On Structural Parameterizations for the 2-Club Problem

Sepp Hartung[1] Christian Komusiewicz[1] André Nichterlein[1]

[1]TU Berlin, Germany

SOFSEM 2013

Problem Definition

s-Club

Input: A graph $G = (V, E)$ and an integer ℓ.

Question: Is there a vertex set $V' \subseteq V$ of size at least ℓ such that $G[V']$ has diameter at most s?

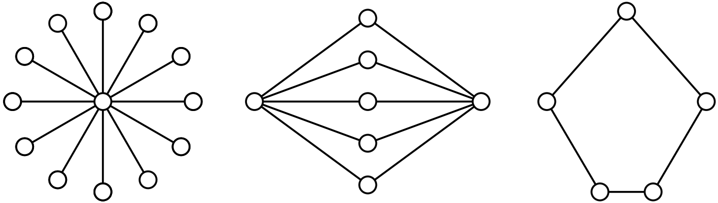
Problem Definition

s-CLUB

Input: A graph $G = (V, E)$ and an integer ℓ.

Question: Is there a vertex set $V' \subseteq V$ of size at least ℓ such that $G[V']$ has diameter at most s?

- 1-CLUB is equivalent to CLIQUE, we focus on 2-CLUB:



star   diamond C_5

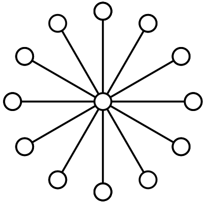
Problem Definition

s-Club

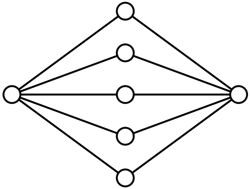
Input: A graph $G = (V, E)$ and an integer ℓ.

Question: Is there a vertex set $V' \subseteq V$ of size at least ℓ such that $G[V']$ has diameter at most s?

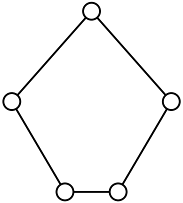
- 1-Club is equivalent to Clique, we focus on 2-Club:



star



diamond



C_5

- max. degree Δ: $\sqrt{\ell - 1} \leq \Delta < \ell - 1$

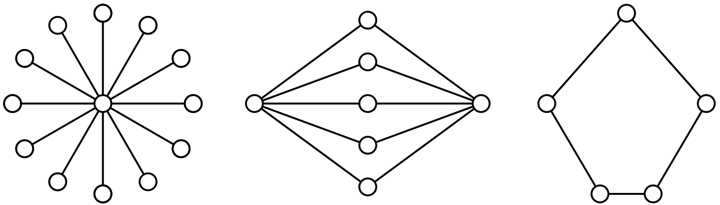
Problem Definition

s-CLUB

Input: A graph $G=(V,E)$ and an integer ℓ.

Question: Is there a vertex set $V' \subseteq V$ of size at least ℓ such that $G[V']$ has diameter at most s?

- 1-CLUB is equivalent to CLIQUE, we focus on 2-CLUB:



star diamond C_5

- max. degree Δ: $\sqrt{\ell - 1} \leq \Delta < \ell - 1$
- non-hereditary graph property

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [Mokken, Quality and Quantity, 1979])

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [MOKKEN, QUALITY AND QUANTITY, 1979])
- experimental studies:

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming
    [Balasundaram et al., J. Comb. Optim., 2005]

Applications and Know Results

Applications:

- analysis of social and biological networks (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming [Balasundaram et al., J. Comb. Optim., 2005]

Applications and Know Results

Applications:

- analysis of social and biological networks (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming [Balasundaram et al., J. Comb. Optim., 2005]

Known Results (for s-Club):

- NP-complete on diameter $s+1$ graphs [Balasundaram et al., J. Comb. Optim., 2005]

Applications and Know Results

Applications:

- analysis of social and biological networks (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming [Balasundaram et al., J. Comb. Optim., 2005]

Known Results (for s-Club):

- NP-complete on diameter $s+1$ graphs [Balasundaram et al., J. Comb. Optim., 2005]
- no poly. algorithm to find an s-club of size $\Delta+2$ (unless $P=NP$) [Butenko & Prokopyev, Tech. Report, 2007]

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming
    [Balasundaram et al., J. Comb. Optim., 2005]

Known Results (for s-Club):

- NP-complete on diameter $s+1$ graphs
  [Balasundaram et al., J. Comb. Optim., 2005]
- no poly. algorithm to find an s-club of size $\Delta+2$ (unless $P=NP$)
  [Butenko & Prokopyev, Tech. Report, 2007]
- inapproximable within factor $|V|^{1/3-\varepsilon}$ for all $\varepsilon > 0$ [Memon, ARES'07]

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming
    [Balasundaram et al., J. Comb. Optim., 2005]

Known Results (for s-Club):

- NP-complete on diameter $s+1$ graphs
  [Balasundaram et al., J. Comb. Optim., 2005]
- no poly. algorithm to find an s-club of size $\Delta+2$ (unless $P=NP$)
  [Butenko & Prokopyev, Tech. Report, 2007]
- inapproximable within factor $|V|^{1/3-\varepsilon}$ for all $\varepsilon>0$ [Memon, ARES'07]
- fixed-parameter algorithmics: [Schäfer et al., Opt. Letters, 2011]

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming
    [Balasundaram et al., J. Comb. Optim., 2005]

Known Results (for s-Club):

- NP-complete on diameter $s+1$ graphs
  [Balasundaram et al., J. Comb. Optim., 2005]
- no poly. algorithm to find an s-club of size $\Delta+2$ (unless $P=NP$)
  [Butenko & Prokopyev, Tech. Report, 2007]
- inapproximable within factor $|V|^{1/3-\varepsilon}$ for all $\varepsilon > 0$ [Memon, ARES'07]
- fixed-parameter algorithmics: [Schäfer et al., Opt. Letters, 2011]
  - $2^{O(\ell \log \ell)}$-time algorithm but no poly kernel wrt. ℓ

Applications and Know Results

Applications:

- analysis of social and biological networks
  (proposed by [Mokken, Quality and Quantity, 1979])
- experimental studies:
  - heuristics [Bourjolly et al., Computers & OR, 2000]
  - integer linear programming
    [Balasundaram et al., J. Comb. Optim., 2005]

Known Results (for s-Club):

- NP-complete on diameter $s+1$ graphs
  [Balasundaram et al., J. Comb. Optim., 2005]
- no poly. algorithm to find an s-club of size $\Delta+2$ (unless $P = NP$)
  [Butenko & Prokopyev, Tech. Report, 2007]
- inapproximable within factor $|V|^{1/3-\varepsilon}$ for all $\varepsilon > 0$ [Memon, ARES'07]
- fixed-parameter algorithmics: [Schäfer et al., Opt. Letters, 2011]
  - $2^{O(\ell \log \ell)}$-time algorithm but no poly kernel wrt. ℓ
  - $2^{|V|-\ell} \cdot n^{O(1)}$ time algorithm for dual size parameter

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?
Is the problem even FPT ($f(k) \cdot n^{O(1)}$-time algorithm)?

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?
Is the problem even FPT ($f(k) \cdot n^{O(1)}$-time algorithm)?

Examples:

1. number of vertices/edges, diameter

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?
Is the problem even FPT ($f(k) \cdot n^{O(1)}$-time algorithm)?

Examples:

1. number of vertices/edges, diameter
2. distance from being non-cyclic: feedback vertex/edge set

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?
Is the problem even FPT ($f(k) \cdot n^{O(1)}$-time algorithm)?

Examples:

1. number of vertices/edges, diameter
2. distance from being non-cyclic: feedback vertex/edge set
3. sparsity: degeneracy

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?
Is the problem even FPT ($f(k) \cdot n^{O(1)}$-time algorithm)?

Examples:

1. number of vertices/edges, diameter
2. distance from being non-cyclic: feedback vertex/edge set
3. sparsity: degeneracy
4. treewidth: Dominating Set, Independent Set and many others are $c^{\text{treewidth}} \cdot n^2$-time solvable for $c \leq 3$

Our Work: Structural Parameterization

A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?
Is the problem even FPT ($f(k) \cdot n^{O(1)}$-time algorithm)?

Examples:

1. number of vertices/edges, diameter
2. distance from being non-cyclic: feedback vertex/edge set
3. sparsity: degeneracy
4. treewidth: Dominating Set, Independent Set and many others are $c^{\text{treewidth}} \cdot n^2$-time solvable for $c \leq 3$
5. chromatic number, size of a vertex cover/independent set/dominating set ect.

Our Work: Structural Parameterization

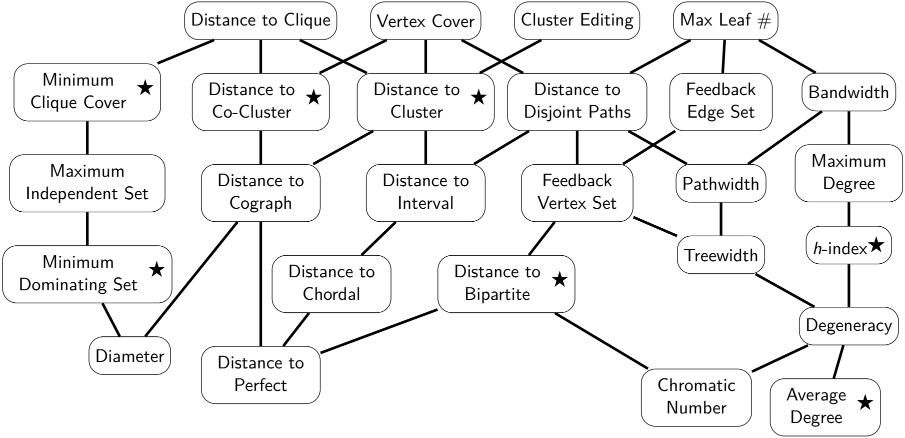
A structural parameter measures the size of some structure in the input graph.

Question: Is the problem polynomial-time solvable if a particular structural parameter k is constant ($n^{f(k)}$-time algorithm)?
Is the problem even FPT ($f(k) \cdot n^{O(1)}$-time algorithm)?

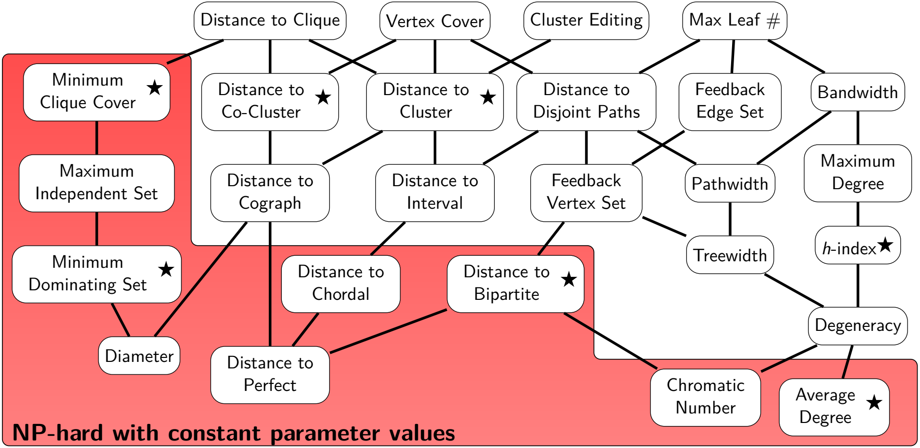
Examples:

1. number of vertices/edges, diameter
2. distance from being non-cyclic: feedback vertex/edge set
3. sparsity: degeneracy
4. treewidth: Dominating Set, Independent Set and many others are $c^{\text{treewidth}} \cdot n^2$-time solvable for $c \leq 3$
5. chromatic number, size of a vertex cover/independent set/dominating set ect.
6. distance to a graph class $\Pi \equiv \#$ vertex deletions to obtain any graph in Π

Our Results for 2-Club

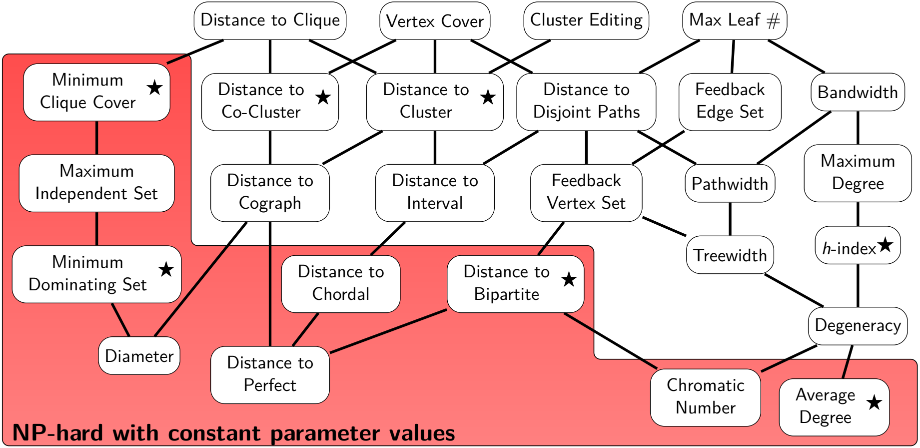


Our Results for 2-Club



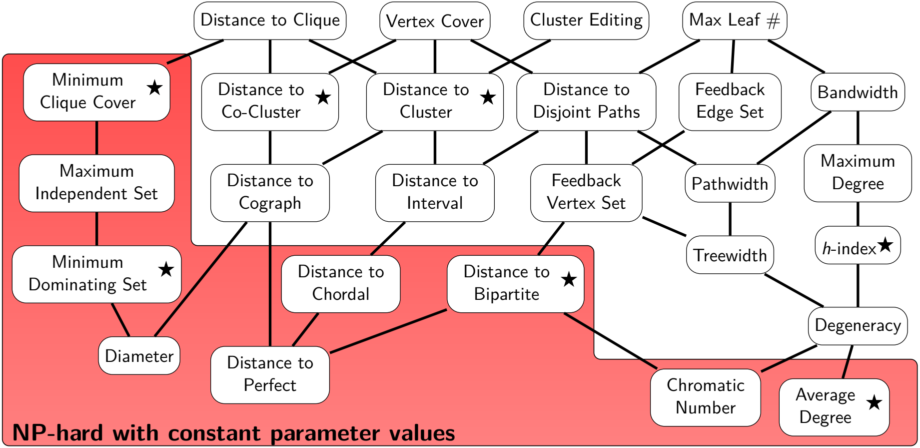
- NP-hard even if graph can be covered by 3 cliques (→ independent set, dominating set at most 3)
- poly-time solvable if independent set at most 2

Our Results for 2-Club

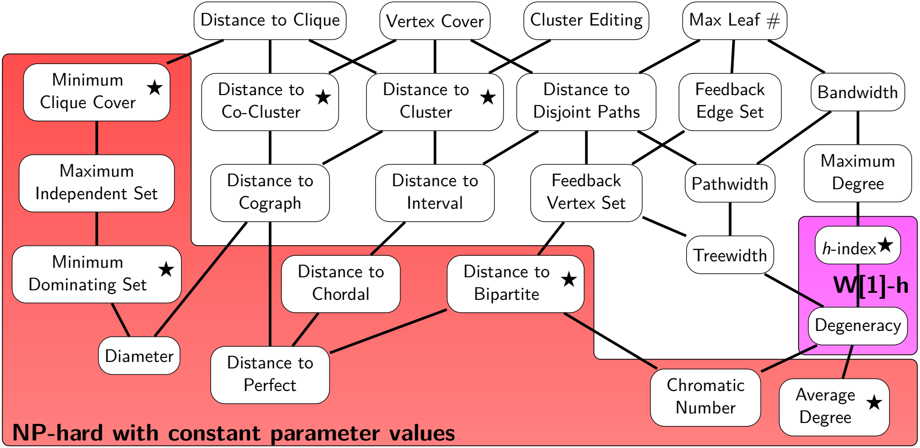


- NP-hard if distance to bipartite is one
- poly-time solvable on bipartite graphs
  [Schäfer 09, diploma thesis]

Our Results for 2-Club

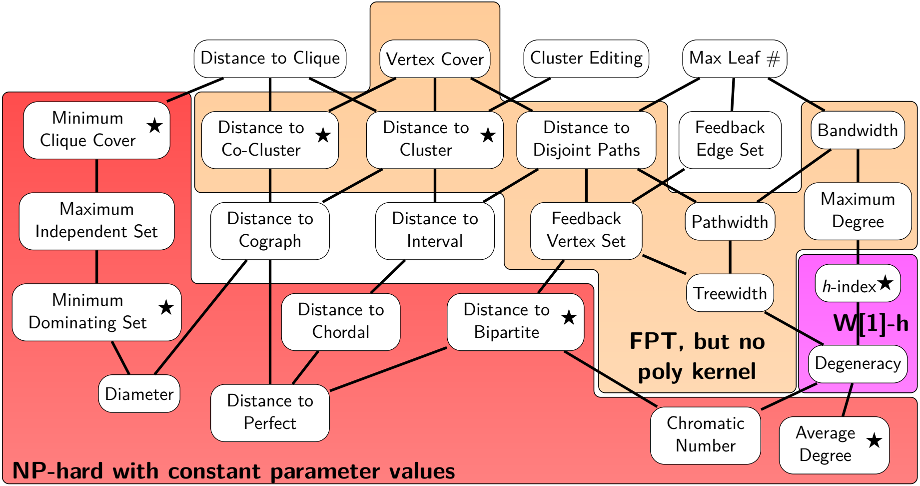


- NP-hard for any constant average degree $\alpha > 2$



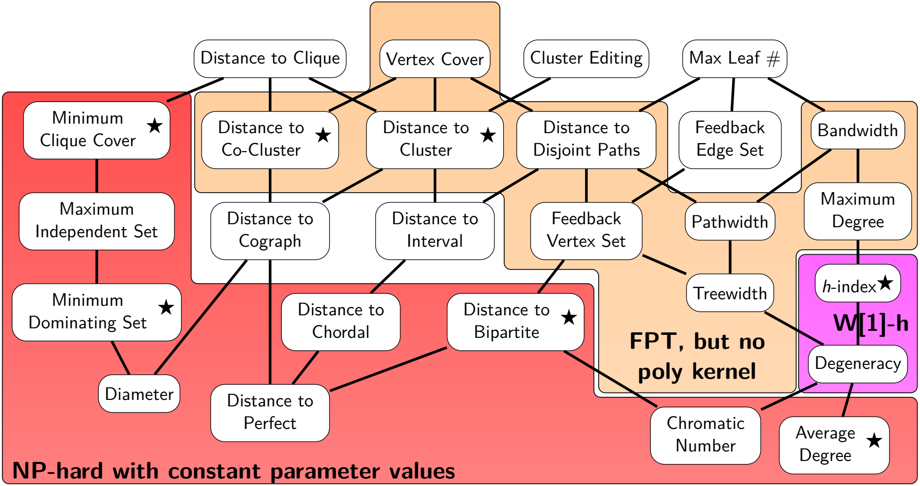
- W[1]-hard w.r.t. h-index of the graphs → no FPT-algorithm
- $n^{f(k)}$-time solvable for k being the h-index

Our Results for 2-Club



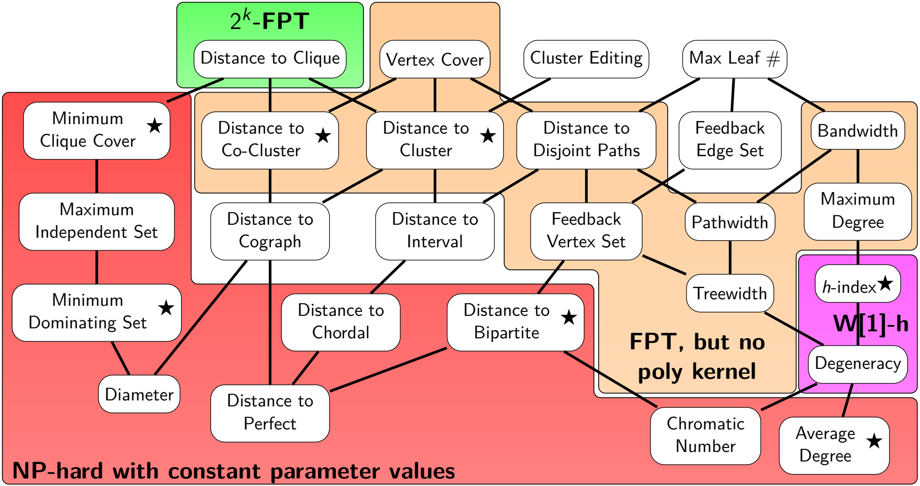
- $2^{2^{\text{treewidth}}}$-time algorithm [H. ET AL. IPEC'12]
- Δ^{Δ}-time algorithm for max degree Δ [SCHÄFER ET AL. OPT. LETTERS 2012]

Our Results for 2-Club



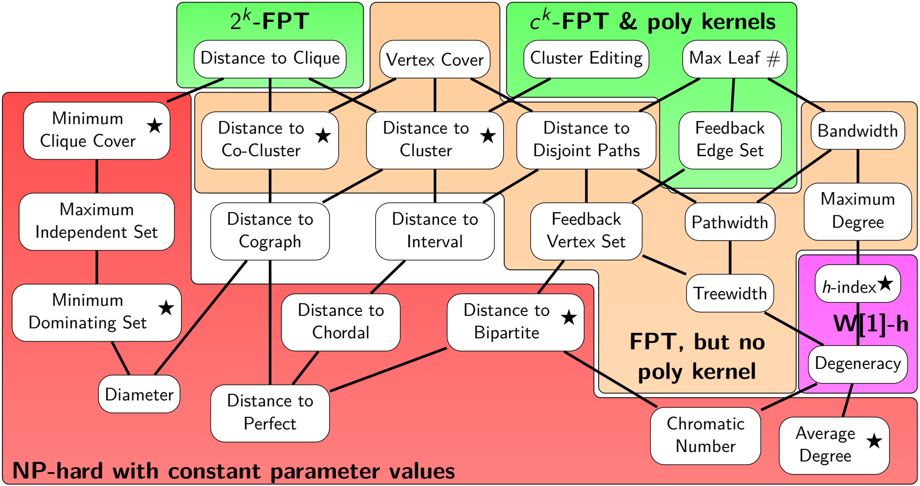
3^{2^k}-time algorithm for k being vertex-deletion distance into cluster graphs or co-cluster graphs.

Our Results for 2-Club



- trivial 2^k-time algorithm for k being distance to 2-club
- no $(2-\varepsilon)^k$-time algorithm [H. IPEC 2012]

Our Results for 2-Club



Poly-time algorithm for constant h-index

h-**index** is the largest number h such that the graph has at least h vertices with degree at least h

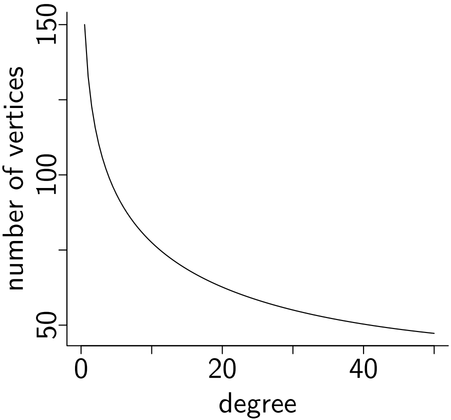
- introduced by [Eppstein and Spiro, WADS 2009]

Poly-time algorithm for constant *h*-index

***h*-index** is the largest number *h* such that the graph has at least *h* vertices with degree at least *h*

- introduced by [Eppstein and Spiro, WADS 2009]

Motivation:
power law degree distribution in social and biological networks [Barabasi & Albert Science 1999 and Jeong et al. Nature 2000]

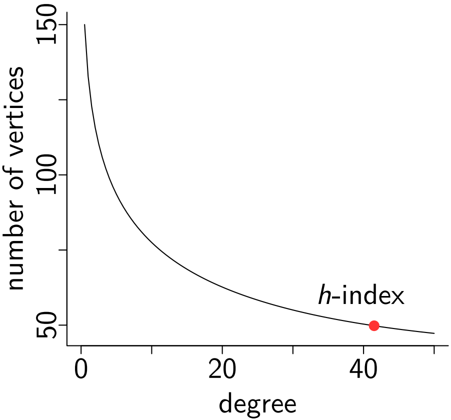


Poly-time algorithm for constant h-index

***h*-index** is the largest number h such that the graph has at least h vertices with degree at least h

- introduced by [Eppstein and Spiro, WADS 2009]

Motivation:
power law degree distribution in social and biological networks [Barabasi & Albert Science 1999 and Jeong et al. Nature 2000]

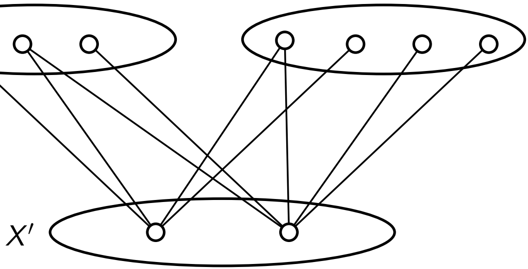


co-Author Citeseer graph with $\approx 230,000$ vertices and max. degree 1372 has h-index 114

Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

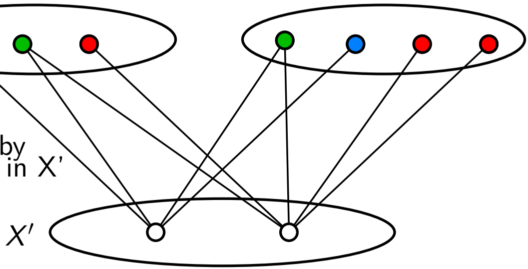


Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

distinguish vertices by
there neighborhood in X'
$\rightarrow$ **twin classes**

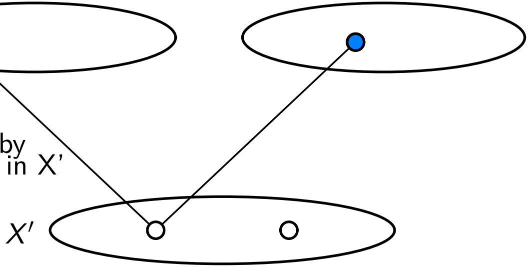


Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

distinguish vertices by
there neighborhood in X'
$\rightarrow$ **twin classes**

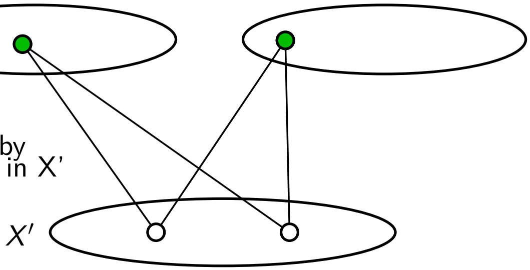


Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

distinguish vertices by
there neighborhood in X'
$\rightarrow$ **twin classes**

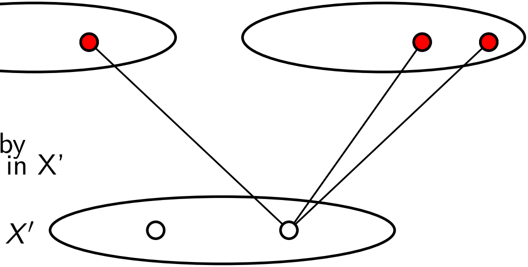


Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

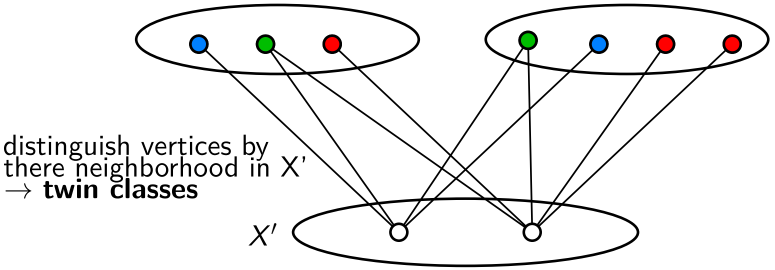
distinguish vertices by
there neighborhood in X'
$\rightarrow$ **twin classes**



Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

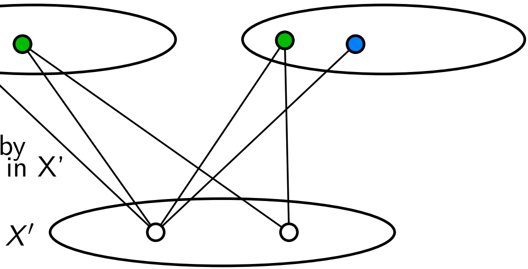


Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
<u>We are left with the following:</u>

distinguish vertices by
there neighborhood in X'
$\rightarrow$ **twin classes**

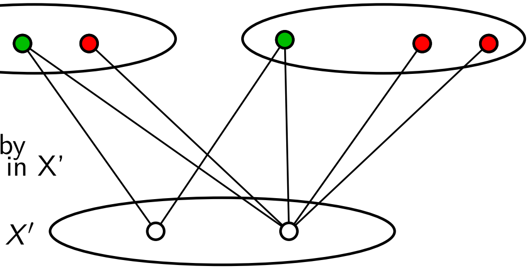


Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

distinguish vertices by
there neighborhood in X'
$\rightarrow$ **twin classes**

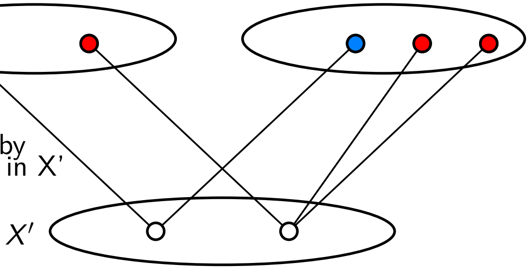


Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:

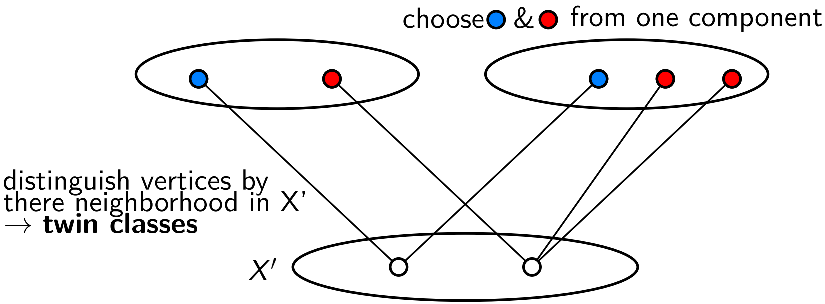
distinguish vertices by
there neighborhood in X'
$\rightarrow$ **twin classes**



Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

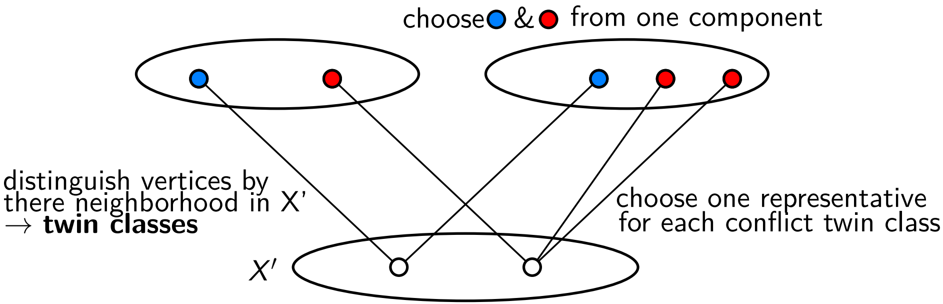
There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
<u>We are left with the following:</u>



Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

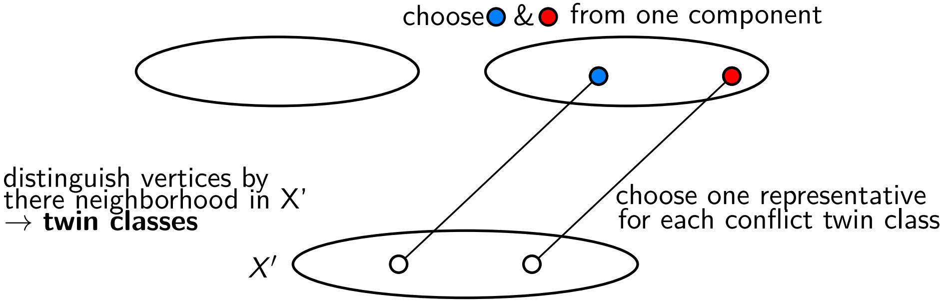
There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:



Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

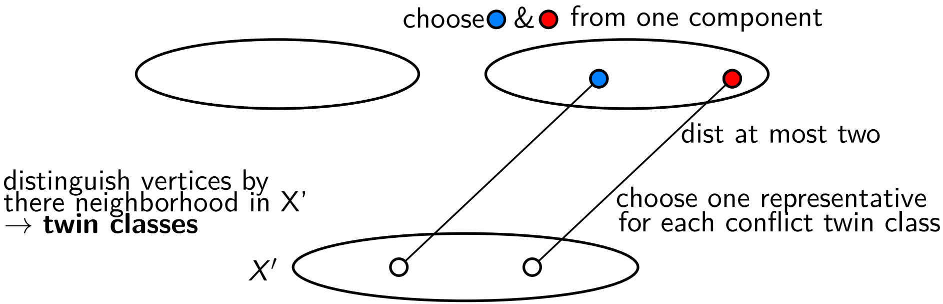
There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:



Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

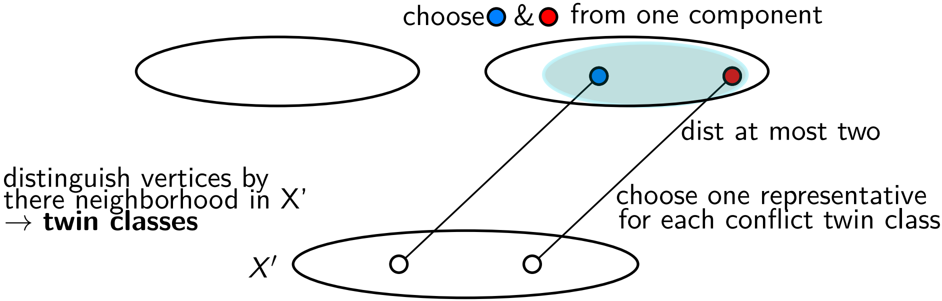
There can be at most h vertices with degree larger than $h \to$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:



Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

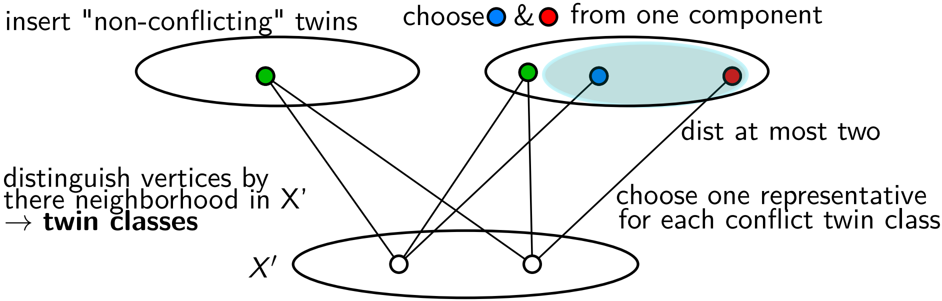
There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
We are left with the following:



Poly-time algorithm for constant h-index II

There is no $O(f(h) \cdot n^{O(1)})$-time algorithm.
Now an $f(h) \cdot n^{2^h}$-time algorithm:

There can be at most h vertices with degree larger than $h \rightarrow$ vertex set X.
Guess the subset $X' \subseteq X$ that is contained in a maximum-size 2-club.
Delete $X \setminus X'$ & all vertices with dist more than two to any of X'.
<u>We are left with the following:</u>



Open Questions

- single-exponential algorithm wrt. vertex cover
- α^k-time algorithm for k being distance to clique with $\alpha < 2$
- poly kernel wrt. distance to clique
- improve $2^{o(\ell \log \ell)}$-time algorithm
- poly-time solvable for constant degeneracy
- FPT wrt. distance to interval graph/cographs
- What about 3-Club?

Open Questions

- single-exponential algorithm wrt. vertex cover
- α^k-time algorithm for k being distance to clique with $\alpha < 2$
- poly kernel wrt. distance to clique
- improve $2^{o(\ell \log \ell)}$-time algorithm
- poly-time solvable for constant degeneracy
- FPT wrt. distance to interval graph/cographs
- What about 3-CLUB?

Thanks!