Cooperative Computing for Autonomous Data Centers Storing Social Network Data

Jon Berry (Sandia National Laboratories)
Mike Collins (Christopher Newport University)
Aaron Kearns (U. New Mexico)
Cynthia A. Phillips (Sandia National Laboratories)
Jared Saia (U. New Mexico)
Randy Smith (Sandia National Laboratories)
A New Distributed Computing Model

Alice and Bob (or more) independently create social graphs $G_A$ and $G_B$.
- Alice and Bob each know nothing of the other’s graph.
- Shared namespace. Overlap at nodes.

Goal: Cooperate to compute algorithms over $G_A$ union $G_B$ with limited sharing: $O(\log^k n)$ total communication for size n graphs, constant k
Another Limited Sharing Model

**Goal:** Cooperate to compute algorithms over $G_A \cup G_B \cup G_C \ldots$

Alice gets no information beyond answer in honest-but-curious model.

- Secure multiparty computation
  - Few players, large data (this context is new)
Motivation

• Company mergers
• National security: connect-the-dots for counterterrorism
• Nodes are people
  – Exploit structure of social networks
Topics

• s-t connectivity
• Planted clique
• Engineering better test sets
Result: Low-communication $s$-$t$ Connectivity

- $s$-$t$ connectivity for social graphs: $O(\log^2 n)$ bits for $n$-node graphs
- $\Omega(n \log n)$ lower bound for general graphs (Hajnal, Maass, Turán)
  - Edges partitioned, 2 parties

Alice

Usually total
Communication large

Bob
Social networks have a giant component: second smallest component of size $O(\log n)$
Social Network Structure

• Normal connection growth (Easley and Kleinberg)
• Observed in social networks (long distance phone call, linkedin, etc)
• Theoretically in Chung-Lu graphs with power law exponent between $1+\varepsilon$ and 3.47

Giant Connected Component
Assumptions

- Alice’s graph $G_A$ and Bob’s graph $G_B$ both have giant components
- These giant components intersect
  - Can verify with $O(\log^2 n)$ communication with high probability if intersect by a constant fraction (say 1%)
Shell expansion

- Like breadth-first-search, “layer” is connected piece in $G_A$ or $G_B$
- Key: don’t explore too much of the graph(s)

Alice

Only send new nodes at each step.

Bob
Low-Sharing s-t Connectivity Algorithm

- Alice and Bob agree on a value $\gamma$ (polylog in n)
  - Algorithm is correct iff $\gamma$ at least size of 2\textsuperscript{nd} largest component
- Do shell expansion (BFS) from both s and t
- Stopping criteria:
  1. s shell merges with t shell (yes)
  2. No new nodes added in some step (no)
  3. Shell merges with giant component of $G_A$ or $G_B$ (yes)
  4. Shell size exceeds $\gamma$. Stop before sending. (yes)

- With a good guess, $\gamma = O(\log n)$, so $O(\log^2 n)$ bits communicated

Also: Secure multi-party communication version of S-T connectivity (IEEE/IPDPS 2015)
S-T connectivity (yes/no) without revealing node names
Topics

• s-t connectivity
• Planted clique
• Engineering better test sets
The Planted Clique Problem

• Find a clique that has been artificially added to a graph
  – Given graph, choose nodes randomly and build a clique

• Can we find a clique that’s a little larger than “native” clique size?

• For Erdos-Renyi, native is log n, can find $\sqrt{n/e}$
  – (Deshpande and Montanari 2013, Alon, Krivelevich, Sudakov, 1998)

• A form of anomaly detection, with other theoretical applications
The Distributed Planted Clique Problem

• When can social network structure help in solving a problem?
• Find a clique that has been artificially added to a graph
  – $O(\log n)$ nodes chosen randomly and builds a clique
  – Adversary assigns clique edges to Alice or Bob
• Can we find a clique that’s a little larger than “native” clique size?
Exploiting Social Network Structure

- Two key assumptions (n-node graph)
  1. Maximum degree is $O(n^{1-\epsilon})$
  2. Clustering coefficient for degree-$d$ nodes is $O\left(\frac{1}{d^2}\right)$

These two assumptions lead to a polynomial-time, polylog-communication algorithm for finding an $O(\log n)$-size planted clique.

For now, please hold off on protests about what one sees in practice (we know this isn’t realistic!)
Assumption: Clustering coefficient for degree-\(d\) nodes is \(O\left(\frac{1}{d^2}\right)\)

- **Strong triadic closure (Easley, Kleinberg):** two strong edges in a wedge implies (at least weak) closure.
  - Reasons: opportunity, trust, social stress

- **Converse of strong triadic closure:** not (both edges strong) implies not (more than coincidental closures)
  - experimental evidence: Kossinets, Watts 2006
Clustering Coefficient Assumption: Social Science Justification (slide 2)

Bounded number of strong human interactions even with social media (Dunbar 2012)

– so bounded number of strong wedges.
– As degree increases, more wedges involve weak pairs
– Social reasons for triadic closure all reduced as strength decreases

– Assumption is implied on average w.h.p by Kolda et al (SISC), where
  \( \xi \) fit from global CC:  
  \[
  c_{\text{avg}}(d) = c_{\text{max}} \exp(-(d - 1) \cdot \xi)
  \]
  
  But the assumption actually isn’t justified at all!
Problems

Experimental validation on some public social networks failed!

Why? Because the clustering coefficient assumption doesn’t hold.
Topics

- s-t connectivity
- Planted clique
- Engineering better test sets
Clustering Coefficient “Rhino Horn”

Human vs Automated

- Networks like Twitter contain a vast amount of non-human behavior
  - You can buy 500 followers for $5 US
  - Economic incentives to manipulate connections
- For applications, we assume that the network owners (e.g. law-enforcement agencies) will have human-only networks
  - Their networks are not public where entities can sign up
  - No cleaning problem
  - Will our distributed algorithms work?
- Our work uses data from SNAP, LAW
  - What cleaning of these networks can we justify?
Human vs Automated

Goal: Clean (enough) non-human behavior to test our algorithms

• Limitation: we have only topology
• Dunbar: Real human relationships require attention
  – Attention can be divided
  – Total attention, time of day, etc, is limited
• Communities with too many “strong” connections may not be human.
  – E.g.: in Twitter-2010, there is a 317-clique of mutual follower relations (with no apparent common ground among nodes)
Some Test Network Desired Properties

- Automated sub-networks are not present
- Edges plausibly represent a social bond
  - Even better if the relationship requires time/effort
- Large size (millions/billions of nodes/edges)
- Approximates a full network snapshot
  - Not ego-networks

We don’t know publicly available social networks with all these
  - Closest: friendster

Given exemplars, could generate more instances with a network generator like BTER.
Varying Strength of Ties

- People “know” about 1500 others by face/name
- Hierarchy of strength

Edge strength

• A notion somewhat like Easley and Kleinberg 2010, and Berry et al., 2011

\[ s(u, v) = \frac{2 \times \# \text{ triangles on}(u, v)}{d_u + d_v - 2} \]

\[ s(u, v) = \frac{2 \times 2}{5 + 6 - 2} = \frac{4}{9} \]

• Idea: Total strength has a constant bound
  – Edge strength a continuum, not just strong/weak
“strength-index” for Nodes (like H-index)

Strength index is the maximum of
\[ \min(r_i, s_i) \]
over all \( i \)

Neighbours sorted by edge strength

\( r_i \), (i/degree) “relative rank”
Suppose strength-index = \( s \);

Dunbar-like constant = \( D \),

\( S = \) Prefix sum of strengths\( \leq s \)

Then:

\[
D \geq S \geq s^2 \times \text{degree}
\]

\[
s \leq \sqrt{\frac{D}{d}}
\]

\( s = \) s-index

\( D = \) Dunbar-like constant

\( d = \) degree

Most important edges are free from tail effects

SSC: “Symmetric Strength Component”
SSC and total strength S are empirically bounded by small constants.
Cleaning Non-Human Nodes

• We assume \( s \leq \sqrt{\frac{D}{d}} \) for entirely-human vertices

• Constant D will depend on the network
• Remove nodes with s above this curve (or edges connecting violators)
• Selecting D
  – Compute average SSC average \( \mu \) and standard deviation \( \sigma \)
  – \( D = \mu + k\sigma \) for user-defined parameter \( k \)
• Nodes above the line for a given \( k \) are \( k\sigma \) violators
YouTube Heat Map

- Before cleaning, $k=3, 6, 12$
• Before cleaning. $k=3,6,12$
Twitter Heat Map

- Before cleaning. k=3, 6, 12
• Before cleaning, $k=3, 6, 12$. Already clean!
Cleaning

- Sometimes small number of vertices have a large fraction of edges

<table>
<thead>
<tr>
<th>Network</th>
<th>percentage of vertices removed</th>
<th>percentage of edges removed</th>
</tr>
</thead>
<tbody>
<tr>
<td>com-youtube(12\sigma)</td>
<td>0.01%</td>
<td>2.5%</td>
</tr>
<tr>
<td>com-youtube(6\sigma)</td>
<td>0.11%</td>
<td>10.76%</td>
</tr>
<tr>
<td>com-youtube(3\sigma)</td>
<td>1.18%</td>
<td>32%</td>
</tr>
<tr>
<td>ljournal-2008(12\sigma)</td>
<td>0.05%</td>
<td>1.57%</td>
</tr>
<tr>
<td>ljournal-2008(6\sigma)</td>
<td>0.14%</td>
<td>3.13%</td>
</tr>
<tr>
<td>ljournal-2008(3\sigma)</td>
<td>0.36%</td>
<td>5.38%</td>
</tr>
<tr>
<td>twitter-2010(12\sigma)</td>
<td>0.02%</td>
<td>26.4%</td>
</tr>
<tr>
<td>twitter-2010(6\sigma)</td>
<td>0.046%</td>
<td>34.3%</td>
</tr>
<tr>
<td>twitter-2010(3\sigma)</td>
<td>0.048%</td>
<td>34.7%</td>
</tr>
</tbody>
</table>
Cleaned LiveJournal

- k=12
LiveJournal: Cleaned Clustering Coefficients

LiveJournal Clustering Coefficients

<table>
<thead>
<tr>
<th>Degree</th>
<th>Original</th>
<th>12-σ cleaned</th>
<th>6-σ cleaned</th>
<th>3-σ cleaned</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>10</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>100</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>1000</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>10000</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>100000</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>
Cleaned Twitter

- $k=3$
Twitter Clustering Coefficients

- Original
- 12-σ cleaned
- 6-σ cleaned
- 3-σ cleaned

Per-Degree Coefficient vs. Degree
Validation Goal

Show empirically that we are not

“throwing out the baby with the bath water”

Working on it......
Summary

• A possible tool for cleaning non-human behavior from some social networks.
• Social network structure enables more efficient algorithms in theory and practice, but requires human-only networks.
• We won’t be able to validate the other networks
• Theory implications are wide open
Write Ups
