Finding Connected Dense $k$-Subgraphs

Xujin Chen
Institute of Applied Mathematics, AMSS, Chinese Academy of Sciences

Joint work with
Xiaodong Hu, Changjun Wang

The 7th Meeting of Asian Association for Algorithms & Computation,
Hangzhou, May 16-17, 2014
1 Results

2 Algorithms
- $O(n^2/k^2)$-approximation for densest connected $k$-subgraphs
- $O(n^2/5)$-approximation for densest connected $k$-subgraphs
- $O(\min\{n^2/k^2, k, n^2/3\})$-apx for heaviest connected $k$-subgraphs

3 Conclusion

4 ...
Density of a graph

Let $G = (V, E)$ be a simple undirected graph with $n$ vertices, $m$ edges, and nonnegative edge weights $w \in \mathbb{Z}_+^E$.

The (weighted) density of $G$ is its average (weighted) degree.

$$\sigma(G) = \frac{\sum_{v \in V} d_G(v)}{|V|} = \frac{2|E|}{|V|}$$

$$\sigma(G, w) = \frac{\sum_{v \in V} d_G^w(v)}{|V|} = \frac{2w(E)}{|V|}$$
Densest (connected) $k$-subgraph problem

Let $G = (V, E)$ be a connected simple undirected graph with $n$ vertices, $m$ edges, and nonnegative edge weights $w \in \mathbb{Z}_E^+$. Let $k \leq n$ be a positive integer. A subgraph of $G$ is called a $k$-subgraph if it has exactly $k$ vertices.

**D$k$SP**
The densest $k$-subgraph problem (D$k$SP) is to find a $k$-subgraph of $G$ that has the maximum density.

**D$Ck$SP**
The densest connected $k$-subgraph problem (D$Ck$SP) is to find a connected $k$-subgraph of $G$ that has the maximum density.
Heaviest (connected) $k$-subgraph problem

- Let $G = (V, E)$ be a connected simple undirected graph with $n$ vertices, $m$ edges, and nonnegative edge weights $w \in \mathbb{Z}_E^+$. Let $k \leq n$ be a positive integer. A subgraph of $G$ is called a $k$-subgraph if it has exactly $k$ vertices.

**DkSP**

The densest $k$-subgraph problem (DkSP) is to find a $k$-subgraph of $G$ that has the maximum density.

The heaviest $k$-subgraph problem (HkSP)

**DCkSP**

The densest connected $k$-subgraph problem (DCkSP) is to find a connected $k$-subgraph of $G$ that has the maximum density.

The heaviest connected $k$-subgraph problem (HCkSP)
Applications

**Detect** important substructures in massive graphs: social networks, protein interaction graphs, world wide web...

In a web graph, hubs (resource lists) and authorities (authoritative pages) on a topic are characterized by large number of links between them.

**Discover** communities in web and social networks, for compressed representation of a graph and for spam detection.
Applications

**Detect** important substructures in massive graphs: social networks, protein interaction graphs, world wide web...

In a web graph, hubs (resource lists) and authorities (authoritative pages) on a topic are characterized by large number of links between them.

**Discover** communities in web and social networks, for compressed representation of a graph and for spam detection.

**Connectivity** requirements are natural in various scenarios ...

If most vertices belong to a dense connected subnetwork, only a few selected inter-hub links are needed to have a short average distance between any two arbitrary vertices in the entire network. Commercial airlines employ this hub-based routing scheme.
Applications

Detect important substructures in massive graphs: social networks, protein interaction graphs, world wide web...

In a web graph, hubs (resource lists) and authorities (authoritative pages) on a topic are characterized by large number of links between them

Discover communities in web and social networks, for compressed representation of a graph and for spam detection

Connectivity requirements are natural in various scenarios...

Fast and effective algorithms for finding dense (connected) subgraphs.

J. Kleinberg
**Hardness**

$D_k\text{SP}$, $D C_k\text{SP}$ and their weighted versions are **strongly NP-hard**

The NP-hardness remains even for

- chordal graphs, triangle-free graphs, comparability graphs \[\text{[Corneil, Perl, DAM’84]}\]
- bipartite graphs of maximum degree 3 \[\text{[Feige, Seltser, 1997]}\]

$H_k\text{SP}$ and $H C_k\text{SP}$ remain **NP-hard** in

- the metric case \[\text{[Ravi et al, OR’94]}\]
- binary weighted case for cographs and split graphs \[\text{[CP84]}\]

**Open**: the status of $D_k\text{SP}$ on planar graphs & interval graphs

**NP-hard**: $D C_k\text{SP}$ on planar graphs \[\text{[Keil, Brecht, JCMCC’91]}\].
Approximation algorithms
- polynomial time
- approximation ratio: \( \sup_{\text{all instances}} \frac{\text{optimal value}}{\text{solution value}} \)

Polynomial time approximation scheme (PTAS)
- for any \( \varepsilon > 0 \), in polynomial time, produces a \((1 + \varepsilon)\)-approximate solution
- the running time is polynomial in \( n \) for every fixed \( \varepsilon \) but can be different for different \( \varepsilon \).
**Inapproximability for D$k$SP**

- **(1 + \(\varepsilon\))-approximation** is at least as hard as refuting random 3-SAT clauses for some \(\varepsilon > 0\) [Feige, STOC’02]
- No **PTAS** assuming NP does not have randomized algorithms that run in sub-exponential time [Khot, SJOC’06]
- No constant factor approximations in polynomial time under Unique Games with Small Set Expansion conjecture [Raghavendra, Steurel, STOC’10], or under certain “average case” hardness assumptions [Alon et al. 2011]

**Approximability for H$k$SP**
Large gap for approximation

Approximability for HkSP

- $O(n^{0.3885})$-approximation [Kortsarz, Peleg, FOCS’93]
- Combinatorial algorithm with approximation ratio $O(n^\delta)$ for some $\delta < 1/3$ [Feige et al. Algorithmica’01]
- $O(n^{1/4+\varepsilon})$-approximation in $n^{O(1/\varepsilon)}$ time, $O(n^{1/4})$-approximation in $n^{O(\log n)}$ time [Bhaskara et al. STOC’10]
- $O(n/k)$-approximation greedy algorithm [Asahiro et al. JOA’00]
- Randomized rounding algorithms by linear and semidefinite programming relaxation approaches
  - an approximation ratio somewhat better than $n/k$ [Feige, Langberg, JOA’01]
  - further improvement for a range of values $k = \Theta(n)$ [Srivastav, Wolf, WAACO’98; Han et al. MP’02]
Better approximations for special cases

- PTAS for the restricted D$k$SP where $m = \Omega(n^2)$ and $k = \Omega(n)$, or each vertex of $G$ has degree $\Omega(n)$ [Arora et al. STOC’95]
- 2-approximation algorithm for D$k$SP on $H$-minor-free graphs, where $H$ is any given fixed graph [Demaine et al. FOCS’05]
- Constant factor approximation for D$k$SP on a large family of intersection graphs: chordal graphs, circular-arc graphs, claw-free graphs, ... [Chen et al. AOA’11]
- PTAS for D$k$SP on unit disk graphs [Chen et al. AOA’11], interval graphs [Nonner, ADS’11], a subclass of chordal graphs [Liazi et al. JOCO’07]
Limited work on the **connected** versions

Existing polynomial time algorithms deal only with special graphical topologies, including:

- 4- and 2-approximation algorithm for metric \( H_kSP \) & \( HC_kSP \)  
  [Ravi et al. OR’94; Hassin et al. ORL’97]

- Exact algorithms for  
  \( H_kSP \) and \( HC_kSP \) on trees  
  [Corneil, Perl, DAM’84]  
  \( D_kSP \) and \( DC_kSP \) on \( h \)-trees, cographs, and split graphs  
  [Corneil, Perl, DAM’84]  
  \( DC_kSP \) on interval graphs whose clique graphs are simple paths  
  [Liazi et al. BCC’05]
The problem of finding a (connected) subgraph (without any cardinality constraint) of maximum weighted density is strongly polynomial time solvable [Goldberg’84]

Finding a weighted densest subgraph with at least $k$ vertices is NP-hard, and admit 2-approximation [Andersen, Chellapilla’09; Khuller, Saha’09]

The approximation for finding a weighted densest subgraph with at most $k$ vertices is as hard as that of $DkSP/HkSP$ up to a constant factor
Our results

Given connected graph $G = (V, E)$ with $|V| = n$, $|E| = m$ and $w \in \mathbb{Z}_+^E$, 
$\sigma_k^*(G) = \text{the maximum density of all } k\text{-subgraphs}$, 
$\sigma_k^*(G, w) = \text{the maximum weighted density of all } k\text{-subgraphs}$; 
let $\text{opt}(G)$ denote the optimal value of the DC$k$SP on $G$ 
let $\text{opt}(G, w)$ denote the optimal value of the HC$k$SP on $G$ with $w$

We design $O(mn \log n)$ time combinatorial approximation algorithms for finding a connected $k$-subgraph $C$ (resp. $C'$) of $G$ such that

$$\frac{\text{opt}(G)}{\sigma(C)} \leq \frac{\sigma_k^*(G)}{\sigma(C)} \leq O\left(\min\{n^{2/5}, n^2 / k^2\}\right)$$

$$\frac{\text{opt}(G, w)}{\sigma(C', w)} \leq \frac{\sigma_k^*(G, w)}{\sigma(C', w)} \leq O\left(\min\{n^{2/3}, n^2 / k^2\}\right)$$
Example for DC$k$SP

\[
\sup_G \frac{\sigma_k^*(G)}{\sigma(G)} \geq \frac{n^{1/3}}{3}
\]

\(n = \ell^3\)
\(k = \ell^2\)

\(\ell\)-clique \(\ell\)-clique \(\ell\)-clique

\(\sigma_k^*(G) = \ell - 1\)

The densest \(k\)-subgraph

\(\text{opt}(G) = \frac{\ell(\ell - 1) + 2(\ell^2 - \ell)}{\ell^2}\)

A densest connected \(k\)-subgraph
Example for DC$k$SP

\[ \sup_G \frac{\sigma_k^*(G)}{\sigma^*(C)} \geq \frac{\ell^2}{3\ell} \geq \frac{n^{1/3}}{3} \]

\[ n = \ell^3 \]

\[ k = \ell^2 \]

\[ \ell\text{-clique} \quad \ell\text{-clique} \quad \ell\text{-clique} \]

The densest $k$-subgraph

\[ \sigma_k^*(G) = \ell - 1 \]

A densest connected $k$-subgraph

\[ \text{opt}(G) = \frac{\ell(\ell-1) + 2(\ell^2 - \ell)}{\ell^2} \]

In contrast to \( \frac{\sigma_k^*(G)}{\sigma^*(C)} \leq O(n^{0.4}) \)
Example for HCKSP

\[
\sup_G \frac{\sigma_k^*(G,w)}{\text{opt}(G,w)} \geq \ell \geq \frac{n^{1/2}}{2}
\]

\[\sigma_k^*(G,w) = 1\]

\[\text{opt}(G,w) = 1/\ell\]
When solution value is compared with the optimum of $D^k\text{SP}$ ($H^k\text{SP}$)...

$$\Omega(n^{1/3}) \leq \max \frac{\sigma^*_k(G)}{\sigma(C)} \leq O(\min\{n^{2/5}, n^2/k^2\})$$

$$\Omega(n^{1/2}) \leq \max \frac{\sigma^*_k(G, w)}{\sigma(C', w)} \leq O(\min\{n^{2/3}, n^2/k^2\})$$
Lower and upper bounds

When solution value is compared with the optimum of $D_kSP$ ($H_kSP$)...

\[ \Omega(n^{1/3}) \leq \max \frac{\sigma_k^*(G)}{\sigma(C)} \leq O(\min\{n^{2/5}, n^2/k^2\}) \]

\[ \Omega(n^{1/2}) \leq \max \frac{\sigma_k^*(G, w)}{\sigma(C', w)} \leq O(\min\{n^{2/3}, n^2/k^2\}) \]

In the following, we focus on the unweighted problem: $DC_kSP$ on connected graph $G = (V, E)$ with $n$ vertices and $m$ edges
Algorithm 1

An $O(n^2/k^2)$-approximation algorithm for DC$k$SP in $O(mn)$ time
Algorithm 1

An $O(n^2/k^2)$-approximation algorithm for DC$k$SP in $O(mn)$ time

For simplicity, we assume $k$ is even.
Removable vertices

The vertices whose removals increase the graph’s density play an important role in our algorithm design.

Definition

A vertex \( v \in V \) is called removable in \( G \) if \( \sigma(G \setminus v) > \sigma(G) \).

Since \( \sigma(G \setminus v) = 2(|E| - d_G(v))/(|V| - 1) \), we have

Lemma

A vertex \( v \in V \) is removable in \( G \) if and only if \( d_G(v) < \sigma(G)/2 \).

It also provides an efficient way to identify removable vertices.
Greedy attachment

Let $S$ and $T$ be disjoint nonempty vertex subsets (or subgraphs) of $G$. $[S, T] = \{uv \in E : u \in S, v \in T\}$.

Greedy attachment

For any positive integer $j \leq |V| - |S|$, a set $S^*$ of $j$ vertices in $G \setminus S$ with maximum $|[S, S^*]|$ can be found in $O(m + n \log n)$ time, for which we have

$$|[S, S^*]| \geq \frac{j}{n} \cdot |[S, V \setminus S]|. \quad (1)$$

If $G[S]$ is connected, then such an $S^*$ can be chosen such that $G[S \cup S^*]$ is connected. We refer to this $S^*$ as a $j$-attachment of $S$ in $G$. 
Algorithm 1

To find a connected $k$-subgraph $C$ with $\sigma(C) \geq \Omega\left(\frac{k^2}{n^2}\right) \cdot \sigma^*_k(G)$, we start with connected $G' \leftarrow G$, and repeatedly delete removable vertices from $G'$ to increase its density without destroying its connectivity.

- If we can reach $G'$ with $|G'| = k$ in this way, we output $C \leftarrow G'$.
- If $\exists$ a removable cut-vertex $r$ in $G'$ such that the densest component $G'_r$ of $G \setminus r$ has $|G'_r| \geq k$ vertices, then we recurse with $G' \leftarrow G'_r$.
- If we stop at a $G'$ without any removable vertices, then

Procedure 1

Construct $C$ from an arbitrary connected $(k/2)$-subgraph by greedily attaching $k/2$ more vertices.

- Otherwise, we find a connected subgraph of $G'$ induced by a set $S$ of at most $k/2$ vertices, and expand the subgraph in two ways ...
Algorithm 1

To find a connected $k$-subgraph $C$ with $\sigma(C) \geq \Omega\left(\frac{k^2}{n^2}\right) \cdot \sigma_k^*(G)$, we start with connected $G' \leftarrow G$, and repeatedly delete removable vertices from $G'$ to increase its density without destroying its connectivity.

- If we can reach $G'$ with $|G'| = k$ in this way, then output $C \leftarrow G'$
- If $\exists$ a removable cut-vertex $r$ in $G'$ with $|G'_r| \geq k$, then recurse
- If $G'$ has no removable vertices, then $C \leftarrow \text{PRC1}(G')$
- Otherwise, $C \leftarrow \text{PRC2}(G')$

Procedure 2

Find a connected subgraph of $G'$ induced by a set $S$ of at most $k/2$ vertices, and expand the subgraph in two ways:

1. attaching $G'_r$ for all removable vertices $r$ of $G'$ contained in $S$, and
2. greedily attaching no more than $k/2$ vertices.

From the resulting connected subgraphs, we choose the one that has more edges, and further expand it to be a connected $k$-subgraph.
Algorithm 1

Input: $G = (V, E)$ with $|V| \geq k$.
Output: a connected $k$-subgraph of $G$, written as $\text{ALG1}(G)$.

1. $G' \leftarrow G$
2. While $|G'| > k$ and $G'$ has a removable vertex $r$ that is not a cut-vertex do
3. $G' \leftarrow G' \setminus r$
4. End-While // any removable vertex of $G'$ is a cut-vertex
5. If $|G'| = k$ then output $\text{ALG1}(G) \leftarrow G'$
6. If $|G'| > k$ and $G'$ has no removable vertices then output $\text{ALG1}(G) \leftarrow \text{PRC1}(G')$
7. If $|G'| > k$ and $|G'_r| < k$ for each removable vertex $r$ of $G'$ then output $\text{ALG1}(G) \leftarrow \text{PRC2}(G')$
8. If $|G'| > k$ and $|G'_r| \geq k$ for some removable vertex $r$ of $G'$ then output $\text{ALG1}(G) \leftarrow \text{ALG1}(G'_r)$
Algorithm 1

**Theorem** $(O(n^2/k^2)$-approximation)

*Algorithm 1 finds in $O(mn)$ time a connected $k$-subgraph $C$ of $G$ such that $\sigma_k^*(G)/\sigma(C) \leq 12n^2/k^2$.***
Algorithms 1, 2, 3, 4

An $O(n^{2/5})$-approximation algorithm for $DCkSP$

in $O(mn \log n)$ time

For simplicity, we assume $k$ is even.
Algorithms 1, 2, 3, 4

An $O(n^{2/5})$-approximation algorithm for DC$k$SP in $O(mn \log n)$ time

For simplicity, we assume $k$ is even.

In view of the $O(n^2/k^2)$-approximation of Algorithm 1, we may focus on the case of $k < n^{4/5}$. (Note that $n^2/k^2 \leq n^{2/5}$ if $k \geq n^{4/5}$.)
Algorithm 2

**Input:** $G = (V, E)$ with $|V| \geq k$.

**Output:** a connected $k$-subgraph of $G$, denoted as $\text{ALG2}(G)$.

1. Find a densest connected subgraph $D$ of $G$
2. If $|D| \leq k$ then Expand $D$ to be a connected $k$-subgraph $H$ of $G$
   Output $\text{ALG2}(G) \leftarrow H$
3. Else Output $\text{ALG2}(G) \leftarrow \text{PRC1}(D)$

**Lemma**

If $k < n^{4/5}$, then $\sigma(\text{ALG2}(G)) \geq \min\{k/(4n), n^{-2/5}\} \cdot \sigma^*(G)$. 

Xujin Chen, Xiaodong Hu, Changjun Wang

Finding Connected Dense $k$-Subgraphs
Algorithm 3

Let $V_h$ be a set of $k/2$ vertices of highest degrees in $G$, and let $d_h = \frac{2}{k} \sum_{v \in V_h} d_G(v)$ denote the average degree of the vertices in $V_h$.

Algorithm 3

**Input:** $G = (V, E)$ with $|V| \geq k$.

**Output:** a connected $k$-subgraph of $G$, denoted as $\text{ALG3}(G)$.

1. $V_h^* \leftarrow$ a $(k/2)$-attachment of $V_h$ in $G$
2. $H \leftarrow$ a densest component of $G[V_h \cup V_h^*]$
3. Output $\text{ALG3}(G) \leftarrow$ a $k$-connected subgraph of $G$ that is expanded from $H$

Lemma

$$\sigma(\text{ALG3}(G)) \geq \frac{\bar{\sigma}}{\sqrt{k}} \geq \frac{\sqrt{k}}{2n} \cdot d_h, \text{ where } \bar{\sigma} = \sigma(G[V_h \cup V_h^*]) \geq \frac{kd_h}{2n}.$$
Algorithm 4

For \( u, v \in V \), let \( W(u, v) = \# \) walks of length 2 from \( u \) to \( v \) in \( G \).

**Algorithm 4**

**Input:** \( G = (V, E) \) with \( |V| \geq k \).

**Output:** a connected \( k \)-subgraph of \( G \), denoted as \( \text{ALG4}(G) \).

1. \( G_\ell \leftarrow G[V \setminus V_h] \).
2. Compute \( W(u, v) \) for all pairs of vertices \( u, v \) in \( G_\ell \).
3. For every \( v \in V \setminus V_h \), construct a connected \( k \)-subgraph \( C^v \) as follows:
   - Sort the vertices \( u \in V \setminus V_h \setminus \{v\} \) with positive \( W(v, u) \) as \( v_1, v_2, \ldots, v_t \) such that \( W(v, v_1) \geq W(v, v_2) \geq \cdots \geq W(v, v_t) > 0 \).
   - \( P^v \leftarrow \{v_1, \ldots, v_{\min\{t,k/2-1\}}\} \)
   - \( B^v \leftarrow \) a set of \( \min\{d_{G_\ell}(v), k/2\} \) neighbors of \( v \) in \( G_\ell \) such that the number of edges between \( B^v \) and \( P^v \) is maximized.
   - \( C^v \leftarrow \) the component of \( G_\ell[\{v\} \cup B^v \cup P^v] \) that contains \( v \)
   - Expand \( C^v \) to be a connected \( k \)-subgraph of \( G \)
4. Output \( \text{ALG4}(G) \leftarrow \) the densest \( C^v \) for \( v \in V \setminus V_h \)
Lemma

If \( k \leq \frac{2}{3} n \), then \( \sigma(\text{ALG4}(G)) \geq \frac{(\sigma^*_k(G) - 2\bar{\sigma})^2}{2\max\{k, 2d_h\} \cdot k} \geq \frac{(\sigma^*_k(G) - 2\bar{\sigma})^2}{6\max\{k, 2d_h\}} \).
\(O(n^{2/5})\)-approximation

**Lemma**

If \( k \leq \frac{2}{3} n \), then 
\[
\sigma(\text{ALG4}(G)) \geq \frac{(\sigma_k^*(G) - 2\bar{\sigma})^2}{2 \max\{k, 2d_h\}} \cdot \frac{k - 2}{k} \geq \frac{(\sigma_k^*(G) - 2\bar{\sigma})^2}{6 \max\{k, 2d_h\}}.
\]

**Theorem**

A connected \(k\)-subgraph \(C\) of \(G\) can be found in \(O(mn \log n)\) time such that 
\[
\frac{\sigma_k^*(G)}{\sigma(C)} \leq O(n^{2/5}).
\]
**Algorithms**

\[ O\left(\min\{\frac{n^2}{k^2}, k, \frac{n^2}{3}\}\right) \]-approximation for HC\(k\)SP in \(O(mn)\) time
Given $G = (V, E)$ with $|V| \leq k$ and $w \in \mathbb{Z}_+^E$, 

- $O(n^2/k^2)$-approximation
- $(2k)$-approximation

1. **For** every $v \in V$ **do**
2. sort the neighbors of $v$ as $v_1, v_2, \ldots, v_t$ such that $w(vv_1) \geq w(vv_2) \geq \cdots \geq w(vv_t)$, where $t = \min\{d_G(v), k-1\}$
3. $C^v \leftarrow G[\{v_1, v_2, \ldots, v_t\}]$
4. **If** $|C^v| < k$, **then** expand it to be a connected $k$-subgraph
5. **End-For**
6. Output the heaviest $C^v$ for all $v \in V$

- $\min\{n^2/k^2, k\} \leq n^{2/3}$
Conclusion

\[ \Omega(n^{1/3}) \leq \frac{\sigma^*_k(G)}{\sigma(C)} \leq O(\min\{n^{2/5}, n^2/k^2\}) \]

\[ \Omega(n^{1/2}) \leq \frac{\sigma^*_k(G, w)}{\sigma(C', w)} \leq O(\min\{n^{2/3}, n^2/k^2\}) \]

Question

\[ \frac{\text{opt}(G)}{\sigma(C)} = ??? \]
Xujin Chen, Xiaodong Hu, Changjun Wang

Finding Connected Dense $k$-Subgraphs

xchen@amss.ac.cn
http://people.ucas.ac.cn/~xchen

Thanks
- Arora, Karger, Karpinski, Polynomial time approximation schemes for dense instances of NP-hard problems, *Proc. STOC*, 1995, 284-293.

- Asahiro, Iwama, Tamaki, Tokuyama, Greedily finding a dense subgraph, *J. Algorithms* 34 (2000), 203-221.

- Bhaskara, Charikar, Chlamtac, Feige, Vijayaraghavan, Detecting high log-densities: an $O(n^{1/4})$ approximation for densest $k$-subgraph, *Proc. STOC*, 2010, 201-210.

- Chen, Fleischer, Li, Densest $k$-subgraph approximation on intersection graphs, *Proc. Approximation and Online Algorithms*, 2011, 83-93.

- Feige, Peleg, Kortsarz, The dense $k$-subgraph problem, *Algorithmica* 29 (1001), 410-421.

- Goldberg, Finding a maximum density subgraph. University of California Berkeley, CA (1984).
- Hassin, Rubinstein, Tamir, Approximation algorithms for maximum dispersion, *Oper. Res. Lett.* 21 (1997), 133-137.

- Khot, Ruling out PTAS for graph min-bisection, dense $k$-subgraph, and bipartite clique, *SIAM J. Comput.* 36 (2006), 1025-1071.

- Kortsarz, Peleg, On choosing a dense subgraph, *Proc. FOCS*, 1993, 692-701.

- Lawler, *Combinatorial Optimization: Networks and Matroids*, Courier Dover Publications, 1976.

- Nonner, PTAS for densest $k$-subgraph in interval graphs, *Proc. Algorithms and Data Structures*, 2011, 631-641.

- Srivastav, Wolf, Finding dense subgraphs with semidefinite programming, *Proc. Int’l Workshop on Approximation Algorithms for Combinatorial Optimization*, 1998, 181-191.