Exercise 5: Tree width and Hypertree width

Database Theory

2020-05-11

Maximilian Marx, David Carral

Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

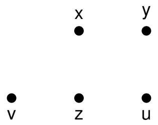
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.



Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

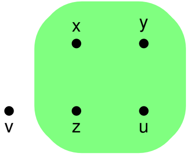
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.



Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

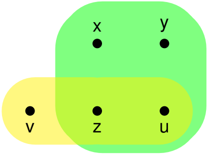
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \land \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \land \mathrm{b}(y, z) \land \mathrm{c}(z, u) \land \mathrm{d}(u, v) \land \mathrm{e}(v, z) \land \mathrm{f}(z, x) \land \mathrm{d}(x, u) \land \mathrm{d}(u, y)\big)$

Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.



Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

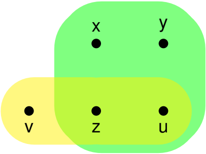
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

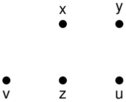
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

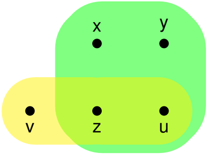
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

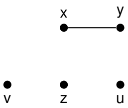
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

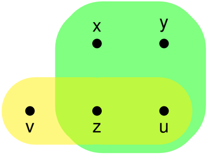
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \land \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \land \mathrm{b}(y, z) \land \mathrm{c}(z, u) \land \mathrm{d}(u, v) \land \mathrm{e}(v, z) \land \mathrm{f}(z, x) \land \mathrm{d}(x, u) \land \mathrm{d}(u, y)\big)$

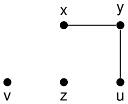
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

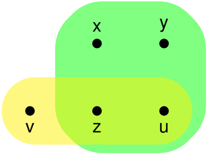
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

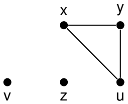
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

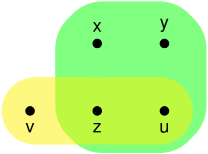
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \land \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \land \mathrm{b}(y, z) \land \mathrm{c}(z, u) \land \mathrm{d}(u, v) \land \mathrm{e}(v, z) \land \mathrm{f}(z, x) \land \mathrm{d}(x, u) \land \mathrm{d}(u, y)\big)$

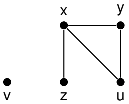
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

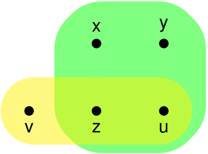
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \land \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \land \mathrm{b}(y, z) \land \mathrm{c}(z, u) \land \mathrm{d}(u, v) \land \mathrm{e}(v, z) \land \mathrm{f}(z, x) \land \mathrm{d}(x, u) \land \mathrm{d}(u, y)\big)$

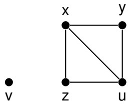
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

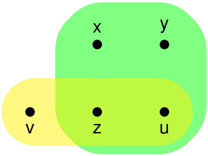
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

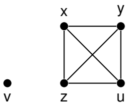
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

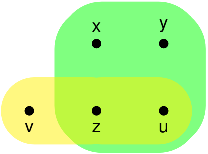
1. $\exists x, y, z, u, v.\ \big(\mathsf{r}(x, y, z, u) \wedge \mathsf{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v.\ \big(\mathsf{a}(x, y) \wedge \mathsf{b}(y, z) \wedge \mathsf{c}(z, u) \wedge \mathsf{d}(u, v) \wedge \mathsf{e}(v, z) \wedge \mathsf{f}(z, x) \wedge \mathsf{d}(x, u) \wedge \mathsf{d}(u, y)\big)$

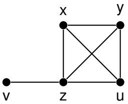
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

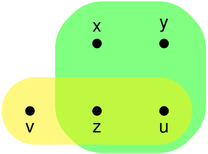
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

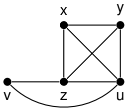
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

Solution.

1.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

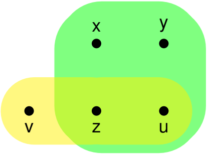
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

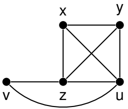
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

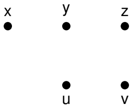
Solution.

1.





2.



Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

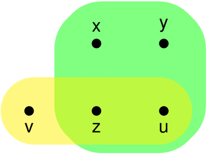
1. $\exists x, y, z, u, v. \big(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v. \big(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\big)$

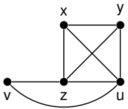
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

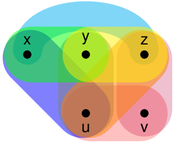
Solution.

1.





2.



Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

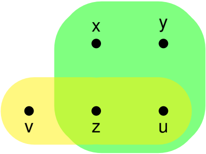
1. $\exists x, y, z, u, v.\ \big(\mathrm{r}(x, y, z, u) \land \mathrm{s}(z, u, v)\big)$
2. $\exists x, y, z, u, v.\ \big(\mathrm{a}(x, y) \land \mathrm{b}(y, z) \land \mathrm{c}(z, u) \land \mathrm{d}(u, v) \land \mathrm{e}(v, z) \land \mathrm{f}(z, x) \land \mathrm{d}(x, u) \land \mathrm{d}(u, y)\big)$

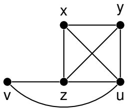
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

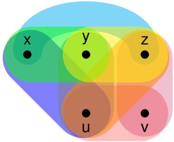
Solution.

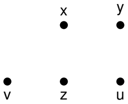
1.





2.





Exercise 1

Exercise. Construct the query hypergraph and the primal graph for the following queries:

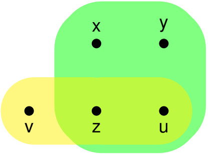
1. $\exists x, y, z, u, v. \left(\mathrm{r}(x, y, z, u) \wedge \mathrm{s}(z, u, v)\right)$
2. $\exists x, y, z, u, v. \left(\mathrm{a}(x, y) \wedge \mathrm{b}(y, z) \wedge \mathrm{c}(z, u) \wedge \mathrm{d}(u, v) \wedge \mathrm{e}(v, z) \wedge \mathrm{f}(z, x) \wedge \mathrm{d}(x, u) \wedge \mathrm{d}(u, y)\right)$

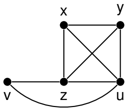
Definition (Lecture 6, Slide 23)

The *primal graph* of a hypergraph G is the undirected graph with the same vertices as G, and an edge connecting two vertices if there is some hyperedge in G that contains these two vertices.

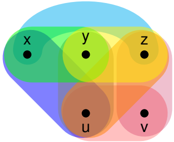
Solution.

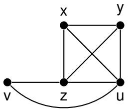
1.





2.

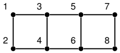




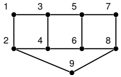
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

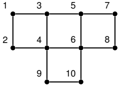
1.



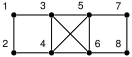
2.



3.



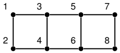
4.



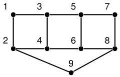
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

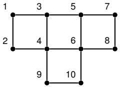
1\.



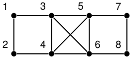
2\.



3\.



4\.

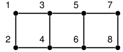


Solution.

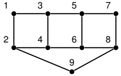
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

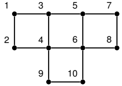
1.



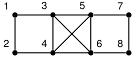
2.



3.

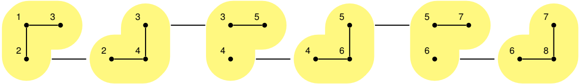


4.



Solution.

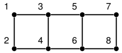
1.



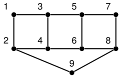
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

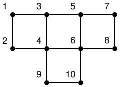
1.



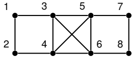
2.



3.

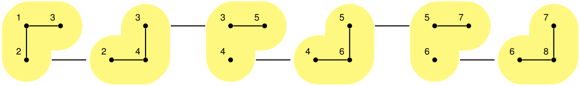


4.



Solution.

1.

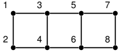


⇝ tree width 2.

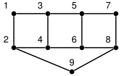
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

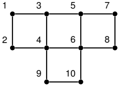
1.



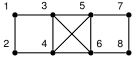
2.



3.

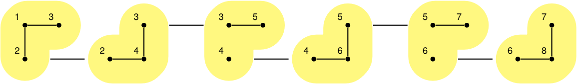


4.



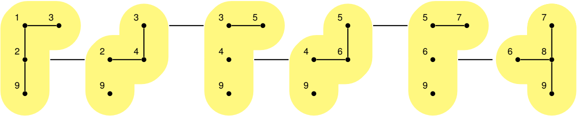
Solution.

1.



⇝ tree width 2.

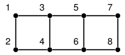
2.



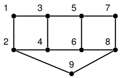
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

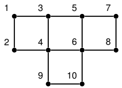
1.



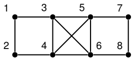
2.



3.

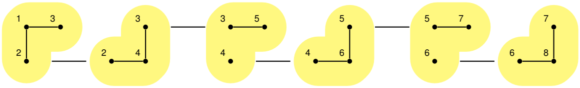


4.



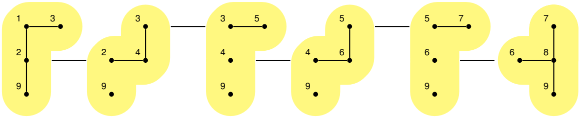
Solution.

1.



⇝ tree width 2.

2.

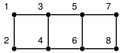


⇝ tree width 3.

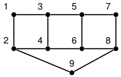
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

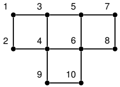
1.



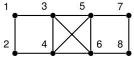
2.



3.

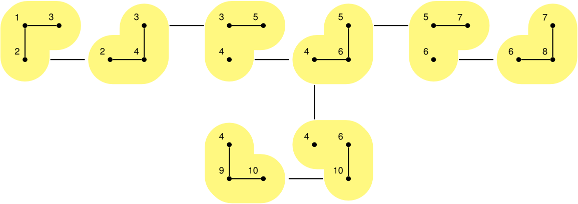


4.



Solution.

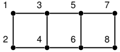
3.



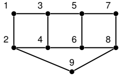
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

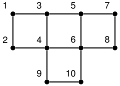
1.



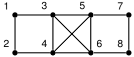
2.



3.

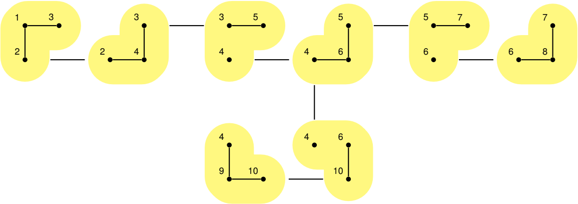


4.



Solution.

3.

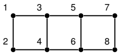


⇝ tree width 2.

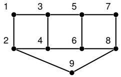
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

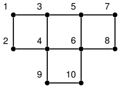
1.



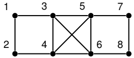
2.



3.

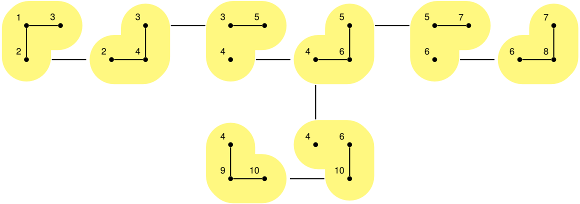


4.



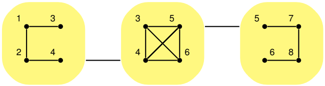
Solution.

3.



⇝ tree width 2.

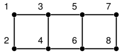
4.



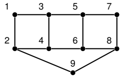
Exercise 2

Exercise. Determine the tree width of each of the following graphs and provide a suitable tree decomposition. Argue why there cannot be a tree decomposition of smaller width.

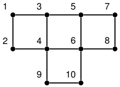
1.



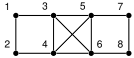
2.



3.

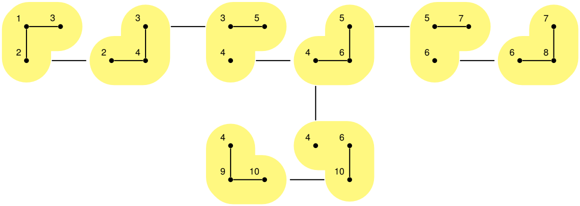


4.



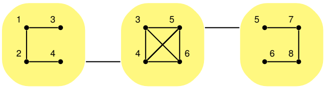
Solution.

3.



⇝ tree width 2.

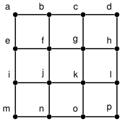
4.



⇝ tree width 3.

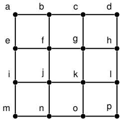
Exercise 3

Exercise. Show that the $n \times n$ grid has a tree width $\leq n$ by finding a suitable tree decomposition of width n. For example, the following 4×4 grid has tree width 4:



Exercise 3

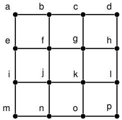
Exercise. Show that the $n \times n$ grid has a tree width $\leq n$ by finding a suitable tree decomposition of width n. For example, the following 4×4 grid has tree width 4:



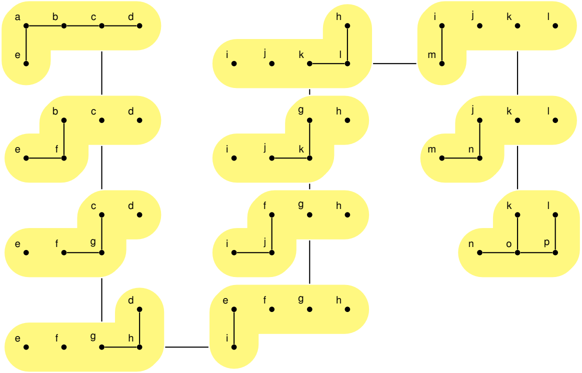
Solution.

Exercise 3

Exercise. Show that the $n \times n$ grid has a tree width $\leq n$ by finding a suitable tree decomposition of width n. For example, the following 4×4 grid has tree width 4:



Solution.



Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ *iff* k *cops have a winning strategy in the cops & robber game on G.*

Solution.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Solution.

- Consider an n-clique.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ *iff* k *cops have a winning strategy in the cops & robber game on G.*

Solution.

- Consider an n-clique.
- Clearly, n cops have a winning strategy: they can occupy every vertex.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Solution.

- Consider an n-clique.
- Clearly, n cops have a winning strategy: they can occupy every vertex.
- Thus, an n-clique has tree width at most $n - 1$.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Solution.

- Consider an n-clique.
- Clearly, n cops have a winning strategy: they can occupy every vertex.
- Thus, an n-clique has tree width at most $n - 1$.
- Consider the cops & robber game with $n - 1$ cops.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Solution.

- Consider an n-clique.
- Clearly, n cops have a winning strategy: they can occupy every vertex.
- Thus, an n-clique has tree width at most $n - 1$.
- Consider the cops & robber game with $n - 1$ cops.
- Every vertex has $n - 1$ neighbours.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Solution.

- Consider an n-clique.
- Clearly, n cops have a winning strategy: they can occupy every vertex.
- Thus, an n-clique has tree width at most $n - 1$.
- Consider the cops & robber game with $n - 1$ cops.
- Every vertex has $n - 1$ neighbours.
- While the cops can occupy all neighbouring vertices, they cannot catch the robber: if they move to the robbers position, one of the neighbouring vertices becomes free.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Solution.

- Consider an n-clique.
- Clearly, n cops have a winning strategy: they can occupy every vertex.
- Thus, an n-clique has tree width at most $n - 1$.
- Consider the cops & robber game with $n - 1$ cops.
- Every vertex has $n - 1$ neighbours.
- While the cops can occupy all neighbouring vertices, they cannot catch the robber: if they move to the robbers position, one of the neighbouring vertices becomes free.
- Thus, the robber wins if there are at most $n - 1$ cops.

Exercise 4

Exercise. Show that a clique (fully connected graph) of size n has tree width $n - 1$.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Solution.

- Consider an n-clique.
- Clearly, n cops have a winning strategy: they can occupy every vertex.
- Thus, an n-clique has tree width at most $n - 1$.
- Consider the cops & robber game with $n - 1$ cops.
- Every vertex has $n - 1$ neighbours.
- While the cops can occupy all neighbouring vertices, they cannot catch the robber: if they move to the robbers position, one of the neighbouring vertices becomes free.
- Thus, the robber wins if there are at most $n - 1$ cops.
- Hence the n-clique cannot have tree width $\leq n - 2$.

Exercise 5

Exercise. Recall that a graph is 3-colourable if one can assign three colours to its vertices in such a way that neighbouring vertices never share the same colour. Let C_3 be the set of all 3-colourable graphs. Are the graphs in C_3 of bounded or unbounded tree width? Explain your answer.

Exercise 5

Exercise. Recall that a graph is 3-colourable if one can assign three colours to its vertices in such a way that neighbouring vertices never share the same colour. Let C_3 be the set of all 3-colourable graphs. Are the graphs in C_3 of bounded or unbounded tree width? Explain your answer.
Solution.

Exercise 5

Exercise. Recall that a graph is 3-colourable if one can assign three colours to its vertices in such a way that neighbouring vertices never share the same colour. Let C_3 be the set of all 3-colourable graphs. Are the graphs in C_3 of bounded or unbounded tree width? Explain your answer.

Solution.

- Any $n \times n$ grid is 2-colourable.

Exercise 5

Exercise. Recall that a graph is 3-colourable if one can assign three colours to its vertices in such a way that neighbouring vertices never share the same colour. Let C_3 be the set of all 3-colourable graphs. Are the graphs in C_3 of bounded or unbounded tree width? Explain your answer.

Solution.

- Any $n \times n$ grid is 2-colourable.
- Hence, C_3 contains all grids.

Exercise 5

Exercise. Recall that a graph is 3-colourable if one can assign three colours to its vertices in such a way that neighbouring vertices never share the same colour. Let C_3 be the set of all 3-colourable graphs. Are the graphs in C_3 of bounded or unbounded tree width? Explain your answer.

Solution.

- Any $n \times n$ grid is 2-colourable.
- Hence, C_3 contains all grids.
- Grids have unbounded tree width.

Exercise 5

Exercise. Recall that a graph is 3-colourable if one can assign three colours to its vertices in such a way that neighbouring vertices never share the same colour. Let C_3 be the set of all 3-colourable graphs. Are the graphs in C_3 of bounded or unbounded tree width? Explain your answer.

Solution.

- Any $n \times n$ grid is 2-colourable.
- Hence, C_3 contains all grids.
- Grids have unbounded tree width.
- Thus, C_3 contains graphs of unbounded tree width.

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Theorem (Lecture 8, Slide 17)

A hypergraph H is of hypertree width $\leq k$ iff k marshals have a winning strategy in the marshals & robber game on H.

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ *iff* k *cops have a winning strategy in the cops & robber game on G.*

Theorem (Lecture 8, Slide 17)

A hypergraph H is of hypertree width $\leq k$ *iff* k *marshals have a winning strategy in the marshals & robber game on H.*

Solution.

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Theorem (Lecture 8, Slide 17)

A hypergraph H is of hypertree width $\leq k$ iff k marshals have a winning strategy in the marshals & robber game on H.

Solution.

1. True: cops don't walk along edges, so deleting edges does not invalidate winning strategies.

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Theorem (Lecture 8, Slide 17)

A hypergraph H is of hypertree width $\leq k$ iff k marshals have a winning strategy in the marshals & robber game on H.

Solution.

1. True: cops don't walk along edges, so deleting edges does not invalidate winning strategies.
2. True: analogous.

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k - 1$ iff k cops have a winning strategy in the cops & robber game on G.

Theorem (Lecture 8, Slide 17)

A hypergraph H is of hypertree width $\leq k$ iff k marshals have a winning strategy in the marshals & robber game on H.

Solution.

1. True: cops don't walk along edges, so deleting edges does not invalidate winning strategies.
2. True: analogous.
3. False: Consider a hypergraph that has a hyperedge containing all vertices. Then the hypergraph is acyclic (i.e., has hypertree width 1), but removing the hyperedge may result in a cyclic hypergraph (i.e., hypertree width > 1).

Exercise 6

Exercise. Decide whether the following claims are true or false. Explain your answer.

1. Deleting an edge from a graph may make the tree width smaller but never larger.
2. Deleting a vertex from a graph (and removing all of its edges) may make the tree width smaller but never larger.
3. Deleting a hyperedge from a hypergraph may make the hypertree width smaller but never larger.
4. Deleting a vertex from a hypergraph (and removing empty edges) may make the hypertree width smaller but never larger.

Theorem (Seymour and Thomas; Lecture 7, Slide 15)

A graph G is of tree width $\leq k-1$ iff k cops have a winning strategy in the cops & robber game on G.

Theorem (Lecture 8, Slide 17)

A hypergraph H is of hypertree width $\leq k$ iff k marshals have a winning strategy in the marshals & robber game on H.

Solution.

1. True: cops don't walk along edges, so deleting edges does not invalidate winning strategies.
2. True: analogous.
3. False: Consider a hypergraph that has a hyperedge containing all vertices. Then the hypergraph is acyclic (i.e., has hypertree width 1), but removing the hyperedge may result in a cyclic hypergraph (i.e., hypertree width > 1).
4. True: marshals don't occupy vertices, but hyperedges, so deleting vertices does not invalidate winning strategies.

Exercise 7

Exercise. The following BCQ corresponds to graph (a) in Exercise 2:

$$\exists x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8.\ \mathrm{r}(x_1, x_2) \wedge \mathrm{r}(x_1, x_3) \wedge \mathrm{r}(x_2, x_4) \wedge \mathrm{r}(x_3, x_4) \wedge \mathrm{r}(x_3, x_5) \wedge \\ \mathrm{r}(x_4, x_6) \wedge \mathrm{r}(x_5, x_6) \wedge \mathrm{r}(x_5, x_7) \wedge \mathrm{r}(x_6, x_8) \wedge \mathrm{r}(x_7, x_8)$$

According to the logical characterisation from the lecture, this query can be expressed in the $\exists$-$\wedge$-fragment of FO using only tree width$+1$ variables. Find such a formula.

Exercise 7

Exercise. The following BCQ corresponds to graph (a) in Exercise 2:

$$\exists x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8.\ \mathsf{r}(x_1, x_2) \wedge \mathsf{r}(x_1, x_3) \wedge \mathsf{r}(x_2, x_4) \wedge \mathsf{r}(x_3, x_4) \wedge \mathsf{r}(x_3, x_5) \wedge \mathsf{r}(x_4, x_6) \wedge \mathsf{r}(x_5, x_6) \wedge \mathsf{r}(x_5, x_7) \wedge \mathsf{r}(x_6, x_8) \wedge \mathsf{r}(x_7, x_8)$$

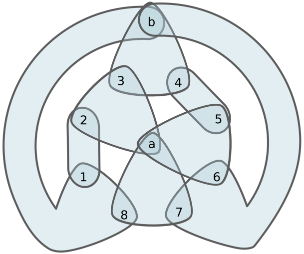
According to the logical characterisation from the lecture, this query can be expressed in the $\exists$-$\wedge$-fragment of FO using only tree width$+1$ variables. Find such a formula.

Solution.

$$\begin{aligned}
&\exists x, y, z.\ \mathsf{r}(x, y) \wedge \mathsf{r}(x, z) \wedge \\
&\quad \big(\exists x.\ \mathsf{r}(y, x) \wedge \mathsf{r}(z, x) \wedge \\
&\quad \big(\exists y.\ \mathsf{r}(z, y) \wedge \\
&\quad \big(\exists z.\ \mathsf{r}(x, z) \wedge \mathsf{r}(y, z) \wedge \\
&\quad \big(\exists x.\ \mathsf{r}(y, x) \wedge \\
&\quad \big(\exists y.\ \mathsf{r}(x, y) \wedge \mathsf{r}(z, y)\big)\big)\big)\big)\big)
\end{aligned}$$

Exercise 8

Exercise. Consider *Adler's Hypergraph*:



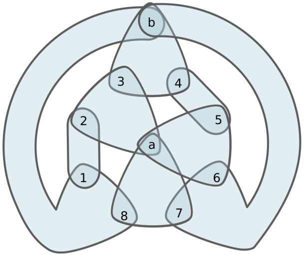
Play the marshals & robber game on this graph.

1. Can one marshal catch the robber?
2. Can two marshals catch the robber?
3. Can three marshals catch the robber?
4. Adler et al. [Eur. J. Comb., 2007] proposed this graph as an example where fewer marshals can win if they are allowed to play non-monotonically, that is, if they are not required to shrink the remaining space in each turn. Can you confirm her findings?

(∗) Can you explain why non-monotone play is unavoidable in one of the above cases if the marshals want to win?

Exercise 8

Exercise. Consider *Adler's Hypergraph*:



Play the marshals & robber game on this graph.

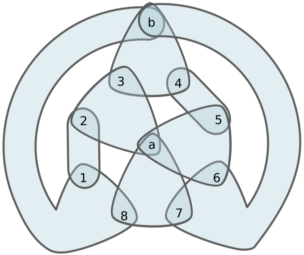
1. Can one marshal catch the robber?
2. Can two marshals catch the robber?
3. Can three marshals catch the robber?
4. Adler et al. [Eur. J. Comb., 2007] proposed this graph as an example where fewer marshals can win if they are allowed to play non-monotonically, that is, if they are not required to shrink the remaining space in each turn. Can you confirm her findings?

(∗) Can you explain why non-monotone play is unavoidable in one of the above cases if the marshals want to win?

Solution.

Exercise 8

Exercise. Consider *Adler's Hypergraph*:



Play the marshals & robber game on this graph.

1. Can one marshal catch the robber?
2. Can two marshals catch the robber?
3. Can three marshals catch the robber?
4. Adler et al. [Eur. J. Comb., 2007] proposed this graph as an example where fewer marshals can win if they are allowed to play non-monotonically, that is, if they are not required to shrink the remaining space in each turn. Can you confirm her findings?

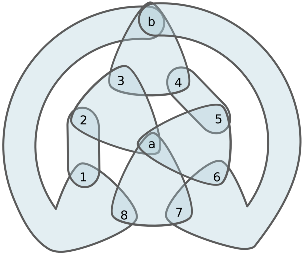
(∗) Can you explain why non-monotone play is unavoidable in one of the above cases if the marshals want to win?

Solution.

1. No.

Exercise 8

Exercise. Consider *Adler's Hypergraph*:



Play the marshals & robber game on this graph.

1. Can one marshal catch the robber?
2. Can two marshals catch the robber?
3. Can three marshals catch the robber?
4. Adler et al. [Eur. J. Comb., 2007] proposed this graph as an example where fewer marshals can win if they are allowed to play non-monotonically, that is, if they are not required to shrink the remaining space in each turn. Can you confirm her findings?

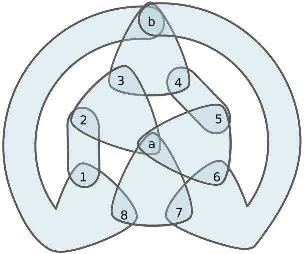
(∗) Can you explain why non-monotone play is unavoidable in one of the above cases if the marshals want to win?

Solution.

1. No.
2. Yes, but only non-monotonically.

Exercise 8

Exercise. Consider *Adler's Hypergraph*:



Play the marshals & robber game on this graph.

1. Can one marshal catch the robber?
2. Can two marshals catch the robber?
3. Can three marshals catch the robber?
4. Adler et al. [Eur. J. Comb., 2007] proposed this graph as an example where fewer marshals can win if they are allowed to play non-monotonically, that is, if they are not required to shrink the remaining space in each turn. Can you confirm her findings?

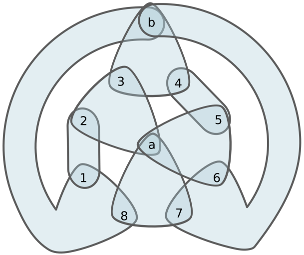
(∗) Can you explain why non-monotone play is unavoidable in one of the above cases if the marshals want to win?

Solution.

1. No.
2. Yes, but only non-monotonically.
3. Yes, even when playing monotonically.

Exercise 8

Exercise. Consider *Adler's Hypergraph*:



Play the marshals & robber game on this graph.

1. Can one marshal catch the robber?
2. Can two marshals catch the robber?
3. Can three marshals catch the robber?
4. Adler et al. [Eur. J. Comb., 2007] proposed this graph as an example where fewer marshals can win if they are allowed to play non-monotonically, that is, if they are not required to shrink the remaining space in each turn. Can you confirm her findings?

(∗) Can you explain why non-monotone play is unavoidable in one of the above cases if the marshals want to win?

Solution.

1. No.
2. Yes, but only non-monotonically.
3. Yes, even when playing monotonically.

(∗) The graph has hypertree width 3, but generalised hypertree width 2.