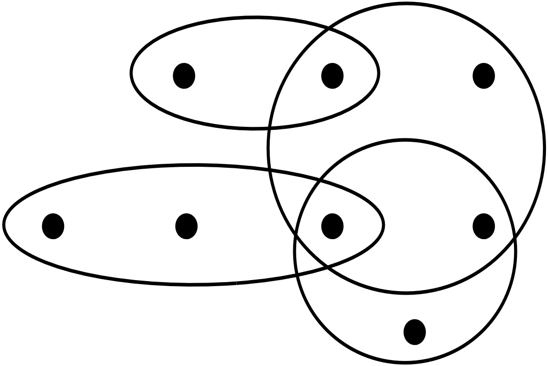
Hypergraph decomposition

Yann Strozecki

Université Paris Sud - Paris 11
Equipe ALGO

Avril 2012, séminaire graphe et logique (LABRI)

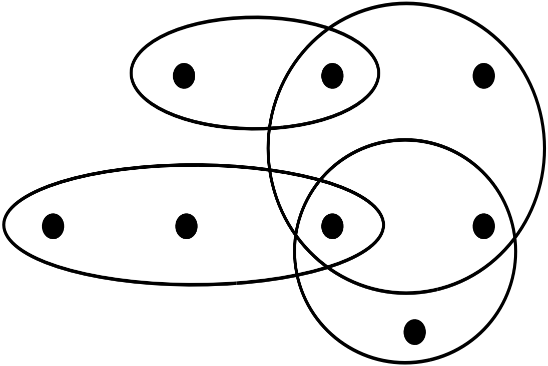
Hypergraphs



Hard problems:

- Hypergraph coloring.

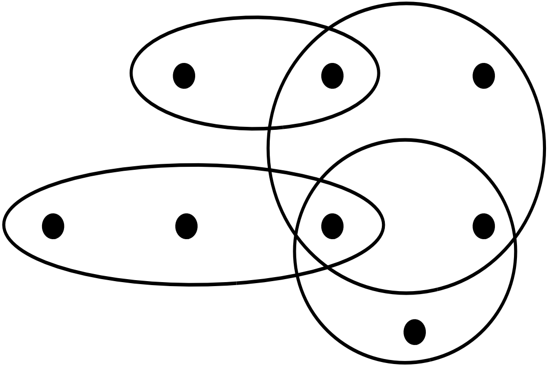
Hypergraphs



Hard problems:

- Hypergraph coloring.
- Minimal edge-covering (3 dimensional matching).

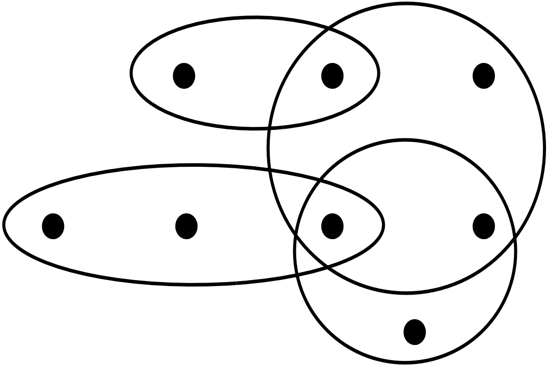
Hypergraphs



Hard problems:

- Hypergraph coloring.
- Minimal edge-covering (3 dimensional matching).
- Finding a spanning tree [Duris, S.].

Hypergraphs



Hard problems:

- Hypergraph coloring.
- Minimal edge-covering (3 dimensional matching).
- Finding a spanning tree [Duris, S.].

Representation of a hypergraph

A hypergraph has up to 2^n edges.
Dense representation often not relevant for complexity issues.

How to represent a hypergraph:

Representation of a hypergraph

A hypergraph has up to 2^n edges.
Dense representation often not relevant for complexity issues.

How to represent a hypergraph:

- Membership oracle, decide whether a set of vertices is an edge.

Representation of a hypergraph

A hypergraph has up to 2^n edges.
Dense representation often not relevant for complexity issues.

How to represent a hypergraph:

- Membership oracle, decide whether a set of vertices is an edge.
- Representable matroids.

Representation of a hypergraph

A hypergraph has up to 2^n edges.
Dense representation often not relevant for complexity issues.

How to represent a hypergraph:

- Membership oracle, decide whether a set of vertices is an edge.
- Representable matroids.
- Sparse hypergraphs: list of edges.

Representation of a hypergraph

A hypergraph has up to 2^n edges.
Dense representation often not relevant for complexity issues.

How to represent a hypergraph:

- Membership oracle, decide whether a set of vertices is an edge.
- Representable matroids.
- Sparse hypergraphs: list of edges.
- Uniform hypergraphs, acyclic hypergraphs.

Representation of a hypergraph

A hypergraph has up to 2^n edges.
Dense representation often not relevant for complexity issues.

How to represent a hypergraph:

- Membership oracle, decide whether a set of vertices is an edge.
- Representable matroids.
- Sparse hypergraphs: list of edges.
- Uniform hypergraphs, acyclic hypergraphs.
- Subclasses: upward closed hypergraphs, matroids, closed uner union, intersection ...

Representation of a hypergraph

A hypergraph has up to 2^n edges.
Dense representation often not relevant for complexity issues.

How to represent a hypergraph:

- Membership oracle, decide whether a set of vertices is an edge.
- Representable matroids.
- Sparse hypergraphs: list of edges.
- Uniform hypergraphs, acyclic hypergraphs.
- Subclasses: upward closed hypergraphs, matroids, closed uner union, intersection ...

Decomposition of a hypergraph

Simple idea, use a graph decomposition : treewidth of a hypergraph.

Alternatively transform the hypergraph into a graph.

Definition

The *Gaifman graph* (or primal graph) $G(H)$ of a hypergraph H is the graph (V, E), where $\{u, v\} \in E$ iff u and v belongs to some hyperedge of H.

Decomposition of a hypergraph

Simple idea, use a graph decomposition : treewidth of a hypergraph.

Alternatively transform the hypergraph into a graph.

Definition

The *Gaifman graph* (or primal graph) $G(H)$ of a hypergraph H is the graph (V, E), where $\{u, v\} \in E$ iff u and v belongs to some hyperedge of H.

Consider the treewidth of the Gaifman graph.

Decomposition of a hypergraph

Simple idea, use a graph decomposition : treewidth of a hypergraph.

Alternatively transform the hypergraph into a graph.

Definition

The *Gaifman graph* (or primal graph) $G(H)$ of a hypergraph H is the graph (V, E), where $\{u, v\} \in E$ iff u and v belongs to some hyperedge of H.

Consider the treewidth of the Gaifman graph.

A hyperedge of size k in the graph becomes a clique of size k in the Gaifman graph: treewidth at least k.

Decomposition of a hypergraph

Simple idea, use a graph decomposition : treewidth of a hypergraph.

Alternatively transform the hypergraph into a graph.

Definition

The *Gaifman graph* (or primal graph) $G(H)$ of a hypergraph H is the graph (V, E), where $\{u, v\} \in E$ iff u and v belongs to some hyperedge of H.

Consider the treewidth of the Gaifman graph.

A hyperedge of size k in the graph becomes a clique of size k in the Gaifman graph: treewidth at least k.

Hypertree width

The tree T is a tree decomposition of width t of the hypergraph H if:

- The nodes of T are labeled by sets of vertices of the hypergraph (or bags).

Hypertree width

The tree T is a tree decomposition of width t of the hypergraph H if:

- The nodes of T are labeled by sets of vertices of the hypergraph (or bags).
- Each set of vertices must be contained in t hyperedges.

Hypertree width

The tree T is a tree decomposition of width t of the hypergraph H if:

- The nodes of T are labeled by sets of vertices of the hypergraph (or bags).
- Each set of vertices must be contained in t hyperedges.
- Let v be in the hyperedges covering a bag, either it is in the bag or it is in no bag under it.

Hypertree width

The tree T is a tree decomposition of width t of the hypergraph H if:

- The nodes of T are labeled by sets of vertices of the hypergraph (or bags).
- Each set of vertices must be contained in t hyperedges.
- Let v be in the hyperedges covering a bag, either it is in the bag or it is in no bag under it.

A conjunctive query of bounded hypertree width can be evaluated in *polynomial time*.

Hypertree width

The tree T is a tree decomposition of width t of the hypergraph H if:

- The nodes of T are labeled by sets of vertices of the hypergraph (or bags).
- Each set of vertices must be contained in t hyperedges.
- Let v be in the hyperedges covering a bag, either it is in the bag or it is in no bag under it.

A conjunctive query of bounded hypertree width can be evaluated in *polynomial time*.

Generalizations: generalized hypertree width, fractional hypertree width, submodular width . . . .

Hypertree width

The tree T is a tree decomposition of width t of the hypergraph H if:

- The nodes of T are labeled by sets of vertices of the hypergraph (or bags).
- Each set of vertices must be contained in t hyperedges.
- Let v be in the hyperedges covering a bag, either it is in the bag or it is in no bag under it.

A conjunctive query of bounded hypertree width can be evaluated in *polynomial time*.

Generalizations: generalized hypertree width, fractional hypertree width, submodular width . . . .

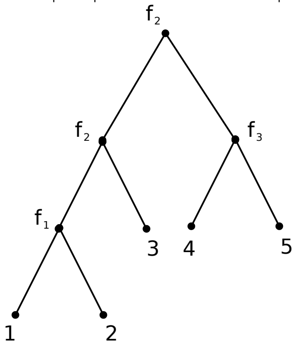
An example of a decomposition

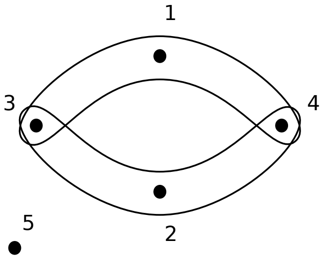
f_1	0	1
0	1	0
1	0	1

f_2	0	1
0	0	1
1	0	0

f_3	0	1
0	0	0
1	1	0

f_4	0	1
0	0	0
1	0	1





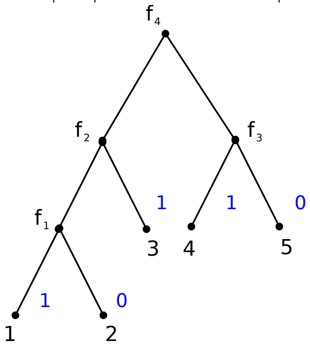
An example of a decomposition

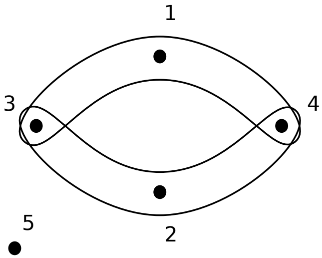
f_1	0	1
0	1	0
1	0	1

f_2	0	1
0	0	1
1	0	0

f_3	0	1
0	0	0
1	1	0

f_4	0	1
0	0	0
1	0	1





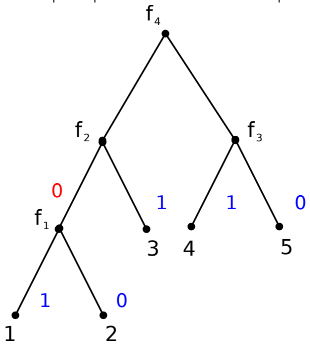
An example of a decomposition

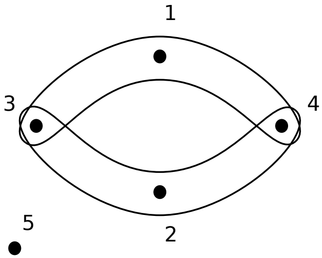
f_1	0	1
0	1	0
1	0	1

f_2	0	1
0	0	1
1	0	0

f_3	0	1
0	0	0
1	1	0

f_4	0	1
0	0	0
1	0	1





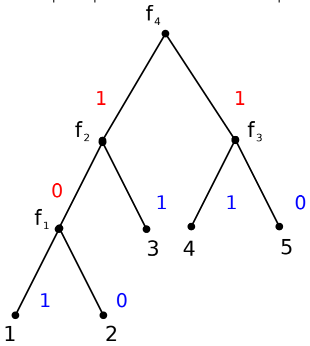
An example of a decomposition

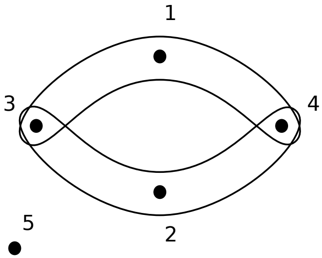
f_1	0	1
0	1	0
1	0	1

f_2	0	1
0	0	1
1	0	0

f_3	0	1
0	0	0
1	1	0

f_4	0	1
0	0	0
1	0	1





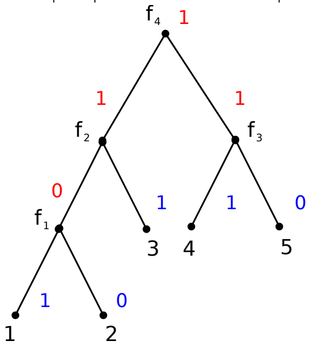
An example of a decomposition

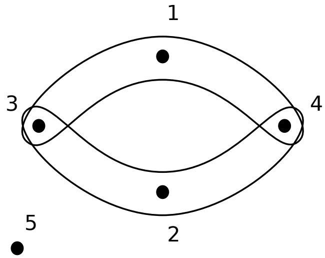
f_1	0	1
0	1	0
1	0	1

f_2	0	1
0	0	1
1	0	0

f_3	0	1
0	0	0
1	1	0

f_4	0	1
0	0	0
1	0	1





Hypergraph Decomposition

F_t is the set of unary and binary functions of domain and codomain $\{0, \ldots, t\}$.

$\mathcal{H}_t$ the set of terms $T(F_t, \{0, \ldots, t\})$.

Definition

Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let X be one of its subsets. The value of the term T where the labels of the leaves not in X are replaced by 0 is called the value of X in T and is denoted by $v(X, T)$.

Hypergraph Decomposition

F_t is the set of unary and binary functions of domain and codomain $\{0, \ldots, t\}$.

$\mathcal{H}_t$ the set of terms $T(F_t, \{0, \ldots, t\})$.

Definition

Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let X be one of its subsets. The value of the term T where the labels of the leaves not in X are replaced by 0 is called the value of X in T and is denoted by $v(X, T)$.

The function v defines a colored hypergraph.

Hypergraph Decomposition

F_t is the set of unary and binary functions of domain and codomain $\{0, \ldots, t\}$.

$\mathcal{H}_t$ the set of terms $T(F_t, \{0, \ldots, t\})$.

Definition

Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let X be one of its subsets. The value of the term T where the labels of the leaves not in X are replaced by 0 is called the value of X in T and is denoted by $v(X, T)$.

The function v defines a colored hypergraph.

Definition

Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let $S \subseteq \{0, \ldots, t\}$. The hypergraph H represented by (T, S) has L for vertices and its set of hyperedges is $\{X \subseteq L \mid v(X, T) \in S\}$.

Hypergraph Decomposition

F_t is the set of unary and binary functions of domain and codomain $\{0, \ldots, t\}$.

$\mathcal{H}_t$ the set of terms $T(F_t, \{0, \ldots, t\})$.

Definition

Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let X be one of its subsets. The value of the term T where the labels of the leaves not in X are replaced by 0 is called the value of X in T and is denoted by $v(X, T)$.

The function v defines a colored hypergraph.

Definition

Let T be a term of $\mathcal{H}_t$, let L be the set of leaves of T and let $S \subseteq \{0, \ldots, t\}$. The hypergraph H represented by (T, S) has L for vertices and its set of hyperedges is $\{X \subseteq L \mid v(X, T) \in S\}$.

Related decomposition notions

Definition

Let H be a hypergraph. The decomposition-width of H, denoted by $\mathrm{dw}(H)$, is the smallest integer t such that H is represented by a term of $\mathcal{H}_t$.

Related to decomposition notions of matroids, given by their sets of circuits.

- Decomposition-width of matroids [Kral].
- Branch-width of representable matroids [Hlineny].
- Series-parallel composition of matroids [S.].
- Can be used to decompose *oriented* matroids.

Related decomposition notions

Definition

Let H be a hypergraph. The decomposition-width of H, denoted by $\mathrm{dw}(H)$, is the smallest integer t such that H is represented by a term of $\mathcal{H}_t$.

Related to decomposition notions of matroids, given by their sets of circuits.

- Decomposition-width of matroids [Kral].
- Branch-width of representable matroids [Hlineny].
- Series-parallel composition of matroids [S.].
- Can be used to decompose *oriented* matroids.

Other restrictions of hypergraphs: uniform hypergraphs, graphs ...

Related decomposition notions

Definition

Let H be a hypergraph. The decomposition-width of H, denoted by $\mathrm{dw}(H)$, is the smallest integer t such that H is represented by a term of $\mathcal{H}_t$.

Related to decomposition notions of matroids, given by their sets of circuits.

- Decomposition-width of matroids [Kral].
- Branch-width of representable matroids [Hlineny].
- Series-parallel composition of matroids [S.].
- Can be used to decompose *oriented* matroids.

Other restrictions of hypergraphs: uniform hypergraphs, graphs . . .

Monadic second order logic

We consider the *Monadic Second Order* (MSO) logic.
Quantification over vertices and set of vertices.

Hyperedge relation: $E(X)$ holds if and only if X is an hyperedge.

Monadic second order logic

We consider the *Monadic Second Order* (MSO) logic.
Quantification over vertices and set of vertices.

Hyperedge relation: $E(X)$ holds if and only if X is an hyperedge.

Examples:

- Clutter: $\forall X, Y[(X \subset Y \wedge E(Y)) \Rightarrow \neg E(X)]$

Monadic second order logic

We consider the *Monadic Second Order* (MSO) logic.
Quantification over vertices and set of vertices.

Hyperedge relation: $E(X)$ holds if and only if X is an hyperedge.

Examples:

- Clutter: $\forall X, Y[(X \subset Y \wedge E(Y)) \Rightarrow \neg E(X)]$
- X is a transversal:
  $\textsc{Transversal}(X) \equiv \forall Y[E(Y) \Rightarrow (X \cap Y \neq \emptyset)]$.

Monadic second order logic

We consider the *Monadic Second Order* (MSO) logic.
Quantification over vertices and set of vertices.

Hyperedge relation: $E(X)$ holds if and only if X is an hyperedge.

Examples:

- Clutter: $\forall X, Y[(X \subset Y \wedge E(Y)) \Rightarrow \neg E(X)]$
- X is a transversal:
  $\textsc{Transversal}(X) \equiv \forall Y[E(Y) \Rightarrow (X \cap Y \neq \emptyset)]$.
- k-coloring:

$$\exists X_1 \dots \exists X_k \bigwedge_{i \neq j}(X_i \cap X_j = \emptyset) \wedge \forall X E(X) \Rightarrow$$
$$[\exists v_1 \exists v_2 (v_1 \in X) \wedge (v_2 \in X) \wedge \bigvee_{i \neq j}(v_1 \in X_i) \wedge (v_2 \in X_j)]$$

Monadic second order logic

We consider the *Monadic Second Order* (MSO) logic.
Quantification over vertices and set of vertices.

Hyperedge relation: $E(X)$ holds if and only if X is an hyperedge.

Examples:

- Clutter: $\forall X, Y[(X \subset Y \wedge E(Y)) \Rightarrow \neg E(X)]$
- X is a transversal:
  $\textsc{Transversal}(X) \equiv \forall Y[E(Y) \Rightarrow (X \cap Y \neq \emptyset)]$.
- k-coloring:

$$\exists X_1 \dots \exists X_k \bigwedge_{i \neq j}(X_i \cap X_j = \emptyset) \wedge \forall X E(X) \Rightarrow$$
$$[\exists v_1 \exists v_2 (v_1 \in X) \wedge (v_2 \in X) \wedge \bigvee_{i \neq j}(v_1 \in X_i) \wedge (v_2 \in X_j)]$$

- Set of circuits of a matroid.

Monadic second order logic

We consider the *Monadic Second Order* (MSO) logic.
Quantification over vertices and set of vertices.

Hyperedge relation: $E(X)$ holds if and only if X is an hyperedge.

Examples:

- Clutter: $\forall X, Y[(X \subset Y \wedge E(Y)) \Rightarrow \neg E(X)]$
- X is a transversal:
  $\textsc{Transversal}(X) \equiv \forall Y[E(Y) \Rightarrow (X \cap Y \neq \emptyset)]$.
- k-coloring:

$$\exists X_1 \ldots \exists X_k \bigwedge_{i \neq j}(X_i \cap X_j = \emptyset) \wedge \forall X E(X) \Rightarrow$$
$$[\exists v_1 \exists v_2 (v_1 \in X) \wedge (v_2 \in X) \wedge \bigvee_{i \neq j}(v_1 \in X_i) \wedge (v_2 \in X_j)]$$

- Set of circuits of a matroid.

Tractability of MSO

Theorem

Let φ be a MSO formula of size l and H a hypergraph with n vertices given by a term of $\mathcal{H}_t$. There is an algorithm which decides whether $H \models \varphi$ in time $f(t, l) \times n$ where f is a computable function.

Idea : the relation E of a hypergraph H can be represented by a MSO formula over its representation by a term of $\mathcal{H}_t$. Equivalently, there is a tree automaton to do that.

Tractability of MSO

Theorem

Let φ be a MSO formula of size l and H a hypergraph with n vertices given by a term of $\mathcal{H}_t$. There is an algorithm which decides whether $H \models \varphi$ in time $f(t,l) \times n$ where f is a computable function.

Idea : the relation E of a hypergraph H can be represented by a MSO formula over its representation by a term of $\mathcal{H}_t$. Equivalently, there is a tree automaton to do that.

Enumeration

Theorem

Let $\varphi(X)$ be a MSO formula of size l with a free variable X and H a hypergraph with n vertices given by a term of $\mathcal{H}_t$. There is an algorithm which lists all satisfying assignments of X in H with delay $f(t,l) \times n$ where f is a computable function.

Minimal transversals are interesting objects in database, boolean circuits and I.A:

$$\textsc{Transversal}(X) \wedge \forall Y [\textsc{Transversal}(Y) \Rightarrow \neg(Y \subsetneq X)].$$

Enumeration

Theorem

Let $\varphi(X)$ be a MSO formula of size l with a free variable X and H a hypergraph with n vertices given by a term of $\mathcal{H}_t$. There is an algorithm which lists all satisfying assignments of X in H with delay $f(t,l) \times n$ where f is a computable function.

Minimal transversals are interesting objects in database, boolean circuits and I.A:

$$\textsc{Transversal}(X) \wedge \forall Y [\textsc{Transversal}(Y) \Rightarrow \neg(Y \subsetneq X)].$$

Complexity of enumerating the minimal transversals: open.

Enumeration

Theorem

Let $\varphi(X)$ be a MSO formula of size l with a free variable X and H a hypergraph with n vertices given by a term of $\mathcal{H}_t$. There is an algorithm which lists all satisfying assignments of X in H with delay $f(t,l) \times n$ where f is a computable function.

Minimal transversals are interesting objects in database, boolean circuits and I.A:

$$\textsc{Transversal}(X) \wedge \forall Y [\textsc{Transversal}(Y) \Rightarrow \neg (Y \subsetneq X)].$$

Complexity of enumerating the minimal transversals: open.

Normal Form

A term of $\mathcal{H}_t$ is in normal form if:

- it contains only binary functions
- it has only 1 as constants

Proposition

Let H be a hypergraph with two or more vertices represented by (T, S) where $T \in \mathcal{H}_t$, then there is $\tilde{T} \in \mathcal{H}_t$ in normal form such that $(\tilde{T}, [t])$ represents H.

Normal Form

A term of $\mathcal{H}_t$ is in normal form if:

- it contains only binary functions
- it has only 1 as constants

Proposition

Let H be a hypergraph with two or more vertices represented by (T, S) where $T \in \mathcal{H}_t$, then there is $\tilde{T} \in \mathcal{H}_t$ in normal form such that $(\tilde{T}, [t])$ represents H.

Idea: local transformation of the decomposition tree.

Normal Form

A term of $\mathcal{H}_t$ is in normal form if:

- it contains only binary functions
- it has only 1 as constants

Proposition

Let H be a hypergraph with two or more vertices represented by (T, S) where $T \in \mathcal{H}_t$, then there is $\tilde{T} \in \mathcal{H}_t$ in normal form such that $(\tilde{T}, [t])$ represents H.

Idea: local transformation of the decomposition tree.

Hypergraphs Operations

Usual operations preserve the decomposition-width.

- **Vertex set restriction:** $H = (V, E)$, $W \subseteq V$ then $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$.

Hypergraphs Operations

Usual operations preserve the decomposition-width.

- **Vertex set restriction:** $H = (V, E)$, $W \subseteq V$ then $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$.
- **Adding or removing an edge:** $\mathrm{dw}(H \setminus \{e_1\}) \leq \mathrm{dw}(H) + 1$ and $\mathrm{dw}(H \cup \{e_1\}) \leq \mathrm{dw}(H) + 1$.

Hypergraphs Operations

Usual operations preserve the decomposition-width.

- **Vertex set restriction:** $H = (V, E)$, $W \subseteq V$ then $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$.
- **Adding or removing an edge:** $\mathrm{dw}(H \setminus \{e_1\}) \leq \mathrm{dw}(H) + 1$ and $\mathrm{dw}(H \cup \{e_1\}) \leq \mathrm{dw}(H) + 1$.
- **Disjoint union:** $m = \max(\mathrm{dw}(H_1), \mathrm{dw}(H_2))$, $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 1$.

Hypergraphs Operations

Usual operations preserve the decomposition-width.

- **Vertex set restriction:** $H = (V, E)$, $W \subseteq V$ then $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$.
- **Adding or removing an edge:** $\mathrm{dw}(H \setminus \{e_1\}) \leq \mathrm{dw}(H) + 1$ and $\mathrm{dw}(H \cup \{e_1\}) \leq \mathrm{dw}(H) + 1$.
- **Disjoint union:** $m = \max(\mathrm{dw}(H_1), \mathrm{dw}(H_2))$, $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 1$.
- **Union with a common point:** $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 2$.

Hypergraphs Operations

Usual operations preserve the decomposition-width.

- **Vertex set restriction:** $H = (V, E)$, $W \subseteq V$ then $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$.
- **Adding or removing an edge:** $\mathrm{dw}(H \setminus \{e_1\}) \leq \mathrm{dw}(H) + 1$ and $\mathrm{dw}(H \cup \{e_1\}) \leq \mathrm{dw}(H) + 1$.
- **Disjoint union:** $m = \max(\mathrm{dw}(H_1), \mathrm{dw}(H_2))$, $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 1$.
- **Union with a common point:** $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 2$.

Does not seem to work: Induced hypergraph, amalgamated sum.

Hypergraphs Operations

Usual operations preserve the decomposition-width.

- **Vertex set restriction:** $H = (V, E)$, $W \subseteq V$ then $\mathrm{dw}(H \times W) \leq \mathrm{dw}(H)$.
- **Adding or removing an edge:** $\mathrm{dw}(H \setminus \{e_1\}) \leq \mathrm{dw}(H) + 1$ and $\mathrm{dw}(H \cup \{e_1\}) \leq \mathrm{dw}(H) + 1$.
- **Disjoint union:** $m = \max(\mathrm{dw}(H_1), \mathrm{dw}(H_2))$, $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 1$.
- **Union with a common point:** $m \leq \mathrm{dw}(H_1 \cup H_2) \leq m + 2$.

Does not seem to work: Induced hypergraph, amalgamated sum.

Bounds on the decomposition-width

Proposition

Let H be a hypergraph with n vertices, then $\mathrm{dw}(H) \leq 2^{\lceil \frac{n}{2} \rceil}$.

Idea of proof: choose a partition of the vertices into two equal parts. Build a term for each with one color for each hyperedge.

Bounds on the decomposition-width

Proposition

Let H be a hypergraph with n vertices, then $\mathrm{dw}(H) \leq 2^{\lceil \frac{n}{2} \rceil}$.

Idea of proof: choose a partition of the vertices into two equal parts. Build a term for each with one color for each hyperedge.

Proposition

For $n \geq 8$, there is a hypergraph H with n vertices such that $\mathrm{dw}(H) > \frac{2^{\lceil \frac{n}{2} \rceil}}{n}$.

Bounds on the decomposition-width

Proposition

Let H be a hypergraph with n vertices, then $\mathrm{dw}(H) \leq 2^{\lceil \frac{n}{2} \rceil}$.

Idea of proof: choose a partition of the vertices into two equal parts. Build a term for each with one color for each hyperedge.

Proposition

For $n \geq 8$, there is a hypergraph H with n vertices such that $\mathrm{dw}(H) > \frac{2^{\lceil \frac{n}{2} \rceil}}{n}$.

Idea of proof: count the terms of $\mathcal{H}_t$.

Bounds on the decomposition-width

Proposition

Let H be a hypergraph with n vertices, then $\mathrm{dw}(H) \leq 2^{\lceil \frac{n}{2} \rceil}$.

Idea of proof: choose a partition of the vertices into two equal parts. Build a term for each with one color for each hyperedge.

Proposition

For $n \geq 8$, there is a hypergraph H with n vertices such that $\mathrm{dw}(H) > \frac{2^{\lceil \frac{n}{2} \rceil}}{n}$.

Idea of proof: count the terms of $\mathcal{H}_t$.

Type of an edge

How to bound the decomposition-width of a hypergraph ?

The type of X a set of vertices with regard to Y ($X \cap Y = \emptyset$):
$\mathrm{type}(X, Y) = \{W \subseteq Y \mid X \cup W \in E\}$.

Type of an edge

How to bound the decomposition-width of a hypergraph ?

The type of X a set of vertices with regard to Y ($X \cap Y = \emptyset$):
$\mathrm{type}(X, Y) = \{W \subseteq Y \mid X \cup W \in E\}$.

$\mathrm{Type}(X) = \{\mathrm{type}(Z, \overline{X}) \mid Z \subseteq X\}$.

Type of an edge

How to bound the decomposition-width of a hypergraph ?

The type of X a set of vertices with regard to Y ($X \cap Y = \emptyset$):
$\text{type}(X, Y) = \{W \subseteq Y \mid X \cup W \in E\}$.

$\text{Type}(X) = \{\text{type}(Z, \overline{X}) \mid Z \subseteq X\}$.

Lemma

Let $T \in \mathcal{H}_t$ be a term which represents the hypergraph H and let T' be one of its subterm. Let L be the set of vertices in T' then $|\text{Type}(L)| \leq t + 1$.

Type of an edge

How to bound the decomposition-width of a hypergraph ?

The type of X a set of vertices with regard to Y ($X \cap Y = \emptyset$):
$\mathrm{type}(X, Y) = \{ W \subseteq Y \mid X \cup W \in E \}$.

$\mathrm{Type}(X) = \{\mathrm{type}(Z, \overline{X}) \mid Z \subseteq X\}$.

Lemma

Let $T \in \mathcal{H}_t$ be a term which represents the hypergraph H and let T' be one of its subterm. Let L be the set of vertices in T' then $|\mathrm{Type}(L)| \leq t + 1$.

Explicit family of large decomposition-width

First family: $H_{k,n} = ([n], \{X \mid |X| = k\})$

Proposition

For all $n > 3k$, we have $\mathrm{dw}(H_{k,n}) = k + 1$.

Idea of proof : for any decomposition T of $H_{k,n}$, find a subterm whose set of leaves L satisfies $n/3 \leq |L| \leq 2n/3$. The type of a set is roughly its size. Thus $\mathrm{Type}(L) = k + 1$.

Explicit family of large decomposition-width

First family: $H_{k,n} = ([n], \{X \mid |X| = k\})$

Proposition

For all $n > 3k$*, we have* $\mathrm{dw}(H_{k,n}) = k + 1$.

Idea of proof : for any decomposition T of $H_{k,n}$, find a subterm whose set of leaves L satisfies $n/3 \leq |L| \leq 2n/3$. The type of a set is roughly its size. Thus $\mathrm{Type}(L) = k + 1$.

Second family: $I_n = ([n], \{X \subseteq [n] \mid |X| \in X\})$

Explicit family of large decomposition-width

First family: $H_{k,n} = ([n], \{X \mid |X| = k\})$

Proposition

For all $n > 3k$*, we have* $\mathrm{dw}(H_{k,n}) = k + 1$.

Idea of proof : for any decomposition T of $H_{k,n}$, find a subterm whose set of leaves L satisfies $n/3 \leq |L| \leq 2n/3$. The type of a set is roughly its size. Thus $\mathrm{Type}(L) = k + 1$.

Second family: $I_n = ([n], \{X \subseteq [n] \mid |X| \in X\})$

Theorem

For all $n > 0$*, we have* $\mathrm{dw}(I_n) \geq 2^{\frac{n}{27}}$.

Explicit family of large decomposition-width

First family: $H_{k,n} = ([n], \{X \mid |X| = k\})$

Proposition

For all $n > 3k$*, we have* $\mathrm{dw}(H_{k,n}) = k + 1$.

Idea of proof : for any decomposition T of $H_{k,n}$, find a subterm whose set of leaves L satisfies $n/3 \leq |L| \leq 2n/3$. The type of a set is roughly its size. Thus $\mathrm{Type}(L) = k + 1$.

Second family: $I_n = ([n], \{X \subseteq [n] \mid |X| \in X\})$

Theorem

For all $n > 0$*, we have* $\mathrm{dw}(I_n) \geq 2^{\frac{n}{27}}$.

Uniform-representation I

A decomposition adapted to k-uniform hypergraphs:

$D = \{(0,0),(k,0),(k+1,0)\} \cup \{(i,j)\}_{0<i<k, 0\leq j\leq t}$
$\mathcal{F}_{k,t}$ is the set of unary and binary functions with domain and codomain D which satisfy for all $(a,b),(c,d) \in D^2$:

- $f((a,b)) = (a,c)$ for some $c \leq t$
- $g((a,b),(c,d)) = (a+c,e)$ for some $e \leq t$ when $a+c<k$
- $g((a,b),(c,d)) = (k,0)$ or $(k+1,0)$ when $a+c=k$
- $g((a,b),(c,d)) = (k+1,0)$ when $a+c>k$

Uniform-representation I

A decomposition adapted to k-uniform hypergraphs:

$D = \{(0,0),(k,0),(k+1,0)\} \cup \{(i,j)\}_{0<i<k, 0\leq j\leq t}$
$\mathcal{F}_{k,t}$ is the set of unary and binary functions with domain and codomain D which satisfy for all $(a,b),(c,d) \in D^2$:

- $f((a,b)) = (a,c)$ for some $c \leq t$
- $g((a,b),(c,d)) = (a+c,e)$ for some $e \leq t$ when $a+c<k$
- $g((a,b),(c,d)) = (k,0)$ or $(k+1,0)$ when $a+c=k$
- $g((a,b),(c,d)) = (k+1,0)$ when $a+c>k$

$\mathcal{H}_{k,t}$: the set of terms $T(\mathcal{F}_{k,t}, \{(1,i)\}_{0\leq i\leq t})$.

Uniform-representation I

A decomposition adapted to k-uniform hypergraphs:

$D = \{(0,0), (k,0), (k+1,0)\} \cup \{(i,j)\}_{0<i<k, 0\leq j\leq t}$
$\mathcal{F}_{k,t}$ is the set of unary and binary functions with domain and codomain D which satisfy for all $(a,b),(c,d) \in D^2$:

- $f((a,b)) = (a,c)$ for some $c \leq t$
- $g((a,b),(c,d)) = (a+c, e)$ for some $e \leq t$ when $a + c < k$
- $g((a,b),(c,d)) = (k,0)$ or $(k+1,0)$ when $a + c = k$
- $g((a,b),(c,d)) = (k+1,0)$ when $a + c > k$

$\mathcal{H}_{k,t}$: the set of terms $T(\mathcal{F}_{k,t}, \{(1,i)\}_{0\leq i\leq t})$.

Uniform-representation II

Uniform decomposition-width: the smallest t such that H is represented by a term of $\mathcal{H}_{k,t}$ denoted by $\mathrm{dw}_u(H)$.

Proposition

All hypergraphs represented by a term of $\mathcal{H}_{k,t}$ are k-uniform and the following holds:

$$\mathrm{dw}_u(H) \leq \mathrm{dw}(H) \leq (k-1)(\mathrm{dw}_u(H)+1)+2.$$

Uniform-representation II

Uniform decomposition-width: the smallest t such that H is represented by a term of $\mathcal{H}_{k,t}$ denoted by $\mathrm{dw}_u(H)$.

Proposition

All hypergraphs represented by a term of $\mathcal{H}_{k,t}$ are k-uniform and the following holds:

$$\mathrm{dw}_u(H) \leq \mathrm{dw}(H) \leq (k-1)(\mathrm{dw}_u(H)+1)+2.$$

Idea: Right part is trivial.
Left part: inductively build a term of $\mathcal{H}_{k,t}$ from a term of $\mathcal{H}_t$ by taking into account the cardinal.

Uniform-representation II

Uniform decomposition-width: the smallest t such that H is represented by a term of $\mathcal{H}_{k,t}$ denoted by $\mathrm{dw}_u(H)$.

Proposition

All hypergraphs represented by a term of $\mathcal{H}_{k,t}$ are k-uniform and the following holds:

$$\mathrm{dw}_u(H) \leq \mathrm{dw}(H) \leq (k-1)(\mathrm{dw}_u(H)+1)+2.$$

Idea: Right part is trivial.
Left part: inductively build a term of $\mathcal{H}_{k,t}$ from a term of $\mathcal{H}_t$ by taking into account the cardinal.

Clique-width

Let $F_{\mathcal{L}}$ be the following set of graph operations:

- The disjoint union of two labeled graphs: $\oplus$.

Clique-width

Let $F_{\mathcal{L}}$ be the following set of graph operations:

- The disjoint union of two labeled graphs: $\oplus$.
- For all $a, b \in \mathcal{L}$, the function which renames every vertex labeled by a into b: $\rho_{a \to b}$.

Clique-width

Let $F_{\mathcal{L}}$ be the following set of graph operations:

- The disjoint union of two labeled graphs: $\oplus$.
- For all $a, b \in \mathcal{L}$, the function which renames every vertex labeled by a into b: $\rho_{a \to b}$.
- For all $a, b \in \mathcal{L}$, the function which adds all edges between the vertices labeled a and those labeled b: $\eta_{a,b}$.

Clique-width

Let $F_{\mathcal{L}}$ be the following set of graph operations:

- The disjoint union of two labeled graphs: $\oplus$.
- For all $a, b \in \mathcal{L}$, the function which renames every vertex labeled by a into b: $\rho_{a \to b}$.
- For all $a, b \in \mathcal{L}$, the function which adds all edges between the vertices labeled a and those labeled b: $\eta_{a,b}$.

G_a the graph with one vertex labeled by a, $G_{\mathcal{L}} = \{G_a \mid a \in \mathcal{L}\}$.

Clique-width

Let $F_{\mathcal{L}}$ be the following set of graph operations:

- The disjoint union of two labeled graphs: $\oplus$.
- For all $a, b \in \mathcal{L}$, the function which renames every vertex labeled by a into b: $\rho_{a \to b}$.
- For all $a, b \in \mathcal{L}$, the function which adds all edges between the vertices labeled a and those labeled b: $\eta_{a,b}$.

G_a the graph with one vertex labeled by a, $G_{\mathcal{L}} = \{G_a \mid a \in \mathcal{L}\}$.

Definition

The clique-width of the graph G, denoted by $\mathrm{cw}(G)$, is the minimum of the $n \in \mathbb{N}$ such that $\exists \gamma, (G, \gamma) \in T(F_{[n]}, G_{[n]})$.

Clique-width

Let $F_{\mathcal{L}}$ be the following set of graph operations:

- The disjoint union of two labeled graphs: $\oplus$.
- For all $a, b \in \mathcal{L}$, the function which renames every vertex labeled by a into b: $\rho_{a \to b}$.
- For all $a, b \in \mathcal{L}$, the function which adds all edges between the vertices labeled a and those labeled b: $\eta_{a,b}$.

G_a the graph with one vertex labeled by a, $G_{\mathcal{L}} = \{G_a \mid a \in \mathcal{L}\}$.

Definition

The clique-width of the graph G, denoted by $\mathrm{cw}(G)$, is the minimum of the $n \in \mathbb{N}$ such that $\exists \gamma, (G, \gamma) \in T(F_{[n]}, G_{[n]})$.

Decomposition-width and clique-width

Theorem

Let G be a graph, then $\mathrm{cw}(G)/2 \leq \mathrm{dw}(G) \leq \mathrm{cw}(G) + 2$.

Proposition

Let G be a graph, then $\mathrm{dw}_u(G) \leq \mathrm{cw}(G) \leq 2\,\mathrm{dw}_u(G)$.

Decomposition-width and clique-width

Theorem

Let G be a graph, then $\mathrm{cw}(G)/2 \leq \mathrm{dw}(G) \leq \mathrm{cw}(G) + 2$.

Proposition

Let G be a graph, then $\mathrm{dw}_u(G) \leq \mathrm{cw}(G) \leq 2\,\mathrm{dw}_u(G)$.

Idea of the proof: Simulation of graph operation by functions of $\mathcal{F}_{k,t}$ and vice versa.
The graph G_i corresponds to a leaf of color $(1, i)$.
Use $2t$-terms which represents t-colored graphs. Different colors for the left and right part (function of $\mathcal{F}_{k,t}$ not symmetric).

Decomposition-width and clique-width

Theorem

Let G be a graph, then $\mathrm{cw}(G)/2 \leq \mathrm{dw}(G) \leq \mathrm{cw}(G) + 2$.

Proposition

Let G be a graph, then $\mathrm{dw}_u(G) \leq \mathrm{cw}(G) \leq 2\,\mathrm{dw}_u(G)$.

Idea of the proof: Simulation of graph operation by functions of $\mathcal{F}_{k,t}$ and vice versa.
The graph G_i corresponds to a leaf of color $(1, i)$.
Use $2t$-terms which represents t-colored graphs. Different colors for the left and right part (function of $\mathcal{F}_{k,t}$ not symmetric).

Computing the decomposition is hard I

The clique-width of a graph is NP hard to approximate [Fellows et al.].

Corollary

The decomposition-width is NP-hard to approximate.

Computing the decomposition is hard I

The clique-width of a graph is NP hard to approximate [Fellows et al.].

Corollary

The decomposition-width is NP*-hard to approximate.*

Simpler problem: a fixed integer k, test whether a hypergraph has decomposition-width k.

Computing the decomposition is hard I

The clique-width of a graph is NP hard to approximate [Fellows et al.].

Corollary

The decomposition-width is NP*-hard to approximate.*

Simpler problem: a fixed integer k, test whether a hypergraph has decomposition-width k.

Even simpler: $k = 1$?

Computing the decomposition is hard I

The clique-width of a graph is NP hard to approximate [Fellows et al.].

Corollary

The decomposition-width is NP-hard to approximate.

Simpler problem: a fixed integer k, test whether a hypergraph has decomposition-width k.

Even simpler: $k = 1$?

Computing the decomposition is hard II

A term of $\mathcal{H}_1$ is a read-once formula (each variable appears only once) built from all possible logical connectors.

Read-once formulas built from the connectors AND, OR and NOT cannot be learned in polynomial time with only membership queries.

Theorem

There is no polynomial time algorithm to compute the decomposition of a hypergraph of decomposition-width 1 given by a membership oracle.

Computing the decomposition is hard II

A term of $\mathcal{H}_1$ is a read-once formula (each variable appears only once) built from all possible logical connectors.

Read-once formulas built from the connectors AND, OR and NOT cannot be learned in polynomial time with only membership queries.

Theorem

There is no polynomial time algorithm to compute the decomposition of a hypergraph of decomposition-width 1 given by a membership oracle.

When the hypergraph of decomposition-width 1 is k-uniform or upward closed, it is possible to compute its decomposition.

Computing the decomposition is hard II

A term of $\mathcal{H}_1$ is a read-once formula (each variable appears only once) built from all possible logical connectors.

Read-once formulas built from the connectors AND, OR and NOT cannot be learned in polynomial time with only membership queries.

Theorem

There is no polynomial time algorithm to compute the decomposition of a hypergraph of decomposition-width 1 given by a membership oracle.

When the hypergraph of decomposition-width 1 is k-uniform or upward closed, it is possible to compute its decomposition.

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?
- Can the decomposition-width be seen as a branch-width (using Type)? restriction ?

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?
- Can the decomposition-width be seen as a branch-width (using Type)? restriction ?
- Is there a class of hypergraphs with a decomposition which can be found in polynomial time ? Acyclic hypergraphs ?

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?
- Can the decomposition-width be seen as a branch-width (using Type)? restriction ?
- Is there a class of hypergraphs with a decomposition which can be found in polynomial time ? Acyclic hypergraphs ?
- Hypergraphs of decomposition-width 1, 2 ?

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?
- Can the decomposition-width be seen as a branch-width (using Type)? restriction ?
- Is there a class of hypergraphs with a decomposition which can be found in polynomial time ? Acyclic hypergraphs ?
- Hypergraphs of decomposition-width 1, 2 ?
- Do local properties of functions turn into global properties of the represented hypergraphs ?

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?
- Can the decomposition-width be seen as a branch-width (using Type)? restriction ?
- Is there a class of hypergraphs with a decomposition which can be found in polynomial time ? Acyclic hypergraphs ?
- Hypergraphs of decomposition-width 1, 2 ?
- Do local properties of functions turn into global properties of the represented hypergraphs ?
- Bounding the decomposition-width of an amalgamated sum ? on which class of hypergraphs ?

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?
- Can the decomposition-width be seen as a branch-width (using Type)? restriction ?
- Is there a class of hypergraphs with a decomposition which can be found in polynomial time ? Acyclic hypergraphs ?
- Hypergraphs of decomposition-width 1, 2 ?
- Do local properties of functions turn into global properties of the represented hypergraphs ?
- Bounding the decomposition-width of an amalgamated sum ? on which class of hypergraphs ?
- Better specialized algorithm ? for the minimal transversals ?

Conclusion

A work in progress, with many open questions:

- To what is related the decomposition-width of uniform graphs ? Tree-width, clique-width of the Gaifman graph ?
- Can the decomposition-width be seen as a branch-width (using Type)? restriction ?
- Is there a class of hypergraphs with a decomposition which can be found in polynomial time ? Acyclic hypergraphs ?
- Hypergraphs of decomposition-width 1, 2 ?
- Do local properties of functions turn into global properties of the represented hypergraphs ?
- Bounding the decomposition-width of an amalgamated sum ? on which class of hypergraphs ?
- Better specialized algorithm ? for the minimal transversals ?

Thanks!