Complex Networks meet Sparsity

Solving Problems Definable in First-Order Logic

Jan Dreier, Philipp Kuinke, Peter Rossmanith

May 25 2021

RWTH Aachen University, TU Wien

Algorithmic Meta-Theorems

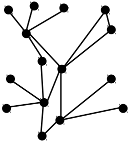
Algorithmic Meta-Theorems

"All problems expressible in a certain logic
can be solved efficiently on certain graph classes"

Algorithmic Meta-Theorems

"All problems expressible in a certain logic can be solved efficiently on certain graph classes"

MSO on treewidth

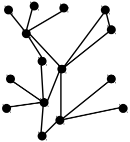


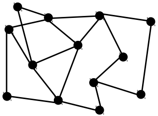
Algorithmic Meta-Theorems

“All problems expressible in a certain logic
can be solved efficiently on certain graph classes”

MSO on treewidth

FO on sparse graphs



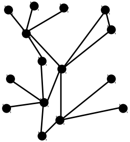


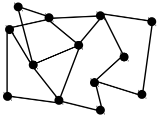
Algorithmic Meta-Theorems

"All problems expressible in a certain logic can be solved efficiently on certain graph classes"

MSO on treewidth

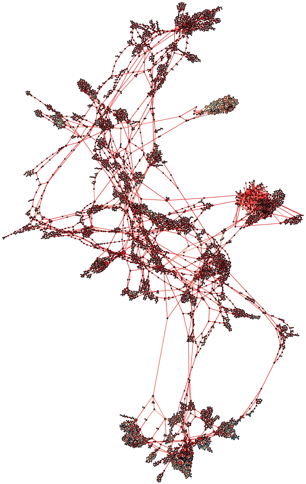
FO on sparse graphs

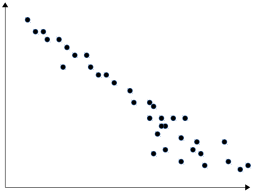


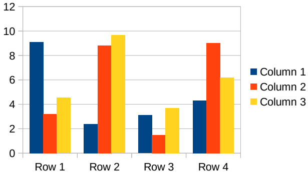


$$\mathcal{C} = \{G_1, \ldots, G_\infty\}$$

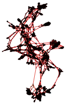
Network Science





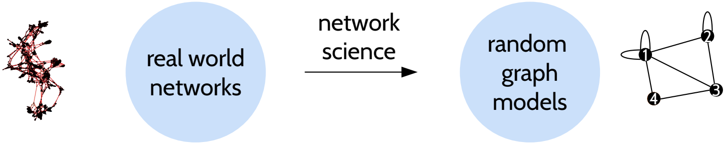


Motivation

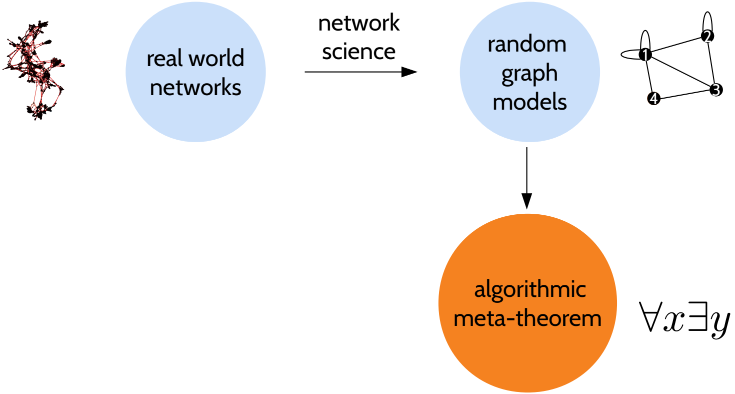


real world
networks

Motivation

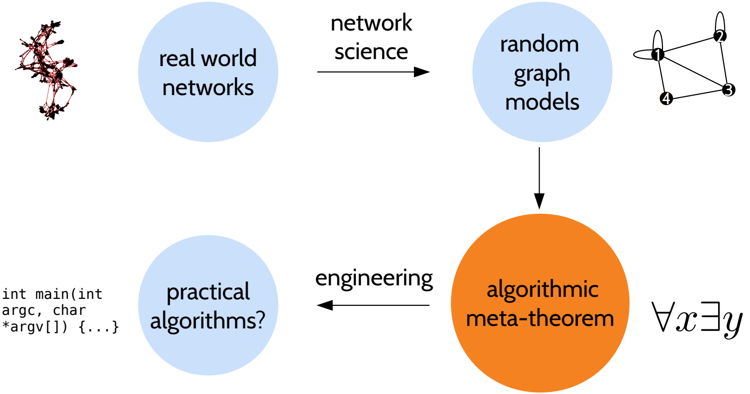

real world networks
network science
random graph models
1
2
3
4

Motivation

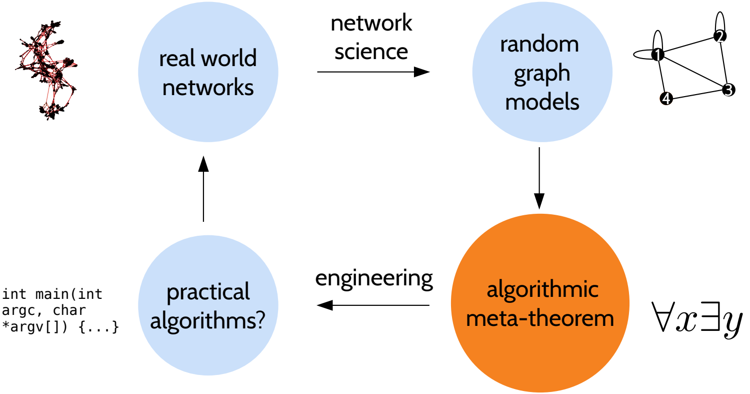


real world networks
network science
random graph models
1
2
3
4
algorithmic meta-theorem
$\forall x \exists y$

Motivation

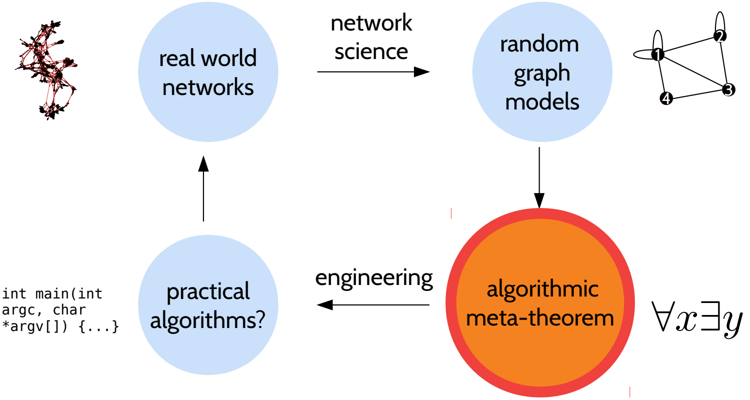

real world networks
network science
random graph models
1
2
3
4
algorithmic meta-theorem
∀x∃y
engineering
practical algorithms?
int main(int argc, char *argv[]) {...}

Motivation



real world networks
network science
random graph models
1
2
3
4
algorithmic meta-theorem
$\forall x \exists y$
engineering
practical algorithms?
int main(int argc, char *argv[]) {...}

Motivation



real world networks
network science
random graph models
1
2
3
4
algorithmic meta-theorem
$\forall x \exists y$
engineering
practical algorithms?
int main(int argc, char *argv[]) {...}

Algorithmic Meta-Theorem

FO Model-Checking [Grohe, Kreutzer, Sieberz 2011]

First-order formulas φ can be evaluated on sparse graphs in time $f(|\varphi|)n^{1+\varepsilon}$.

Algorithmic Meta-Theorem

FO Model-Checking [Grohe, Kreutzer, Siebertz 2011]

First-order formulas φ can be evaluated on sparse graphs in time $f(|\varphi|)n^{1+\varepsilon}$.

- dominating set of size k:

$$\exists x_1 \ldots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

Algorithmic Meta-Theorem

FO Model-Checking [Grohe, Kreutzer, Sieberz 2011]

First-order formulas φ can be evaluated on sparse graphs in time $f(|\varphi|)n^{1+\varepsilon}$.

- dominating set of size k:

$$\exists x_1 \ldots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

- independent set of size k:

$$\exists x_1 \ldots \exists x_k \bigwedge_{i,j} x_i \not\sim x_j \wedge x_i \neq x_j$$

Algorithmic Meta-Theorem

FO Model-Checking [Grohe, Kreutzer, Sieberz 2011]

First-order formulas φ can be evaluated on sparse graphs in time $f(|\varphi|)n^{1+\varepsilon}$.

- dominating set of size k:

$$\exists x_1 \dots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

- independent set of size k:

$$\exists x_1 \dots \exists x_k \bigwedge_{i,j} x_i \not\sim x_j \wedge x_i \neq x_j$$

best algorithms on general graphs: $n^{O(k)}$

Algorithmic Meta-Theorem

FO Model-Checking [Grohe, Kreutzer, Sieberz 2011]

First-order formulas φ can be evaluated on sparse graphs in time $f(|\varphi|)n^{1+\varepsilon}$.

- dominating set of size k:

$$\exists x_1 \ldots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

- independent set of size k:

$$\exists x_1 \ldots \exists x_k \bigwedge_{i,j} x_i \not\sim x_j \wedge x_i \neq x_j$$

best algorithms on general graphs: $n^{O(k)}$

on sparse graphs: $f(k)n^{1+\varepsilon}$

First-Order Logic

can be expressed:

can be expressed:

- dominating set of size k

can be expressed:

- dominating set of size k
- independent set of size k

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k
- fundamentals of SQL database queries

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k
- fundamentals of SQL database queries

on general graphs: $n^{O(k)}$, on sparse graphs: $f(k)n^{1+\varepsilon}$

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k
- fundamentals of SQL database queries

on general graphs: $n^{O(k)}$, on sparse graphs: $f(k)n^{1+\varepsilon}$

---

cannot be expressed:

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k
- fundamentals of SQL database queries

on general graphs: $n^{O(k)}$, on sparse graphs: $f(k)n^{1+\varepsilon}$

---

cannot be expressed:

- are two vertices in the same connected component?

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k
- fundamentals of SQL database queries

on general graphs: $n^{O(k)}$, on sparse graphs: $f(k)n^{1+\varepsilon}$

---

cannot be expressed:

- are two vertices in the same connected component?
- do two vertices have distance at most $\log(n)$?

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k
- fundamentals of SQL database queries

on general graphs: $n^{O(k)}$, on sparse graphs: $f(k)n^{1+\varepsilon}$

---

cannot be expressed:

- are two vertices in the same connected component?
- do two vertices have distance at most $\log(n)$?
- is something acyclic?

First-Order Logic

can be expressed:

- dominating set of size k
- independent set of size k
- motif counting of size k
- fundamentals of SQL database queries

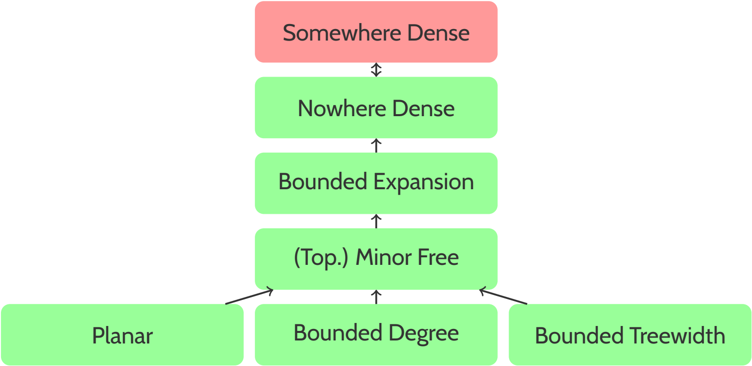
on general graphs: $n^{O(k)}$, on *sparse graphs*: $f(k)n^{1+\varepsilon}$

---

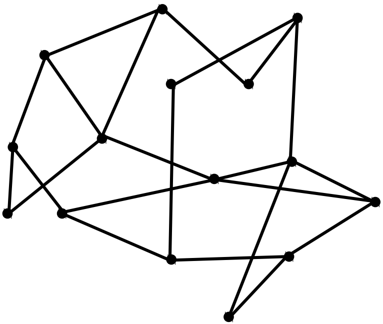
cannot be expressed:

- are two vertices in the same connected component?
- do two vertices have distance at most $\log(n)$?
- is something acyclic?

Sparse Graph Classes



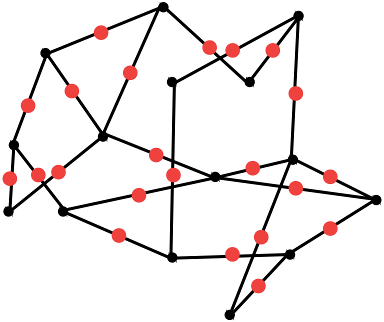
What Are Sparse Graphs?



$$\frac{|E|}{|V|} \leq \ c$$

for every graph
in the graph class.

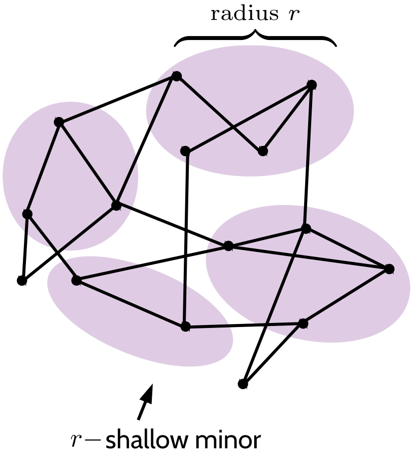
What Are Sparse Graphs?



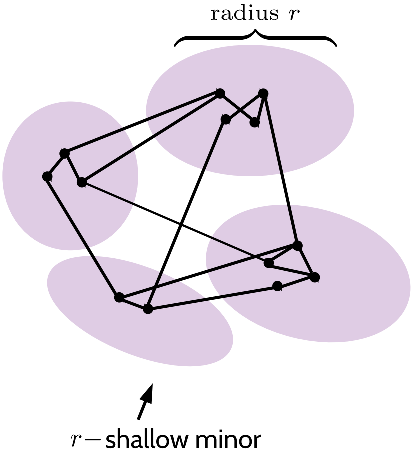
$$\frac{|E|}{|V|} \leq 2$$

for every graph
in the graph class.

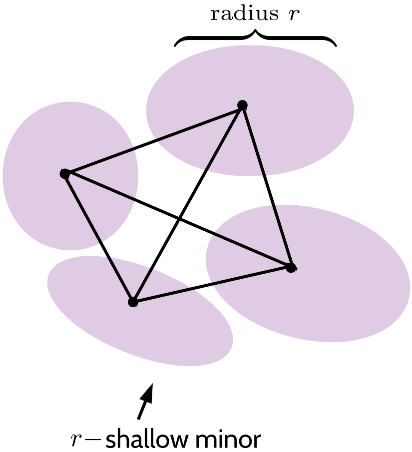
What Are Sparse Graphs?



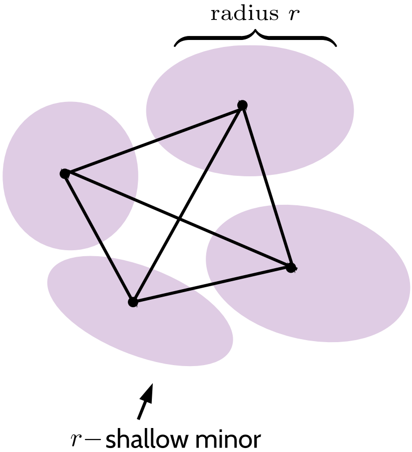
What Are Sparse Graphs?



What Are Sparse Graphs?



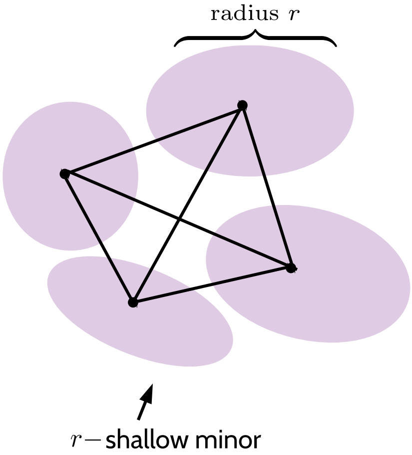
What Are Sparse Graphs?



$$\frac{|E|}{|V|} \leq f(r)$$

for every r-shallow minor of every graph in the graph class.

What Are Sparse Graphs?



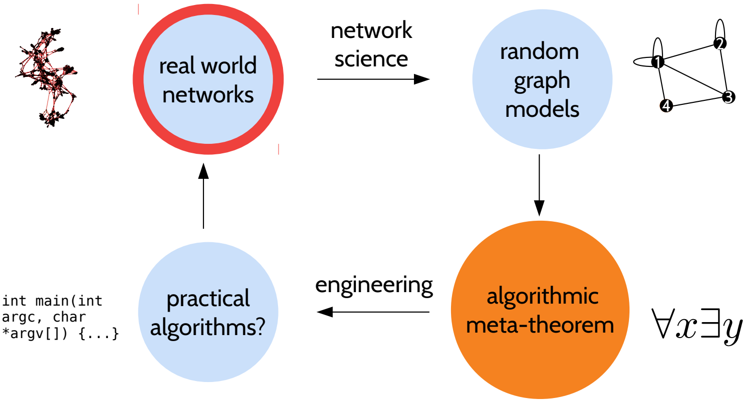
$$\frac{|E|}{|V|} \leq f(r)$$

for every r-shallow minor of every graph in the graph class.

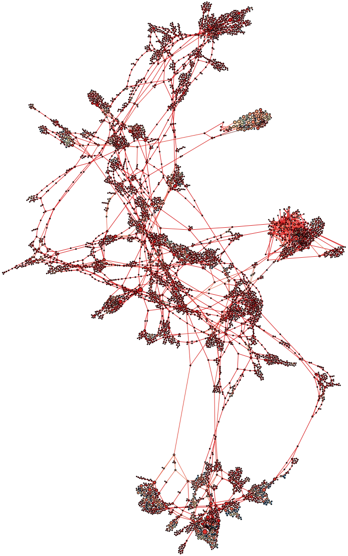
real world: for small r (such as 1,2,3) sparse (w.r.t. gen. col. numbers)

[Nadara, Pilipczuk, Rabinovich, Reidl, Siebertz 2019]

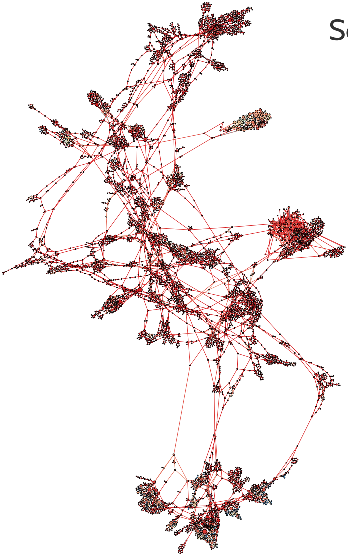
The Real World



The Real World

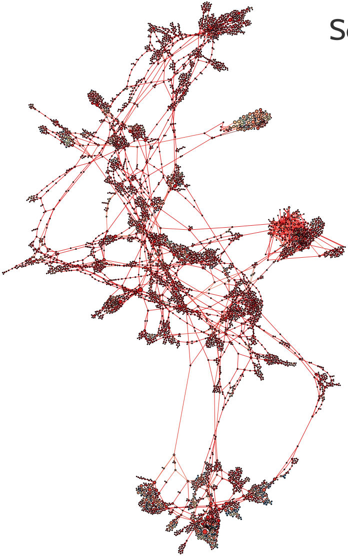


The Real World



Some central properties:

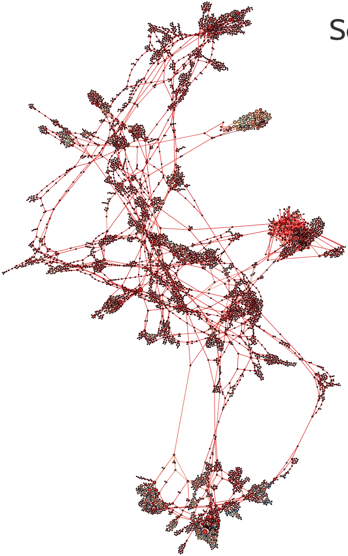
The Real World



Some central properties:

- **Skewed degree distribution**
  Fraction of vertices with degree k proportional to $k^{-\alpha}$ with $2 \leq \alpha \leq 3$?

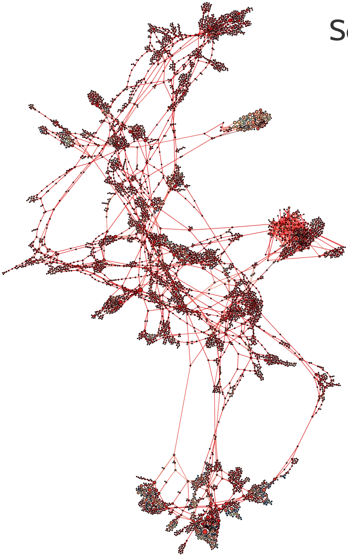
The Real World



Some central properties:

- **Skewed degree distribution**
  Fraction of vertices with degree k proportional to $k^{-\alpha}$ with $2 \leq \alpha \leq 3$?
- **Clustered**
  If we have a common friend we are likely friends as well

The Real World

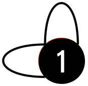


Some central properties:

- **Skewed degree distribution**
  Fraction of vertices with degree k proportional to $k^{-\alpha}$ with $2 \leq \alpha \leq 3$?
- **Clustered**
  If we have a common friend we are likely friends as well
- **Small-world property**
  Everyone is close to everyone

Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



Example: Preferential Attachment Model

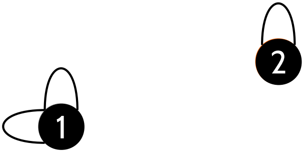
Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.

Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.

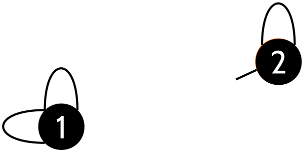
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



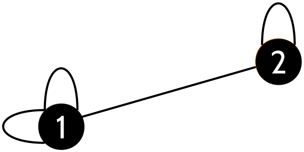
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



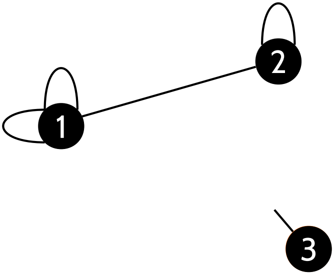
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



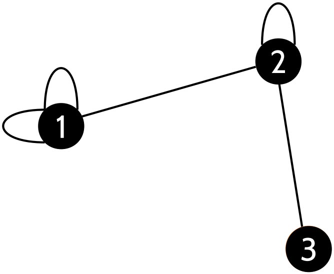
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



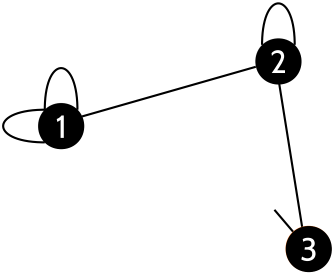
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



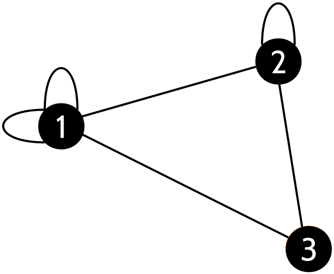
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



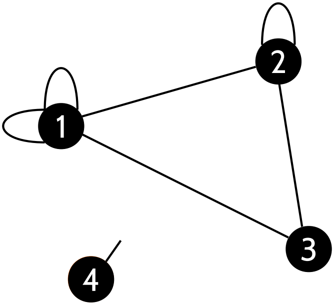
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



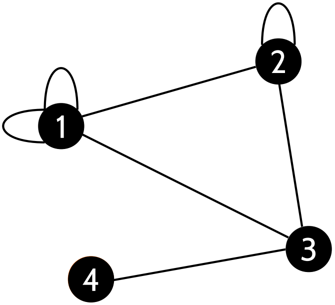
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



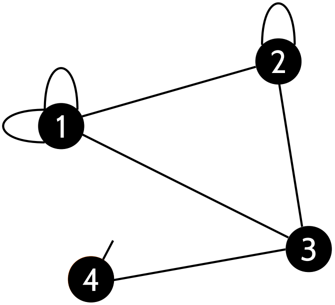
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



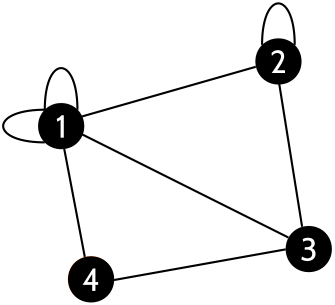
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



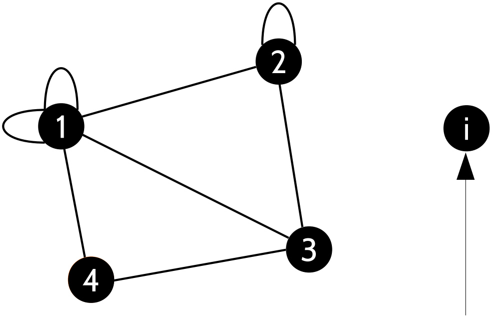
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



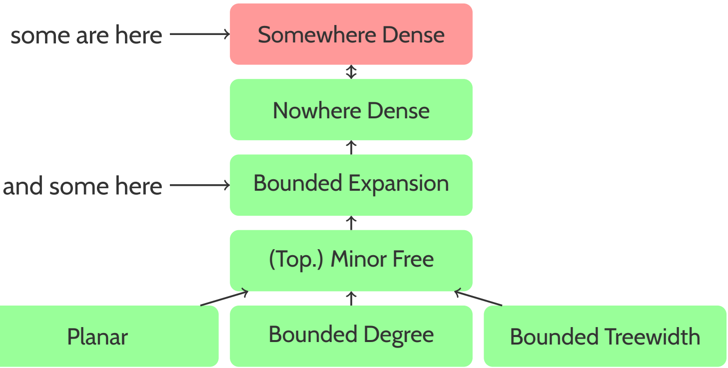
Example: Preferential Attachment Model

Introduced by Barabási and Albert in 1999 to explain the structure of the world wide web.



$$\text{expected degree} \approx \sqrt{\frac{n}{i}}$$

Sparsity of Random Graph Models



[Grohe 2001], [Farrell et. al. 2015], [Demaine et. al. 2019], [Dreier et. al. 2020]

Results

Our Result

A random graph model is *3-power-law-bounded* if (roughly speaking):

[Dreier, Kuinke, Rossmanith 2020]

Our Result

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ **with** $2 \leq \alpha \leq 3$

[Dreier, Kuinke, Rossmanith 2020]

Our Result

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ with $2 \leq \alpha \leq 3$

- unclustered
  real networks: typically clustered

[Dreier, Kuinke, Rossmanith 2020]

Our Result

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ with $2 \leq \alpha \leq 3$
- unclustered
  real networks: typically clustered

Theorem

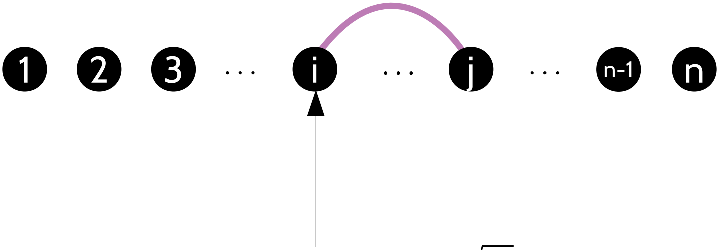
Given a first-order sentence φ and a graph G sampled from a 3-power-law-bounded model, one can decide whether φ is true on G in expected time $f(|\varphi|)n^{1+\varepsilon}$ for every $\varepsilon > 0$.

[Dreier, Kuinke, Rossmanith 2020]

Example: Chung–Lu Model

A more direct way to get a desirable degree distribution.

$$\frac{1}{\sqrt{i \cdot j}}$$



$$\text{expected degree} \approx \sqrt{\frac{n}{i}}$$

α-power-law-boundedness

A random graph model with vertices $1, \ldots, n$ is 3-*power-law-bounded* if the probability that some subset of edges $E \subseteq \binom{\{1,\ldots,n\}}{2}$ is present is at most

α-power-law-boundedness

A random graph model with vertices $1, \ldots, n$ is 3-*power-law-bounded* if the probability that some subset of edges $E \subseteq \binom{\{1,\ldots,n\}}{2}$ is present is at most

$$\prod_{ij \in E} \frac{1}{\sqrt{i \cdot j}}.$$

α-power-law-boundedness

A random graph model with vertices $1, \ldots, n$ is 3-*power-law-bounded* if the probability that some subset of edges $E \subseteq \binom{\{1,\ldots,n\}}{2}$ is present is at most

$$\log(n)^{O(|E|^2)} \prod_{ij \in E} \frac{1}{\sqrt{i \cdot j}}.$$

α-power-law-boundedness

A random graph model with vertices $1, \ldots, n$ is 3-*power-law-bounded* if the probability that some subset of edges $E \subseteq \binom{\{1,\ldots,n\}}{2}$ is present is at most

$$\log(n)^{O(|E|^2)} \prod_{ij \in E} \frac{1}{\sqrt{i \cdot j}}.$$

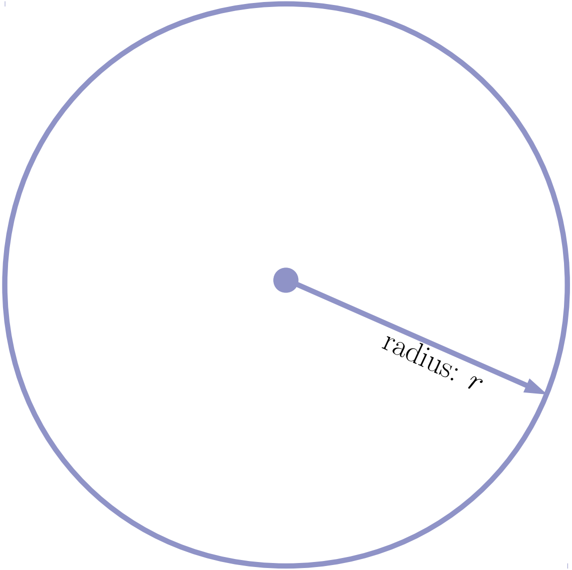
✓

- Preferential attachment model
- Chung–Lu model
- Erdös–Rényi model
- Configuration model
- …

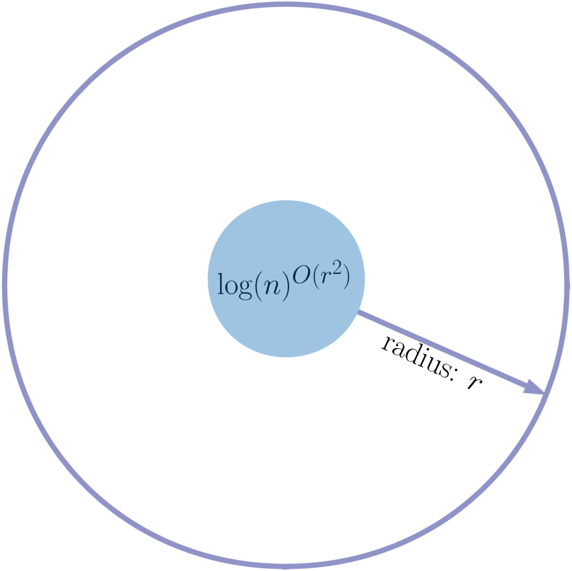
✗

- Hyperbolic random graph model
- random intersection model
- Watts–Strogatz model
- Kleinberg model
- …

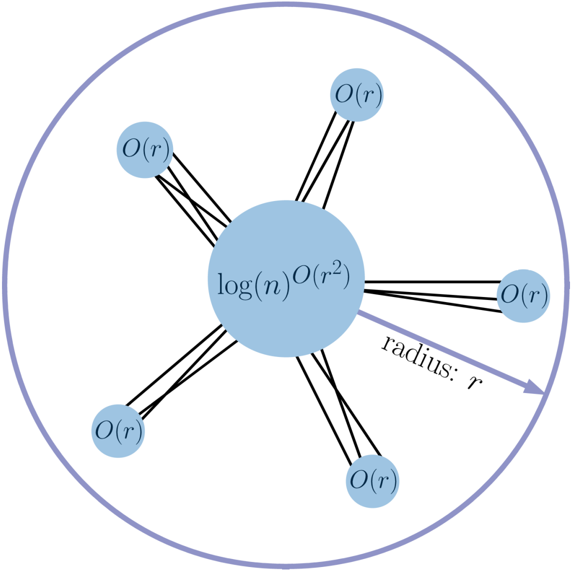
Asymptotic Structure of 3-power-law-bounded models


radius: r

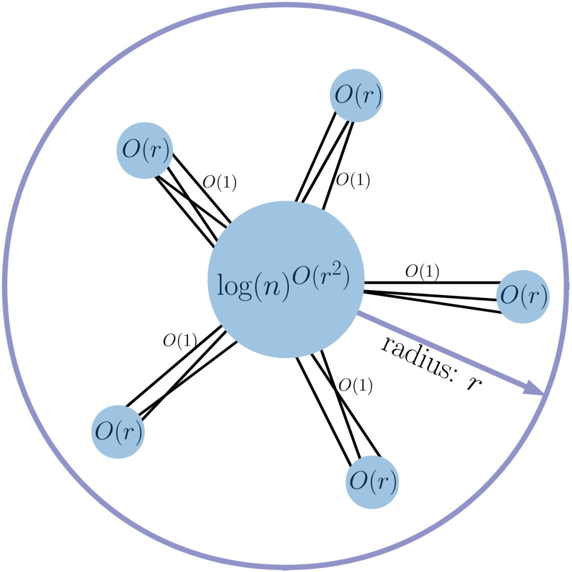
Asymptotic Structure of 3-power-law-bounded models



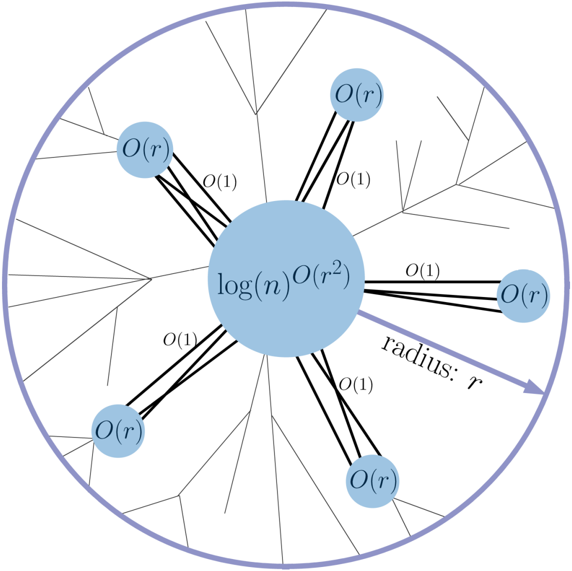
Asymptotic Structure of 3-power-law-bounded models



Asymptotic Structure of 3-power-law-bounded models



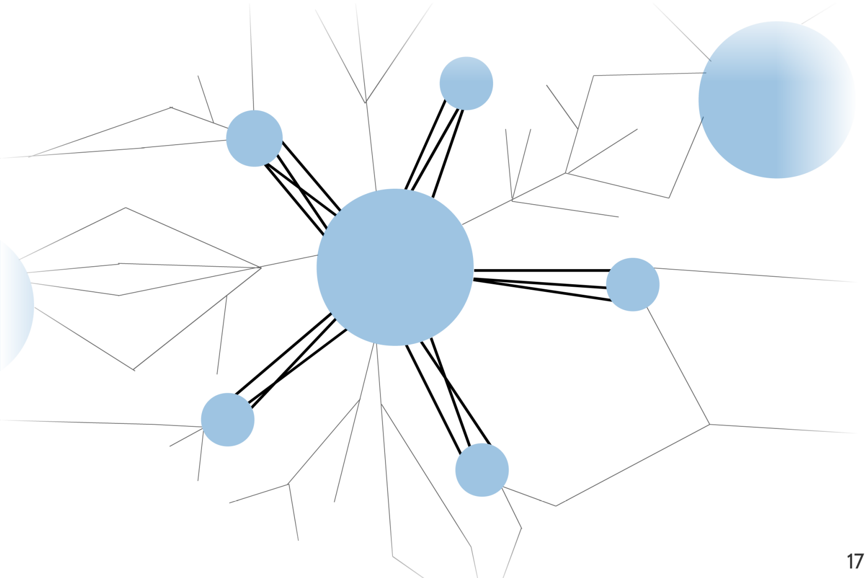
Asymptotic Structure of 3-power-law-bounded models



Algorithm

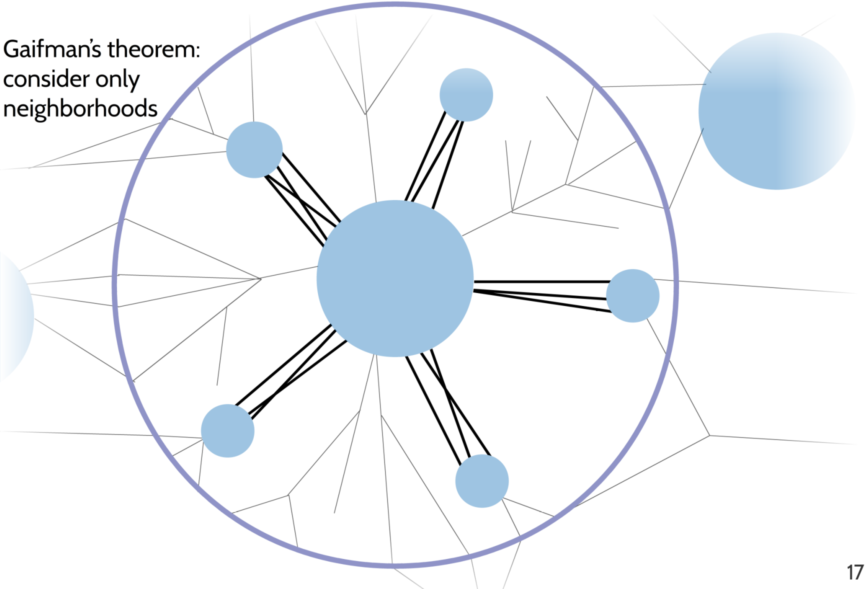
Input: graph sampled from 3-power-law-bounded model

Algorithm



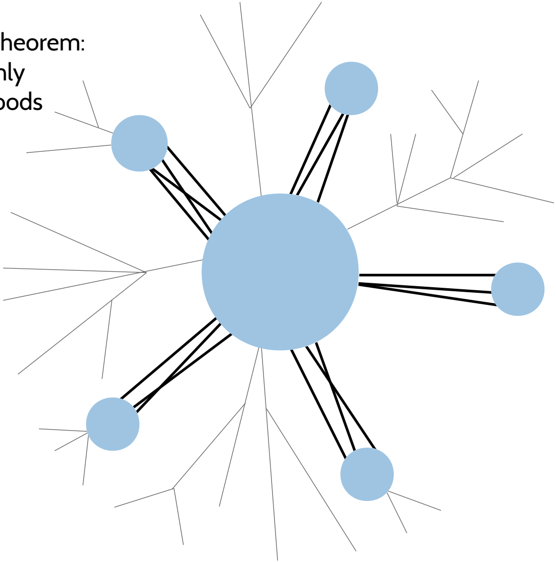
Algorithm

Gaifman's theorem:
consider only
neighborhoods



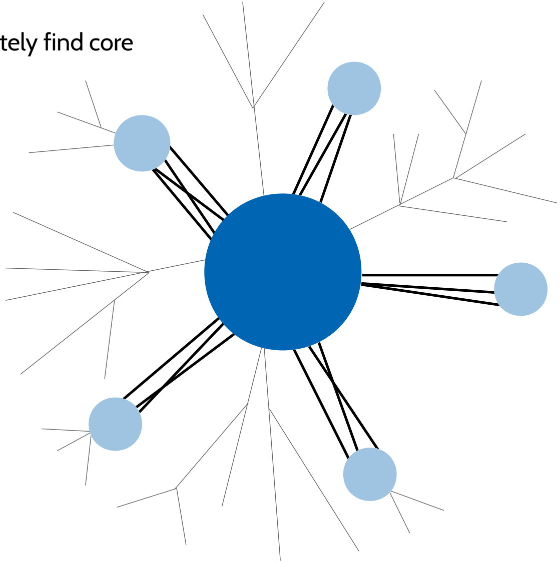
Algorithm

Gaifman's theorem:
consider only
neighborhoods



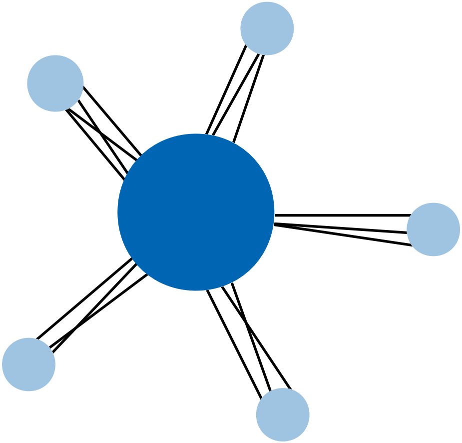
Algorithm

approximately find core



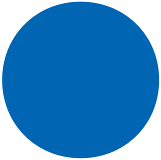
Algorithm

prune trees



Algorithm

prune protrusions

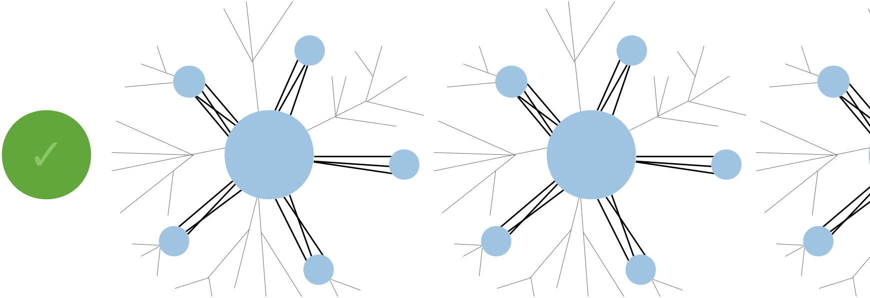


Algorithm

use brute force on core

Algorithm

repeat for every neighborhood

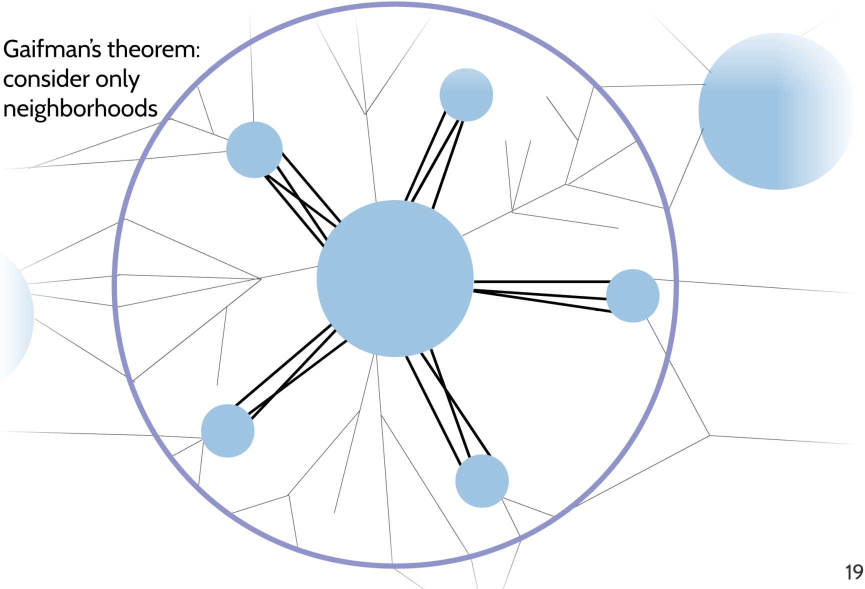


Algorithm

Done!

Practical Considerations I

Gaifman's theorem: consider only neighborhoods



Practical Considerations II

- our run time:

Practical Considerations II

- our run time:

$$\underbrace{2^{2^{2^{2^{2^{2^{2^{2^{2^{2}}}}}}}}}}_{|\varphi|} \cdot n^{1+\varepsilon}$$

Practical Considerations II

- our run time:

$$\underbrace{2^{2^{2^{2^{2^{2^{2^{2^{2^{2}}}}}}}}}}_{|\varphi|} \cdot n^{1+\varepsilon}$$

- under worst-case complexity this is optimal

Practical Considerations II

- our run time:

$$\underbrace{2^{2^{2^{2^{2^{2^{2^{2^{2^{2}}}}}}}}}}_{|\varphi|} \cdot n^{1+\varepsilon}$$

- under worst-case complexity this is optimal
- open question: can we do better in the average-case?

Summary (so far)

A random graph model is *3-power-law-bounded* if (roughly speaking):

Summary (so far)

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ **with** $2 \leq \alpha \leq 3$

Summary (so far)

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ with $2 \leq \alpha \leq 3$

- unclustered
  real networks: typically clustered

Summary (so far)

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ with $2 \leq \alpha \leq 3$
- unclustered
  real networks: typically clustered

Theorem

Given a first-order sentence φ and a graph G sampled from a 3-power-law-bounded model, one can decide whether φ is true on G in expected time $f(|\varphi|)n^{1+\varepsilon}$ for every $\varepsilon > 0$.

Summary (so far)

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ with $2 \leq \alpha \leq 3$

  $\boxed{\text{optimal}}$

- unclustered
  real networks: typically clustered

Theorem

Given a first-order sentence φ and a graph G sampled from a 3-power-law-bounded model, one can decide whether φ is true on G in expected time $f(|\varphi|)n^{1+\varepsilon}$ for every $\varepsilon > 0$.

Summary (so far)

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ with $2 \leq \alpha \leq 3$

  optimal

- unclustered
  real networks: typically clustered

  ???

Theorem

Given a first-order sentence φ and a graph G sampled from a 3-power-law-bounded model, one can decide whether φ is true on G in expected time $f(|\varphi|)n^{1+\varepsilon}$ for every $\varepsilon > 0$.

Summary (so far)

A random graph model is *3-power-law-bounded* if (roughly speaking):

- fraction of vertices with degree k is $O(k^{-3})$
  real networks: typically $k^{-\alpha}$ with $2 \leq \alpha \leq 3$

  optimal

- unclustered
  real networks: typically clustered

  ???

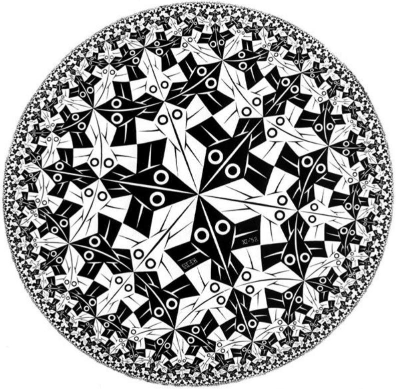
Theorem

Given a first-order sentence φ and a graph G sampled from a 3-power-law-bounded model, one can decide whether φ is true on G in expected time $f(|\varphi|)n^{1+\varepsilon}$ for every $\varepsilon > 0$.

Big Question: model-checking on clustered models?

Want a Challenge?

Can we solve the problem on *Hyperbolic Random Graphs*?



[Cirle Limit I, M.C. Escher]

Lower Bounds

Average Case Complexity

- worst-case: P and NP

Average Case Complexity

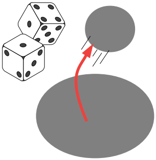
- worst-case: P and NP
- average-case: avgP and (NP, PComp)?

Average Case Complexity

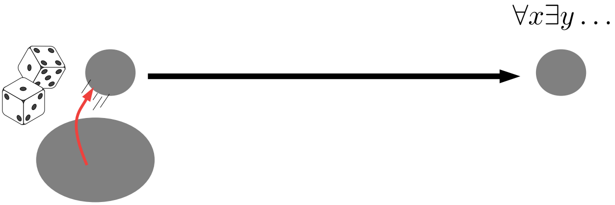
- worst-case: P and NP
- average-case: avgP and (NP, PComp)?
- expected time example:

$$\sum_{x \in \mathbf{N}} \mathrm{P}[X = x] \cdot \text{time}(x) = O(1)$$

With Colors



With Colors

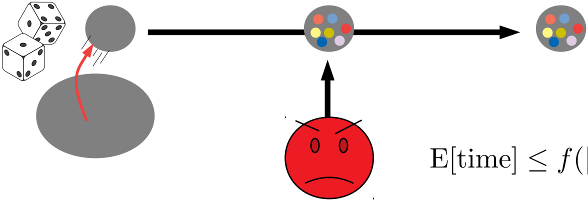


With Colors

$\forall x \exists y \ldots$

$\mathrm{E}[\text{time}] \le f(|\varphi|)n^{1+\varepsilon}?$

With Colors

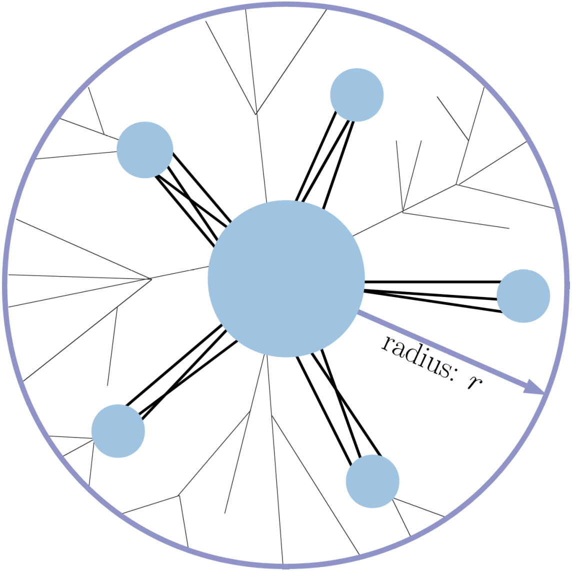


$\forall x \exists y \ldots$

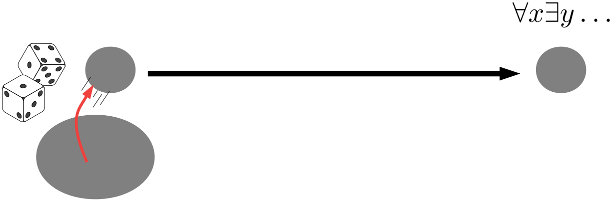
$\mathrm{E}[\text{time}] \leq f(|\varphi|)n^{1+\varepsilon}?$

What if $\alpha < 3$?

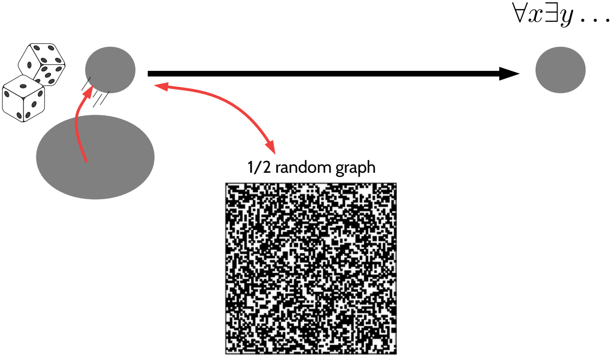
[Dreier, Rossmanith 2019]



Without Colors



Without Colors

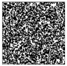


Zero-One Laws

1/2 random graph

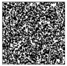
Zero-One Laws

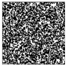
1/2 random graph

$\Pr[$  $\in$  $] \to 1$

Zero-One Laws

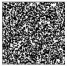
1/2 random graph

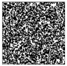
$\Pr[$  $\in$  $] \to 1$

$\Pr[$ apex vertex $\in$  $] \to 0$

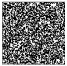
Zero-One Laws

1/2 random graph

Pr[ $\in$ ] $\to 1$

Pr[apex vertex $\in$ ] $\to 0$

---

for all φ Pr[ $\models \varphi$] $\to \{0, 1\}$

Average-Case Hierarchy Collapse

- worst-case
  - W[1]-complete:

$$\exists x_1 \ldots \exists x_k$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[2]-complete:

$$\exists x_1 \dots \exists x_k \forall y$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[3]-complete:

$$\exists x_1 \ldots \exists x_k \forall y \exists z$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \ldots \exists x_k \forall y \exists z \forall w$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \ldots \exists x_k \forall y \exists z \forall w$$

- average-case on 1/2 random graph

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \dots \exists x_k \forall y \exists z \forall w$$

- average-case on 1/2 random graph
  - FPT (i.e., $f(|\varphi|)n$ time):

$$\exists x_1 \dots \exists x_k$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \ldots \exists x_k \forall y \exists z \forall w$$

- average-case on 1/2 random graph
  - FPT (i.e., $f(|\varphi|)n$ time):

$$\exists x_1 \ldots \exists x_k$$

  - equally hard:

$$\exists x_1 \ldots \exists x_k \forall y$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \ldots \exists x_k \forall y \exists z \forall w$$

- average-case on 1/2 random graph
  - FPT (i.e., $f(|\varphi|)n$ time):

$$\exists x_1 \ldots \exists x_k$$

  - equally hard:

$$\exists x_1 \ldots \exists x_k \forall y \exists z$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \ldots \exists x_k \forall y \exists z \forall w$$

- average-case on 1/2 random graph
  - FPT (i.e., $f(|\varphi|)n$ time):

$$\exists x_1 \ldots \exists x_k$$

  - equally hard:

$$\exists x_1 \ldots \exists x_k \forall y \exists z \forall w$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \dots \exists x_k \forall y \exists z \forall w$$

- average-case on 1/2 random graph
  - FPT (i.e., $f(|\varphi|)n$ time):

$$\exists x_1 \dots \exists x_k$$

  - equally hard:

$$\exists x_1 \dots \exists x_k \forall y \exists z \forall w \exists u$$

[Dreier, Lotze, Rossmanith 2020]

Average-Case Hierarchy Collapse

- worst-case
  - W[4]-complete:

$$\exists x_1 \dots \exists x_k \forall y \exists z \forall w$$

- average-case on 1/2 random graph
  - FPT (i.e., $f(|\varphi|)n$ time):

$$\exists x_1 \dots \exists x_k$$

  - equally hard:

$$\exists x_1 \dots \exists x_k \forall y \exists z \forall w \exists u \dots$$

[Dreier, Lotze, Rossmanith 2020]

Summary

- Solve all problems definable in FO logic on unclustered graphs that are not too skewed.

Summary

- Solve all problems definable in FO logic on unclustered graphs that are not too skewed.
- How about clustered models?

Summary

- Solve all problems definable in FO logic on unclustered graphs that are not too skewed.
- How about clustered models?
- How about about (uncolored) lower bounds?

Thanks!

dreier@ac.tuwien.ac.at