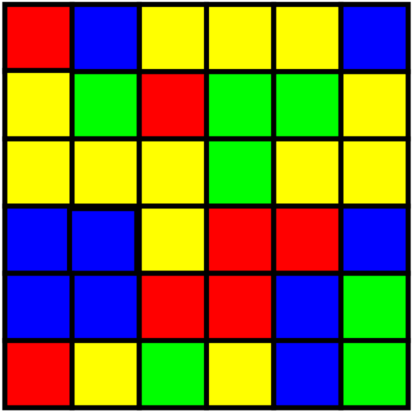
# Spanning trees and the complexity of flood-filling games

Kitty Meeks   Alex Scott

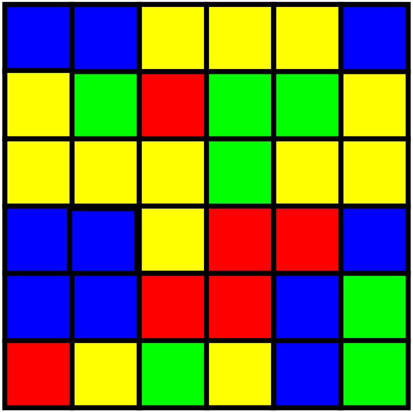
Mathematical Institute
University of Oxford

FUN 2012, Venice

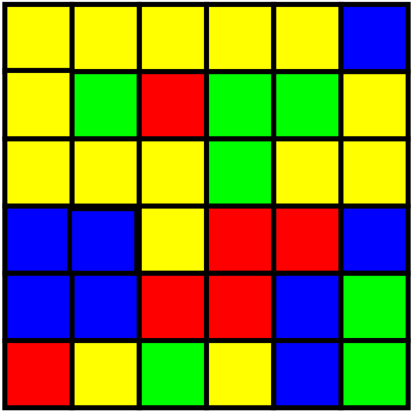
# The original Flood-It game



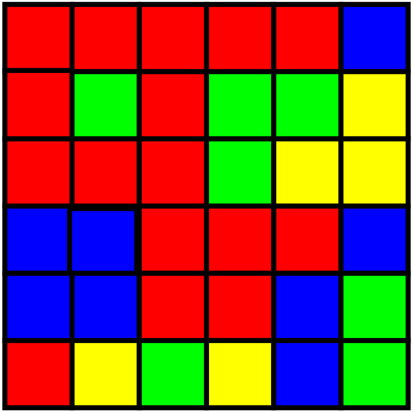
# The original Flood-It game



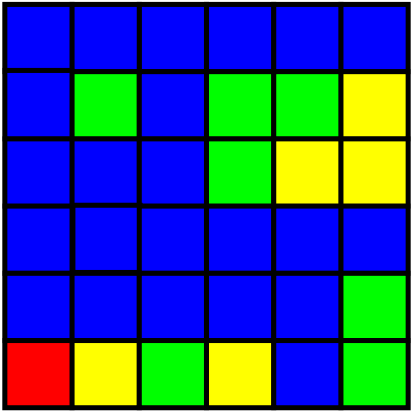
# The original Flood-It game



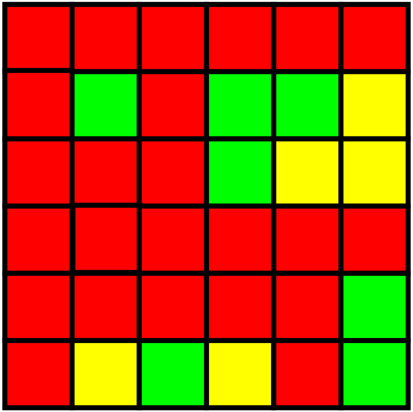
# The original Flood-It game



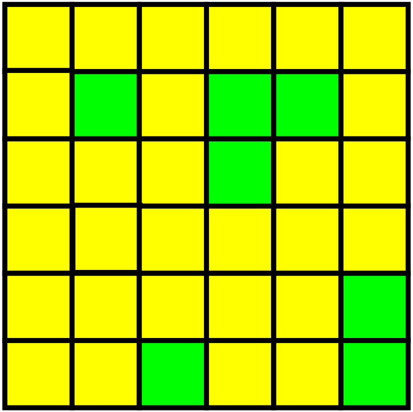
# The original Flood-It game



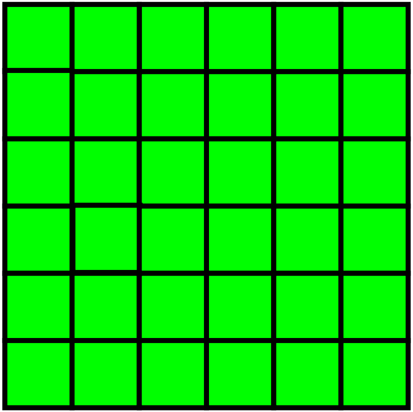
# The original Flood-It game



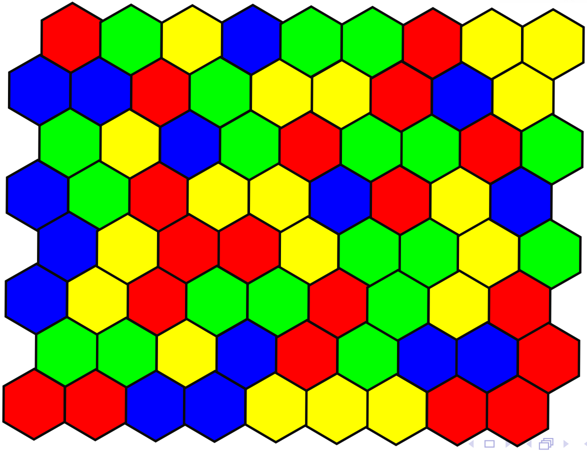
# The original Flood-It game



# The original Flood-It game



# The Honey-Bee game

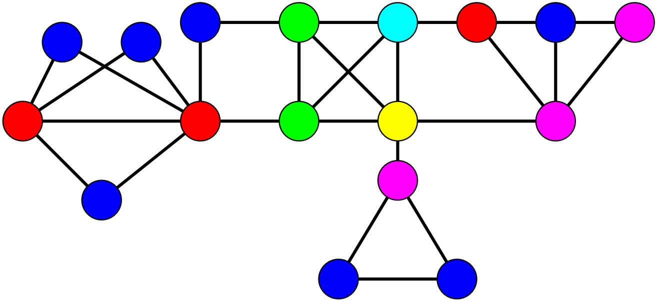


# Generalising to arbitrary graphs

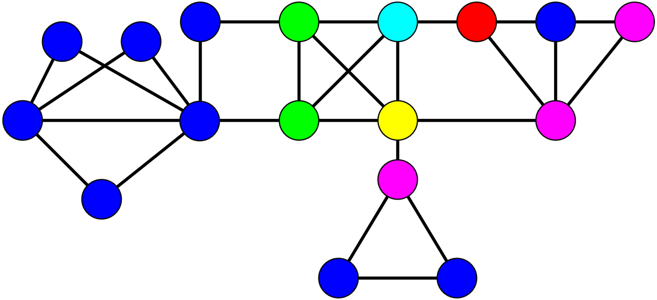
# Generalising to arbitrary graphs





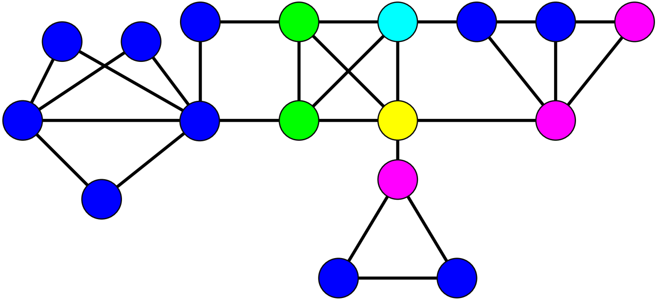
Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

# Generalising to arbitrary graphs



Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

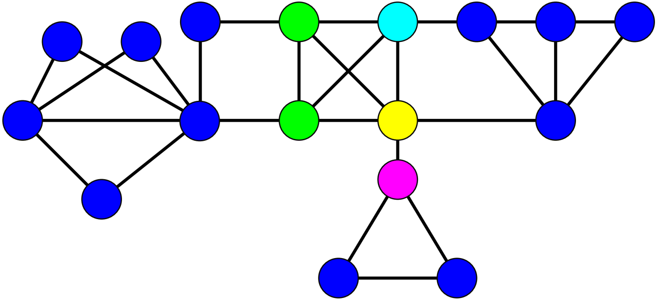
# Generalising to arbitrary graphs



Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

# Generalising to arbitrary graphs

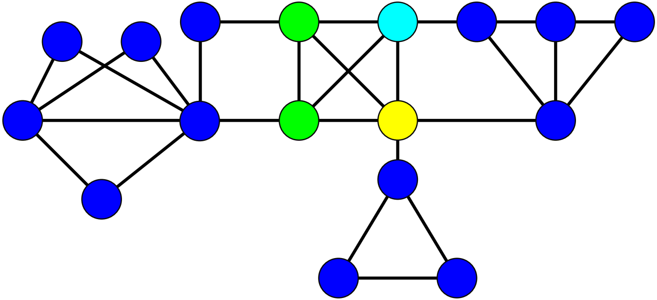
  



Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

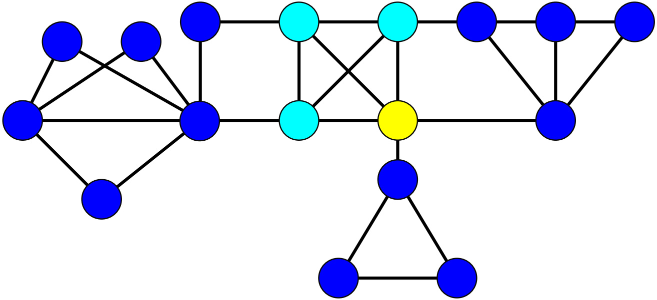
# Generalising to arbitrary graphs





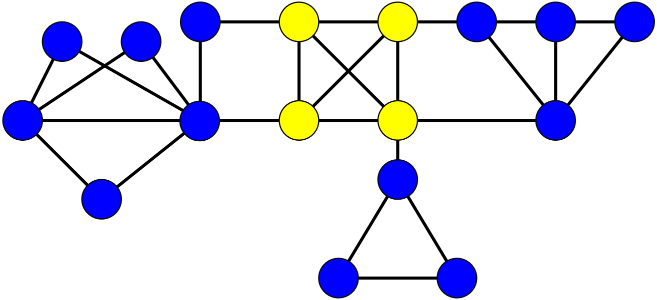
Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

# Generalising to arbitrary graphs



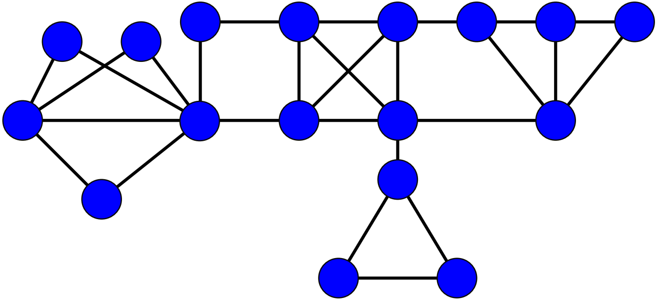
Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

# Generalising to arbitrary graphs



Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

# Generalising to arbitrary graphs



Sources: brighton-hove.dbprimary.com, englishclub.com, picgifs.com

# Existing results concerning Free Flood It

Given a coloured graph $G$, Free Flood It is the problem of determining the minimum number of moves required to flood $G$, when we are allowed to make moves anywhere in the graph.

# Existing results concerning Free Flood It

Given a coloured graph $G$, Free Flood It is the problem of determining the minimum number of moves required to flood $G$, when we are allowed to make moves anywhere in the graph.

Free Flood It is NP-hard when restricted to

- $n \times n$ or $3 \times n$ grids,

# Existing results concerning Free Flood It

Given a coloured graph $G$, Free Flood It is the problem of determining the minimum number of moves required to flood $G$, when we are allowed to make moves anywhere in the graph.

Free Flood It is NP-hard when restricted to

- $n \times n$ or $3 \times n$ grids,
- trees,

# Existing results concerning Free Flood It

Given a coloured graph $G$, Free Flood It is the problem of determining the minimum number of moves required to flood $G$, when we are allowed to make moves anywhere in the graph.
Free Flood It is NP-hard when restricted to

- $n \times n$ or $3 \times n$ grids,
- trees,
- series-parallel graphs.

# Existing results concerning FREE FLOOD IT

Given a coloured graph $G$, FREE FLOOD IT is the problem of determining the minimum number of moves required to flood $G$, when we are allowed to make moves anywhere in the graph.

FREE FLOOD IT is NP-hard when restricted to

- $n \times n$ or $3 \times n$ grids,
- trees,
- series-parallel graphs.

FREE FLOOD IT can be solved in polynomial time when restricted to

- paths
- cycles
- co-comparability graphs

# Existing results concerning FREE FLOOD IT

Given a coloured graph $G$, FREE FLOOD IT is the problem of determining the minimum number of moves required to flood $G$, when we are allowed to make moves anywhere in the graph.
FREE FLOOD IT is NP-hard when restricted to

- $n \times n$ or $3 \times n$ grids,
- trees,
- series-parallel graphs.

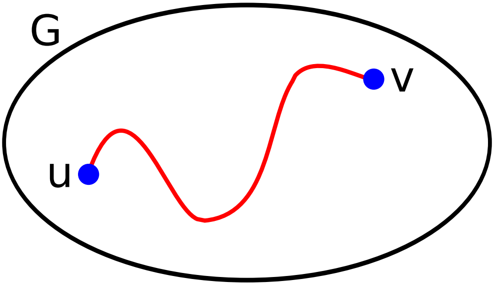
FREE FLOOD IT can be solved in polynomial time when restricted to

- paths
- cycles
- co-comparability graphs

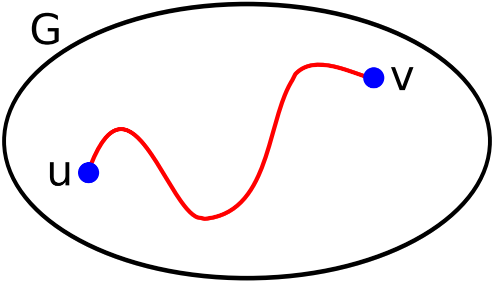
or

- if only two colours are used

# Connecting pairs of vertices



# Connecting pairs of vertices




Using this fact, we can compute in time $O(|V|^3|E|c^2)$ the number of moves required to connect any given pair of vertices in a graph $G = (V, E)$ coloured with $c$ colours.

# Spanning trees

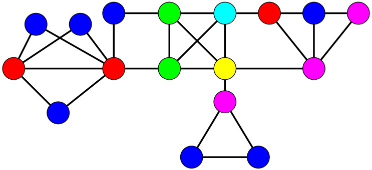
**Theorem**

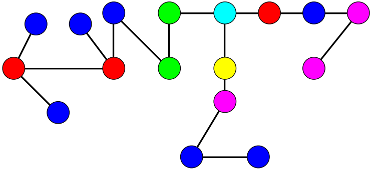
*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

# Spanning trees

**Theorem**

*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

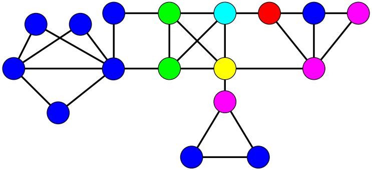


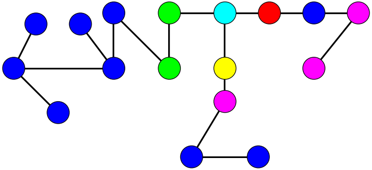


# Spanning trees

**Theorem**

*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

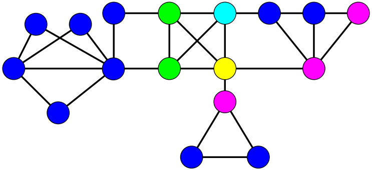


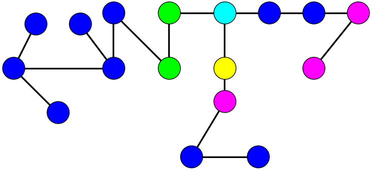


# Spanning trees

**Theorem**

*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

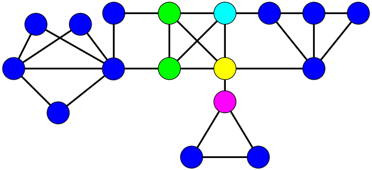


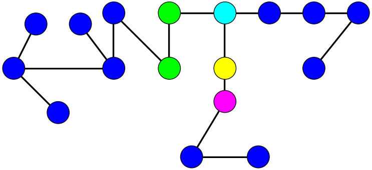


# Spanning trees

**Theorem**

*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

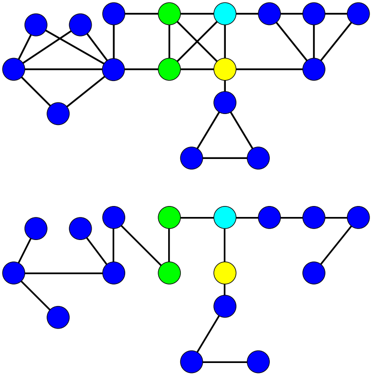




# Spanning trees

**Theorem**

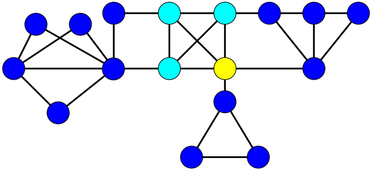
*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

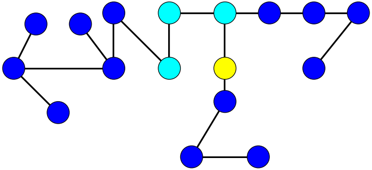


# Spanning trees

**Theorem**

*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

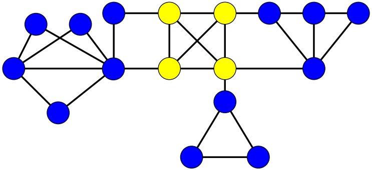


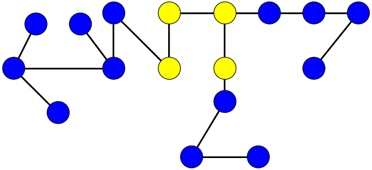


# Spanning trees

**Theorem**

*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*

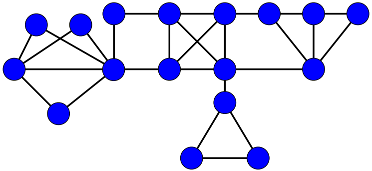


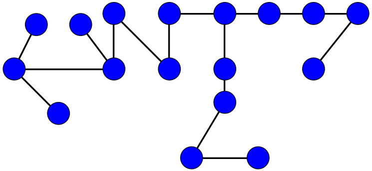


# Spanning trees

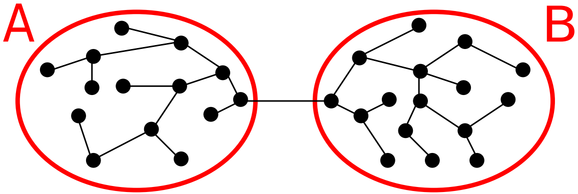
## Theorem

*The number of moves required to flood a coloured graph $G$ is equal to the minimum, taken over all spanning trees $T$ of $G$, of the number of moves required to flood $T$.*





# Proof of spanning trees result: key step




The number of moves required to flood $T$ with colour $d$ is at most the sum of the numbers of moves required to flood $A$ and $B$ respectively with colour $d$.

# This is useless!

- In general, a graph has an exponential number of spanning trees.

# This is useless!

- In general, a graph has an exponential number of spanning trees.
- Besides, FREE FLOOD IT is still NP-hard even on trees.

# ... or is it?

## … or is it?

$$P \neq NP$$

# ... or is it?

$$\cancel{P \neq NP}$$

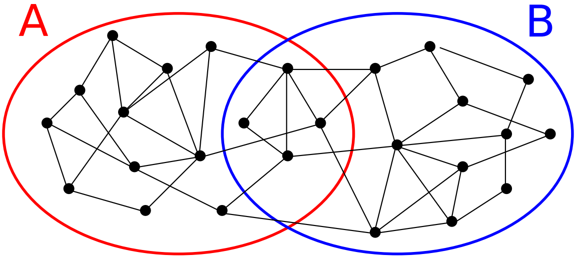
# ... or is it?

Source: finditinscotland.com

# ... or is it?

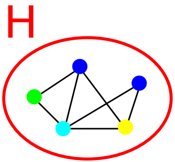
Source: finditinscotland.com

# ... or is it?



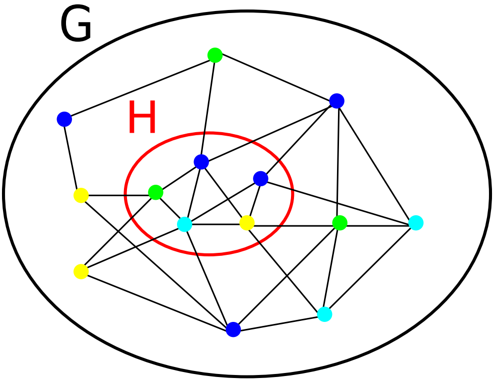

The number of moves required to flood $G$ with colour $d$ is at most the sum of the numbers of moves required to flood $A$ and $B$ respectively with colour $d$.

# ... or is it?




The number of moves required to flood a subgraph doesn't increase when we play in a larger graph.

# ... or is it?

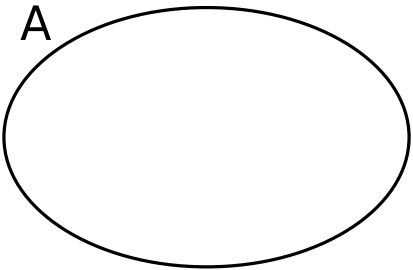



The number of moves required to flood a subgraph doesn't increase when we play in a larger graph.

# Application I: Graphs with polynomially many connected subgraphs

**Theorem**

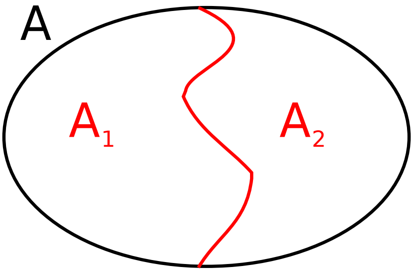
Free Flood It *can be solved in polynomial time on graphs that have only a polynomial number of connected subgraphs.*



# Application I: Graphs with polynomially many connected subgraphs

**Theorem**

FREE FLOOD IT *can be solved in polynomial time on graphs that have only a polynomial number of connected subgraphs.*



# Application I: Graphs with polynomially many connected subgraphs

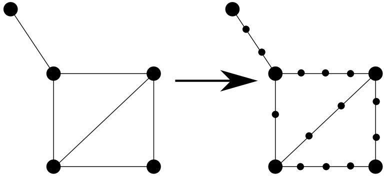
Classes of graphs with only a polynomial number of connected subgraphs include:

- paths
- cycles

# Application I: Graphs with polynomially many connected subgraphs

Classes of graphs with only a polynomial number of connected subgraphs include:

- paths
- cycles
- subdivisions of any fixed graph $H$



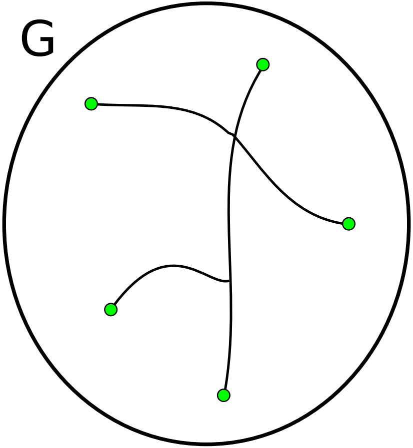
# Application II: Connecting $k$ points

Given a coloured graph $G$ and a subset $U$ of at most $k$ vertices, $k$-LINKING FLOOD IT is the problem of determining the number of moves required to create a single monochromatic component containing $U$.

**Theorem**

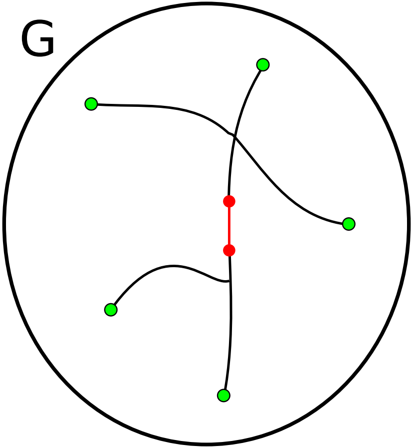
*$k$-LINKING FLOOD IT can be solved in time $O(|V|^{k+3}|E|c^2 2^k)$ on a graph $G = (V, E)$ coloured with $c$ colours.*

# Application II: Connecting $k$ points

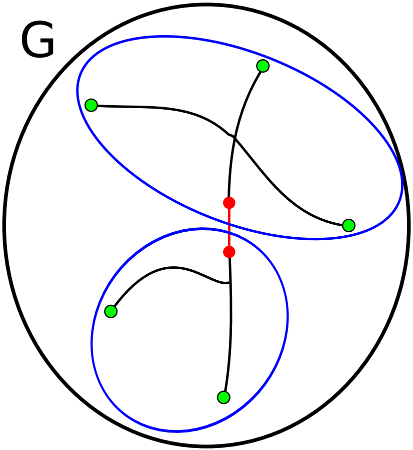



The number of moves required to connect $U$ is equal to the minimum, taken over all subtrees $T$ of $G$ that contain $U$, of the number of moves required to flood $T$.

# Application II: Connecting $k$ points



# Application II: Connecting $k$ points



# Conclusions

- We can analyse flood filling problems by considering only trees.

# Conclusions

- We can analyse flood filling problems by considering only trees.
- This allows us to prove nice complexity results:

# Conclusions

- We can analyse flood filling problems by considering only trees.
- This allows us to prove nice complexity results:
  - Free Flood It is solvable in polynomial time on graphs with polynomially many connected subgraphs.

# Conclusions

- We can analyse flood filling problems by considering only trees.
- This allows us to prove nice complexity results:
  - FREE FLOOD IT is solvable in polynomial time on graphs with polynomially many connected subgraphs.
  - $k$-LINKING FLOOD IT is solvable in polynomial time on arbitrary graphs (for fixed $k$).

# Open Problems

- Is $k$-Linking Flood It fixed parameter tractable, parameterised by $k$?

# Open Problems

- Is $k$-Linking Flood It fixed parameter tractable, parameterised by $k$?
- On what other minor-closed classes of trees is Free Flood It solvable in polynomial time?

# Open Problems

- Is $k$-Linking Flood It fixed parameter tractable, parameterised by $k$?
- On what other minor-closed classes of trees is Free Flood It solvable in polynomial time?
- Extremal problems...

# Open Problems

- Is $k$-Linking Flood It fixed parameter tractable, parameterised by $k$?
- On what other minor-closed classes of trees is Free Flood It solvable in polynomial time?
- Extremal problems...
- Does the Loch Ness Monster exist?

Thank you