Metric properties of large graphs
Propriétés métriques des grands graphes

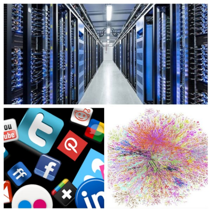
PhD Candidate: Guillaume Ducoffe
Advisor: David Coudert

Université Côte d'Azur, Inria, CNRS, I3S, France

December 9th, 2016

Goals for Network Algorithms: **Scalability**

Growing size of communication networks



Social networks (Facebook $\geq$ 1.79 billion users)

Data Centers (Microsoft $\geq$ 1 million servers)

the Internet ($\geq$ 55811 Autonomous Systems)

"Efficient" algorithms on these graphs?

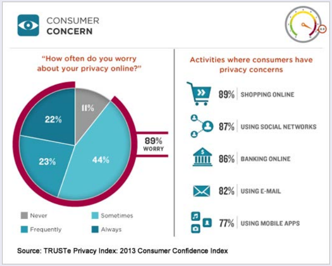
~~polynomial~~ $\rightarrow$ quasi-linear time
~~quadratic~~ $\rightarrow$ (sub)linear space

First issue

need for revisiting textbook (polynomial) graph algorithms

Goals for Network Algorithms: **Privacy**

Raise of **privacy concerns** online



Online discrimination (Machine Learning, heuristics)

Violation of data policies (ex: Google App Education)

Second issue

differential privacy: **preventing data leakage**

Web's transparency: **monitoring data use**

Main lines of the thesis

Information propagation in networks $\Longrightarrow$ combinatorial problems on graphs

Finer-grained complexity analysis of graph problems

NP-hardness, **complexity in P**, *parallel complexity*, *query complexity*, ...

Part I: Metric tree-likeness in graphs

(with COATI team)

- Study of **geometric properties** of the (shortest) path distribution
- Computation of related parameters (**hyperbolicity**, **treelength**, **treebreadth**, treewidth)

algorithmic graph theory

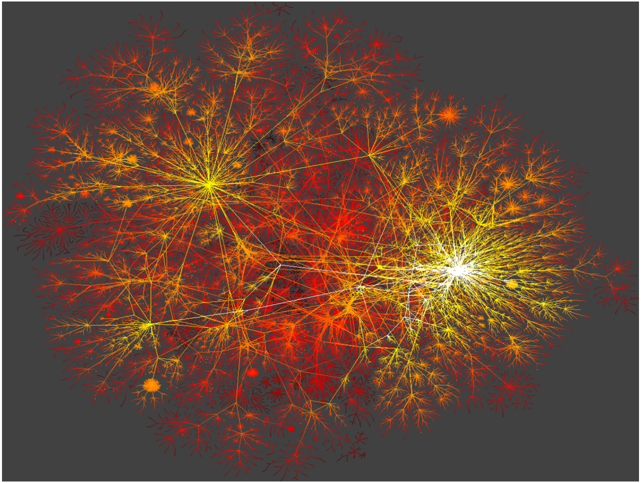
Part II: Privacy at large scale in social graphs

(with Social Networks lab, Columbia)

- Solution concepts for **dynamics of communities**
- Ad Targeting Identification

game and learning theory

Metric tree-likeness in graphs



Skitter data depicting a macroscopic snapshot of Internet connectivity, with selected backbone ISPs (Internet Service Provider) colored separately. By K. C. Claffy (http://www.caida.org/publications/papers/bydate/index.xml)

Key notions

treelikeness $\sim$ closeness of a graph to a tree (w.r.t. some property)

Motivation: optimization problems easier to solve

Tree decompositions [Robertson and Seymour'86]

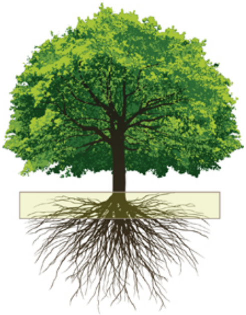
Representation of a graph as a tree preserving **connectivity properties**.

Algorithm on the tree representations

Gromov hyperbolicity [Gromov'87]

(Local) closeness of the graph metric to a **tree metric**.

f(hyperbolicity)-approximation for distance problems on graphs



Gromov hyperbolicity

Definition

G is δ-hyperbolic $\iff$ every 4-tuple $u, v, x, y \in V(G)$ can be **mapped to the nodes of a tree** (possibly edge-weighted) with distortion:

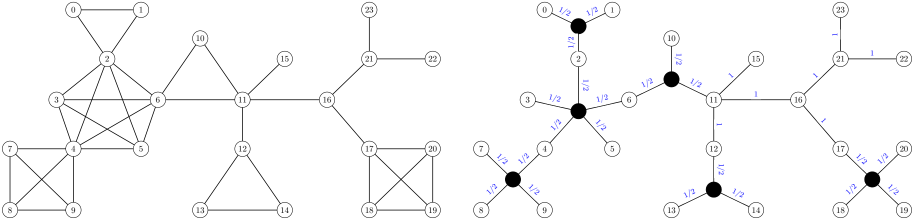
$$\max_{s,t \in \{u,v,x,y\}} |dist_G(s,t) - dist_T(\varphi(s), \varphi(t))| \leq \delta.$$

Trees are 0*-hyperbolic*

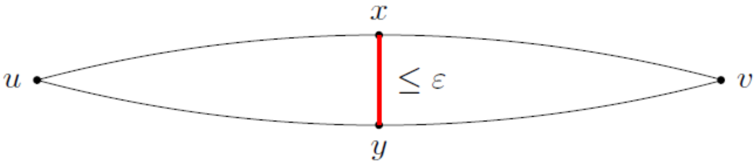
Cliques are 0*-hyperbolic*

Examples

- Block graphs are 0-hyperbolic



- Cycle C_n with n vertices is $\lfloor n/4 \rfloor$-hyperbolic



$$2\delta \geq \varepsilon = \lfloor n/2 \rfloor$$

On computing Gromov hyperbolicity

Four-point definition [Gromov'87]

The hyperbolicity of a connected graph $G = (V, E)$, denoted by $\delta(G)$, is equal to the smallest δ such that for every 4-tuple u, v, x, y of V:

$$dist_G(u,v) + dist_G(x,y) \leq \max\{dist_G(u,x) + dist_G(v,y), dist_G(u,y) + dist_G(v,x)\} + 2\delta$$

Computing hyperbolicity

State of the art:

- *combinatorial algorithms* in $\mathcal{O}(n^4)$-time

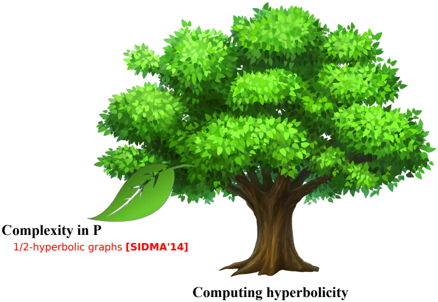
  [Cohen, Coudert, Lancin'15]

  [Borassi, Coudert, Crescenzi, Marino'15]

- in $\mathcal{O}(n^{3.69})$-time (using matrix product)

  [Fournier and Vigneron'15]

Recognition of graphs with small hyperbolicity



Related work

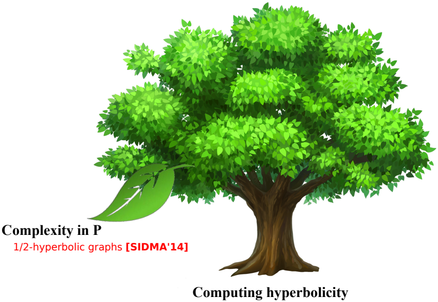
- 0-hyperbolic graphs are block-graphs
  $\longrightarrow \mathcal{O}(n+m)$-time recognition.

  [Howorka'79]

- Deciding $\delta(G) \leq 1$ cannot be done in $\mathcal{O}(n^{2-\varepsilon})$-time (under SETH)

  [Borassi, Crescenzi, Habib'16]

Recognition of graphs with small hyperbolicity



Related work

- 0-hyperbolic graphs are block-graphs
  $\longrightarrow \mathcal{O}(n+m)$-time recognition.

  [Howorka'79]

- **Contribution:** *Recognition of 1/2-hyperbolic graphs*

  [Coudert and D. SIDMA'14]

- Deciding $\delta(G) \leq 1$ cannot be done in $\mathcal{O}(n^{2-\varepsilon})$-time (under SETH)

  [Borassi, Crescenzi, Habib'16]

Subcubic equivalence

both problems can be solved in truly subcubic-time or none of them can.

Theorem [Coudert and D. SIDMA'14]

The two following problems are **subcubic equivalent**:

- deciding whether a graph has hyperbolicity equal to $1/2$;
- deciding whether a graph contains an induced cycle of length four.

no combinatorial truly subcubic algorithm is likely to exist

Subcubic equivalence

both problems can be solved in truly subcubic-time or none of them can.

Theorem [Coudert and D. SIDMA'14]

The two following problems are **subcubic equivalent**:

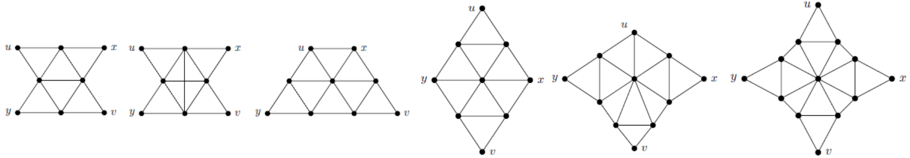
- deciding whether a graph has hyperbolicity equal to $1/2$;
- deciding whether a graph contains an induced cycle of length four.

no combinatorial truly subcubic algorithm is likely to exist

Key ingredients:

- characterization by forbidden isometric subgraphs **[Bandelt and Chepoi'03]**

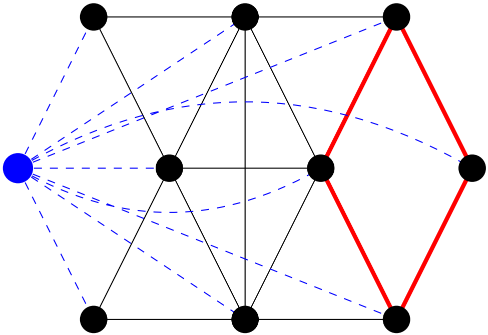
no cycles $C_n,\ n \notin \{3,5\} + \ldots +$



- (modified) graph powers

C_4-free detection $\propto$ 1/2-hyperbolic recognition

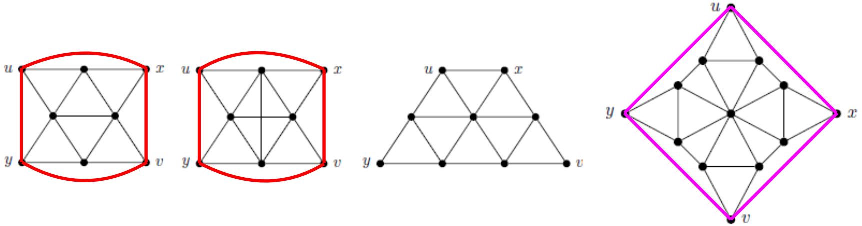
- Observation: G 1/2-hyperbolic $\implies G$ C_4-free
- Remove all other obstructions by lowering $diam(G)$ to 2
  $\longrightarrow$ by adding a **universal vertex**



1/2-hyperbolic recognition $\propto$ C_4-free detection

Reinterpret obstructions as C_4's in (modified) graph powers

- $\delta(G) = 1/2 \Longrightarrow G^j,\ j \geq 1$ and $G^{[2]}$ (modified square) are C_4-free
- obstructions to $\delta(G) = 1/2$ of size $\leq c \Longrightarrow C_4's$ in $G^{\mathcal{O}(c)}$ or $G^{[2]}$



$1/2$-hyperbolic recognition $\propto$ C_4-free detection

Theorem [Coudert and D. SIDMA'14]

$G = (V, E)$ is $1/2$-hyperbolic if and only if none of the graphs $G^j, j \geq 1$ and $G^{[2]}$ contain an induced cycle of length four.

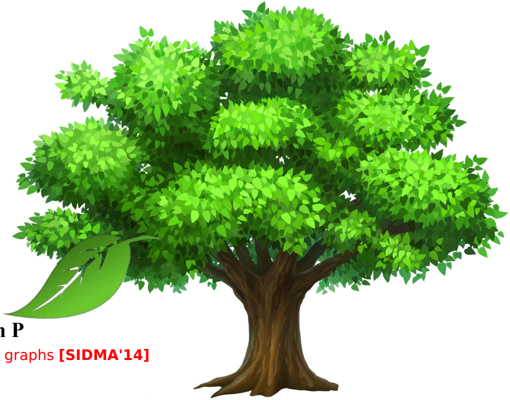
- Problem: $\mathcal{O}(n)$ powers to test

- Solution: **Use a c-factor approx**

$\implies$ obstructions to $\delta(G) \leq 1/2$ have size $\mathcal{O}(c)$

$\implies$ $\mathcal{O}(c)$ modified powers to test

Improved algorithms in some graph classes

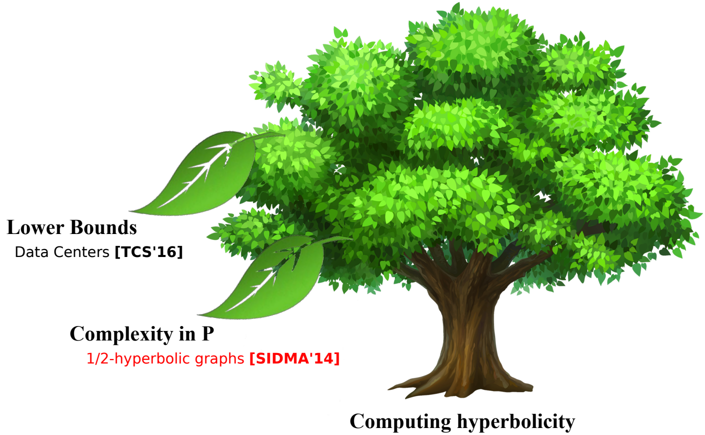


Complexity in P

1/2-hyperbolic graphs **[SIDMA'14]**

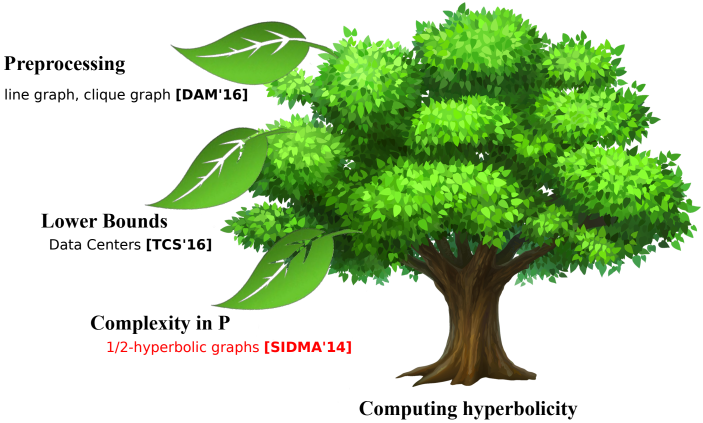
Computing hyperbolicity

Improved algorithms in some graph classes



Lower bounds: new techniques for graph hyperbolicity
$\longrightarrow$ applications to Data Center networks **[Coudert and D. TCS'16]**

Improved algorithms in some graph classes

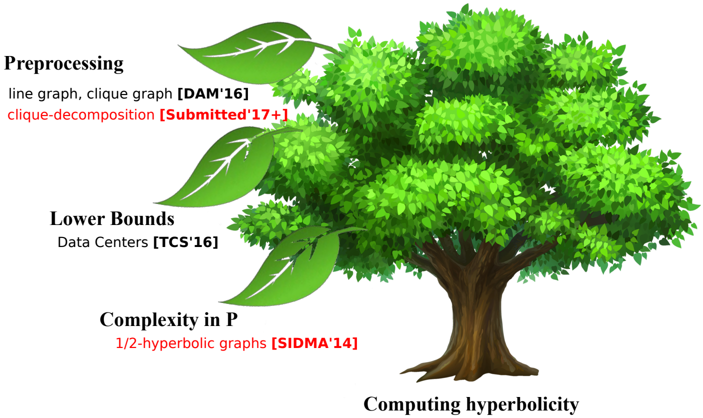


Lower bounds: new techniques for graph hyperbolicity

$\longrightarrow$ applications to Data Center networks **[Coudert and D. TCS'16]**

Preprocessing: preservation of hyp. under graph decompositions

Improved algorithms in some graph classes



Lower bounds: new techniques for graph hyperbolicity

$\longrightarrow$ applications to Data Center networks [Coudert and D. TCS'16]

Preprocessing: preservation of hyp. under graph decompositions

$\longrightarrow$ **clique-decomposition** [Cohen, Coudert, D., Lancin Submitted'17+]

Preservation of hyperbolicity under graph decomposition

Related work

preservation under *modular* and *split* decompositions

edge cutsets inducing complete bipartite subgraphs **[Soto'11]**

Preservation of hyperbolicity under graph decomposition

Related work

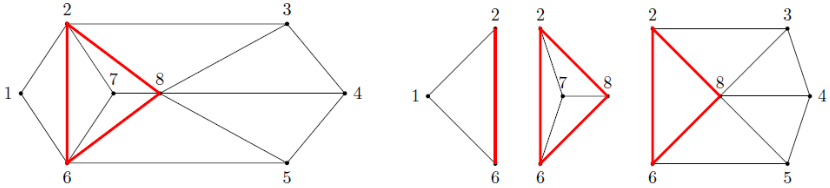
preservation under *modular* and *split* decompositions

edge cutsets inducing complete bipartite subgraphs **[Soto'11]**

Our approach

Clique-decomposition: decomposition of the graph in its **atoms**, *i.e.*, inclusion maximal subgraphs with no clique-separators.

(in $\mathcal{O}(nm)$-time **[Tarjan'85]**)

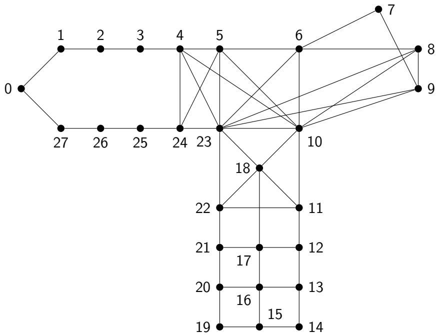


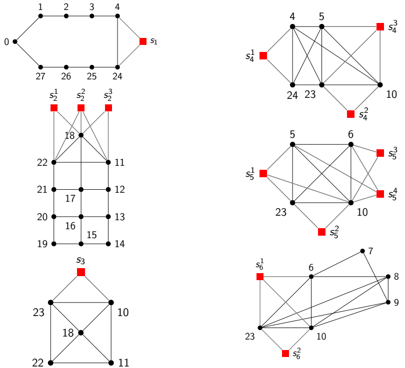
(a) A graph G. (b) The clique-decomposition of G.

Clique-decomposition and hyperbolicity

Theorem [Cohen, Coudert, D., Lancin Submitted'17+]

Let $G = (V, E)$ and let δ^* be the maximum hyperbolicity over the atoms of G. Then, $\delta^* \leq \delta(G) \leq \delta^* + 1$ and the bounds are sharp.





Clique-decomposition and hyperbolicity

Improvements

- **Exact computation** by modifying the atoms (in $\mathcal{O}(nm)$-time)
- **Linear-time algorithm** for computing $\delta(G)$ in outerplanar graphs
- **Finer-grained complexity analysis of clique-decomposition**

[Coudert and D. Submitted'17+]

Two ingredients

- distortion of hyperbolicity under disconnection by **bounded-diameter** separators
- **atoms represent the bags of a tree decomposition**

Tree decompositions

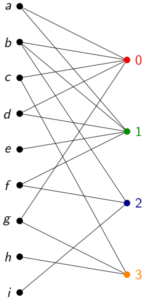
Tree decompositions

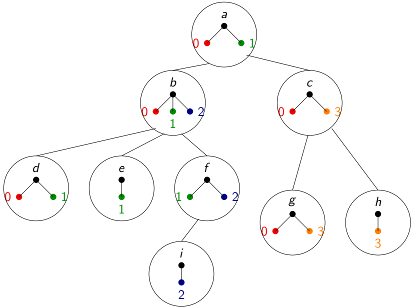
Representation of a graph as a tree preserving connectivity properties.

nodes of the tree $\sim$ subgraphs of G (*bags*)

the decomposition spans all the vertices and all the edges

edges of the tree $\sim$ separators of G



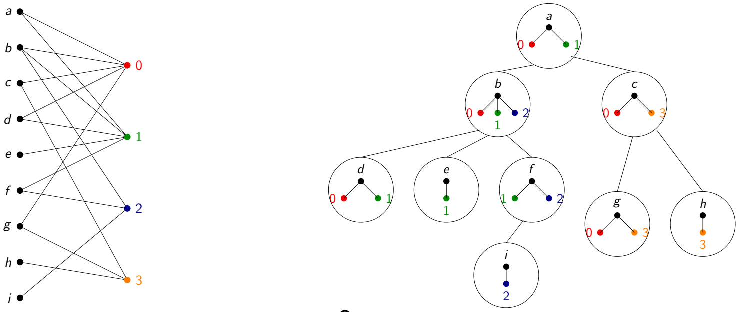


Optimizing the properties of tree decompositions

- minimizing the **size** of bags

  width = max size of bags -1

  treewidth = min width of tree decompositions



$tw = 3$

Optimizing the properties of tree decompositions

- minimizing the **size** of bags

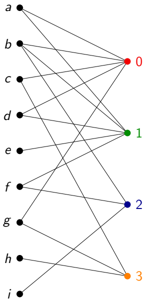
  width = max size of bags −1

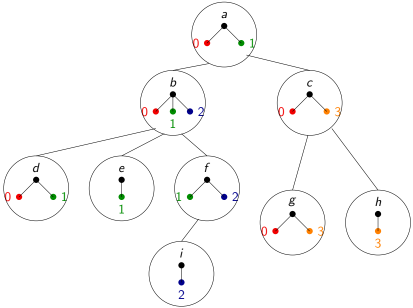
  treewidth = min width of tree decompositions

- minimizing the **diameter** of bags in the graph

  length = max diameter of bags

  treelength = min length of tree decompositions



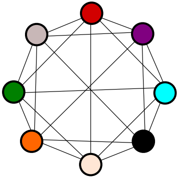


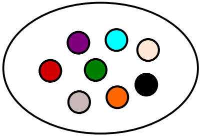
$tl = 2$

Treelength vs. Treewidth: Uncomparability

- **treewidth $\gg$ treelength.**

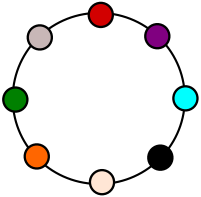
  Complete graph K_n: treewidth $n-1$, treelength 1.

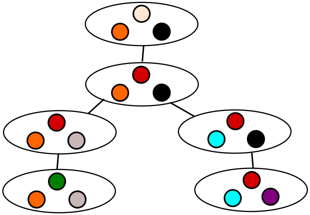




- **treewidth $\ll$ treelength.**

  Cycle C_n: treewidth 2, treelength $\lceil \frac{n}{3} \rceil$.

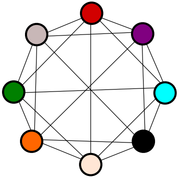


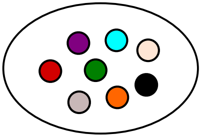


Treelength vs. Treewidth: Uncomparability

- **treewidth $\gg$ treelength.**

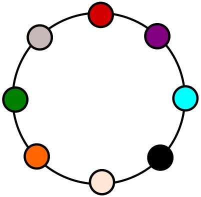
  Complete graph K_n: treewidth $n-1$, treelength 1.

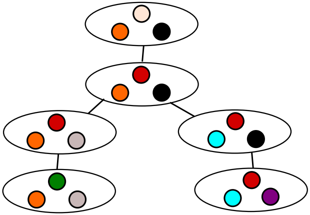




- **treewidth $\ll$ treelength.**

  Cycle C_n: treewidth 2, treelength $\lceil \frac{n}{3} \rceil$.





Relationship with hyperbolicity: $\delta \leq tl \leq 2\delta \cdot \log n + 1$

Treelength vs. Treewidth: Complexity

- $tw \leq k$?
  - exact: in $k^{\mathcal{O}(k^3)} \cdot n$-time **[Bodlaender'96]**
  - 5-approximation: in $2^{\mathcal{O}(k)} \cdot n$-time **[Bodlaender et al.'13]**
  - $\sqrt{tw}$-approximation: in $n^{\mathcal{O}(1)}$-time **[Feige, Hajiaghayi, Lee'08]**
- $tl \leq k$?
  - NP-complete for every $k \geq 2$ **[Lokshtanov'10]**
  - 3-approximation: in $\mathcal{O}(n+m)$-time **[Dourisboure and Gavoille'07]**

Treelength "easier" to approximate than treewidth

Our result

Related work

$tw(G) < 12 \cdot tl(G)$ if G is planar **[Dieng and Gavoille'09]**

$tl(G) \leq \lfloor k/2 \rfloor$ if G is k-chordal **[Dourisboure and Gavoille'07]**

Theorem [Coudert, D., Nisse SIDMA'16]

For every **apex-minor free** graph G with bounded **shortest maximal cycle basis** we have that $tl(G) = \Theta(tw(G))$.

Improves on **[Diestel and Müller'14]**

Our result

Related work

$tw(G) < 12 \cdot tl(G)$ if G is planar **[Dieng and Gavoille'09]**

$tl(G) \leq \lfloor k/2 \rfloor$ if G is k-chordal **[Dourisboure and Gavoille'07]**

Theorem [Coudert, D., Nisse SIDMA'16]

For every **apex-minor free** graph G with bounded **shortest maximal cycle basis** we have that $tl(G) = \Theta(tw(G))$.

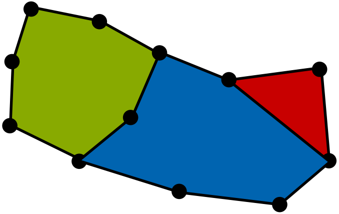
More precisely:

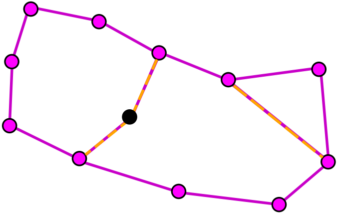
- $tw(G) \leq 72\sqrt{2}(g+1)^{3/2} \cdot tl(G) + \mathcal{O}(g^2)$ *if G has* **genus** *at most g*
- $tl(G) \leq \lfloor \ell/2 \rfloor \cdot (tw(G) - 1)$ *if G has shortest maximal cycle basis ℓ*

Improves on **[Diestel and Müller'14]**

Shortest maximal cycle basis

Cycle space: Eulerian subgraphs + *symmetric difference* on the edges





Cycle basis: Basis of the cycle space composed of cycles

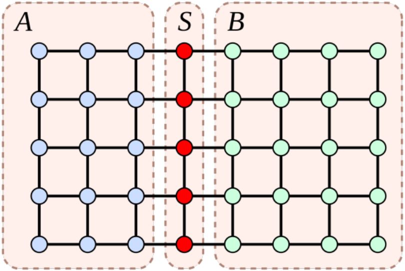
G has shortest maximal cycle basis $\leq \ell \iff$ the cycles of length at most ℓ in G generate the cycle space

generalizes chordality + longest isometric cycle

Diameter of minimal separators

tree decomposition $\sim$ *family of pairwise parallel minimal separators*

[Parra and Scheffler'97]



Theorem [Coudert, D., Nisse SIDMA'16]

Every minimal separator S has **diameter** $\leq \lfloor \ell/2 \rfloor \cdot (|S| - 1)$

$$\forall S,\ diam(S) \leq c \cdot |S| \Longrightarrow tl(G) \leq c \cdot tw(G)$$

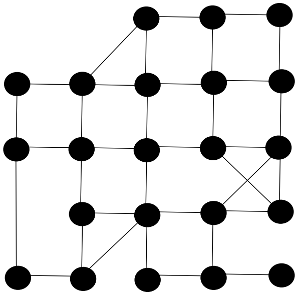
Diameter of minimal separators

$\mathcal{G}_\ell$ class of graphs with shortest maximal cycle basis $\leq \ell$

Choose $G \in \mathcal{G}_\ell$ a **minimum counter-example**

$\exists$ S min sep of G s.t.:

- S is a **stable set** of size $|S| \geq 2$
- all the vertices in S are pairwise at distance $> \lfloor \ell/2 \rfloor$.



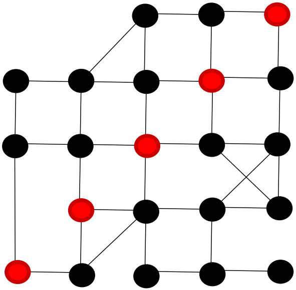
Diameter of minimal separators

$\mathcal{G}_\ell$ class of graphs with shortest maximal cycle basis $\leq \ell$

Choose $G \in \mathcal{G}_\ell$ a **minimum counter-example**

$\exists$ S min sep of G s.t.:

- S is a **stable set** of size $|S| \geq 2$
- all the vertices in S are pairwise at distance $> \lfloor \ell/2 \rfloor$.



Pick a minimal separator S

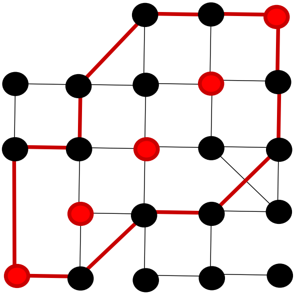
Diameter of minimal separators

$\mathcal{G}_\ell$ class of graphs with shortest maximal cycle basis $\leq \ell$

Choose $G \in \mathcal{G}_\ell$ a **minimum counter-example**

$\exists$ S min sep of G s.t.:

- S is a **stable set** of size $|S| \geq 2$
- all the vertices in S are pairwise at distance $> \lfloor \ell/2 \rfloor$.



Pick a minimal separator S

Connect two components of $G[S]$

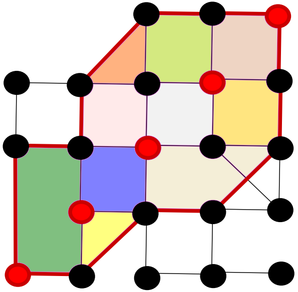
Diameter of minimal separators

$\mathcal{G}_\ell$ class of graphs with shortest maximal cycle basis $\leq \ell$

Choose $G \in \mathcal{G}_\ell$ a **minimum counter-example**

$\exists$ S min sep of G s.t.:

- S is a **stable set** of size $|S| \geq 2$
- all the vertices in S are pairwise at distance $> \lfloor \ell/2 \rfloor$.



Pick a minimal separator S

Connect two components of $G[S]$

Symmetric difference of cycles of length $\leq \ell$

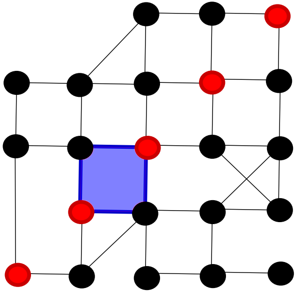
Diameter of minimal separators

$\mathcal{G}_\ell$ class of graphs with shortest maximal cycle basis $\leq \ell$

Choose $G \in \mathcal{G}_\ell$ a **minimum counter-example**

$\exists$ S min sep of G s.t.:

- S is a **stable set** of size $|S| \geq 2$
- all the vertices in S are pairwise at distance $> \lfloor \ell/2 \rfloor$.



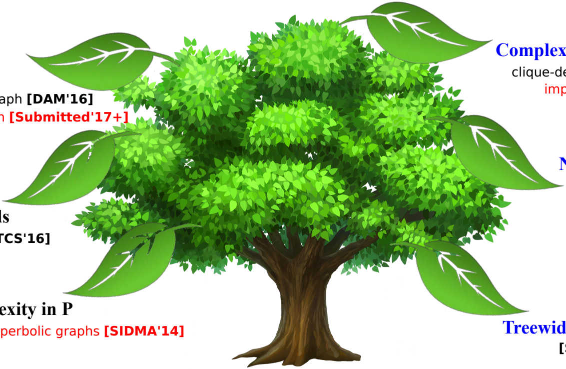
Pick a minimal separator S

Connect two components of $G[S]$

Symmetric difference of cycles of length $\leq \ell$

Two components of $G[S]$ at distance $\leq \lfloor \ell/2 \rfloor$

Conclusion for this part



Preprocessing

line graph, clique graph **[DAM'16]**

clique-decomposition **[Submitted'17+]**

Lower Bounds

Data Centers **[TCS'16]**

Complexity in P

1/2-hyperbolic graphs **[SIDMA'14]**

Complexity in P

clique-decomposition

improved algorithms **[Submitted'17+]**

NP-hardness

treebreadth

pathbreadth **[IWOCA'16]**

pathlength

Treewidth vs. Treelength

[SIDMA'16]

Computing hyperbolicity

Computing tree decompositions

Conclusion for this part

- *Finer-grained complexity of polynomial problems*
  (hyperbolicity, clique-decomposition)
- *Relationship between treewidth and treelength*

Open problems

- Computing tree decompositions of width $\mathcal{O}(tl(G))$
- Recognizing graphs with **large** hyperbolicity
- Extension of the concepts to **directed graphs**

Privacy at large scale in social graphs

(http://www.computerweekly.com/)

Modeling online communities

Information-sharing in social networks [Kleinberg and Ligett'13]

Every user is in **one** community

Communities = Partition of the users

Goals for a user:

- Avoid **conflicts** with users
- **Maximize size** of her community

Game on a **conflict graph**

users ⟷ nodes

conflicts ⟷ edges

Extension to **edge-weighted** graphs (not presented)

Coloring games in graphs

input: graph $G = (V, E)$.

$$\begin{aligned} \text{vertices in } V &\longleftrightarrow \text{agents of the game} \\ \text{(proper) vertex-colorings of } G &\longleftrightarrow \text{configurations of the game} \\ \text{color of a vertex} &\longleftrightarrow \text{strategy of an agent} \\ \text{utility function} &\longleftrightarrow \#\text{vertices in her color class} \end{aligned}$$

Coloring games in graphs

input: graph $G = (V, E)$.

vertices in V $\longleftrightarrow$ agents of the game
(proper) vertex-colorings of G $\longleftrightarrow$ configurations of the game
color of a vertex $\longleftrightarrow$ strategy of an agent
utility function $\longleftrightarrow$ #vertices in her color class

Coloring games in graphs

input: graph $G = (V, E)$.

vertices in V	$\longleftrightarrow$	agents of the game
(proper) vertex-colorings of G	$\longleftrightarrow$	configurations of the game
color of a vertex	$\longleftrightarrow$	strategy of an agent
utility function	$\longleftrightarrow$	#vertices in her color class

Coloring games in graphs

input: graph $G = (V, E)$.

vertices in V	$\longleftrightarrow$	agents of the game
(proper) vertex-colorings of G	$\longleftrightarrow$	configurations of the game
color of a vertex	$\longleftrightarrow$	strategy of an agent
utility function	$\longleftrightarrow$	#vertices in her color class

Coloring games in graphs

input: graph $G = (V, E)$.

vertices in V	$\longleftrightarrow$	agents of the game
(proper) vertex-colorings of G	$\longleftrightarrow$	configurations of the game
color of a vertex	$\longleftrightarrow$	strategy of an agent
utility function	$\longleftrightarrow$	#vertices in her color class

Better-response: change color **one by one** (if beneficial)

Coloring games in graphs

<u>input</u>: graph $G = (V, E)$.

vertices in V	$\longleftrightarrow$	agents of the game
(proper) vertex-colorings of G	$\longleftrightarrow$	configurations of the game
color of a vertex	$\longleftrightarrow$	strategy of an agent
utility function	$\longleftrightarrow$	#vertices in her color class

Better-response: change color **one by one** (if beneficial)

Coloring games in graphs

input: graph $G = (V, E)$.

vertices in V	$\longleftrightarrow$	agents of the game
(proper) vertex-colorings of G	$\longleftrightarrow$	configurations of the game
color of a vertex	$\longleftrightarrow$	strategy of an agent
utility function	$\longleftrightarrow$	#vertices in her color class

What about *coalitions*?

Better-response: change color **one by one** (if beneficial)

Coloring games in graphs

input: graph $G = (V, E)$.

$$\begin{aligned} \text{vertices in } V &\longleftrightarrow \text{agents of the game} \\ \text{(proper) vertex-colorings of } G &\longleftrightarrow \text{configurations of the game} \\ \text{color of a vertex} &\longleftrightarrow \text{strategy of an agent} \\ \text{utility function} &\longleftrightarrow \#\text{vertices in her color class} \end{aligned}$$

What about *coalitions*?

Better-response: change color **k by k** (if beneficial)

Coloring games in graphs

input: graph $G = (V, E)$.

vertices in V	$\longleftrightarrow$	agents of the game
(proper) vertex-colorings of G	$\longleftrightarrow$	configurations of the game
color of a vertex	$\longleftrightarrow$	strategy of an agent
utility function	$\longleftrightarrow$	#vertices in her color class

What about *coalitions*?

Better-response: change color **k by k** (if beneficial)

Local process and individual optimization

k-deviations

Any subset of $\leq k$ agents joining the same color class – or creating a new one – so that all the agents in the subset increase their utility.

Local process and individual optimization

k-deviations

Any subset of $\leq k$ agents joining the same color class – or creating a new one – so that all the agents in the subset increase their utility.

Equilibria

- The coloring is k-stable iff, there is no k-deviation.
  - A k-stable coloring is a ***k*-strong Nash equilibrium**
  - A 1-stable coloring is a **Nash equilibrium**
- A graph is called k-stable when there exists a k-stable coloring.

Local process and individual optimization

k-deviations

Any subset of $\leq k$ agents joining the same color class – or creating a new one – so that all the agents in the subset increase their utility.

Equilibria

- The coloring is k-stable iff, there is no k-deviation.
  - A k-stable coloring is a ***k*-strong Nash equilibrium**
  - A 1-stable coloring is a **Nash equilibrium**
- A graph is called k-stable when there exists a k-stable coloring.

Existence? Time of convergence?

State of the art: complexity of coloring games

Theorem [Panagopoulou and Spirakis'08] [Kleinberg and Ligett'13]

For every $G = (V, E)$, the better-response dynamic converges to a Nash equilibrium ($k = 1$) within $\mathcal{O}(|V|^2)$ steps.

Potential game: $\sum$ *utilities*

State of the art: complexity of coloring games

Theorem [Panagopoulou and Spirakis'08] [Kleinberg and Ligett'13]

For every $G = (V, E)$, the better-response dynamic converges to a Nash equilibrium ($k = 1$) within $\mathcal{O}(|V|^2)$ steps.

Potential game: $\sum$ *utilities*

Theorem [Escoffier, Gourvès, Monnot'10] [Kleinberg and Ligett'13]

For every $G = (V, E)$, **for every** $k \leq 3$**,** the better-response dynamic converges to a k**-strong** Nash equilibrium within $\mathcal{O}(|V|^3)$ steps.

Potential game: $\sum$ *(utilities)*^{2}

State of the art: complexity of coloring games

Theorem [Panagopoulou and Spirakis'08] [Kleinberg and Ligett'13]

For every $G = (V, E)$, the better-response dynamic converges to a Nash equilibrium ($k = 1$) within $\mathcal{O}(|V|^2)$ steps.

Potential game: $\sum$ utilities

Theorem [Escoffier, Gourvès, Monnot'10] [Kleinberg and Ligett'13]

For every $G = (V, E)$, **for every** $k \leq 3$**,** the better-response dynamic converges to a k-**strong** Nash equilibrium within $\mathcal{O}(|V|^3)$ steps.

Potential game: $\sum$ (utilities)^2

Conjecture [Escoffier, Gourvès, Monnot'10]

For every $G = (V, E)$, for every $k \geq 1$, the better-response dynamic converges to a k-strong Nash equilibrium within $\mathcal{O}(|V|^2)$ steps.

No **polynomial** *potential* **[Kleinberg and Ligett'13]**

Our contributions: Better-response dynamics (1/2)

Theorem [D., Mazauric, Chaintreau SUGC'13]

For every $G = (V, E)$, for every $k \geq 1$, the better-response dynamic converges to a k-strong Nash equilibrium within $\exp[\mathcal{O}(\sqrt{n})]$ steps.

Exponential *potential*

Theorem [D., Mazauric, Chaintreau SUGC'13]

For every $G = (V, E)$ with $|V| = \binom{m}{2} + r$ nodes, **for every $k \leq 2$**, the better-response dynamic converges to a k-strong Nash equilibrium within at most $2\binom{m+1}{3} + mr = \Theta(|V|^{3/2})$ steps and **this is sharp**.

Worst-case: $E = \emptyset$

Reinterpret colorings as **integer partitions**

Our contributions: Better-response dynamics (2/2)

Conjecture [Escoffier, Gourvès, Monnot'10]

For every $G = (V, E)$, for every $k \geq 1$, the better-response dynamic converges to a k-strong Nash equilibrium within $\mathcal{O}(|V|^2)$ steps.

Theorem [D., Mazauric, Chaintreau SUGC'13]

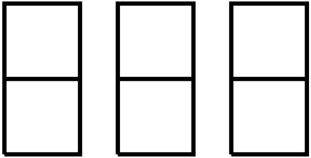
There are graphs $G = (V, E)$ such that for every $k \geq 4$, the better-response dynamic converges to a k-strong Nash equilibrium within **superpolynomial** $\Omega(|V|^{\Theta(\log |V|)})$ steps in the worst case.

Based on ***cascading*** *sequences of 4-deviations*

Superpolynomial cascading sequences for $k \geq 4$

no edges $\Longrightarrow$ longest chain in a DAG

square $\longleftrightarrow$ **node**
heap $\longleftrightarrow$ **color class**

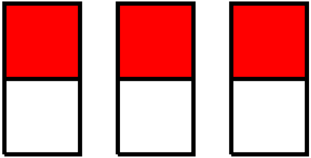


as k grows, new types of deviations can occur

Superpolynomial cascading sequences for $k \geq 4$

no edges $\Longrightarrow$ longest chain in a DAG

square $\longleftrightarrow$ **node**
heap $\longleftrightarrow$ **color class**



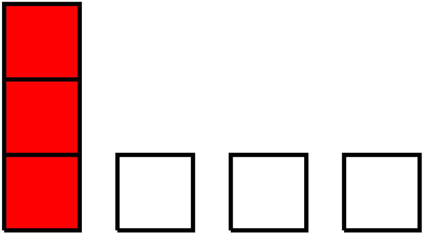
as k grows, new types of deviations can occur

Superpolynomial cascading sequences for $k \geq 4$

<u>no edges</u> $\Longrightarrow$ longest chain in a DAG

square $\longleftrightarrow$ **node**

heap $\longleftrightarrow$ **color class**



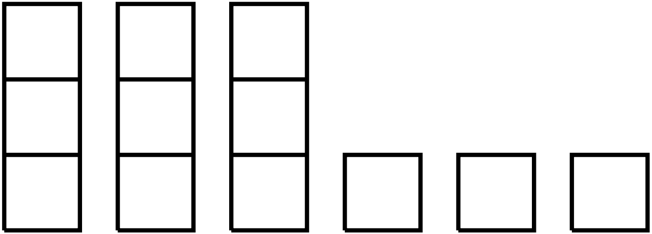
as k grows, new types of deviations can occur

Superpolynomial cascading sequences for $k \geq 4$

no edges $\implies$ longest chain in a DAG

square $\longleftrightarrow$ **node**

heap $\longleftrightarrow$ **color class**



as k grows, new types of deviations can occur

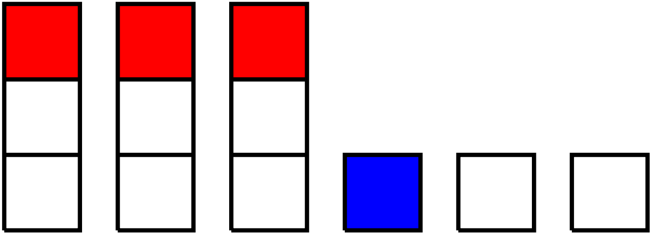
recursive construction of sequences

Superpolynomial cascading sequences for $k \geq 4$

no edges $\Longrightarrow$ longest chain in a DAG

square $\longleftrightarrow$ **node**

heap $\longleftrightarrow$ **color class**



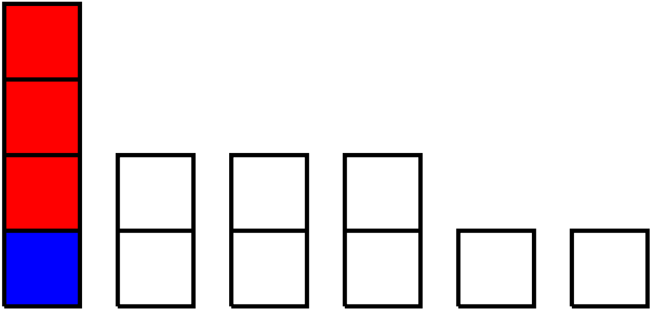
as k grows, new types of deviations can occur

recursive construction of sequences

Superpolynomial cascading sequences for $k \geq 4$

no edges $\implies$ longest chain in a DAG

square $\longleftrightarrow$ **node**
heap $\longleftrightarrow$ **color class**

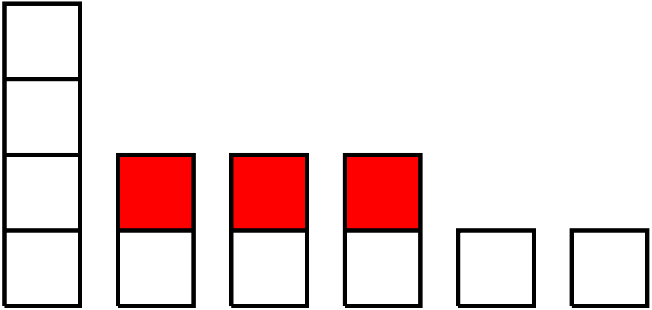


as k grows, new types of deviations can occur
recursive construction of sequences

Superpolynomial cascading sequences for $k \geq 4$

no edges $\implies$ longest chain in a DAG

square $\longleftrightarrow$ **node**
heap $\longleftrightarrow$ **color class**



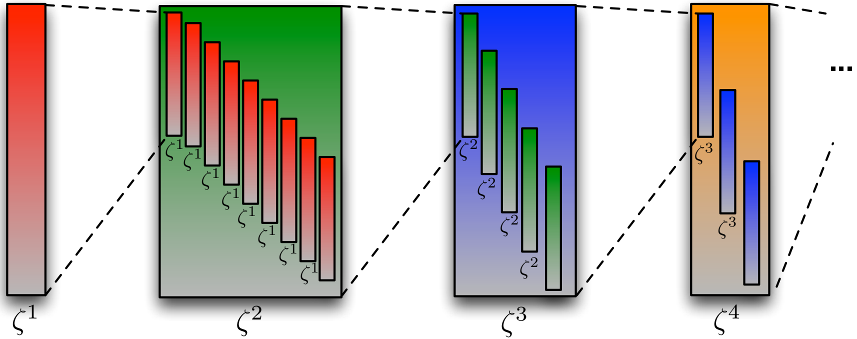
as k grows, new types of deviations can occur
recursive construction of sequences

Superpolynomial cascading sequences for $k \geq 4$

no edges $\implies$ longest chain in a DAG

square $\longleftrightarrow$ **node**

heap $\longleftrightarrow$ **color class**



as k grows, new types of deviations can occur

recursive construction of sequences

Our contributions: Parallel complexity

Need for better understanding of the complexity of coloring games

- *Parallel complexity classes*

  NC^i: $\mathcal{O}(\log^i n)$-time with $poly(n)$ processors **[Bloch'97][Cook'83]**

Theorem [D. SAGT'16]

Computing a Nash equilibrium for coloring games is P-hard under NC^1-reductions.

Our contributions: Parallel complexity

Need for better understanding of the complexity of coloring games

- *Parallel complexity classes*

NC^i: $\mathcal{O}(\log^i n)$-time with $poly(n)$ processors **[Bloch'97][Cook'83]**

Theorem [D. SAGT'16]

Computing a Nash equilibrium for coloring games is P-hard under NC^1-reductions.

Consequences:

- the problem is **inherently sequential**
- it cannot be solved in polytime and **polylogarithmic workspace**
- **Distributed algorithms**: processors = vertices + edges

  $\longrightarrow$ no protocol with polylogarithmic **communication complexity** and **local computation time**.

Conclusion for this part

Coloring games:

- Complexity of better-response dynamics

Exact convergence time for $k \leq 2$

Superpolynomial lower-bound for $k \geq 4$

- Parallel complexity

Coloring games are inherently sequential

Open problems:

- Parallel complexity of graphical games
- Complexity of computing 4-**stable colorings**

Conclusion

Summary of the thesis

Analysis of large-scale networks: **Metric treelikeness**

- **Complexity in P**
  (*conditional lower-bounds*)

- Graph decompositions
  (*line graph*, **tree decompositions**, *clique-decomposition*)

- Algebraic methods
  (*cycle basis*, *graph endomorphisms*)

tools from algorithmic graph theory

Summary of the thesis

Dynamics of information flows: **Privacy** and **Web's transparency**

- Potential games
- Combinatorics on *integer partitions*
  (*longest sequences in better-response dynamics*)
- **Parallel complexity**
- PAC-learning
  (*Ad Targeting Identification*)

tools from algorithmic game theory and learning theory

Perspectives

- Relationships between treelength and *graph minor decompositions*

  FPT algorithms?

  Constructive relationship between treewidth and treelength?

- Random models for **directed** social networks (Twitter, . . .)

  Metric treelikeness in directed graphs?

- Finer-grained complexity of graphical games

  Parallel complexity of ***unweighted*** *games and implications for* ***weighted*** *games.*

Any questions?