Constraint Solving via Fractional Edge Covers

Dániel Marx

Joint work with Martin Grohe

Humboldt-Universität zu Berlin

Institut für Informatik

MathCSP Workshop

March 22, 2006, Oxford

Constraint Satisfaction Problems (CSP)

A CSP instance is given by describing the

- variables,
- domain of the variables,
- constraints on the variables.

Task: Find an assignment that satisfies every constraint.

$$I = C_1(x_1, x_2, x_3) \wedge C_2(x_2, x_4) \wedge C_3(x_1, x_3, x_4)$$

Representation issues

How are the constraints represented in the input?

- full truth table
- listing the satisfying tuples
- formula/circuit
- oracle

Does not really matter if the constraints have small arities.

Representation issues

How are the constraints represented in the input?

- full truth table
- listing the satisfying tuples
- formula/circuit
- oracle

Does not really matter if the constraints have small arities.

In this talk: Each constraint is given by listing all the tuples that satisfy it.

Motivation: Applications in database theory & AI.
Constraints are known databases, “satisfying” means “appears in the database.”

Tractable structures

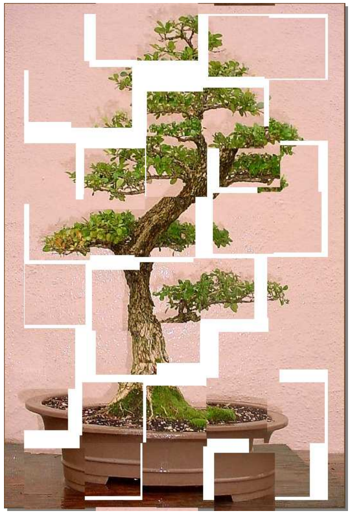
Our aim: identify structural properties that can make a CSP instance tractable.

- bounded tree width
- bounded (generalized) hypertree width
- bounded fractional edge cover number
- bounded fractional hypertree width

Tractable structures

Our aim: identify structural properties that can make a CSP instance tractable.

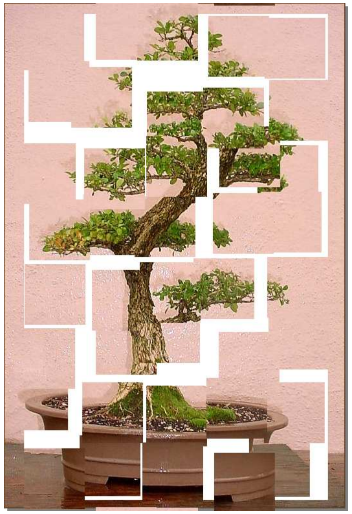
- bounded tree width
- bounded (generalized) hypertree width
- bounded fractional edge cover number
- bounded fractional hypertree width



Tractable structures

Our aim: identify structural properties that can make a CSP instance tractable.

- bounded tree width
- bounded (generalized) hypertree width
- bounded fractional edge cover number
- bounded fractional hypertree width

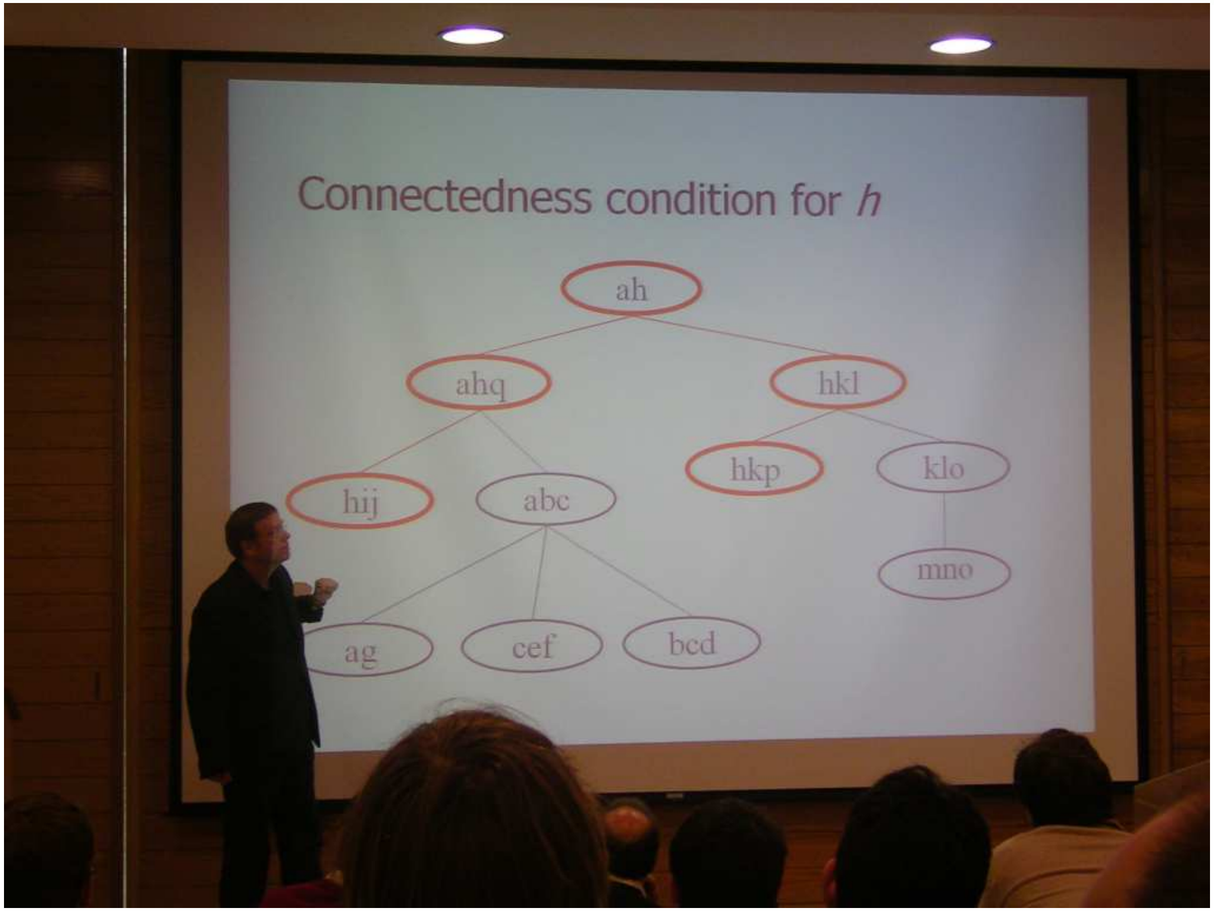


Hypergraph of an instance: vertices are variables, edges are constraint scopes.

If $\mathcal{H}$ is a class of hypergraphs, then CSP$(\mathcal{H})$ is the CSP problem restricted to instances whose hypergraph is in $\mathcal{H}$.

Task: Identify classes $\mathcal{H}$ such that CSP$(\mathcal{H})$ is polynomial-time solvable.

Tree width—reminder

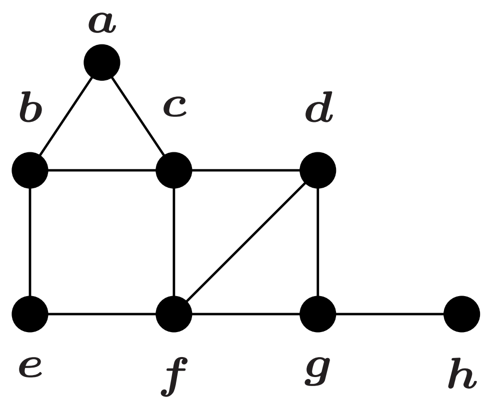


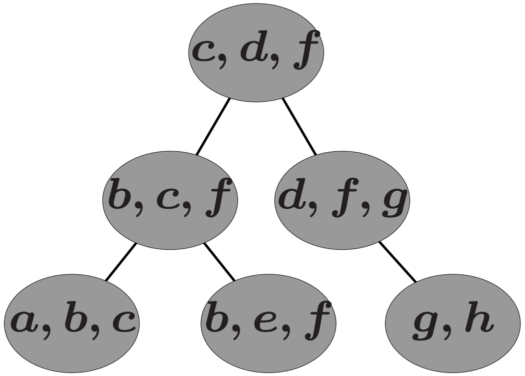
Tree width

Tree width: A measure of how “tree-like” the hypergraph is. (Introduced by Robertson and Seymour.)

Tree decomposition: Bags of vertices are arranged in a tree structure satisfying the following properties:

1. For every edge e, there is a bag containing the vertices of e.
2. For every vertex v, the bags containing v form a connected subtree.





Width of the decomposition: size of the largest bag minus 1.

Tree width: width of the best decomposition.

Generalized hypertree width

In a **generalized hypertree decomposition** [Gottlob et al. '99] of width w, bags of vertices are arranged in a tree structure such that

1. For every edge e, there is a bag containing the vertices of e.
2. For every vertex v, the bags containing v form a connected subtree.
3. For each bag, w edges are given (called the **guards**) that cover the bag.

Generalized hypertree width: width of the best decomposition.

Generalized hypertree width

Theorem: [Gottlob et al. '99] For every w, there is a polynomial-time algorithm for solving CSP on instances with hypergraphs having generalized hypertree width at most w.

Algorithm: Bottom up dynamic programming. There are at most $||I||^w$ possible satisfying assignments for each bag.

Generalized hypertree width

Theorem: [Gottlob et al. '99] For every w, there is a polynomial-time algorithm for solving CSP on instances with hypergraphs having generalized hypertree width at most w.

Algorithm: Bottom up dynamic programming. There are at most $||I||^w$ possible satisfying assignments for each bag.

Generalization: Is there some more general property that makes the number of satisfying assignments of a bag polynomial?

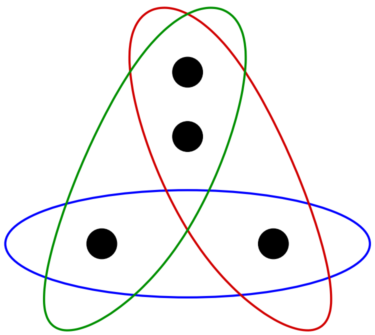
(Fractional) edge covering

An **edge cover** of a hypergraph is a subset of the edges such that every vertex is covered by at least one edge.

$\varrho(H)$: size of the smallest edge cover.

A **fractional edge cover** is a weight assignment to the edges such that every vertex is covered by total weight at least 1.

$\varrho^*(H)$: smallest total weight of a fractional edge cover.



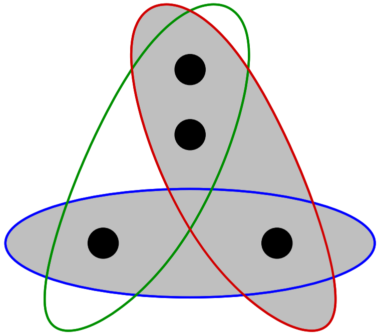
(Fractional) edge covering

An **edge cover** of a hypergraph is a subset of the edges such that every vertex is covered by at least one edge.

$\varrho(H)$: size of the smallest edge cover.

A **fractional edge cover** is a weight assignment to the edges such that every vertex is covered by total weight at least 1.

$\varrho^*(H)$: smallest total weight of a fractional edge cover.



$\varrho(H) = 2$

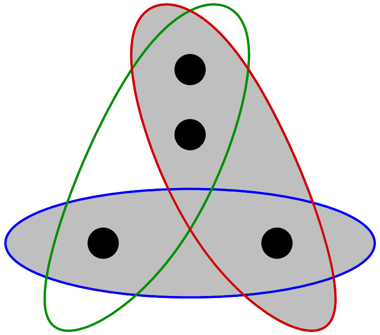
(Fractional) edge covering

An **edge cover** of a hypergraph is a subset of the edges such that every vertex is covered by at least one edge.

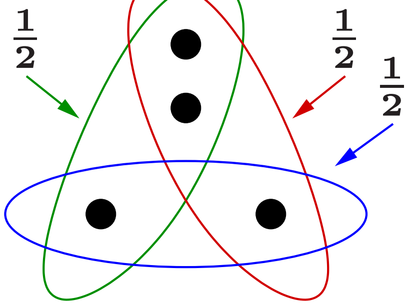
$\varrho(H)$: size of the smallest edge cover.

A **fractional edge cover** is a weight assignment to the edges such that every vertex is covered by total weight at least 1.

$\varrho^*(H)$: smallest total weight of a fractional edge cover.



$\varrho(H) = 2$



$\varrho^*(H) = 1.5$

Edge covers vs. fractional edge covers

Fact: It is NP-hard to determine the edge cover number $\varrho(H)$.

Fact: The fractional edge cover number $\varrho^*(H)$ can be determined in polynomial time using linear progamming.

The gap between $\varrho(H)$ and $\varrho^*(H)$ can be arbitrarily large.

Edge covers vs. fractional edge covers

Fact: It is NP-hard to determine the edge cover number $\varrho(H)$.
Fact: The fractional edge cover number $\varrho^*(H)$ can be determined in polynomial time using linear progamming.

The gap between $\varrho(H)$ and $\varrho^*(H)$ can be arbitrarily large.

Example:
$\binom{2k}{k}$ vertices: all the possible strings with k 0's and k 1's.
$2k$ hyperedges: edge E_i contains the vertices with 1 at the i-th position.

Edge covers vs. fractional edge covers

Fact: It is NP-hard to determine the edge cover number $\varrho(H)$.
Fact: The fractional edge cover number $\varrho^*(H)$ can be determined in polynomial time using linear progamming.

The gap between $\varrho(H)$ and $\varrho^*(H)$ can be arbitrarily large.

Example:
$\binom{2k}{k}$ vertices: all the possible strings with k 0's and k 1's.
$2k$ hyperedges: edge E_i contains the vertices with 1 at the i-th position.

Edge cover: if only k edges are selected, then there is a vertex that contains 1's only at the remaining k positions, hence not covered $\Rightarrow \varrho(H) \geq k+1$.

Fractional edge cover: assign weight $1/k$ to each edge, each vertex is covered by exactly k edges $\Rightarrow \varrho^*(H) \leq 2k \cdot 1/k = 2$.

CSP and fractional edge covering

Lemma: [easy] If the hypergraph of instance I has edge cover number w, then there are at most $\|I\|^w$ satisfying assignments.
Proof: Assume that $C_1, \ldots, C_w$ cover the instance. Fixing a satisfying assignment for each C_i determines all the variables.

Lemma: If the hypergraph of instance I has fractional edge cover number w, then there are at most $\|I\|^w$ satisfying assignments (and they can be enumerated in polynomial time).
Proof: By Shearer's Lemma.

Corollary: CSP($\mathcal{H}$) is polynomial-time solvable if $\mathcal{H}$ has bounded fractional edge cover number.

CSP and fractional edge covering

Lemma: [easy] If the hypergraph of instance I has edge cover number w, then there are at most $\|I\|^w$ satisfying assignments.
Proof: Assume that $C_1, \ldots, C_w$ cover the instance. Fixing a satisfying assignment for each C_i determines all the variables.

Lemma: If the hypergraph of instance I has fractional edge cover number w, then there are at most $\|I\|^w$ satisfying assignments (and they can be enumerated in polynomial time).
Proof: By Shearer's Lemma.

Corollary: CSP$(\mathcal{H})$ is polynomial-time solvable if $\mathcal{H}$ has bounded fractional edge cover number.

Remark: $\|I\|^w$ is tight, hence if the fractional edge cover number can be unbounded, then there is no polynomial bound on the number of solutions.

Shearer's Lemma—combinatorial version

Shearer's Lemma: Let $H = (V, E)$ be a hypergraph, and let A_1, A_2, . . . , A_p be (not necessarily distinct) subsets of V such that each $v \in V$ is contained in at least q of the A_i's. Denote by E_i the edge set of the hypergraph projected to A_i. Then

$$|E| \leq \prod_{i=1}^{p} |E_i|^{1/q}.$$

Shearer's Lemma—combinatorial version

Shearer's Lemma: Let $H = (V, E)$ be a hypergraph, and let $A_1, A_2, \ldots,$ A_p be (not necessarily distinct) subsets of V such that each $v \in V$ is contained in at least q of the A_i's. Denote by E_i the edge set of the hypergraph projected to A_i. Then

$$|E| \leq \prod_{i=1}^{p} |E_i|^{1/q}.$$

Example:

$$E = \{1, 13, 2, 23, 234, 24\} \; q = 2$$

$$A_1 = 123 \qquad A_2 = 124 \qquad A_3 = 34$$

$$E_1 = \{1, 13, 2, 23\} \quad E_2 = \{1, 2, 24\} \quad E_3 = \{\emptyset, 3, 4, 34\}$$

$$6 = |E| \leq (|E_1| \cdot |E_2| \cdot |E_3|)^{1/q} = (4 \cdot 3 \cdot 4)^{1/2} = 6.928$$

Shearer's Lemma—entropy version

Shearer's Lemma: Assume we have the following random variables:

- $X_1, \ldots, X_n$,
- $Y_1, \ldots, Y_m$, where each $Y_i = (X_{i_1}, \ldots, X_{i_k})$ is a combination of some X_i's,
- $X = (X_1, \ldots, X_n)$.

If each X_j appears in at least q of the Y_i's, then $H(X) \leq \frac{1}{q} \sum H(Y_i)$.

Entropy: "information content"

$H(X) = -\sum_x P(X = x) \log_2 P(X = x)$

Bounding the number of solutions

Lemma: If the hypergraph of instance I has fractional edge cover number w, then there are at most $\|I\|^w$ satisfying assignments.

Example: Let $C_1(x_1, x_2) \wedge C_2(x_2, x_3) \wedge C_3(x_1, x_3)$ be an instance where each constraint is satisfied by at most n pairs.

Fractonal edge cover number: $3/2 \Rightarrow$ we have to show that there are at most $n^{3/2}$ solutions.

Bounding the number of solutions

Lemma: If the hypergraph of instance I has fractional edge cover number w, then there are at most $\|I\|^w$ satisfying assignments.

Example: Let $C_1(x_1, x_2) \wedge C_2(x_2, x_3) \wedge C_3(x_1, x_3)$ be an instance where each constraint is satisfied by at most n pairs.

Fractonal edge cover number: $3/2 \Rightarrow$ we have to show that there are at most $n^{3/2}$ solutions.

Let $X = (x_1, x_2, x_3)$ be a random variable with uniform distribution over the **satisfying assignments** of the instance.

$Y_1 = (x_1, x_2)$ $Y_2 = (x_2, x_3)$ $Y_3 = (x_1, x_3)$
$H(Y_i) \leq \log_2 n$ (has at most n different values)
$H(X) \leq \frac{1}{2}(H(Y_1) + H(Y_2) + H(Y_3)) \leq \frac{3}{2} \log_2 n$

X has uniform distribution, hence it has $2^{H(X)} = 2^{\frac{3}{2}\log_2 n} = n^{3/2}$ different values.

Fractional hypertree width

In a **fractional hypertree decomposition** of width w, bags of vertices are arranged in a tree structure such that

1. For every edge e, there is a bag containing the vertices of e.
2. For every vertex v, the bags containing v form a connected subtree.
3. A fractional edge cover of weight w is given for each bag.

Fractional hypertree width: width of the best decomposition.

Note: fractional hypertree width $\leq$ generalized hypertree width

Fractional hypertree width

In a **fractional hypertree decomposition** of width w, bags of vertices are arranged in a tree structure such that

1. For every edge e, there is a bag containing the vertices of e.
2. For every vertex v, the bags containing v form a connected subtree.
3. A fractional edge cover of weight w is given for each bag.

Fractional hypertree width: width of the best decomposition.

Note: fractional hypertree width $\leq$ generalized hypertree width

Theorem: For every w, there is a polynomial-time algorithm for solving CSP if a fractional hypertree decomposition of width at most w is given in the input.

Determining fractional hypertree width

? Currently we do not know if deciding fractional hypertree width $\leq w$ is possible in polynomial time for every fixed value of w.

For the applications, an approximate form would be sufficient:

Conjecture: There are functions $f_1(w)$, $f_2(w)$ such that for every w, there is an algorithm that constructs in time $n^{f_1(w)}$ a fractional hypertree decompositioni of width $\leq f_2(w)$ for hypergraphs having fractional hypertree width $\leq w$.

Determining fractional hypertree width

? Currently we do not know if deciding fractional hypertree width $\leq w$ is possible in polynomial time for every fixed value of w.

For the applications, an approximate form would be sufficient:

Conjecture: There are functions $f_1(w)$, $f_2(w)$ such that for every w, there is an algorithm that constructs in time $n^{f_1(w)}$ a fractional hypertree decompositioni of width $\leq f_2(w)$ for hypergraphs having fractional hypertree width $\leq w$.

Two possible approaches:

- Separator-based approach. Problem: given sets X, Y, we have to find a separator that can be fractionally edge covered with weight $\leq w$.
- Game-theoretic approach.

Law enforcement on graphs

Robber and Cops Game: k cops try to capture a robber in the graph.

- In each step, the cops can move from vertex to vertex arbitrarily with helicopters.
- The robber moves infinitely fast, and sees where the cops will land.
- The robber cannot go through the vertices blocked by the cops.

Law enforcement on graphs

Robber and Cops Game: k cops try to capture a robber in the graph.

- In each step, the cops can move from vertex to vertex arbitrarily with helicopters.
- The robber moves infinitely fast, and sees where the cops will land.
- The robber cannot go through the vertices blocked by the cops.

Law enforcement on graphs

Robber and Cops Game: k cops try to capture a robber in the graph.

- In each step, the cops can move from vertex to vertex arbitrarily with helicopters.
- The robber moves infinitely fast, and sees where the cops will land.
- The robber cannot go through the vertices blocked by the cops.

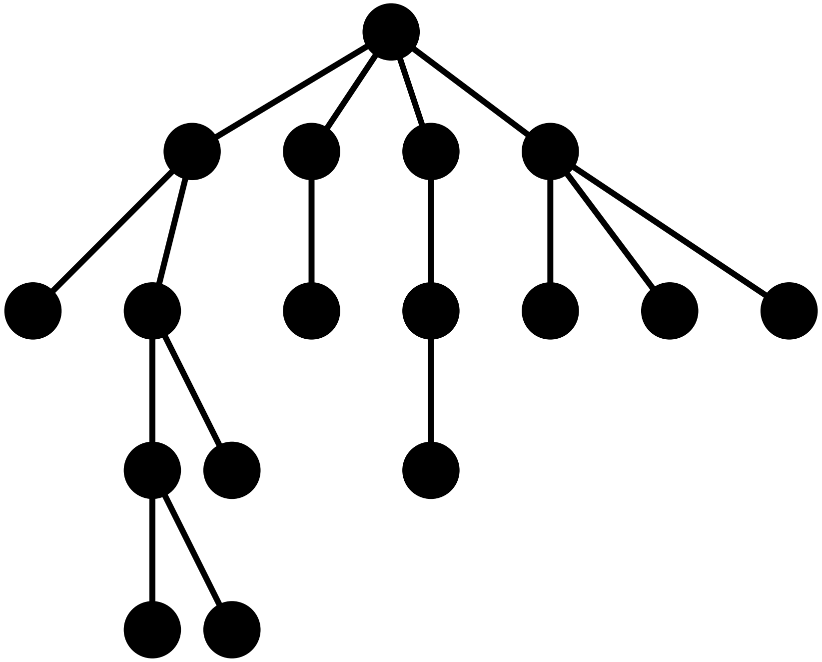
Theorem: [Seymour and Thomas '93]

k cops can win the game $\iff$ the tree width of the graph is at most $k-1$.

The winner of the game can be determined in $n^{O(k)}$ time $\Rightarrow$ tree width $\leq k$ can be checked in polynomial time for fixed k.

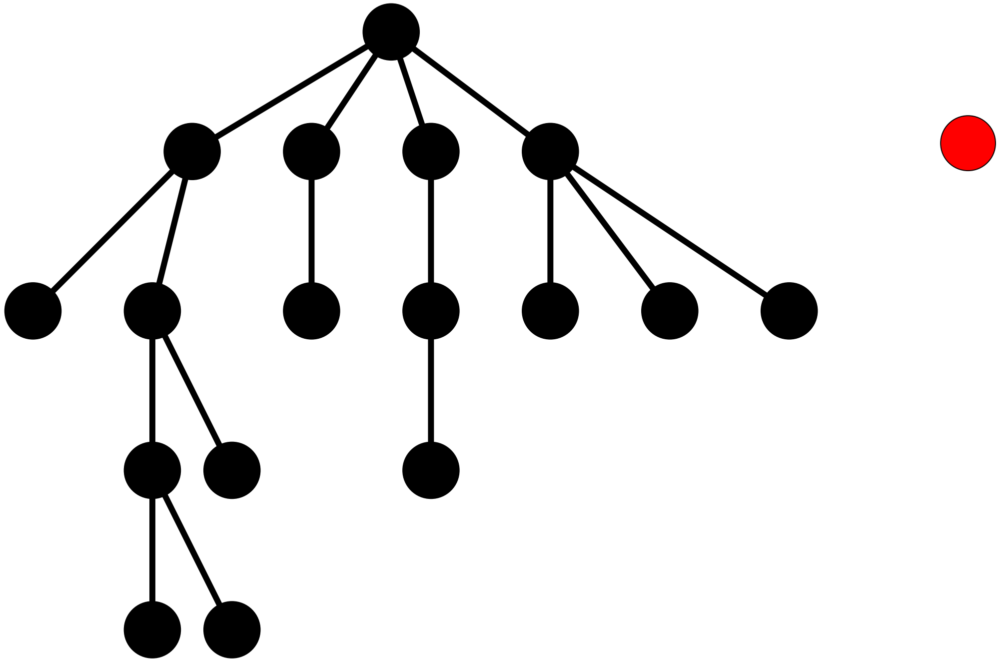
The Robber and Cops game (cont.)

Example: 2 cops have a winning strategy in a tree.



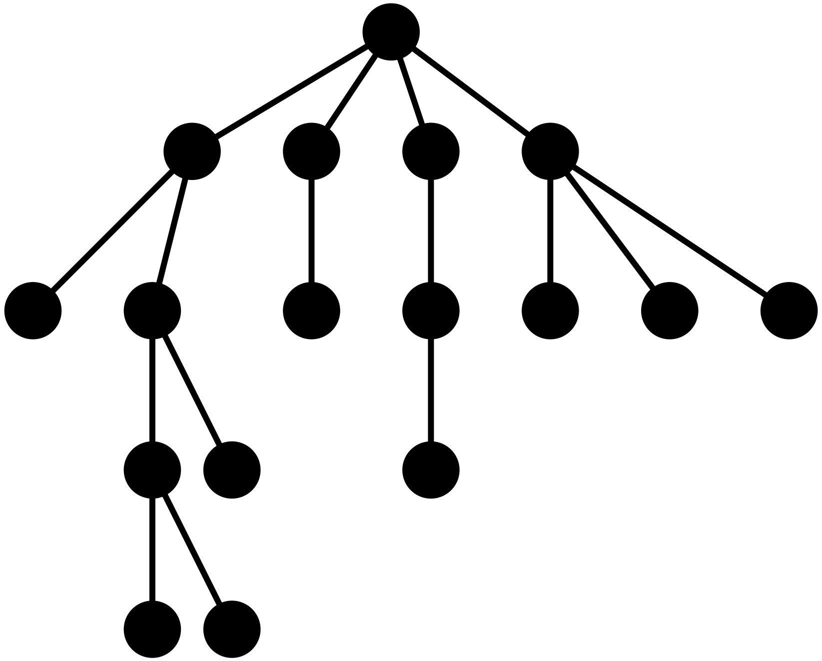
The Robber and Cops game (cont.)

Example: 2 cops have a winning strategy in a tree.



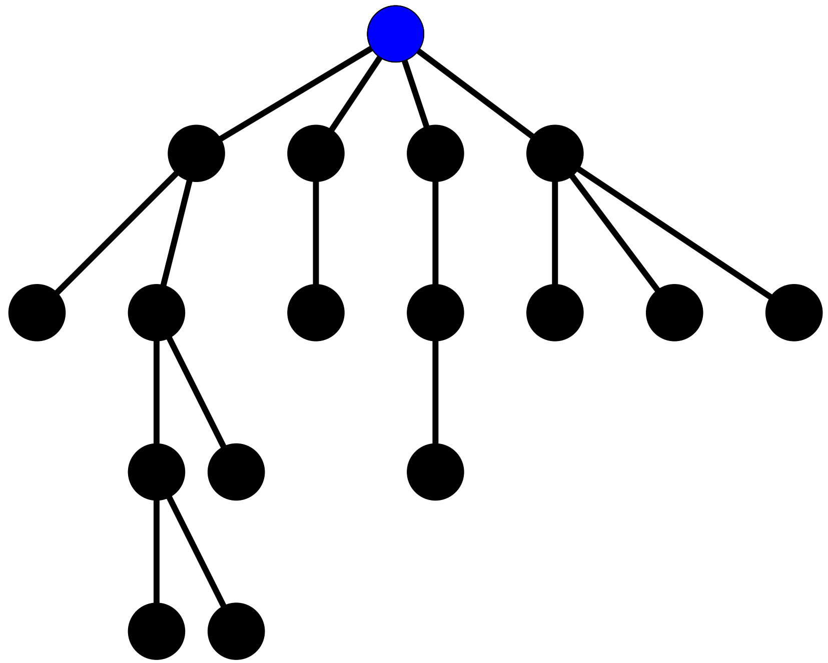
The Robber and Cops game (cont.)

Example: 2 cops have a winning strategy in a tree.



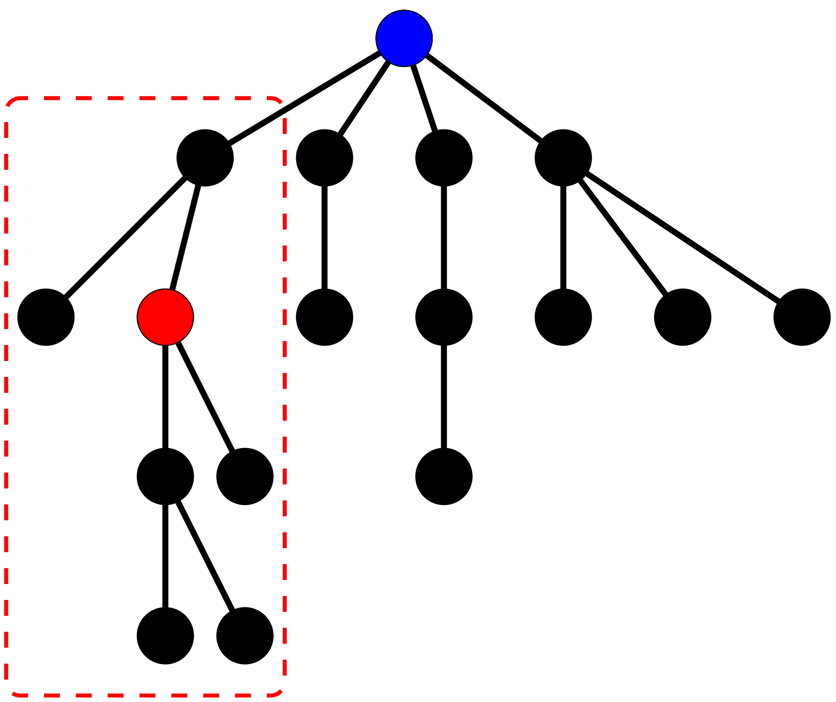
The Robber and Cops game (cont.)

Example: 2 cops have a winning strategy in a tree.



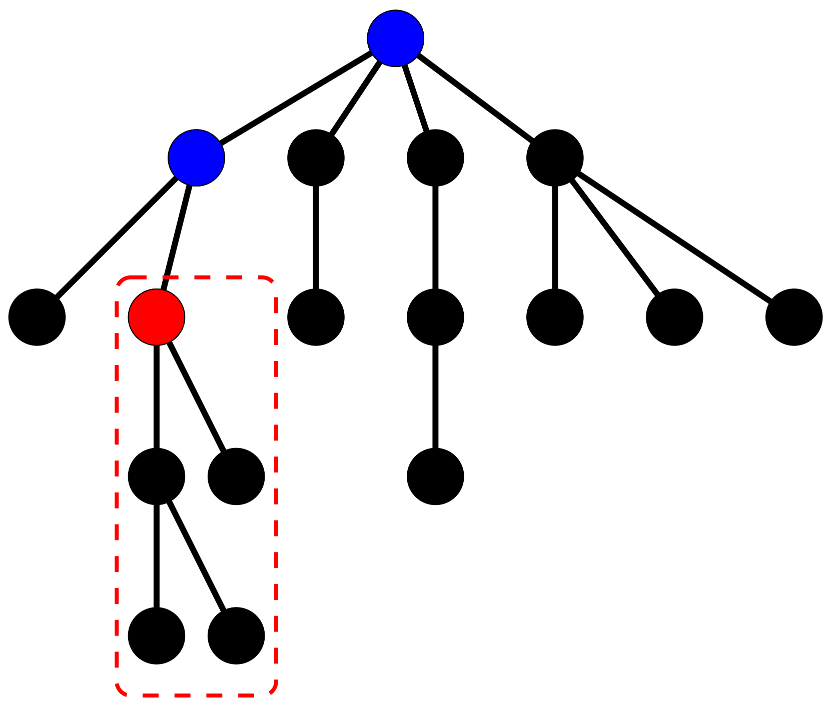
The Robber and Cops game (cont.)

Example: 2 cops have a winning strategy in a tree.



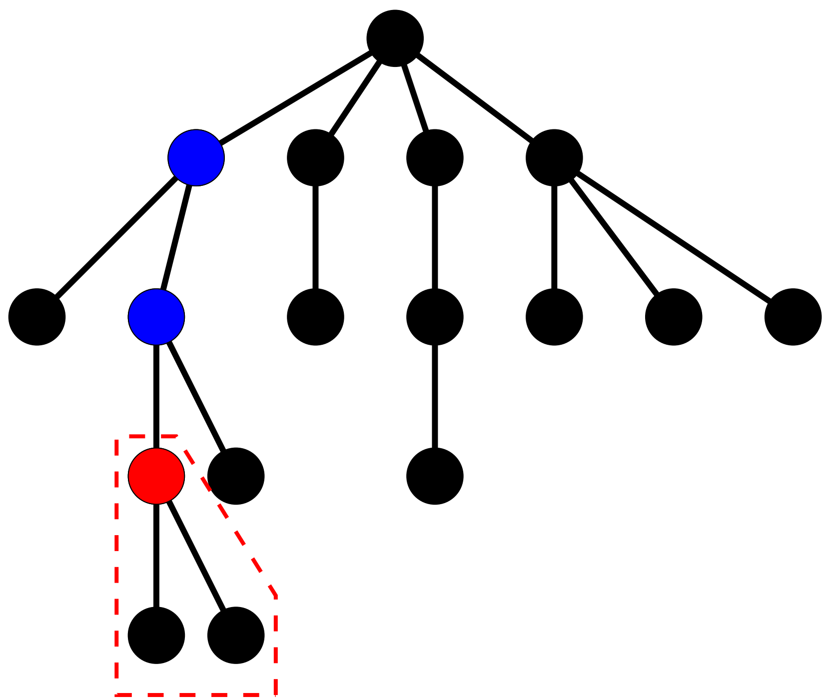
The Robber and Cops game (cont.)

Example: 2 cops have a winning strategy in a tree.



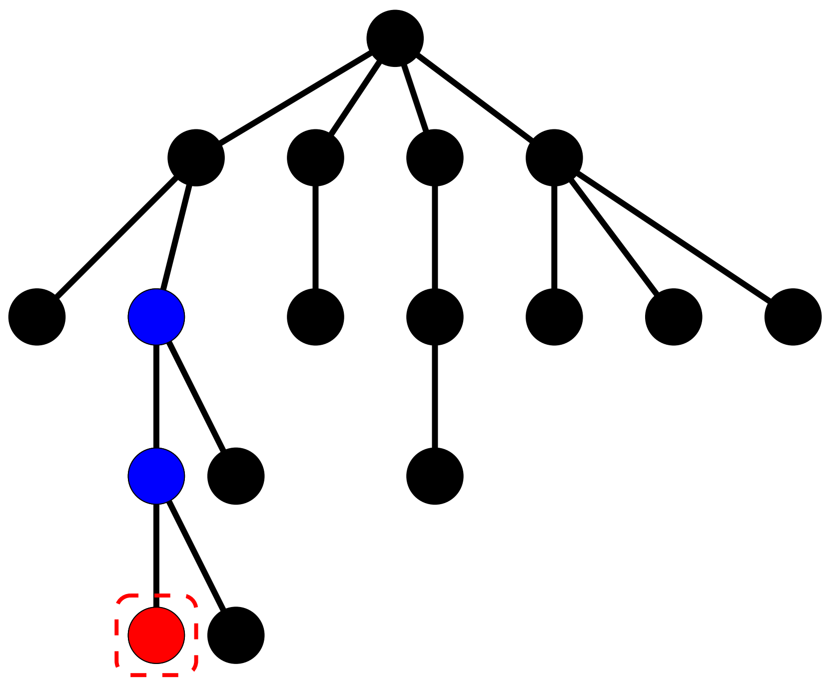
The Robber and Cops game (cont.)

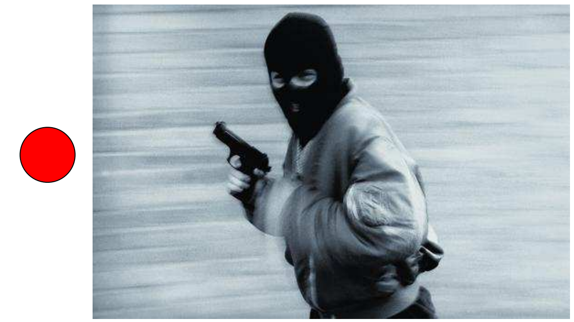
Example: 2 cops have a winning strategy in a tree.



The Robber and Cops game (cont.)

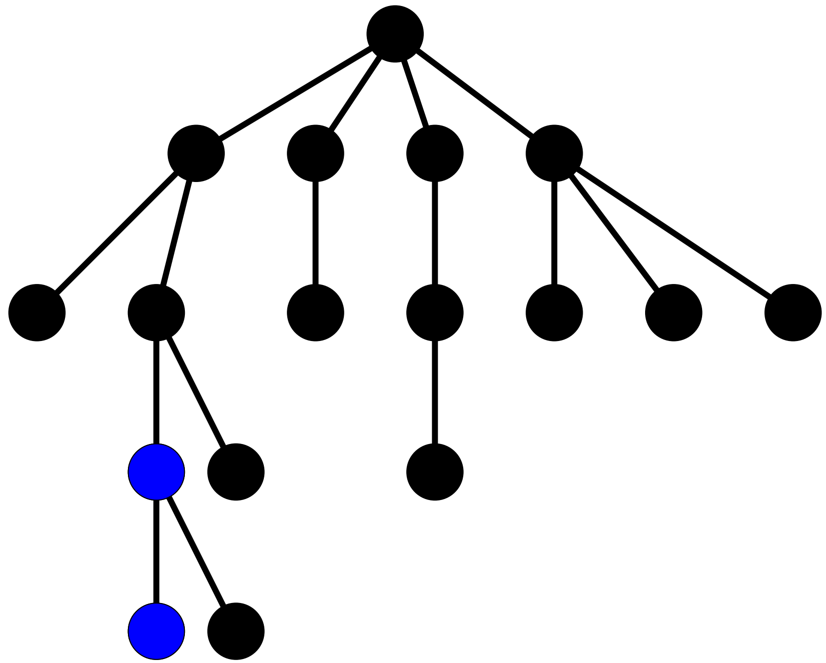
Example: 2 cops have a winning strategy in a tree.





The Robber and Cops game (cont.)

Example: 2 cops have a winning strategy in a tree.



Law enforcement on hypergraphs

Robber and Marshals Game:

Played on a hypergraph, a marshal can occupy an edge blocking all the vertices of the edge at the same time.

Theorem: [Adler et al. '05] k marshals can win the game if generalized hypertree width is $\leq k$, and they cannot win the game if generalized hypertree width is $\geq 3k+1$.

$\Rightarrow n^{O(k)}$ algorithm for approximating generalized hypertree width:

Theorem: [Adler et al. '05] There is an $n^{O(k)}$ time algorithm that constructs a generalized hypertree decomposition of width $\leq 3k$ if generalized hypertree width is $\leq k$.

Law enforcement on hypergraphs

Robber and Army Game:

A general has k battalions. A battalion can be divided arbitrarily, each part can be assigned to an edge. A vertex is blocked if it is covered by one full battalion.

Theorem: k battalions can win the game if fractional hypertree width is $\leq k$, and they cannot win the game if fractional hypertree width is $\geq 3k + 2$.

! We don't know how to turn this result into an algorithm (there are too many army positions).

Law enforcement on hypergraphs

Robber and Army Game:

A general has k battalions. A battalion can be divided arbitrarily, each part can be assigned to an edge. A vertex is blocked if it is covered by one full battalion.

Theorem: k battalions can win the game if fractional hypertree width is $\leq k$, and they cannot win the game if fractional hypertree width is $\geq 3k+2$.

! We don't know how to turn this result into an algorithm (there are too many army positions).

But maybe not so many:

Conjecture: If hypergraph H has fractional hypertree width w, then for every $r \leq w$ there are at most $|V(H)+E(H)|^{O(w)}$ maximal r-covered sets. Furthermore, there is a polynomial-time algorithm that enumerates all these sets.

Dichotomy?

Given a class of hypergraphs $\mathcal{H}$, CSP($\mathcal{H}$) is the problem restricted to instances with hypergraphs in $\mathcal{H}$.

Holy Grail: Determine all those classes of hypergraphs that make CSP($\mathcal{H}$) polynomial-time solvable.

- Is there a hypergraph property more general than bounded fractional hypertree width that makes CSP polynomial-time solvable?
- Is it possible to show that there is no polynomial-time algorithm for CSP($\mathcal{H}$) if $\mathcal{H}$ has unbounded fractional hypertree width? (modulo some comlexity-theoretic assumption)

Dichotomy?

Given a class of hypergraphs $\mathcal{H}$, CSP($\mathcal{H}$) is the problem restricted to instances with hypergraphs in $\mathcal{H}$.

Holy Grail: Determine all those classes of hypergraphs that make CSP($\mathcal{H}$) polynomial-time solvable.

- Is there a hypergraph property more general than bounded fractional hypertree width that makes CSP polynomial-time solvable?
- Is it possible to show that there is no polynomial-time algorithm for CSP($\mathcal{H}$) if $\mathcal{H}$ has unbounded fractional hypertree width? (modulo some comlexity-theoretic assumption)

Theorem: [Grohe '03] If $\mathcal{H}$ has bounded edge size, then

CSP($\mathcal{H}$) is polynomial-time solvable $\iff$ $\mathcal{H}$ has bounded tree width

(assuming FPT $\neq$ W[1]).

Conclusions

- CSP where constraints are represented as lists of satisfying tuples.
- Bounded tree width and bounded hypertree width make the problem polynomial-time solvable.
- **New:** Bounded fractional edge cover number.
- **New:** fractional hypertree width.
- **Open:** finding fractional hypertree decompositions.
- Robber and Army Game: equivalent to fractional hypertree width (up to a constant factor).
- **Open:** Are there other classes of hypergraphs where CSP is easy? Can we prove that bounded fractional hypertree width is best possible?

Conclusions