Approximation Algorithms as “Experimental Probes” of Informatics Graphs

Michael W. Mahoney
Stanford University

(For more info, see:
http://cs.stanford.edu/people/mmahoney/
or Google on “Michael Mahoney”)
Networks and networked data

Lots of “networked” data!!

- technological networks
  - AS, power-grid, road networks

- biological networks
  - food-web, protein networks

- social networks
  - collaboration networks, friendships

- information networks
  - co-citation, blog cross-postings, advertiser-bidded phrase graphs...

- language networks
  - semantic networks...

- ...

Interaction graph model of networks:

- **Nodes** represent “entities”
- **Edges** represent “interaction” between pairs of entities
Questions of interest ...

What are degree distributions, clustering coefficients, diameters, etc.?

Heavy-tailed, small-world, expander, geometry+rewiring, local-global decompositions, ...

Are there natural clusters, communities, partitions, etc.?

Concept-based clusters, link-based clusters, density-based clusters, ...

(e.g., isolated micro-markets with sufficient money/clicks with sufficient coherence)

How do networks grow, evolve, respond to perturbations, etc.?

Preferential attachment, copying, HOT, shrinking diameters, ...

How do dynamic processes - search, diffusion, etc. - behave on networks?

Decentralized search, undirected diffusion, cascading epidemics, ...

How best to do learning, e.g., classification, regression, ranking, etc.?

Information retrieval, machine learning, ...
Micro-markets in sponsored search

Goal: Find isolated markets/clusters (in an advertiser-bidded phrase bipartite graph) with sufficient money/clicks with sufficient coherence.

Ques: Is this even possible?

What is the CTR and advertiser ROI of sports gambling keywords?
What do these networks “look” like?
Clustering and Community Finding

• Linear (Low-rank) methods
  If Gaussian, then low-rank space is good.

• Kernel (non-linear) methods
  If low-dimensional manifold, then kernels are good

• Hierarchical methods
  Top-down and bottom-up -- common in the social sciences

• Graph partitioning methods
  Define “edge counting” metric -- conductance, expansion, modularity, etc. -- in interaction graph, then optimize!

“It is a matter of common experience that communities exist in networks ... Although not precisely defined, communities are usually thought of as sets of nodes with better connections amongst its members than with the rest of the world.”
Approximation algorithms as experimental probes?

The usual *modus operandi* for approximation algorithms for general problems:

• define an objective, the numerical value of which is intractable to compute
• develop approximation algorithm that returns approximation to that number
• graph achieving the approximation may be unrelated to the graph achieving the exact optimum.

But, for **randomized approximation algorithms with a geometric flavor** (e.g. matrix algorithms, regression algorithms, eigenvector algorithms; duality algorithms, etc):

• often can approximate the vector achieving the exact solution
• randomized algorithms compute an ensemble of answers -- the details of which depend on choices made by the algorithm
• maybe compare different approximation algorithms for the same problem.
Probing Large Networks with Approximation Algorithms

**Idea:** Use approximation algorithms for NP-hard graph partitioning problems as experimental probes of network structure.

- Spectral - (quadratic approx) - confuses “long paths” with “deep cuts”
- Multi-commodity flow - (log(n) approx) - difficulty with expanders
- SDP - (sqrt(log(n)) approx) - best in theory
- Metis - (multi-resolution for mesh-like graphs) - common in practice
- X+MQI - post-processing step on, e.g., Spectral of Metis

**Metis+MQI** - best conductance (empirically)

**Local Spectral** - connected and tighter sets (empirically, regularized communities!)

*We are not interested in partitions per se, but in probing network structure.*
Analogy: What does a protein look like?

Three possible representations (all-atom; backbone; and solvent-accessible surface) of the three-dimensional structure of the protein triose phosphate isomerase.

Experimental Procedure:

- Generate a bunch of output data by using the unseen object to filter a known input signal.
- Reconstruct the unseen object given the output signal and what we know about the artifactual properties of the input signal.
Communities, Conductance, and NCPPs

Let $A$ be the adjacency matrix of $G=(V,E)$.

The conductance $\phi$ of a set $S$ of nodes is:

$$\phi(S) = \frac{\sum_{i \in S, j \notin S} A_{ij}}{\min\{A(S), A(\overline{S})\}}$$

The **Network Community Profile (NCP) Plot** of the graph is:

$$\Phi(k) = \min_{S \subset V, |S|=k} \phi(S)$$

*Just as conductance captures the "gestalt" notion of cluster/community quality, the NCP plot measures cluster/community quality as a function of size.*

*NCP is intractable to compute --> use approximation algorithms!*
Widely-studied small social networks

Zachary’s karate club

Newman’s Network Science
“Low-dimensional” graphs (and expanders)

d-dimensional meshes

RoadNet-CA
NCPP for common generative models

Preferential Attachment

Copying Model

RB Hierarchical

Geometric PA
What do large networks look like?

Downward sloping NCPP

- small social networks (validation)
- "low-dimensional" networks (intuition)
- hierarchical networks (model building)
- existing generative models (incl. community models)

Natural interpretation in terms of isoperimetry

- implicit in modeling with low-dimensional spaces, manifolds, k-means, etc.

Large social/information networks are very very different

- We examined more than 70 large social and information networks
- We developed principled methods to interrogate large networks
- Previous community work: on small social networks (hundreds, thousands)
Typical example of our findings

Leskovec, Lang, Dasgupta, and Mahoney (WWW 2008 & arXiv 2008)

General relativity collaboration network
(4,158 nodes, 13,422 edges)
Focus on the red curves (local spectral algorithm) - blue (Metis+Flow), green (Bag of whiskers), and black (randomly rewired network) for consistency and cross-validation.
How do we know this plot it “correct”?

• **Lower Bound Result**
  Spectral and SDP lower bounds for large partitions

• **Modeling Result**
  Very sparse Erdos-Renyi (or PLRG wth \( \beta \in (2,3) \)) gets imbalanced deep cuts

• **Structural Result**
  Small barely-connected “whiskers” responsible for minimum

• **Algorithmic Result**
  Ensemble of sets returned by different algorithms are very different
  Spectral vs. flow vs. bag-of-whiskers heuristic
  Spectral method implicitly regularizes, gets more meaningful communities
Regularized and non-regularized communities (1 of 2)

- Metis+MQI (red) gives sets with better conductance.
- Local Spectral (blue) gives tighter and more well-rounded sets.
Regularized and non-regularized communities (2 of 2)

Two ca. 500 node communities from Local Spectral Algorithm:

Two ca. 500 node communities from Metis+MQI:
OSVV “spectral-flow” partitioning


**SPECTRAL**
- 2nd eigenvector
- Spectral cut
- Optimal cut

**GOOD CASE**

**BAD CASE: LONG PATHS**

**OSVV**

SPECTRAL STEP

FLOW IMPROVEMENT STEP

\[ G_1 = G + M_1 \]
Initial evaluation of OSVV

Classes of Graphs:

- **GM (Guattery-Miller)** graph where eigenvector methods fail.
- **PLAN** - Expander with planted bisections where LR is known to fail.
- **WING** - finite element mesh
- **RND** - Random Geometric Graph
- Random geometric graph with random edges added

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>GM100.10</td>
<td>0.016</td>
<td>0.500</td>
<td>0.778</td>
<td>0.026</td>
<td>0.037</td>
<td>0.131</td>
<td>0.358</td>
<td>0.711</td>
<td>1.102</td>
</tr>
<tr>
<td>OSVV-10.10.10</td>
<td>0.016</td>
<td>0.500</td>
<td>0.781</td>
<td>0.027</td>
<td>0.037</td>
<td>0.131</td>
<td>0.362</td>
<td>0.707</td>
<td>1.095</td>
</tr>
<tr>
<td>OSVV-1.10.10</td>
<td>0.016</td>
<td>0.716</td>
<td>0.793</td>
<td>0.027</td>
<td>0.037</td>
<td>0.131</td>
<td>0.379</td>
<td>1.000</td>
<td>1.141</td>
</tr>
<tr>
<td>METIS</td>
<td>0.016</td>
<td>0.500</td>
<td>0.785</td>
<td>0.027</td>
<td>0.037</td>
<td>0.143</td>
<td>0.372</td>
<td>0.725</td>
<td>1.060</td>
</tr>
<tr>
<td>LR</td>
<td>0.016</td>
<td>1.120</td>
<td>1.475</td>
<td>0.027</td>
<td>0.037</td>
<td>0.125</td>
<td>0.405</td>
<td>0.758</td>
<td>1.140</td>
</tr>
<tr>
<td>SPECFLOW</td>
<td>0.020</td>
<td>0.500</td>
<td>0.709</td>
<td>0.026</td>
<td>0.037</td>
<td>0.143</td>
<td>0.348</td>
<td>0.731</td>
<td>1.146</td>
</tr>
<tr>
<td>METIS</td>
<td>0.026</td>
<td>0.703</td>
<td>0.801</td>
<td>0.030</td>
<td>0.048</td>
<td>0.180</td>
<td>0.463</td>
<td>0.842</td>
<td>1.123</td>
</tr>
<tr>
<td>SPECTRAL</td>
<td>0.020</td>
<td>0.597</td>
<td>0.856</td>
<td>0.032</td>
<td>0.056</td>
<td>0.328</td>
<td>0.651</td>
<td>1.000</td>
<td>1.761</td>
</tr>
</tbody>
</table>

**Fig. 2.** The best score found by multiple tries (see caption of Figure 3) of each algorithm. First and 2nd-place for each graph are highlighted in red and blue respectively. Scores are given to 3 decimal digits. OSVV parameters are described as OSVV – \( \eta, init. s \)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>GM100.10</td>
<td>713.8</td>
<td>367.4</td>
<td>640.9</td>
<td>816.6</td>
<td>1953.4</td>
<td>953.8</td>
<td>1344.4</td>
<td>1315.8</td>
<td>1912.9</td>
</tr>
<tr>
<td>OSVV-10.10.10</td>
<td>563.1</td>
<td>399.5</td>
<td>437.0</td>
<td>2802.5</td>
<td>880.8</td>
<td>101.4</td>
<td>360.9</td>
<td>495.4</td>
<td>850.7</td>
</tr>
<tr>
<td>OSVV-1.10.10</td>
<td>425.6</td>
<td>2675.0</td>
<td>3030.0</td>
<td>1201.6</td>
<td>691.5</td>
<td>136.6</td>
<td>411.0</td>
<td>87.3</td>
<td>422.8</td>
</tr>
<tr>
<td>METIS</td>
<td>104.9</td>
<td>681.5</td>
<td>639.6</td>
<td>1019.4</td>
<td>109.6</td>
<td>109.7</td>
<td>893.8</td>
<td>283.5</td>
<td>327.8</td>
</tr>
<tr>
<td>LR</td>
<td>187.2</td>
<td>659.8</td>
<td>657.5</td>
<td>8521.1</td>
<td>412.6</td>
<td>509.2</td>
<td>699.0</td>
<td>1173.2</td>
<td>1657.5</td>
</tr>
<tr>
<td>SPECFLOW</td>
<td>200.3</td>
<td>636.2</td>
<td>580.7</td>
<td>4887.3</td>
<td>688.0</td>
<td>630.2</td>
<td>641.5</td>
<td>723.6</td>
<td>798.2</td>
</tr>
<tr>
<td>METIS</td>
<td>0.01</td>
<td>0.66</td>
<td>0.67</td>
<td>0.09</td>
<td>0.01</td>
<td>0.01</td>
<td>0.02</td>
<td>0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>SPECTRAL</td>
<td>7.1</td>
<td>3.2</td>
<td>3.3</td>
<td>3.3</td>
<td>9.0</td>
<td>1.1</td>
<td>3.1</td>
<td>2.3</td>
<td>2.5</td>
</tr>
</tbody>
</table>

**Fig. 3.** Total run time in seconds for OSVV – \( \eta, init. s \) (10 tries), METISR (10000 tries), LR (10 tries), SPECFLOW (Eigensolver + 1000 flow roundings), METIS (1 try), SPECTRAL (Eigensolver + 3 sweep roundings).
Conclusions

Approximation Algorithms as Experimental Probes of Informatics Graphs

• Powerful tools to ask precise questions of large graphs
• Use statistical and regularization properties of ensembles of “approximate solution graphs” to infer properties of original network
• Community structure in real informatics graphs -- very different than small commonly-studied graphs and existing generative models