Canonical Decompositions in Monadically Stable and Bounded Shrubdepth Graph Classes

Pierre Ohlmann, Michał Pilipczuk, Wojciech Przybyszewski, Szymon Toruńczyk

ICALP 2023

Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

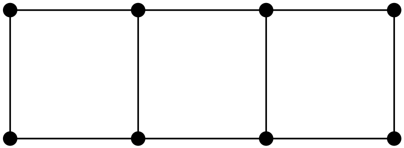
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:



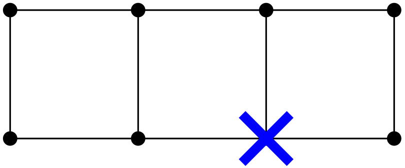
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:

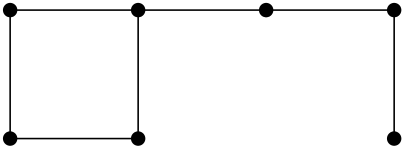


Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.
Example play of the radius-2 Splitter game:



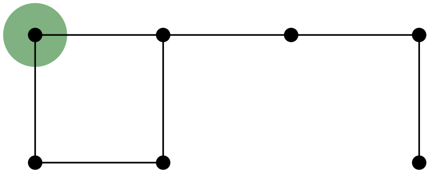
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:



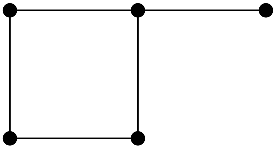
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:



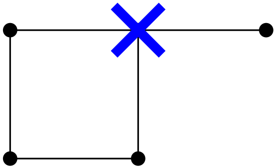
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:

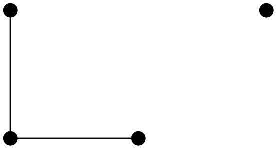


Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.
Example play of the radius-2 Splitter game:



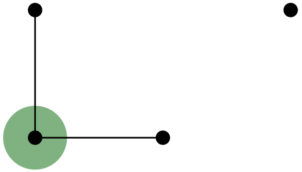
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:

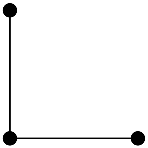


Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.
Example play of the radius-2 Splitter game:



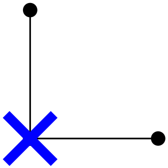
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:



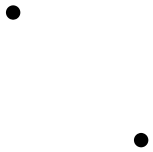
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:



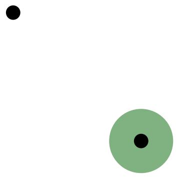
Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.

Example play of the radius-2 Splitter game:



Splitter Game

The radius-r Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i - v$.

Splitter wins once G_i has size 1.
Example play of the radius-2 Splitter game:

Observation: We don't need to assume that G is a finite graph for this definition to make sense.

Progressing moves in the Splitter game

Theorem. [Ohlmann, Pilipczuk, P., Toruńczyk, 2023]

For every nowhere dense class of graphs $\mathscr{C}$ and every $r \in \mathbb{N}$ there exists a constant c such that for every $G \in \mathscr{C}$ Splitter has at most c progressing moves in the radius-r Splitter game on G.

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Proof.

Assume statement doesn't hold. Denote $V(G) = \{v_i : i \in I\}$.

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Proof.

Assume statement doesn't hold. Denote $V(G) = \{v_i : i \in I\}$.

Consider the following theory over the signature that consists of constant symbols $\{v_i : i \in I\} \cup \{v_\infty\}$ and one binary relation E:

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Proof.

Assume statement doesn't hold. Denote $V(G) = \{v_i : i \in I\}$.

Consider the following theory over the signature that consists of constant symbols $\{v_i : i \in I\} \cup \{v_\infty\}$ and one binary relation E:

- $v_i \neq v_j$ for every $i, j \in I$;

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Proof.

Assume statement doesn't hold. Denote $V(G) = \{v_i : i \in I\}$.

Consider the following theory over the signature that consists of constant symbols $\{v_i : i \in I\} \cup \{v_\infty\}$ and one binary relation E:

- $v_i \neq v_j$ for every $i, j \in I$;
- $E(v_i, v_j)$ for every $(v_i, v_j) \in E(G)$ and $\neg E(v_i, v_j)$ for every $(v_i, v_j) \notin E(G)$;

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Proof.

Assume statement doesn't hold. Denote $V(G) = \{v_i : i \in I\}$.

Consider the following theory over the signature that consists of constant symbols $\{v_i : i \in I\} \cup \{v_\infty\}$ and one binary relation E:

- $v_i \neq v_j$ for every $i, j \in I$;
- $E(v_i, v_j)$ for every $(v_i, v_j) \in E(G)$ and $\neg E(v_i, v_j)$ for every $(v_i, v_j) \notin E(G)$;
- Splitter wins the radius-r game in d rounds if he plays optimally;

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Proof.

Assume statement doesn't hold. Denote $V(G) = \{v_i : i \in I\}$.

Consider the following theory over the signature that consists of constant symbols $\{v_i : i \in I\} \cup \{v_\infty\}$ and one binary relation E:

- $v_i \neq v_j$ for every $i, j \in I$;
- $E(v_i, v_j)$ for every $(v_i, v_j) \in E(G)$ and $\neg E(v_i, v_j)$ for every $(v_i, v_j) \notin E(G)$;
- Splitter wins the radius-r game in d rounds if he plays optimally;
- $v_\infty \neq v_i$ for every $i \in I$;

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.

Proof.

Assume statement doesn't hold. Denote $V(G) = \{v_i : i \in I\}$.

Consider the following theory over the signature that consists of constant symbols $\{v_i : i \in I\} \cup \{v_\infty\}$ and one binary relation E:

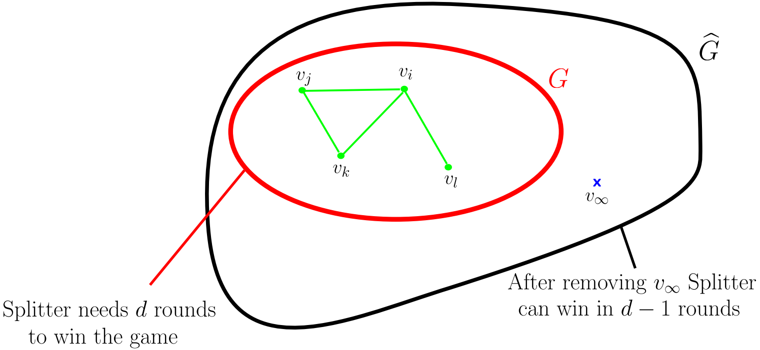
- $v_i \neq v_j$ for every $i, j \in I$;
- $E(v_i, v_j)$ for every $(v_i, v_j) \in E(G)$ and $\neg E(v_i, v_j)$ for every $(v_i, v_j) \notin E(G)$;
- Splitter wins the radius-r game in d rounds if he plays optimally;
- $v_\infty \neq v_i$ for every $i \in I$;
- v_∞ is a progressing move.

□

Progressing moves in infinite graphs

Lemma.

For every infinite graph G and $r \in \mathbb{N}$ if Splitter can win the radius-r Splitter game on G in d rounds, then he has finitely many progressing moves.



Progressing moves in infinite graphs

Theorem. [Ohlmann, Pilipczuk, P., Toruńczyk, 2023]

For every nowhere dense class of graphs $\mathscr{C}$ and every $r \in \mathbb{N}$ there exists a constant c such that for every $G \in \mathscr{C}$ Splitter has at most c progressing moves in the radius-r Splitter game on G.

Progressing moves in infinite graphs

Theorem. [Ohlmann, Pilipczuk, P., Toruńczyk, 2023]

For every nowhere dense class of graphs $\mathscr{C}$ and every $r \in \mathbb{N}$ there exists a constant c such that for every $G \in \mathscr{C}$ Splitter has at most c progressing moves in the radius-r Splitter game on G.

Proof.

Assume the statement is not true. Consider the following theory:

- Splitter can win the radius-r game in at most d rounds;

Progressing moves in infinite graphs

Theorem. [Ohlmann, Pilipczuk, P., Toruńczyk, 2023]

For every nowhere dense class of graphs $\mathscr{C}$ and every $r \in \mathbb{N}$ there exists a constant c such that for every $G \in \mathscr{C}$ Splitter has at most c progressing moves in the radius-r Splitter game on G.

Proof.

Assume the statement is not true. Consider the following theory:

- Splitter can win the radius-r game in at most d rounds;
- there are at least m progressing moves for every $m \in \mathbb{N}$.

Progressing moves in infinite graphs

Theorem. [Ohlmann, Pilipczuk, P., Toruńczyk, 2023]

For every nowhere dense class of graphs $\mathscr{C}$ and every $r \in \mathbb{N}$ there exists a constant c such that for every $G \in \mathscr{C}$ Splitter has at most c progressing moves in the radius-r Splitter game on G.

Proof.

Assume the statement is not true. Consider the following theory:

- Splitter can win the radius-r game in at most d rounds;
- there are at least m progressing moves for every $m \in \mathbb{N}$.

Compactness yields a model that contradicts the previous lemma. □

Treedepth

Question: What happens if we take $r = \infty$?

Treedepth

Question: What happens if we take $r = \infty$?

The radius-∞ Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a connected component in $G_i - v$.

Splitter wins once G_i has size 1.

Treedepth

Question: What happens if we take $r = \infty$?

The radius-∞ Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a connected component in $G_i - v$.

Splitter wins once G_i has size 1.

Defintion.

A graph G has *treedepth d* if Splitter wins the radius-∞ game on G in d rounds if he plays optimally.

Treedepth

Question: What happens if we take $r = \infty$?

The radius-∞ Splitter game is played on a graph G_1. In round i

1. Splitter chooses a vertex v to delete
2. Localizer chooses G_{i+1} as a connected component in $G_i - v$.

Splitter wins once G_i has size 1.

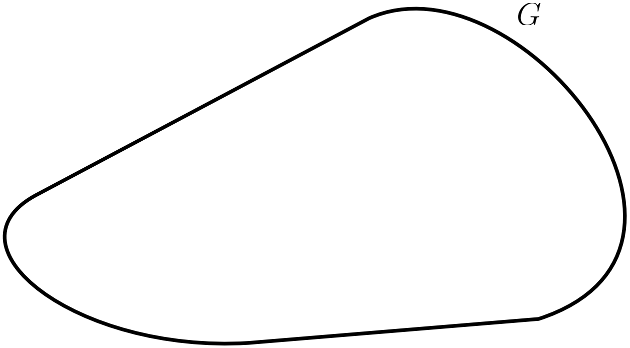
Defintion.

A graph G has *treedepth* d if Splitter wins the radius-∞ game on G in d rounds if he plays optimally.

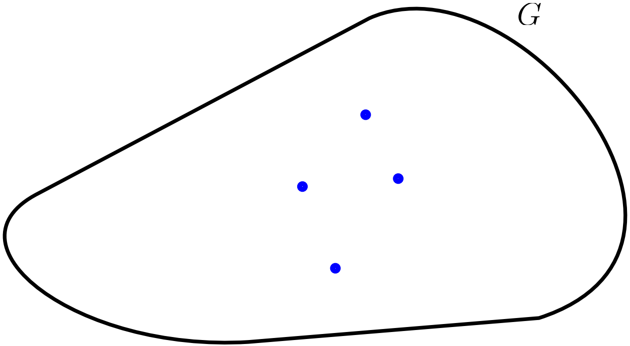
Fact.

There exists a function $f : \mathbb{N} \to \mathbb{N}$ such that for every graph G of treedepth d there exists at most $f(d)$ vertices v such that treedepth of every connected component of $G - \{v\}$ is at most $d - 1$.

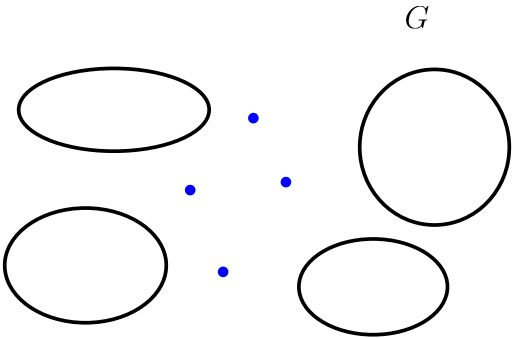
Cannonical decomoposition of graphs of bounded treedepth



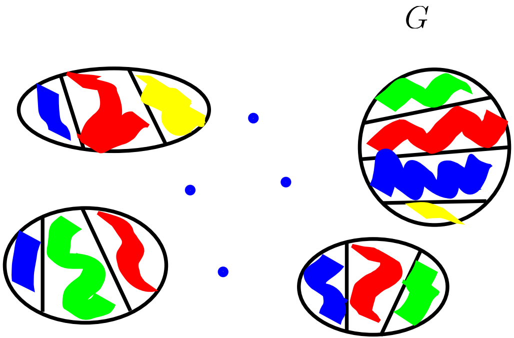
Cannonical decomoposition of graphs of bounded treedepth



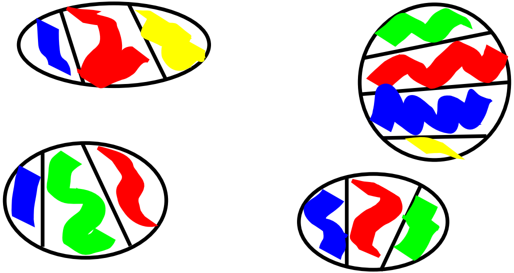
Cannonical decomoposition of graphs of bounded treedepth



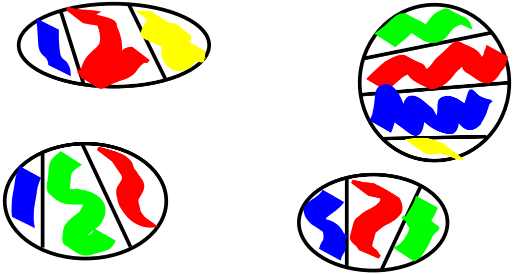
Cannonical decomoposition of graphs of bounded treedepth



Cannonical decomoposition of graphs of bounded treedepth



Cannonical decomoposition of graphs of bounded treedepth



Observation: This yields a decomposition algorithm working in time $f(d) \cdot n^2$ on graphs of treedepth at most d.

Graph isomorphism for bounded treedepth

Theorem. [Bouland, Dawar, Kopczyński, 2012]

Graph isomorphism can be solved on graphs of treedepth at most d in time $f(d) \cdot n^3 \cdot \log n$.

Graph isomorphism for bounded treedepth

Theorem. [Bouland, Dawar, Kopczyński, 2012]

Graph isomorphism can be solved on graphs of treedepth at most d in time $f(d) \cdot n^3 \cdot \log n$.

Remark: The running time can be further improved to $f(d) \cdot n \cdot \log^2 n$.

Beyond sparsity

1. We found canonical moves for Splitter in the Splitter game
2. and applied them for the radius-∞ Splitter game
3. to obtain canonical decompositions and a graph isomorphism algorithm for graphs of bounded treedepth.

Beyond sparsity

1. We want to find canonical moves for Flipper in the Flipper game
2. and apply them for the radius-∞ Flipper game
3. to obtain canonical decompositions and a graph isomorphism algorithm for graphs of bounded shrubdepth.

Beyond sparsity

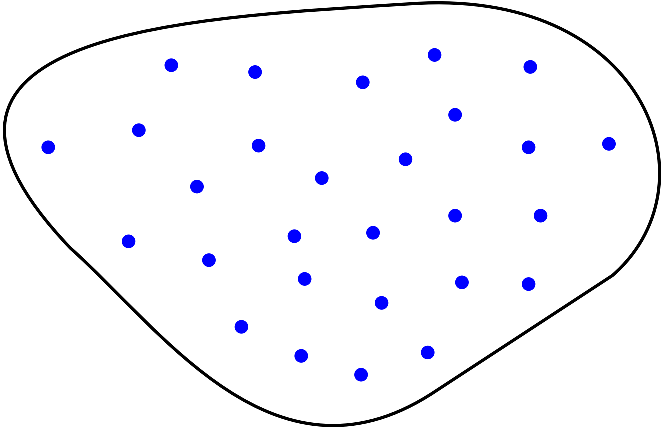
1. We want to find canonical moves for Flipper in the Flipper game
2. and apply them for the radius-∞ Flipper game
3. to obtain canonical decompositions and a graph isomorphism algorithm for graphs of bounded shrubdepth.

Defintion. [Ganian, Hliněný, Nešetřil, Obdržálek, Ossona de Mendez, 2017]

A graph G has *shrubdepth* at most d if Flipper can win the radius-∞ game on G in at most d rounds.

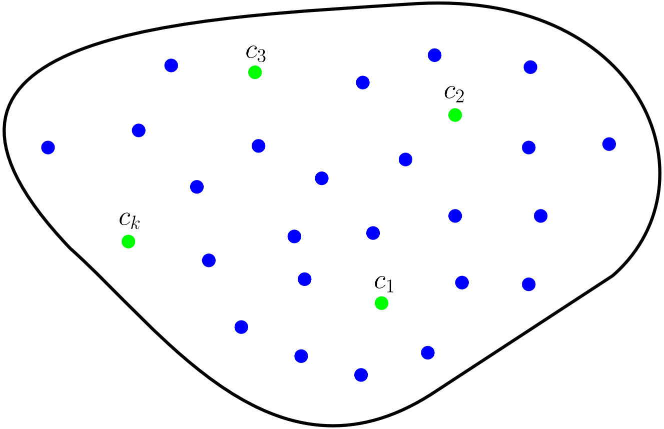
Finitary Substitute Lemma

$G \in \mathcal{C}$ $\phi(x_1, x_2; y_1, \ldots, y_k)$ ψ



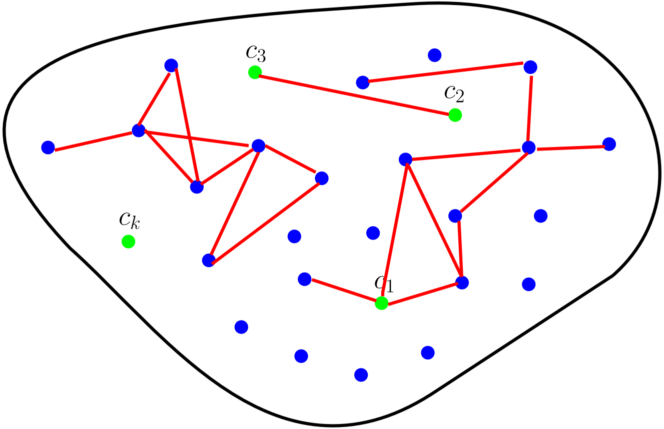
Finitary Substitute Lemma

$G \in \mathcal{C}$ $\phi(x_1, x_2; c_1, \ldots, c_k)$ ψ



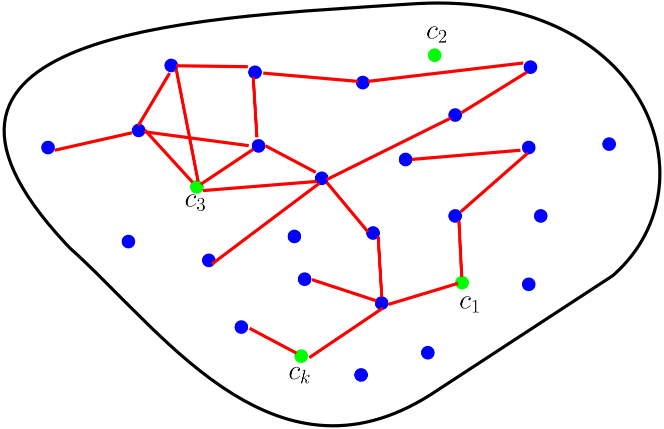
Finitary Substitute Lemma

$G \in \mathcal{C}$ $\qquad$ $\phi(x_1, x_2; c_1, \ldots, c_k)$ $\qquad$ ψ



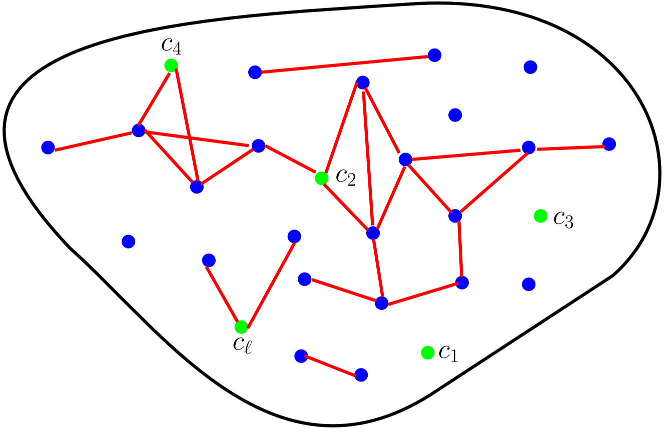
Finitary Substitute Lemma

$G \in \mathcal{C}$ $\phi(x_1, x_2; c_1, \ldots, c_k)$ ψ



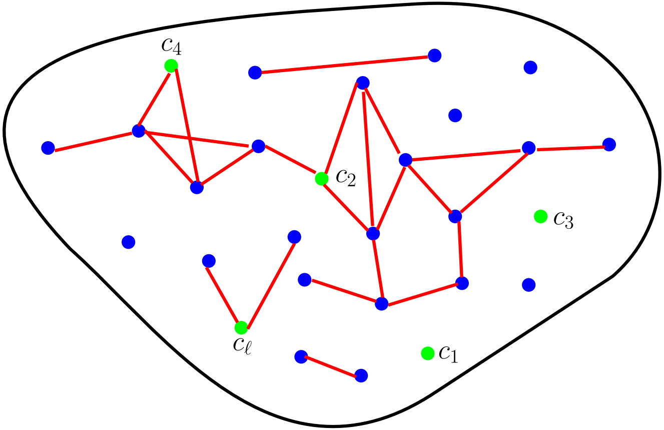
Finitary Substitute Lemma

$G \in \mathcal{C}$ $\phi'(x_1; x_2; y_1, \ldots, y_\ell)$ ψ



Finitary Substitute Lemma

$G \in \mathcal{C}$ $\phi'(x_1; x_2; y_1, \ldots, y_\ell)$ ψ



Proof uses a number of tools from stability theory [Shelah], most importantly properties of forking independence in stable theories.

Main theorem

Theorem. [Ohlmann, Pilipczuk, P., Toruńczyk, 2023]

Graph isomorphism can be solved on graphs of shrubdepth at most d in time $f(d) \cdot n^2$.