Density based clustering in dynamic and abstract representations of large networks

Klaus Holzapfel

Lehrstuhl für Effiziente Algorithmen
Fakultät für Informatik an der Technische Universität München

Jahrestreffen 22. – 24. Juli 2002 (Konstanz)
Kleinberg [STOC 00]

Transportation Problem
(only local information)
Algorithm: $O(\log^2 n)$

Kleinberg [J. ACM 99]

Hubs and Authorities

Achlioptas, Fiat, Karlin, McSherry [FOCS 01]

Web Search via Hub Synthesis
Practical Usage of Abstracting Data

- VLSI Design
  - placement
  - routing and wiring
- Transportation Problems
  - telephone network
  - road network
- Clustering
  - 3-D data representation (simulators)
  - speech recognition
  - web communities
How to cluster data?

- many internal edges (density)
- few external edges (cut)
- different short paths (connectivity)

**Problem:** DENSE $k$-SUBGRAPH-PROBLEM

**Input:** Graph $G$, $k \in \mathbb{N}$

**Output:** Subgraph $G'$ having maximum number of edges w.r.t. all subgraphs of size $k$

- (variable) decision problem $\mathcal{NP}$-complete
- $O(n^{\frac{1}{3} - \epsilon})$-approximation [Feige, Kortsarz, Peleg, 2001]
**Problem:** \( \gamma \text{-DENSE SUBGRAPH-PROBLEM (\( \gamma \text{-DSP} \))} \\
**Input:** Graph \( G \), \( k \in \mathbb{N} \) \\
**Output:** Does there exist a subgraph \( G' \) of size \( k \) having at least \( \gamma(k) \) edges

- \( \gamma(k) = \binom{k}{2} \) \( \gamma \text{-DSP} = \text{CLIQUE} \in \mathcal{NP}\text{-c} \)
- \( \gamma(k) = 0 \) \( \gamma \text{-DSP} \in \mathcal{P} \)

Where is the threshold?
## Results – Overview

<table>
<thead>
<tr>
<th></th>
<th>$\mathcal{P}$</th>
<th>$\mathcal{NP}$-$\mathcal{C}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Asahiro et.al. 2002]</td>
<td>$\gamma(k) = k$</td>
<td>$\gamma(k) = \Theta(k^{1+\epsilon})$</td>
</tr>
<tr>
<td>[Feige, Seltser 1997]</td>
<td>$\gamma(k) = k + k^{\epsilon}$</td>
<td></td>
</tr>
<tr>
<td>[H et.al. 2002]</td>
<td>$\gamma(k) = k + O(1)$</td>
<td>$\gamma(k) = k + \Theta(k^{\epsilon})$</td>
</tr>
</tbody>
</table>
$NP$-completeness
**Theorem.** The $\gamma$-DSP is $\mathcal{NP}$-complete for $\gamma(k) = k + \Theta(k^\epsilon)$ ($\gamma$ must be be computable in polynomial time; $0 < \epsilon < 2$).

Proof sketch: \* **CLIQUE**$\frac{1}{2}$ \$ \preceq_p^m $ $\gamma$-DSP

\[
N(r) = \gamma(k + t\binom{k}{2} + r) - (t + 1)\binom{k}{2}
\]

\[
r = 30D^2k
\]

\[
t = \left\lceil (6D)^{3\epsilon^{-1}} k^2(1-\epsilon)^{-1} \epsilon^{-1} \right\rceil
\]
Polynomial Time Algorithm
**Polynomial Time Algorithm**

Consider: \( \gamma(k) = k + \mathcal{O}(1) \)

**Definition:** \( \text{excess}(G) = |E(G)| - |V(G)| \)

---

**Problem:** \texttt{EXCESS-}c-\texttt{SUBGRAPH}

**Input:** Graph \( G \), \( k \in \mathbb{N} \)

**Output:** Does \( G \) contain a subgraph \( G' \) of size \( k \) and \( \text{excess}(G') = c \)?

---

**Theorem.** Given \( G \) and \( k \in \mathbb{N} \), the problem \texttt{EXCESS-}c-\texttt{SUBGRAPH} can be solved in time \( \mathcal{O}(|V|^{2c+3}) \).
Proof — Sketch

- Sort the connected components by their excess (decr.)

\[
\begin{array}{cccccccc}
G_1 & G_2 & \cdots & G_j & \cdots & G_{j+1} & \cdots & G_r \\
\text{excess } & \geq 0 & & & & & & \text{excess } = -1
\end{array}
\]

(1) negative components are necessary

(2) positive components are sufficient

→ compute \( A_i[...] \) and use dynamic programming

| # of vertices | 1 | 2 | 3 | \ldots | 5 | \ldots | |V(G_i)| |
|---------------|---|---|---|---------|---|---------|----------|
| max. excess   | -1| -1| 0 | \ldots | 2 | \ldots | \*        |

\* = min(excess(\( G_i \)), c + 1)
Proof — How to find $A_i[x]$

Consider a vertex minimal subgraph $G_{\text{min}}$ with excess $c$

For each $v \in V(G_{\text{min}})$, the degree of $v$ in $G_{\text{min}}$ is:

$$\sum_{v \in V(G_{\text{min}})} \text{deg}_{G_{\text{min}}}(v) = 2\|E(G_{\text{min}})\| = 2(\|V(G_{\text{min}})\| + c)$$

In $G_{\text{min}}$, there is no vertex with degree less than 2, thus:

$$\sum_{v \in V(G_{\text{min}})} (\text{deg}_{G_{\text{min}}}(v) - 2) = 2(\|V(G_{\text{min}})\| + c) - 2\|V(G_{\text{min}})\| = 2c$$

Therefore, the number of vertices with degree $\geq 3$ is at most $2c$, i.e. $O(n^{2c})$ possible combinations.

$\Rightarrow$ enumeration possible in polynomial time

Each such combination can be tested using parallel BFS.

Calculation of $A_i$ can be done in time $O(n^{2c+3})$. 
**Theorem.** Let $\gamma : \mathbb{N} \rightarrow \mathbb{N}$ be a function that is computable in polynomial time:

1. If $\gamma(k) = k + \mathcal{O}(1)$ then $\gamma$-DSP is in $\mathcal{P}$.

2. If $\gamma(k) = k + \Theta(k^\epsilon)$, for some rational number $0 < \epsilon < 2$, then $\gamma$-DSP is $\mathcal{NP}$-complete.
Directed Graphs

How to measure density in directed graphs?

- directed graphs [Kannan, Vinay, 1999]:

\[ \delta(G) = \frac{2|E(G)|}{|V(G)|} \Rightarrow \delta'(G) = \max_{S,T \subseteq V(G)} \left( \frac{E(S, T)}{\sqrt{|S||T|}} \right) \]

- evaluating existence of good hubs and authorities
- \(S\) and \(T\) not disjoint
- how to proceed when searching for dense bipartite graphs?
Future Work (1)

Where to go? What to do next?

<table>
<thead>
<tr>
<th>World Wide Web</th>
<th>General Research</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detecting Web Communities</td>
<td>Density-based Clustering</td>
</tr>
<tr>
<td>Grouping of Communities</td>
<td>Generation of Hierarchies</td>
</tr>
</tbody>
</table>

- *Trawling the Web for Emerging Cyber Communities*  
  [Kumar, Raghavan, Rajagopalan, Tomkins, 1999] WWW8

- *An approach to build a cyber-community hierarchy*  
How to utilize a Hierarchy of Web Communities

Web search for: Fibonacci
Algorithmic — How good can problems be approximated within different types of hierarchies and graph classes?

- shortest paths, local vs. global
- distance and connectivity
- searching and similarity

Dynamic aspects in hierarchies — Real world systems are not static; objects and relationships vary over time.

- recognition of emerging / dissolving clusters
- re-calibration of cluster properties (weight, size, ...)
- local vs. global recalculation