2, \ldots, p_k\}$: a computed partition
- $\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$: ground-truth partition
- a : # **pairs of nodes** in a same community according to $\mathcal{P}$ and $\mathcal{R}$
- b : # **pairs of nodes** in a same community according to $\mathcal{P}$ and in different communities in $\mathcal{R}$
- c : # **pairs of nodes** in a different communities according to $\mathcal{P}$ and in same community in $\mathcal{R}$
- d : # **pairs of nodes** in a different communities in $\mathcal{P}$ and $\mathcal{R}$
-
- $rand(\mathcal{P}, \mathcal{R}) = \frac{a+d}{a+b+c+d}$
- $ARI = \frac{C_n^2(a+d) - [(a+b)(a+c)+(c+d)(b+d)]}{(C_n^2)^2 - [(a+b)(a+c)+(c+d)(b+d)]}$
- E(ARI)=0
- rappel : $C_n^k = \frac{n!}{k!(n-k)!}$

MUTUAL INFORMATION

$$I(X,Y) = H(X) + H(Y) - H(X,Y)$$

$$H(X) = \sum_{i=1}^{n_x} p(x_i) \times log(\frac{1}{p(x_i)})$$

Normalisation : $NMI(U,V) = \frac{I(U,V)}{\sqrt{H(U)H(V)}}$

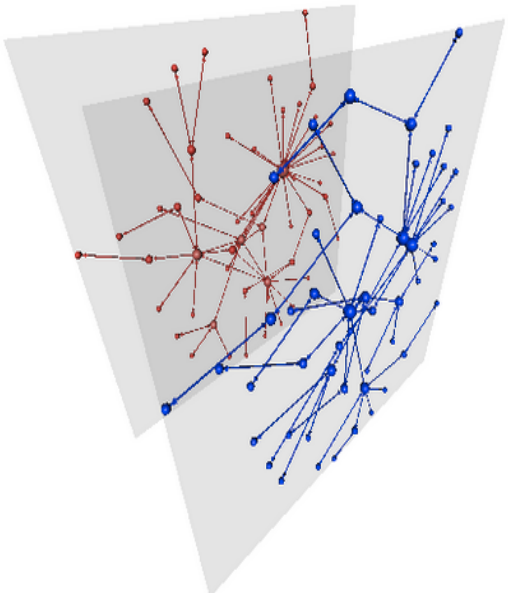
Challenges

- Real-world networks are **Dynamic**
- Real-world networks are *Heterogeneous*
- Nodes have usually semantic attributes
- Scale problem
- **Towards multiplex network mining**

MULTIPLEX NETWORK

Definition

A set of nodes related by different types of relations



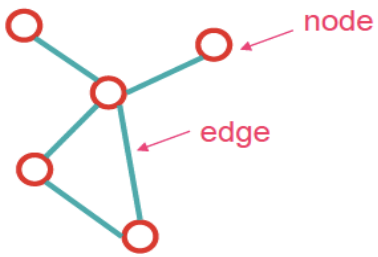
Source: muxviz

Motivation

- Real networks are **dynamic**.
- Real networks are **heterogeneous**.
- Nodes are usually **qualified** by a set of attributes.

MULTIPLEX NETWORK: NOTATIONS

$G =< V, E_1, \ldots, E_\alpha : E_k \subseteq V \times V \; \forall k \in \{1, \ldots, \alpha\} >$



- V: set of nodes (a.k.a. vertices, actors, sites)
- E_k: set of edges of type k (a.k.a. ties, links, bonds)

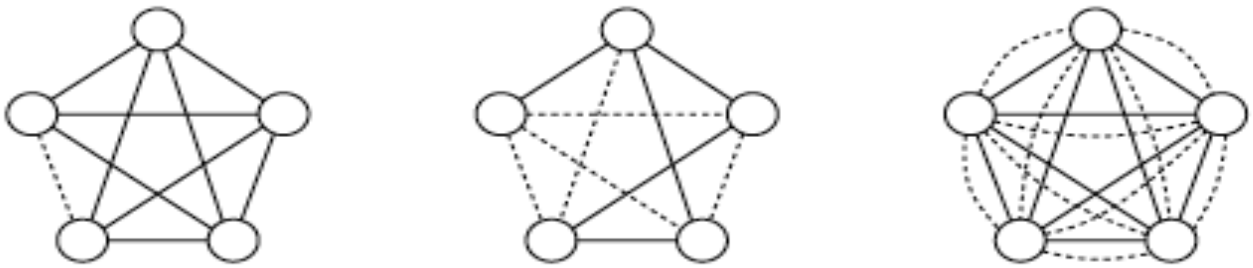
Notations

- $A^{[k]}$ Adjacency Matrix of slice k : $a_{ij}^{[k]} \neq 0$ si les nœuds $(v_i, v_j) \in E_k$, 0 otherwise.
- $m^{[k]} = |E_k|$. We have often $m \sim n$
- Neighbor's of v in slice k: $\Gamma(v)^{[k]} = \{x \in V : (x, v) \in E_k\}$.
- Node degree in slice k: $d_v^k =\| \Gamma(v)^{[k]} \|$
- Total degree of node i: $d_v^{tot} = \sum_{s=1}^{\alpha} d_v^{[s]}$
- All neighbors of v : $\Gamma(v)^{tot} = \cup_{s \in \{1,\ldots,\alpha\}} \Gamma(c)^{[s]}$

COMMUNITY ?

Some definitions

- A dense subgraph loosely coupled to other modules in the network
- A community is a set of nodes seen as one by nodes outside the community
- A subgraph where almost all nodes are linked to other nodes in the community.



What is a dense subgraph in a multiplex network ?

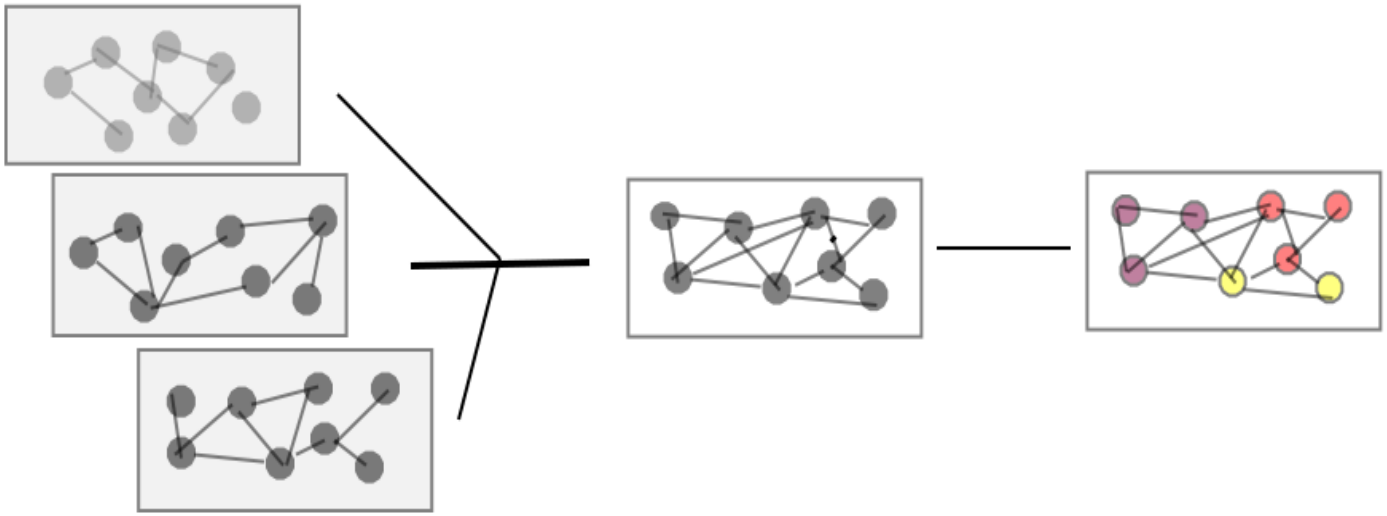
BerlingerioCG11

COMMUNITY DETECTION IN MULTIPLEX NETWORKS

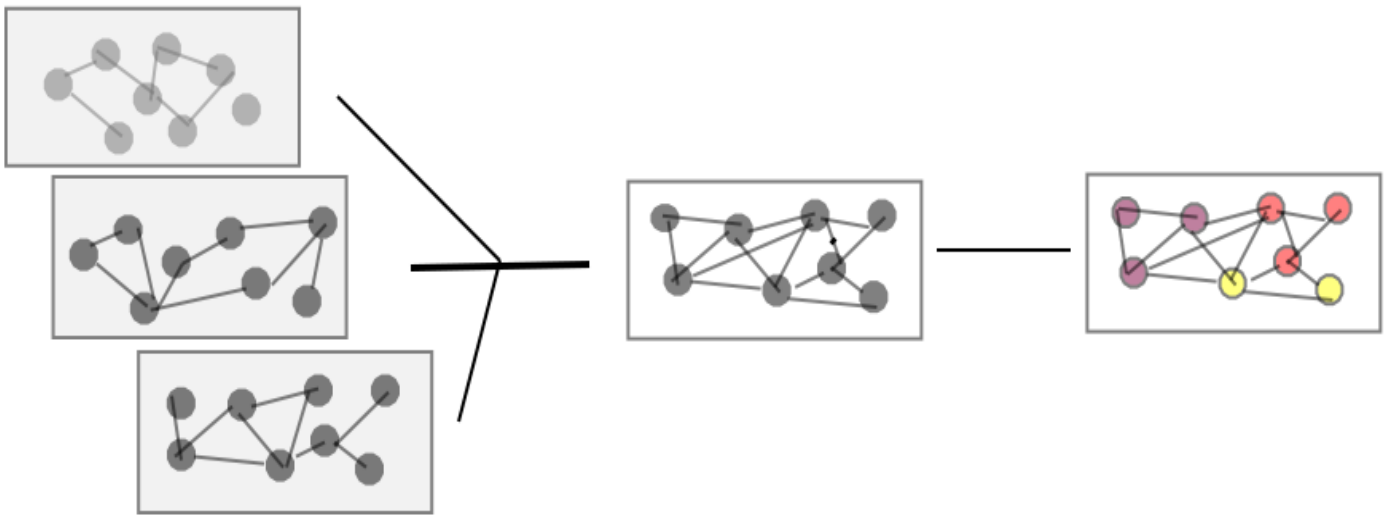
Approaches

1. **Transformation into a monoplex community detection problem**
   - Layer aggregation approaches.
   - Ensemble clustering approaches
   - Hypergraph transformation based approaches
2. **Generalization of monoplex oriented algorithms to multiplex networks.**
   - Generalized-modularity optimization
   - **Seed-centric approaches**

Layer aggregation



LAYER AGGREGATION



Aggregation functions

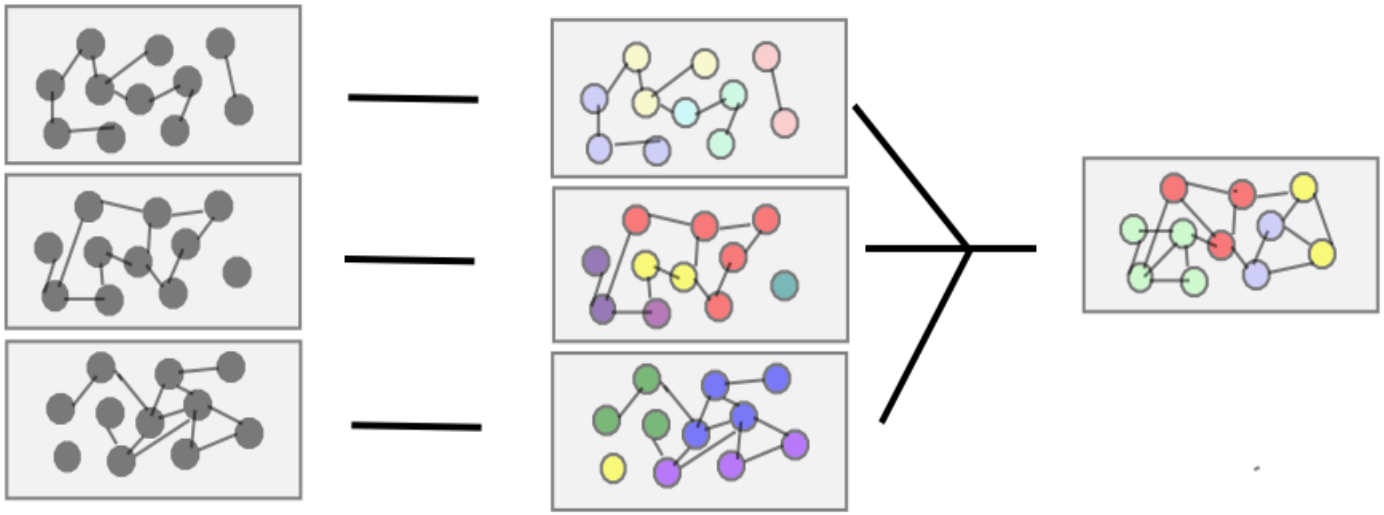
$$A_{ij} = \begin{cases} 1 & \exists 1 \leq l \leq \alpha : A_{ij}^{[l]} \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$A_{ij} = \frac{1}{\alpha} \sum_{k=1}^{\alpha} w_k A_{ij}^{[k]}$$

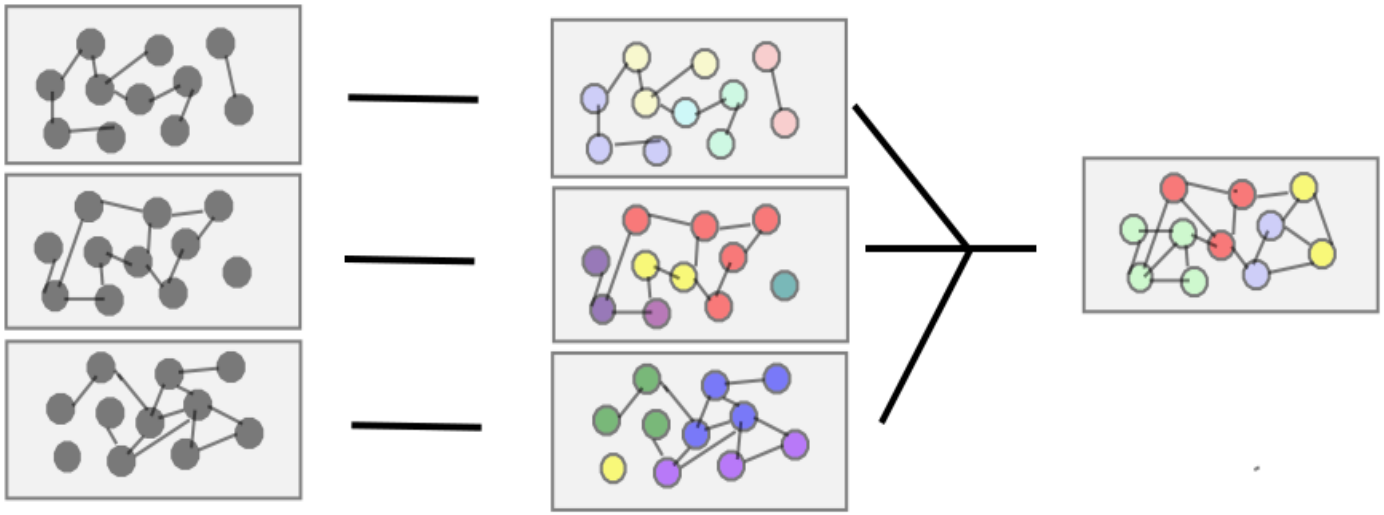
$$A_{ij} = \parallel \{d : A_{ij}^{[d]} \neq 0\} \parallel$$

$$A_{ij} = sim(v_i, v_j)$$

ENSEMBLE CLUSTERING APPROACHES



ENSEMBLE CLUSTERING APPROACHES



Ensemble Clustering — Strehl2003

- CSPA: Cluster-based Similarity Partitioning Algorithm
- HGPA: HyperGraph-Partitioning Algorithm
- MCLA: Meta-Clustering Algorithm
- ...

K-UNIFORM HYPERGRAPH TRANSFORMATION KIVELA2013MULTILAYER

Principle

- A k-uniform hypergraph is a hypergraph in which the cardinality of each hyperedge is exactly k
- Mapping a multiplex to a **3-uniform hypergraph** $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ such that :

$$\mathcal{V} = V \cup \{1, \ldots, \alpha\}$$
$$(u, v, i) \in \mathcal{E} \text{ if } \exists l : A_{uv}^{[l]} \neq 0, u, v \in V, i \in \{1, \ldots, \alpha\}$$

- Apply community detection approaches in Hypergraphs (Ex. tensor factorization approaches)

MULTIPLEX MODULARITY

Generalized modularity — mucha2010community

- $$Q_{multiplex}(P) = \frac{1}{2\mu} \sum_{c \in P} \sum_{\substack{i,j \in c \\ k,l:1 \to \alpha}} \left(\left(A_{ij}^{[s]} - \lambda_k \frac{d_i^{[k]} d_j^{[k]}}{2m^{[k]}} \right) \delta_{kl} + \delta_{ij} C_{ij}^{kl} \right)$$
- $\mu = \sum_{\substack{j \in V \\ k,l:1 \to \alpha}} m^{[k]} + C_{jk}^{l}$
- C_{ij}^{kl} Inter slice coupling $= 0 \; \forall i \neq j$

MODULARITY OPTIMIZATION LIMITATIONS

Hypothesis

- The best partition of a graph is the one that maximize the modularity.
- If a network has a community structure, then it is possible to find a precise partition with maximal modularity.
- If a network has a community structure, then partitions having high modularity values are structurally similar.

All three hypothesis do not hold Good10, LAN11a.

MUXLICOD

Multiplex degree centrality [?]

$$d_i^{multiplex} = -\sum_{k=1}^{\alpha} \frac{d_i^{[k]}}{d_i^{[tot]}} log\left(\frac{d_i^{[k]}}{d_i^{[tot]}}\right)$$

Multiplex shortest path

$$SP(u,v)^{multiplex} = \frac{\sum_{k=1}^{\alpha} SP(u,v)^{[k]}}{\alpha}$$

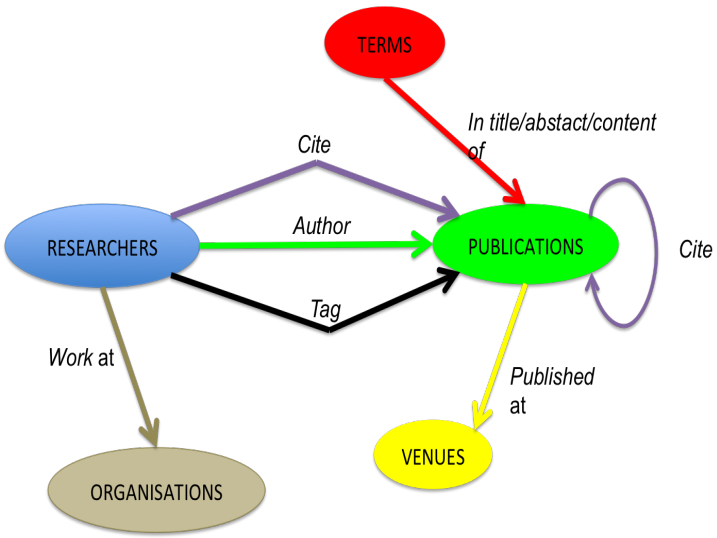
Multiplex neighborhood

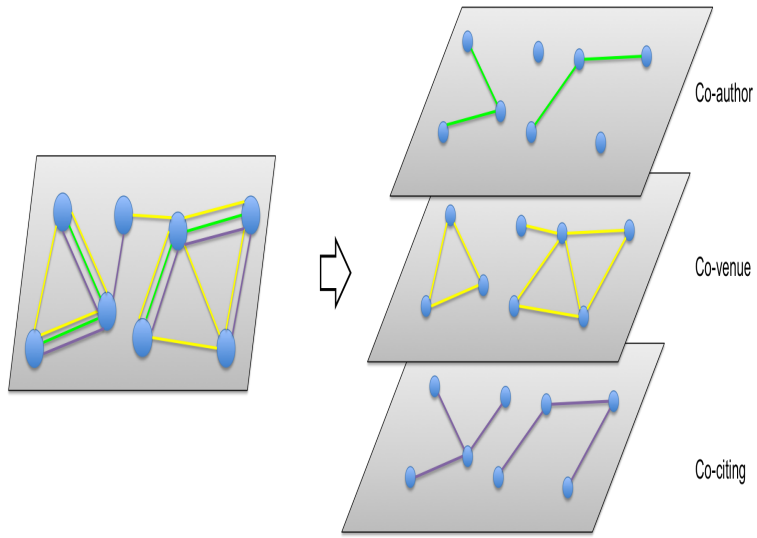
$$\Gamma^{mux}(v) = \{x \in \Gamma(v)^{tot} : \frac{\Gamma(v)^{tot} \cap \Gamma(x)^{tot}}{\Gamma(v)^{tot} \cup \Gamma(x)^{tot}} \geq \delta\}$$

EXPERIMENTS

Datasets

1. European airports network
2. Bibliographical network (DBLP) : Three layer network : co-authorship, co-venue, co-citing.





EVALUATION CRITERIA

1. Multiplex modularity
2. Redundancy [**?**]

$$\rho(c) = \sum_{(u,v)\in\bar{\bar{P}}_c} \frac{\| \{k : \exists A_{uv}^{[k]} \neq 0\} \|}{\alpha \times \| P_c \|}$$

$\bar{\bar{P}}$ the set of couple (u, v) which are directly connected in at least two layers

RESULTS: LAYER AGGREGATION APPROACHES

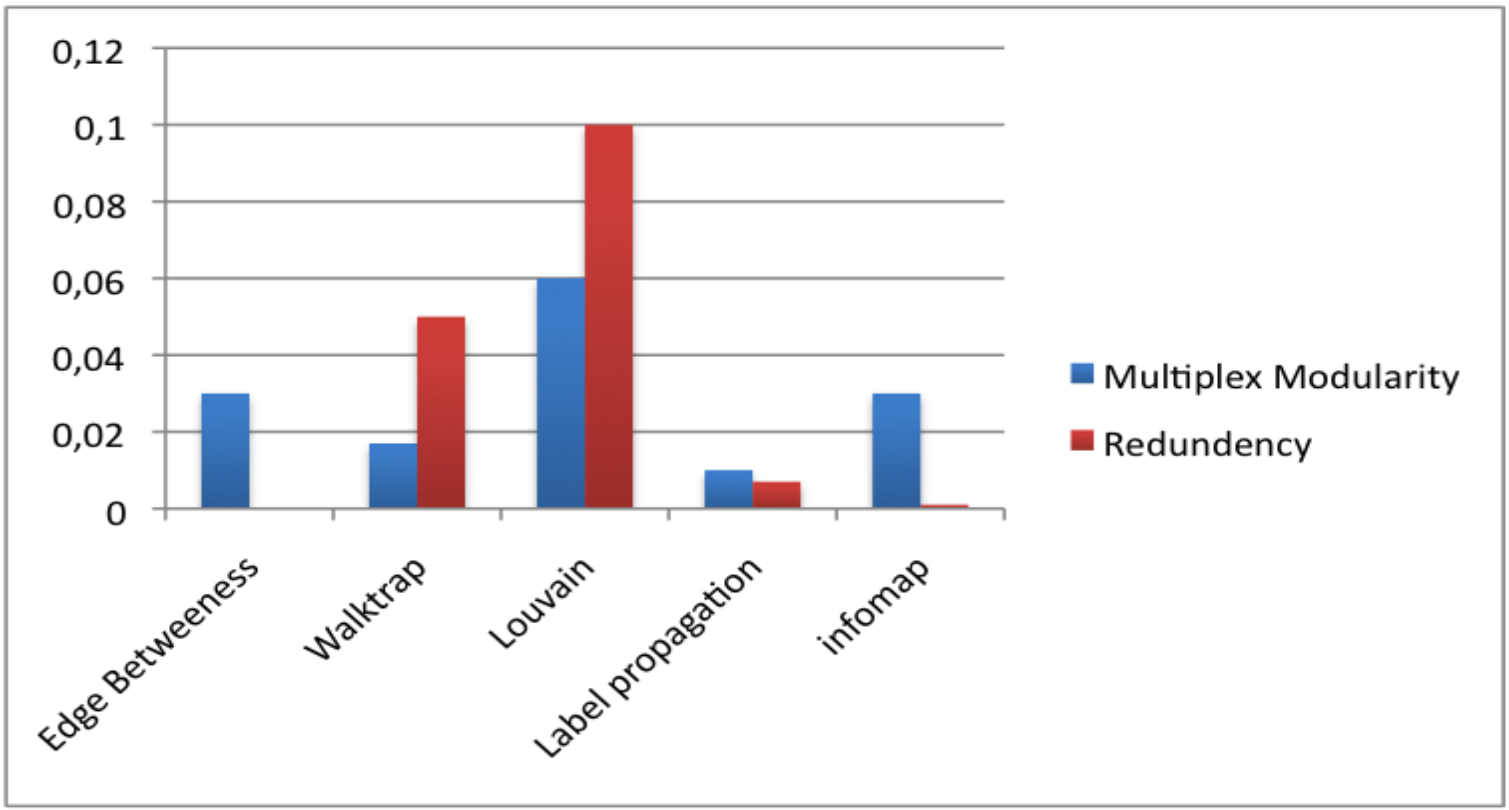


Figure: European airports multiplex network

RESULTS: PARTITION AGGREGATION APPROACHES

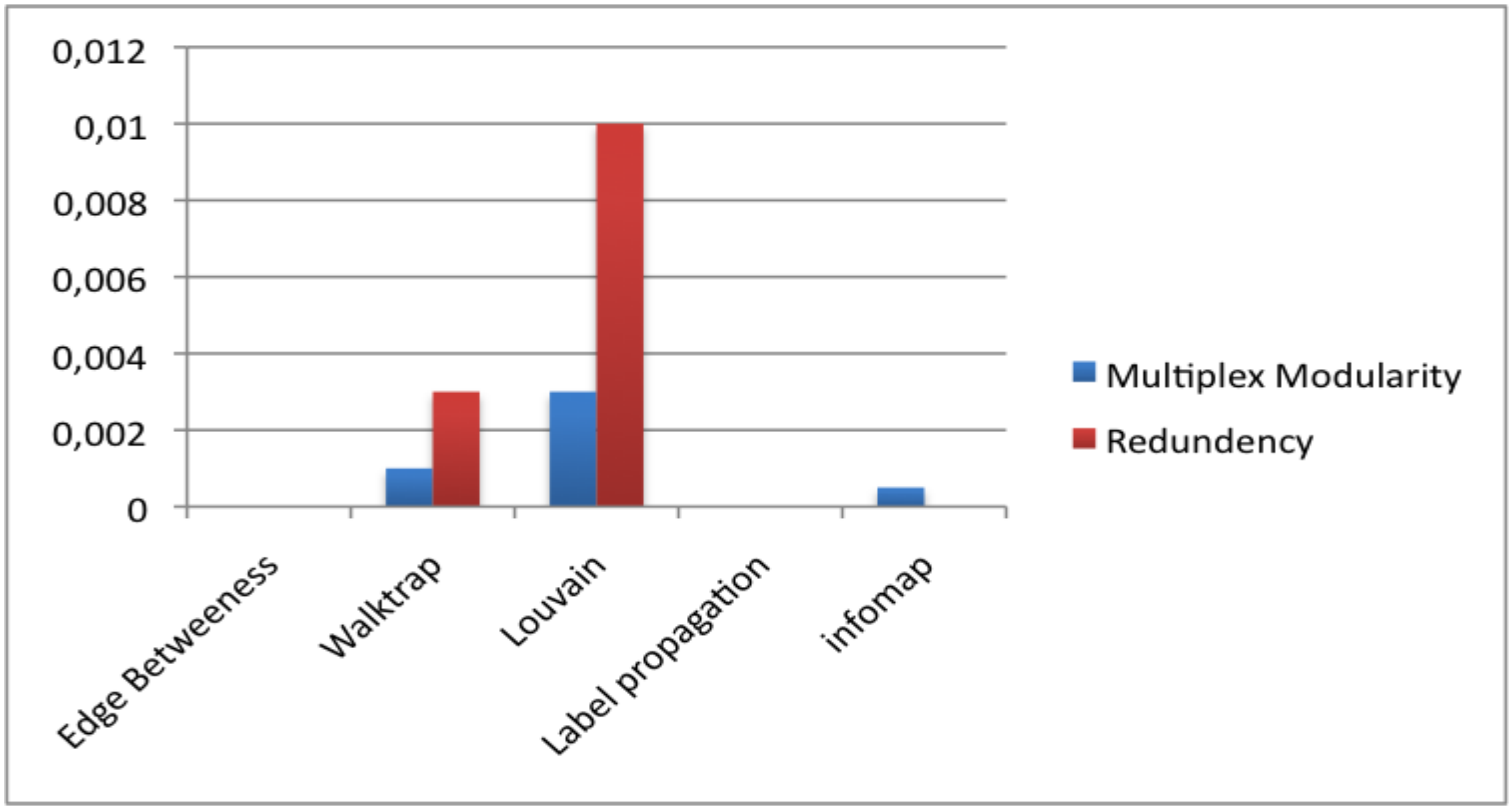


Figure: European airports multiplex network

RESULTS: MUXLICOD

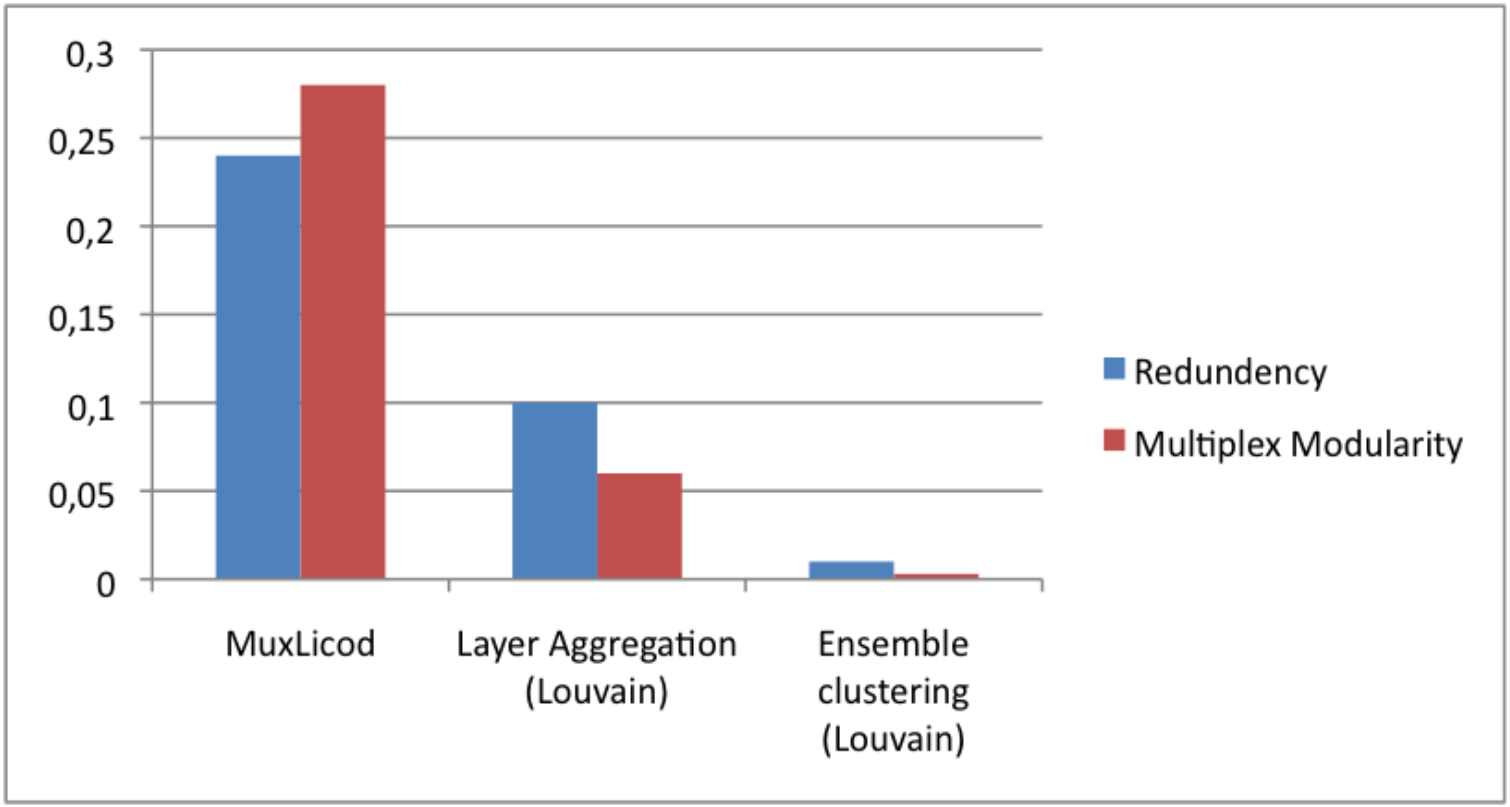


Figure: European airports multiplex network

RESULTS: MUXLICOD

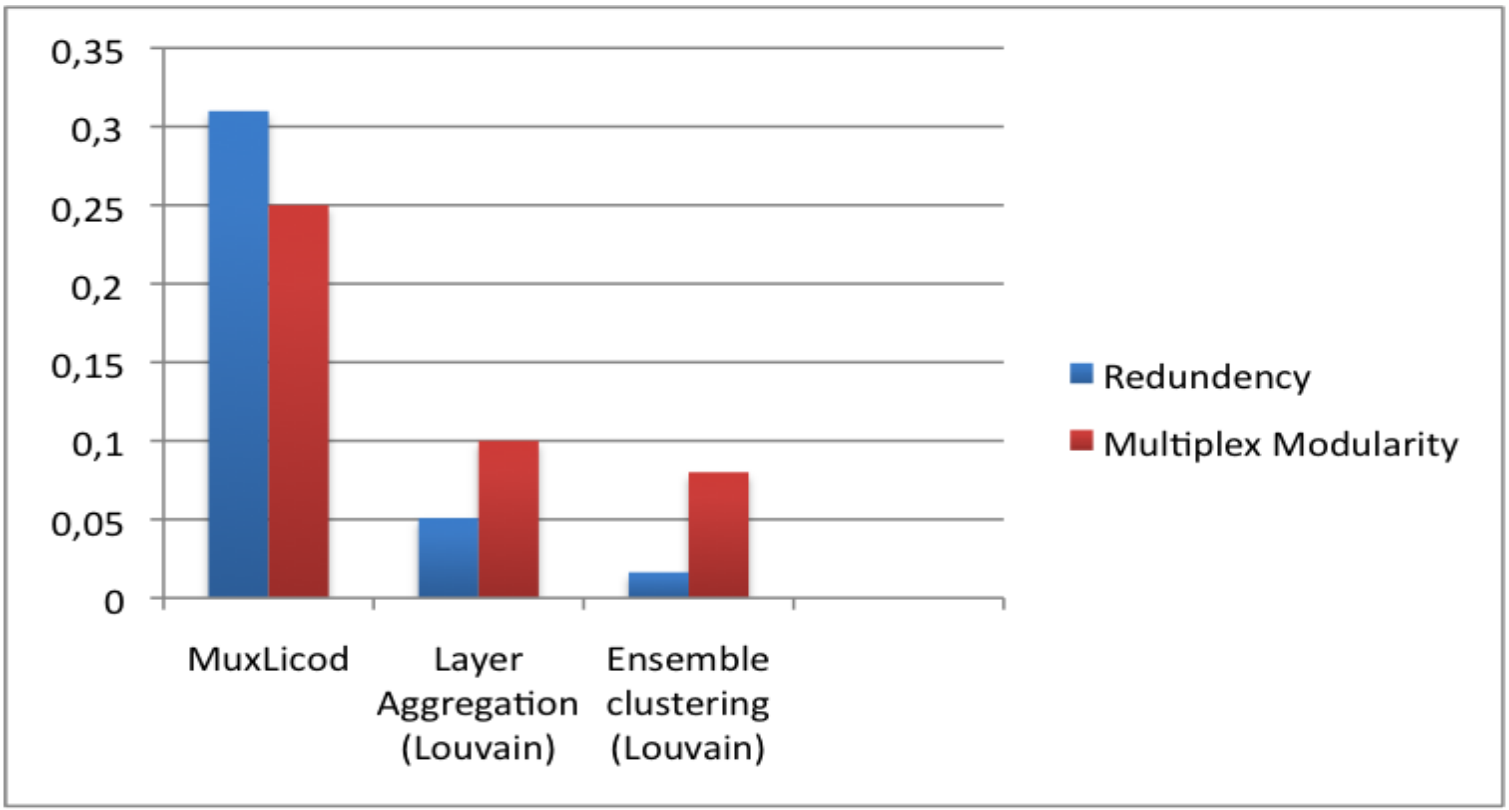


Figure: DBLP 3-layer multiplex

CONCLUSIONS I

- Networks are everywhere (ou presque)
- Network analysis can be applied to different kind of tasks
- Community detection can help in different **analysis** and **application** tasks
- **Challenges** : Scale-problems, **multiplex network mining**
  Problems : Evaluation and interoperation of computed communities.

Bibliography I


J. P. Bagrow, *Evaluating local community methods in networks*, J. Stat. Mech. **2008** (2008), no. 5, P05001.


Vincent D Blondel, Jean-loup Guillaume, and Etienne Lefebvre, *Fast unfolding of communities in large networks*, Journal of Statistical Mechanics: Theory and Experiment **2008** (2008), P10008.


Gennaro Cordasco and Luisa Gargano, *Label propagation algorithm: a semi-synchronous approach*, IJSNM **1** (2012), no. 1, 3–26.


S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, and E. Shir, *A model of internet topology using k-shell decomposition*, PNAS **104** (2007), no. 27, 11150–11154.


Aaron Clauset, *Finding local community structure in networks*, Physical Review E (2005).


Jiyang Chen, Osmar R. Zaïane, and Randy Goebel, *Local community identification in social networks*, ASONAM, 2009, pp. 237–242.

Bibliography II


B. H. Good, Y.-A. de Montjoye, and A. Clauset., *The performance of modularity maximization in practical contexts.*, Physical Review **E** (2010), no. 81, 046106.


M. Girvan and M. E. J. Newman, *Community structure in social and biological networks*, PNAS **99** (2002), no. 12, 7821–7826.


Rushed Kanawati, *Licod: Leaders identification for community detection in complex networks*, SocialCom/PASSAT, 2011, pp. 577–582.


Reihaneh Rabbany Khorasgani, Jiyang Chen, and Osmar R Zaiane, *Top leaders community detection approach in information networks*, 4th SNA-KDD Workshop on Social Network Mining and Analysis (Washington D.C.), 2010.


Andrea Lancichinetti and Santo Fortunato, *Limits of modularity maximization in community detection*, CoRR **abs/1107.1** (2011).


———, *Consensus clustering in complex networks*, Sci. Rep. **2** (2012).

Bibliography III


Ian X.Y. Leung, Pan Hui, Pietro Lio, and Jon Crowcroft, *Towards real-time community detection in large networks*, Phys. Phy. E. **79** (2009), no. 6, 066107.


Feng Luo, James Zijun Wang, and Eric Promislow, *Exploring local community structures in large networks*, Web Intelligence and Agent Systems **6** (2008), no. 4, 387–400.


Marina Meila, *Comparing clusterings by the variation of information*, COLT (Bernhard Schölkopf and Manfred K. Warmuth, eds.), Lecture Notes in Computer Science, vol. 2777, Springer, 2003, pp. 173–187.


Michael Ovelgönne and Andreas Geyer-Schulz, *Cluster cores and modularity maximization*, ICDM Workshops, 2010, pp. 1204–1213.


Clara Pizzuti, *A multiobjective genetic algorithm to find communities in complex networks*, IEEE Trans. Evolutionary Computation **16** (2012), no. 3, 418–430.

Bibliography IV


Pascal Pons and Matthieu Latapy, *Computing communities in large networks using random walks*, J. Graph Algorithms Appl. **10** (2006), no. 2, 191–218.


Usha N. Raghavan, Roka Albert, and Soundar Kumara, *Near linear time algorithm to detect community structures in large-scale networks*, Physical Review E **76** (2007), 1–12.


Lovro Subelj and Marko Bajec, *Robust network community detection using balanced propagation*, European Physics Journal B **81** (2011), no. 3, 353–362.


Massoud Seifi and Jean-Loup Guillaume, *Community cores in evolving networks*, WWW (Companion Volume) (Alain Mille, Fabien L. Gandon, Jacques Misselis, Michael Rabinovich, and Steffen Staab, eds.), ACM, 2012, pp. 1173–1180.


D Shah and T Zaman, *Community detection in networks: The leader-follower algorithm*, Workshop on Networks Across Disciplines in Theory and Applications, NIPS, 2010.

Bibliography V

Zied Yakoubi and Rushed Kanawati, *Interactions in complex systems*, ch. Community detection in complex interaction networks: Seed-based approaches, Cambridge Scholar Publishing, 2014.

Jaewon Yang and Jure Leskovec, *Defining and evaluating network communities based on ground-truth*, ICDM (Mohammed Javeed Zaki, Arno Siebes, Jeffrey Xu Yu, Bart Goethals, Geoffrey I. Webb, and Xindong Wu, eds.), IEEE Computer Society, 2012, pp. 745–754.