Entropy-based Graph Clustering:

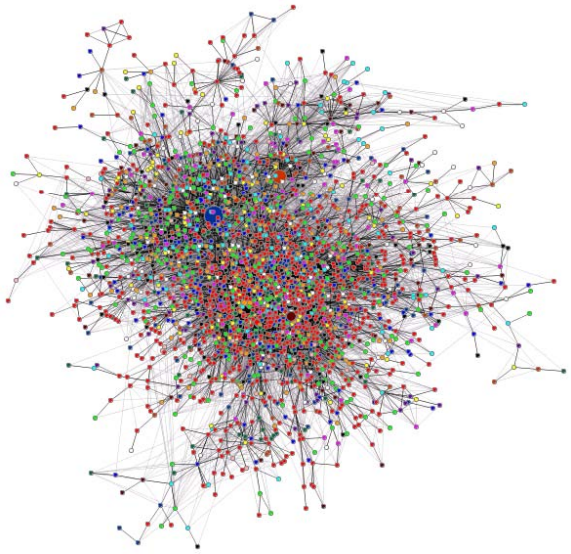
Application to Biological and Social Networks

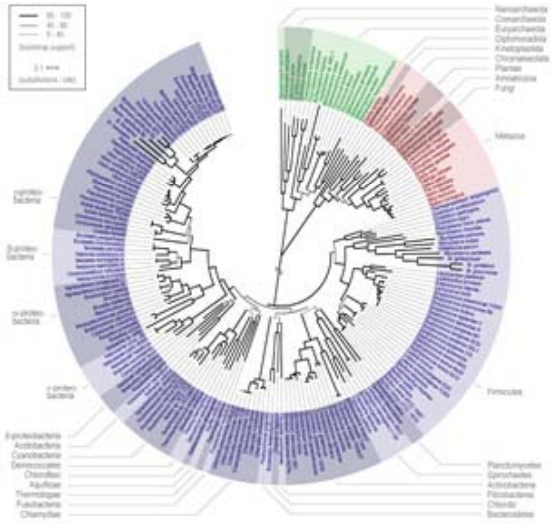
Edward C Kenley **Young-Rae Cho**

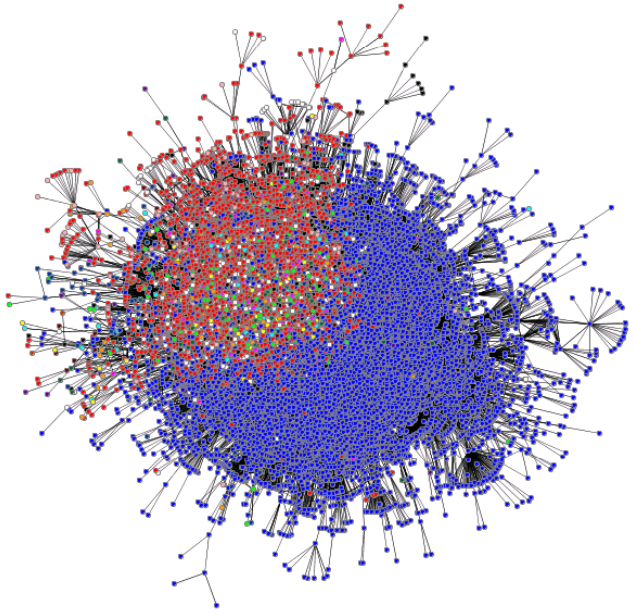
Department of Computer Science
Baylor University

Complex Systems

- Definition
  - Dynamically evolving networks that have a large number of objects with complex connectivity
- Examples
  - Social networks
  - Biological networks
  - Telecommunication networks
  - WWW







Clustering of Complex Systems

BAYLOR UNIVERSITY

- Purpose
  - Predicting groups of objects that closely communicate with each other
    - Groups having the same subjects (called communities)
    - Groups having the same functions (called modules)
- Issues
  - Large scale → Scalability / Efficiency
  - Complex connectivity → Robustness / Accuracy
- Approaches
  - Density-based methods
    - Searching densely connected sub-graphs
  - Partition-based methods
    - Detecting the best partition
  - Hierarchical methods
    - Merging sub-graphs iteratively, or dividing sub-graphs recursively

New Definition: Graph Entropy

BAYLOR UNIVERSITY

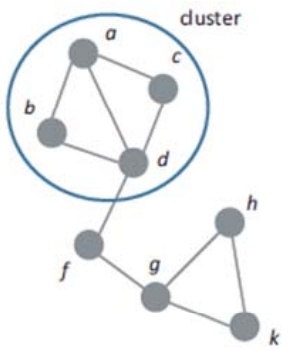
- Assumptions
  - Entropy in a graph represents structural stability
  - Isolated complete sub-graphs as clusters have the most stable status (i.e., lowest entropy)
  - Random reconnections decrease stability (i.e., increase entropy)

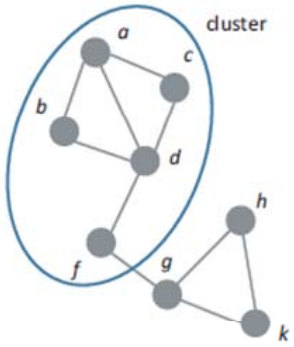
- General Notations
  - Inner links of v given G'(V',E'): edges from v to the vertices in V'
  - $p_i(v)$: probability of v having inner links
  - Outer links of v given G'(V',E'): edges from v to the vertices not in V'
  - $p_o(v)$: probability of v having outer links

- Definitions
  - Vertex Entropy: $e(v) = - p_i(v) \log_2 p_i(v) - p_o(v) \log_2 p_o(v)$
  - Graph Entropy: $e(G(V,E)) = \Sigma_{v \in V} e(v)$

Examples of Graph Entropy

- Examples





e(a) = e(b) = e(c) = 0
e(d) = 0.81
e(f) = 1.00
e(g) = e(h) = e(k) = 0

e(G) = 1.81

e(a) = e(b) = e(c) = e(d) = 0
e(f) = 1.00
e(g) = 0.92
e(h) = e(k) = 0

e(G) = 1.92

{a,b,c,d} gives **higher structural stability** to G than {a,b,c,d,f}

Entropy-Based Graph Clustering

BAYLOR UNIVERSITY

- Algorithm
  - (1) Select a random seed vertex, and form an initial cluster with the seed and its neighbors
  - (2) Remove a vertex on the inner boundary of the cluster if it decreases graph entropy
  - (3) Repeat step (2) until graph entropy is minimal
  - (4) Add a vertex on the outer boundary of the cluster if it decreases graph entropy
  - (5) Repeat step (4) until graph entropy is minimal
  - (6) Output the cluster
  - (7) Repeat steps (1) to (6) until there is no more vertex left

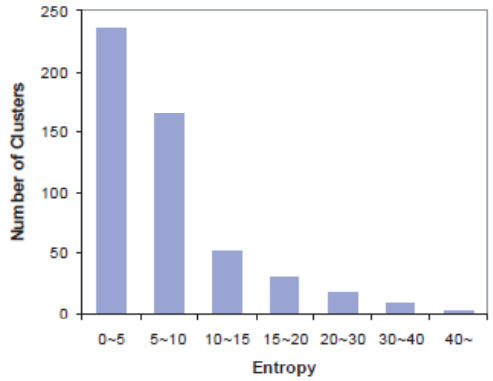
- Features
  - Local optimization → No parameters
  - Randomness
  - Low-density clusters
  - Overlapping clusters

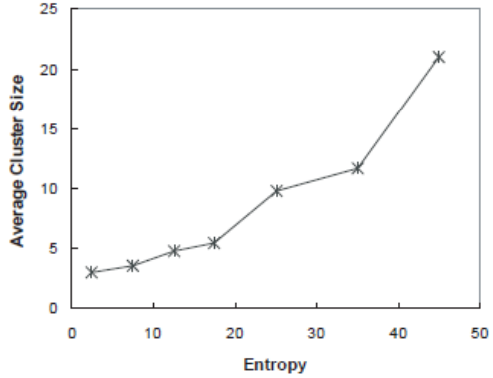
Application to Biological Networks

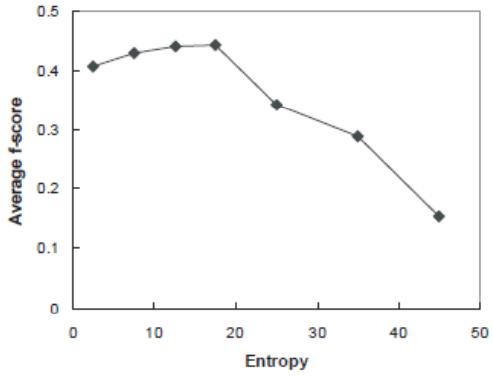
- Dataset
  - Yeast genome-wide protein-protein interaction data from DIP
  - 4,928 proteins (vertices) and 17,186 interactions (edges)

- Clustering Accuracy Evaluation Method
  - Comparison with known protein complex data
  - *f*-measure score: mean of recall and precision, $Recall = \frac{|X \cap P_i|}{|P_i|}$ $Precision = \frac{|X \cap P_i|}{|X|}$

- Results







Revisions of Entropy-Based Clustering

- Priority on Seed Selection in Step (1)
  - Selecting a seed vertex in the descending order of degree
  - Selecting a seed vertex in the descending order of clustering coefficient
- Priority on Seed Growth in Step (4)
  - Adding a vertex on the outer boundary in the ascending order of vertex entropy
- Increase of Cluster Overlapping Rates
  - Allowing random selection of a seed vertex from previous clusters
- Post-Process
  - Filtering out the clusters with graph entropy higher than a threshold

Accuracy Improvement

BAYLOR UNIVERSITY

- Results

	before post-process			after post-process		
	# clusters	avg. size	avg. *f*-score	# clusters	avg. size	avg. *f*-score
original algorithm	501	3.82	0.414	474	3.50	0.420
degree-based seed selection	442	4.02	0.416	413	3.60	**0.423**
coef.-based seed selection	551	3.73	0.415	522	3.39	0.422
degree-based seed selection AND entropy-based seed growth	442	4.03	0.416	413	3.61	**0.424**
coef.-based seed selection AND entropy-based seed growth	551	3.74	0.415	522	3.39	0.422

- Comparison with other competing methods

method	# clusters	avg. size	avg. *f*-score
Entropy-Based Clustering	413	3.61	0.424
MCL	428	5.42	0.419
CNM	40	58.55	0.391

Application to Social / Telecommunication Networks

BAYLOR UNIVERSITY

- Dataset
  - AS (Autonomous System) link network
  - YouTube video link network
  - MySpace social network

dataset	# vertices	# edges	density (%)
AS links	45,744	323,009	0.031
YouTube	321,683	505,845	0.001
MySpace	100,000	6,854,231	0.137

- Clustering Accuracy Evaluation Method
  - *p*-value of cumulative hypergeometric distribution for each vertex in a cluster
  - Averaging *p*-values of all vertices in a cluster

$$p(v) = \sum_{i=|N(v)\cap V'|}^{\min(N(v),|V'|)} \frac{\binom{|N(v)|}{i} \times \binom{|V|-|N(v)|}{|V'|-i}}{\binom{|V|}{|V'|}}$$

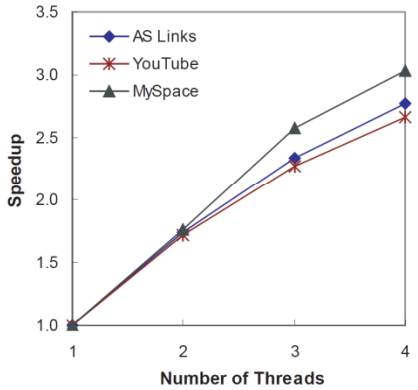
- Results

method	AS links			YouTube			MySpace		
	# clusters	avg. size	avg. –log *p*	# clusters	avg. size	avg. –log *p*	# clusters	avg. size	avg. –log *p*
Entropy-Based	824	63.1	**5.49**	25,897	8.8	-	6,505	6.2	-
MCL	979	46.7	4.31	23,001	14.0	-	117	853.8	-

Efficiency Improvement

- Parallelization
  - Multiple seed growth concurrently producing different clusters
  - Issues
    - Avoiding duplicate cluster generation
    - Sharing resource of candidate seeds

- Results
  - Close to linear increase of speed-up
    as the increment of the number of threads



- Runtime Comparison with other competing methods

method	AS links	YouTube	MySpace
Entropy-Based (1 thread)	162	75	50,028
Entropy-Based (4 threads)	58	28	16,519
MCL	6,983	317	3,764
CNM	414	118	5,334

Conclusion

- Significance
  - Systematic approach to analyze complex systems
  - Information theoretic model to formulate modularity
  - Local optimization algorithm for graph clustering
  - Applications to diverse areas:
    - Computational systems biology
    - Social network analysis
    - National security
    - Internet flow analysis

- Future Works
  - Further revisions of entropy-based graph clustering algorithm
    - Revising the seed selection step, e.g., starting from cliques as initial seed clusters
    - Revising the seed growth step, e.g., alternating vertex addition and removal