Reachability Substitutes for Planar Digraphs

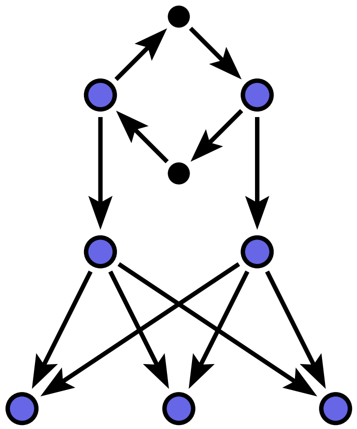
Martin Kutz

Max-Planck Institut für Informatik, Saarbrücken

Joint work with
Irit Katriel (MPII Saarbrücken) and
Martin Skutella (Universität Dortmund)

Reachability Substitutes

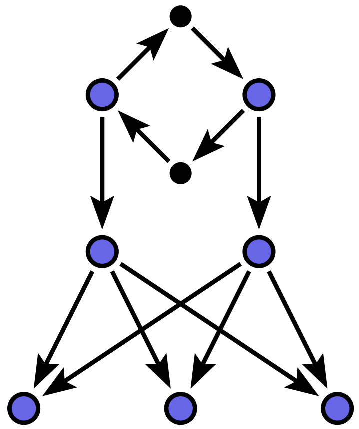
Given a digraph $G = (V, E)$ with a set of vertices U marked "interesting".



Reachability Substitutes

Given a digraph $G = (V, E)$ with a set of vertices U marked "interesting".

How efficiently can we represent the reachabilities in U?



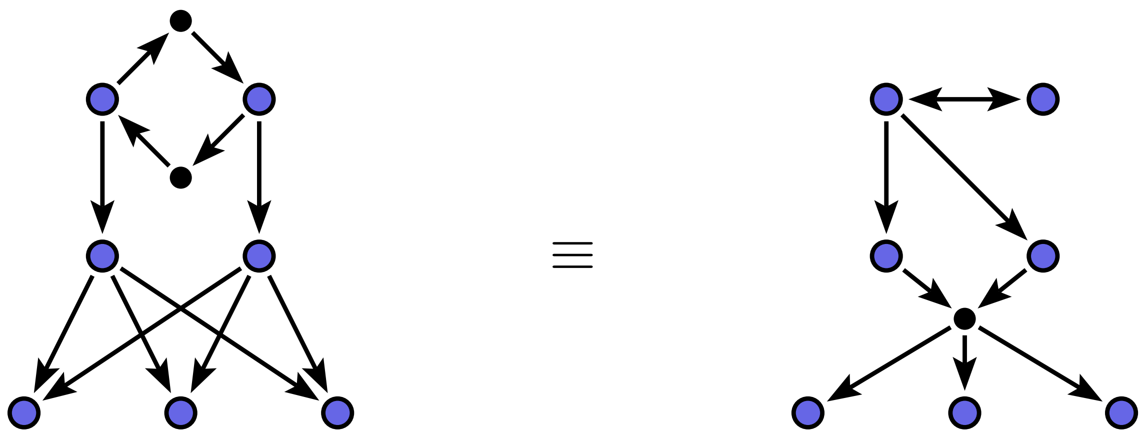
Reachability Substitutes

Given a digraph $G = (V, E)$ with a set of vertices U marked "interesting".

How efficiently can we represent the reachabilities in U?

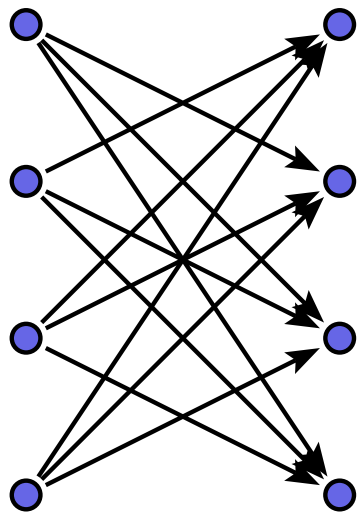
Def. Two digraphs $G = (V, E)$ and $G' = (V', E')$ are reachability substitutes for each other (w.r.t. U) if for all $u, v \in U \subseteq V, V'$:

$$u \overset{G}{\rightsquigarrow} v \quad \text{iff} \quad u \overset{G'}{\rightsquigarrow} v$$



Bad News

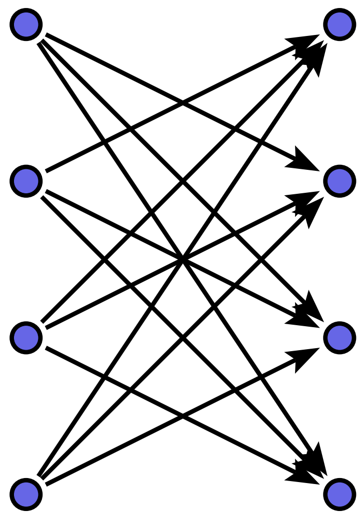
Theorem. Almost all digraphs with k interesting vertices have only RSs of size $\Omega(k^2/\log k)$.



Example:
$\vec{K}_{4,4}$ – matching
is incompressible

Bad News

Theorem. Almost all digraphs with k interesting vertices have only RSs of size $\Omega(k^2/\log k)$.



Example:
$\vec{K}_{4,4}$ – matching
is incompressible

Theorem. Finding a minimum RS (size = $|V| + |E|$) for a given digraph is NP-hard.

Planar Digraphs

How complex can planar reachabilities be?

Planar Digraphs

How complex can planarly induced reachabilities be?

Planar Digraphs

How complex can planarly induced reachabilities be?

Main Theorem. Any planar digraph $G = (V, E)$ with k interesting vertices has a reachability substitute of size $O(k \log^2 k)$.

Planar Digraphs

How complex can planarly induced reachabilities be?

Main Theorem. Any planar digraph $G = (V, E)$ with k interesting vertices has a reachability substitute of size $O(k \log^2 k)$.

Observe: bound in $k = |U|$, not $|V|$. (So Euler won't help.)
The containing/defining digraph G may be arbitrarily large!

Planar Digraphs

How complex can planarly induced reachabilities be?

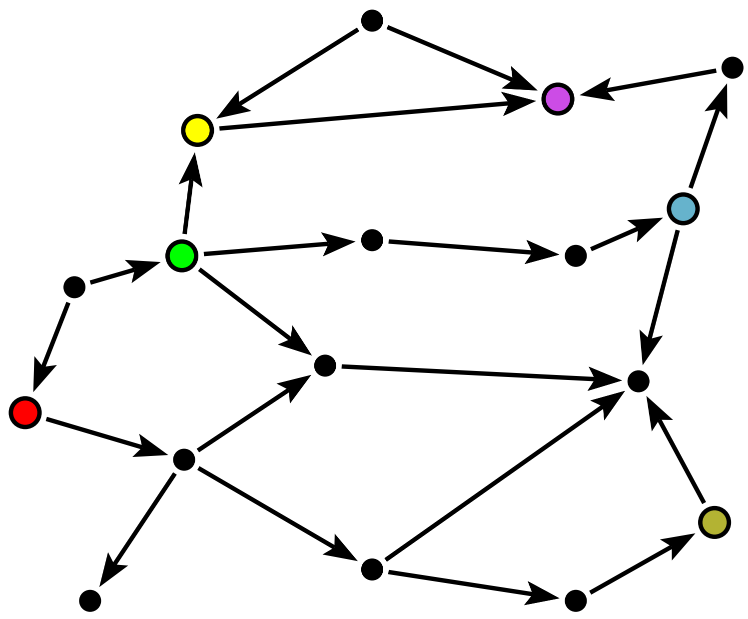
Main Theorem. Any planar digraph $G = (V, E)$ with k interesting vertices has a reachability substitute of size $O(k \log^2 k)$.

Observe: bound in $k = |U|$, not $|V|$. (So Euler won't help.)
The containing/defining digraph G may be arbitrarily large!

Previous result [Subramanian, 1993]: If all interesting vertices lie on a constant number of faces then there is a substitute of size $O(k \log k)$.

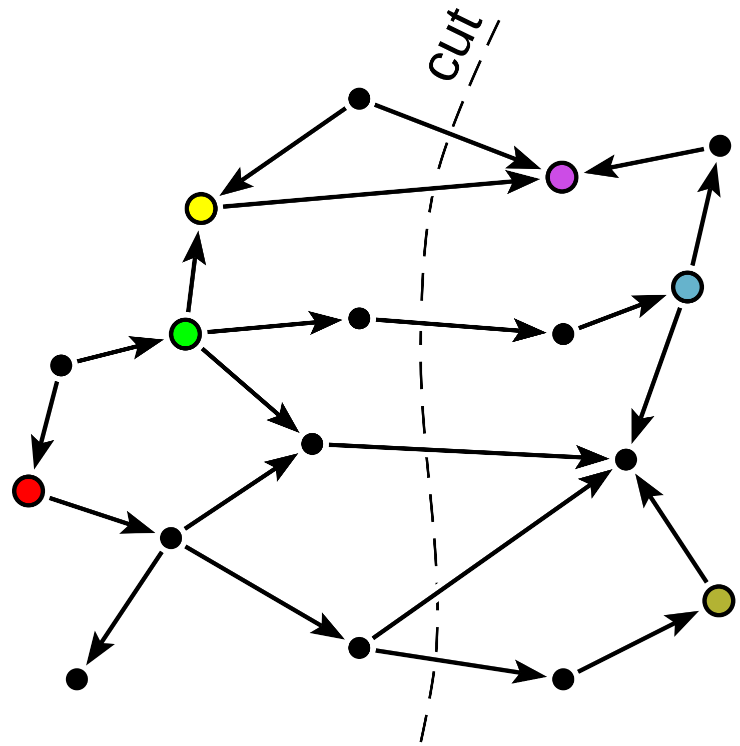
Tools & Techniques

- separation (balanced directed cuts)



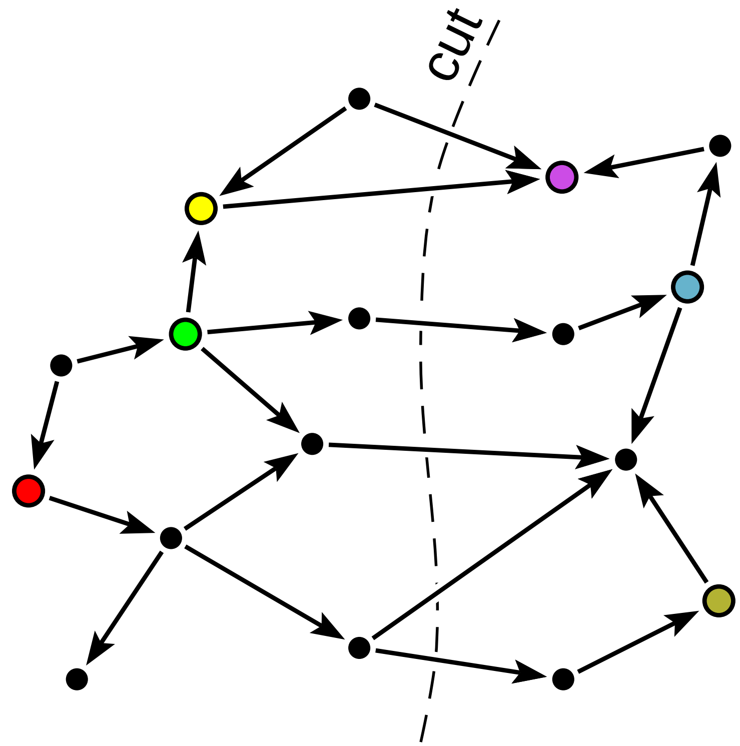
Tools & Techniques

- separation (balanced directed cuts)



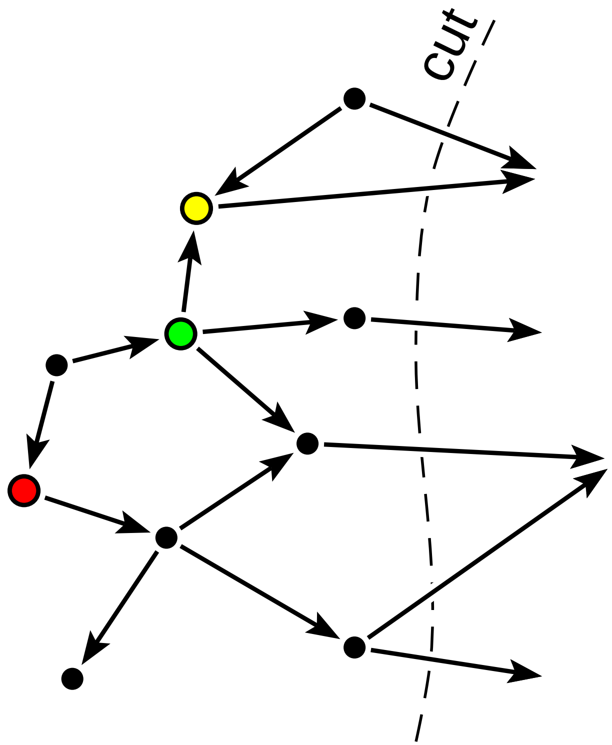
Tools & Techniques

- separation (balanced directed cuts)
- representing reachabilities to / from the cut



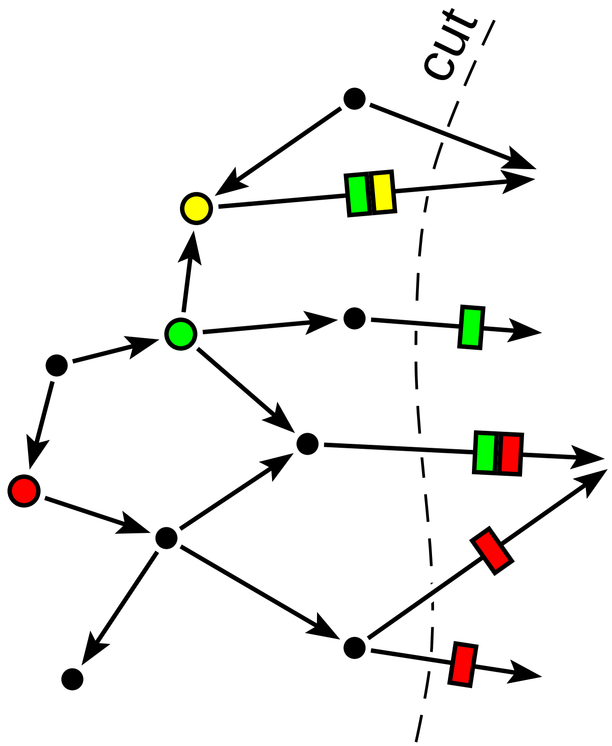
Tools & Techniques

- separation (balanced directed cuts)
- representing reachabilities to / from the cut



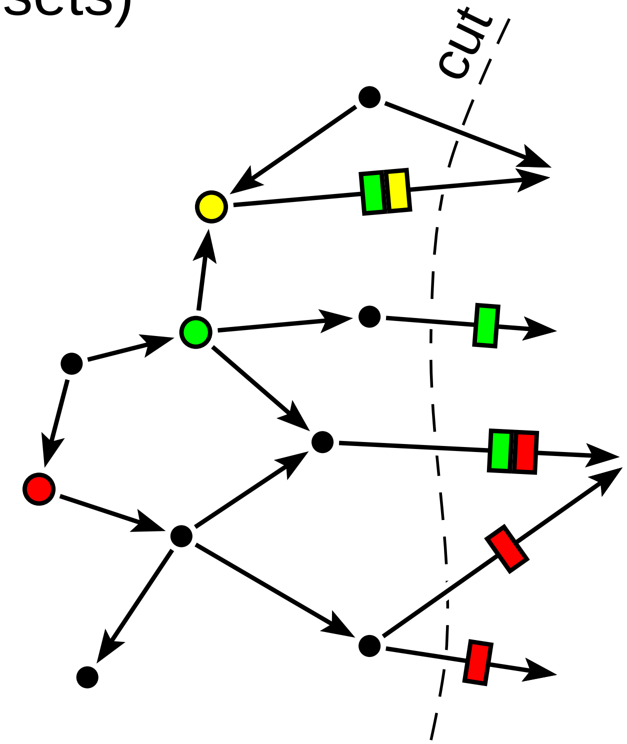
Tools & Techniques

- separation   (balanced directed cuts)
- representing reachabilities to / from the cut



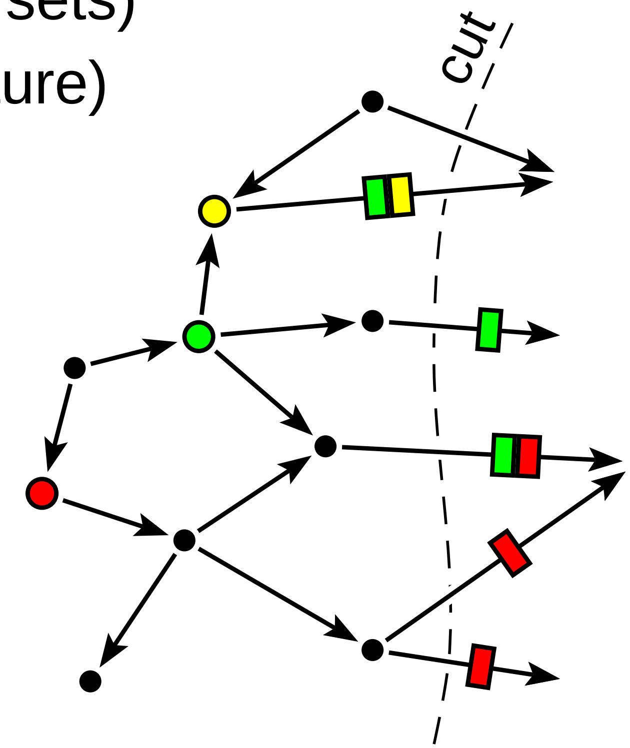
Tools & Techniques

- separation (balanced directed cuts)
- representing reachabilities to / from the cut
  - type bound (how many color sets)



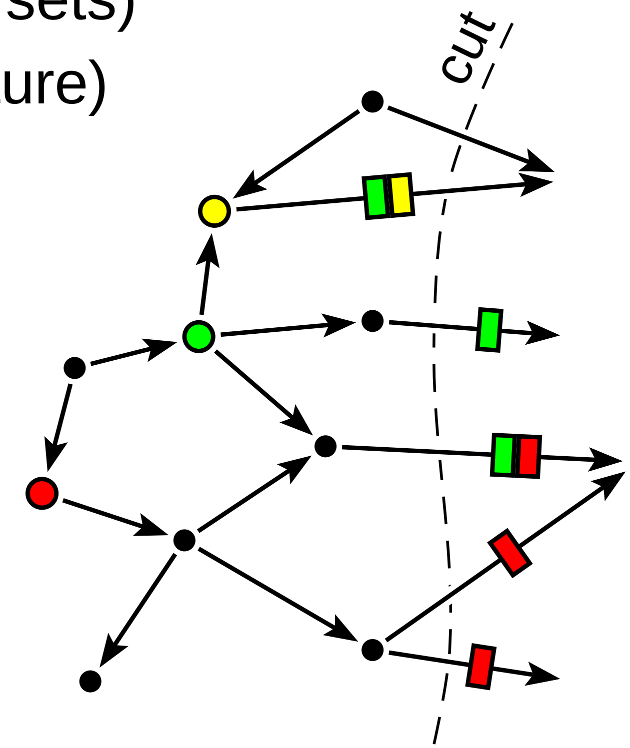
Tools & Techniques

- separation (balanced directed cuts)
- representing reachabilities to / from the cut
  - type bound (how many color sets)
  - new encoding (interval structure)



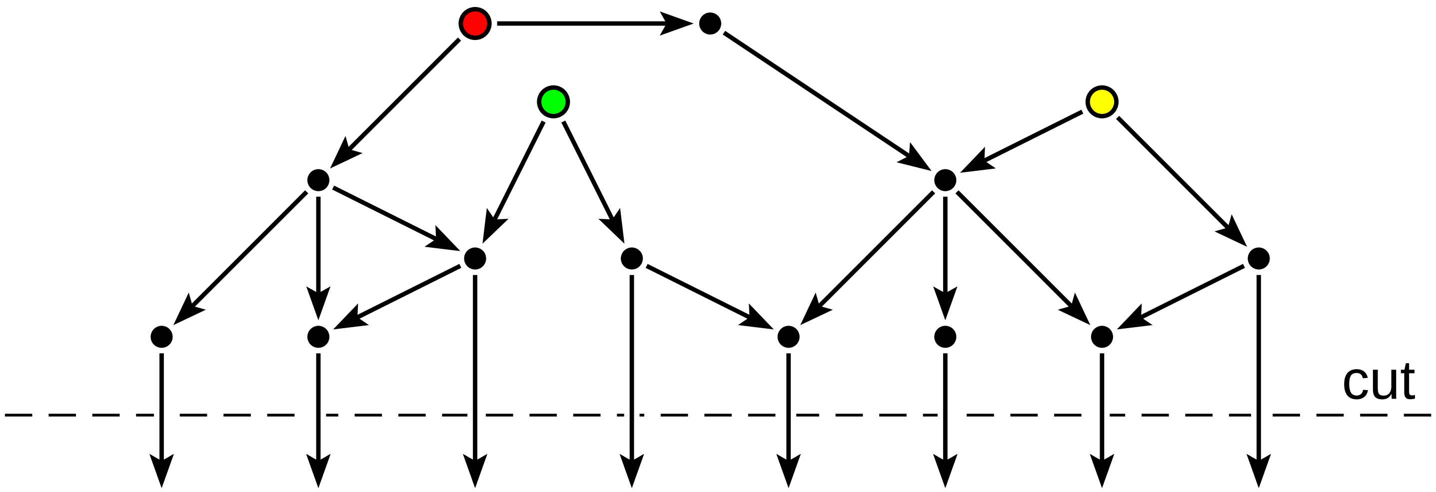
Tools & Techniques

- separation (balanced directed cuts)
- representing reachabilities to / from the cut
  - type bound (how many color sets)
  - new encoding (interval structure)
- recurse

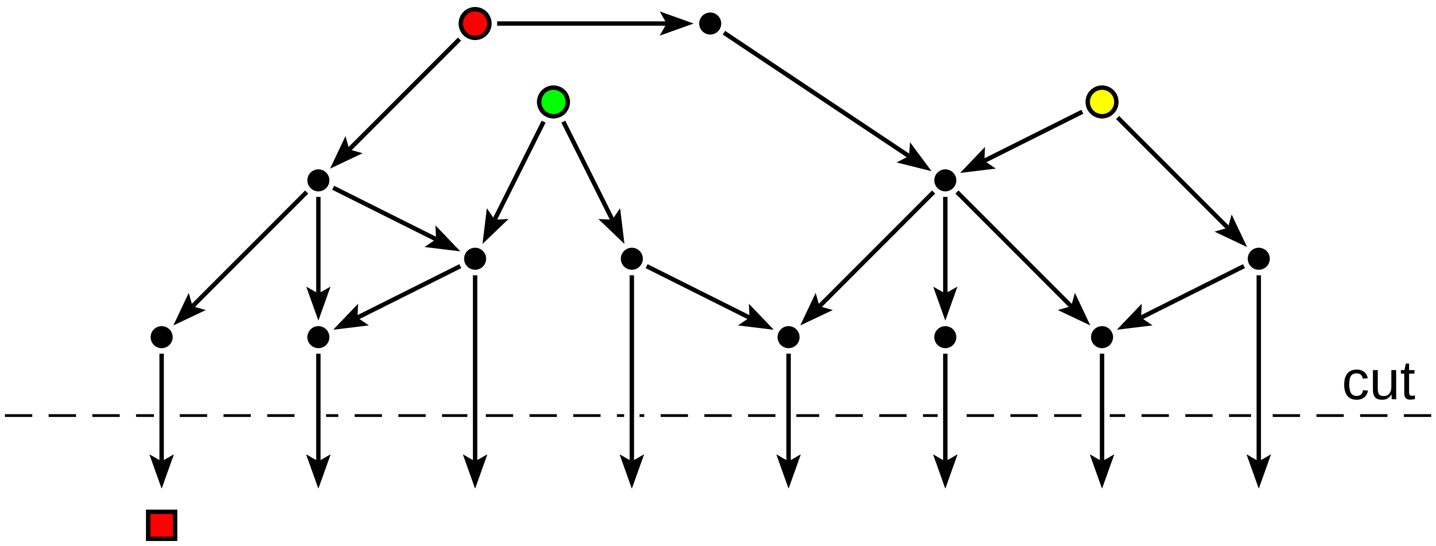


For simplicity, we consider only dags.

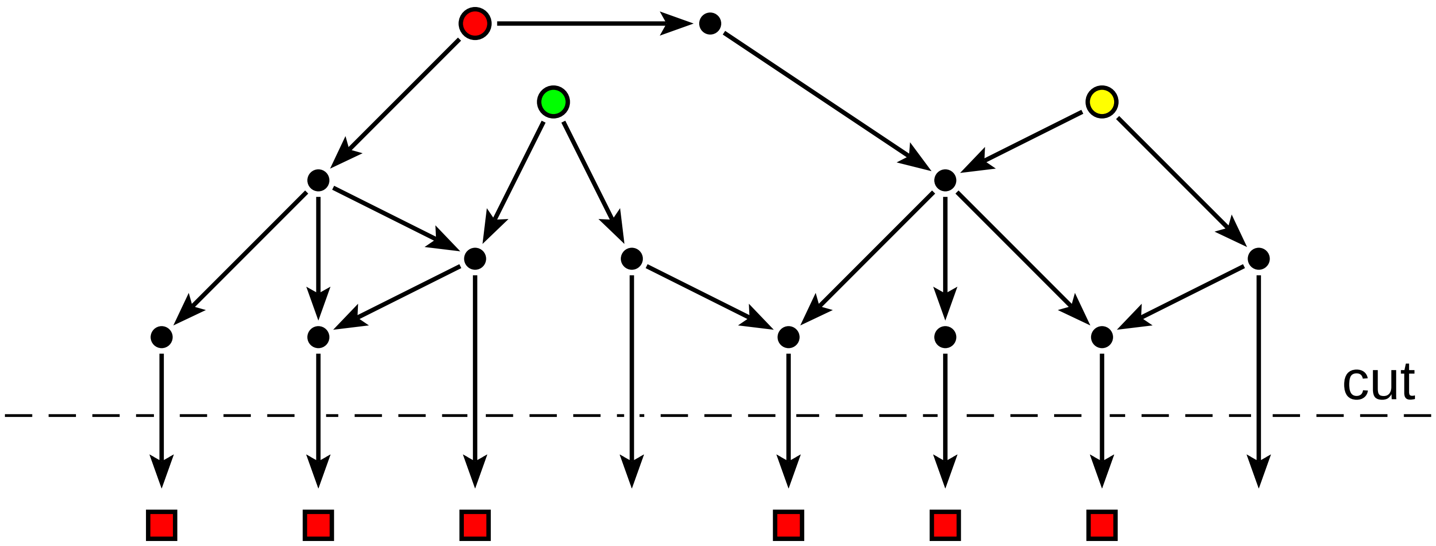
Types Along the Cut



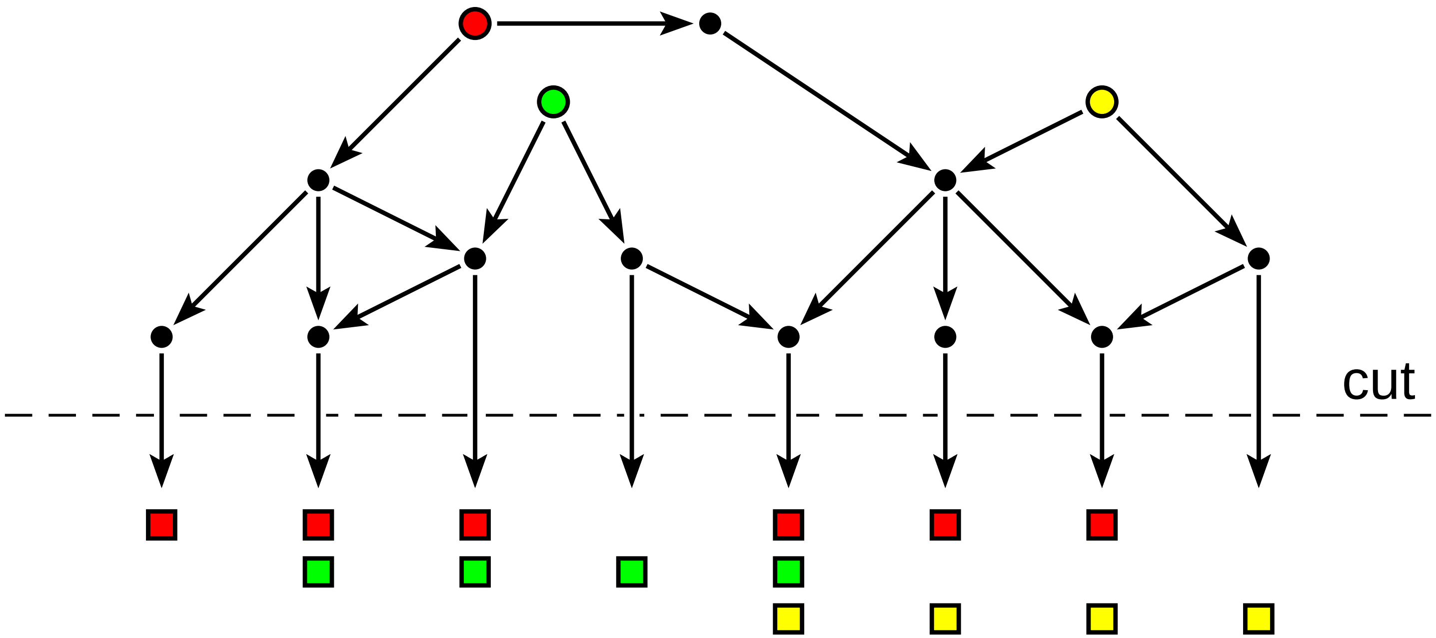
Types Along the Cut



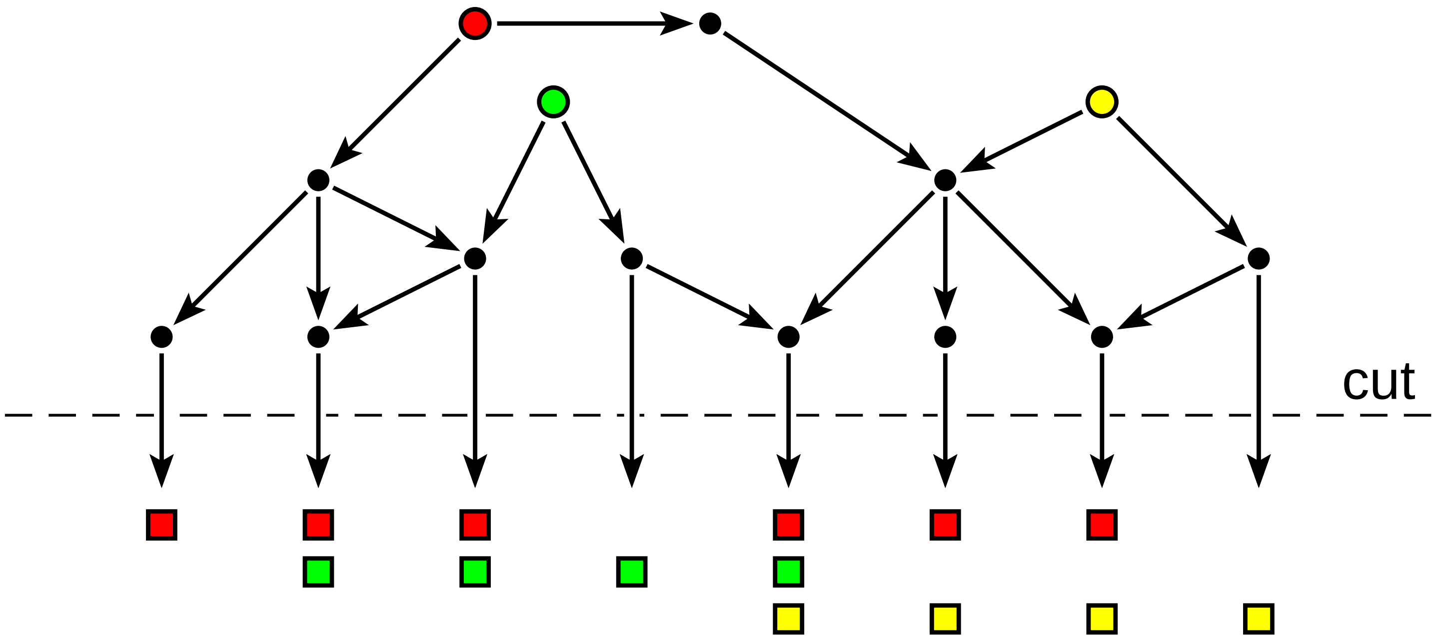
Types Along the Cut



Types Along the Cut

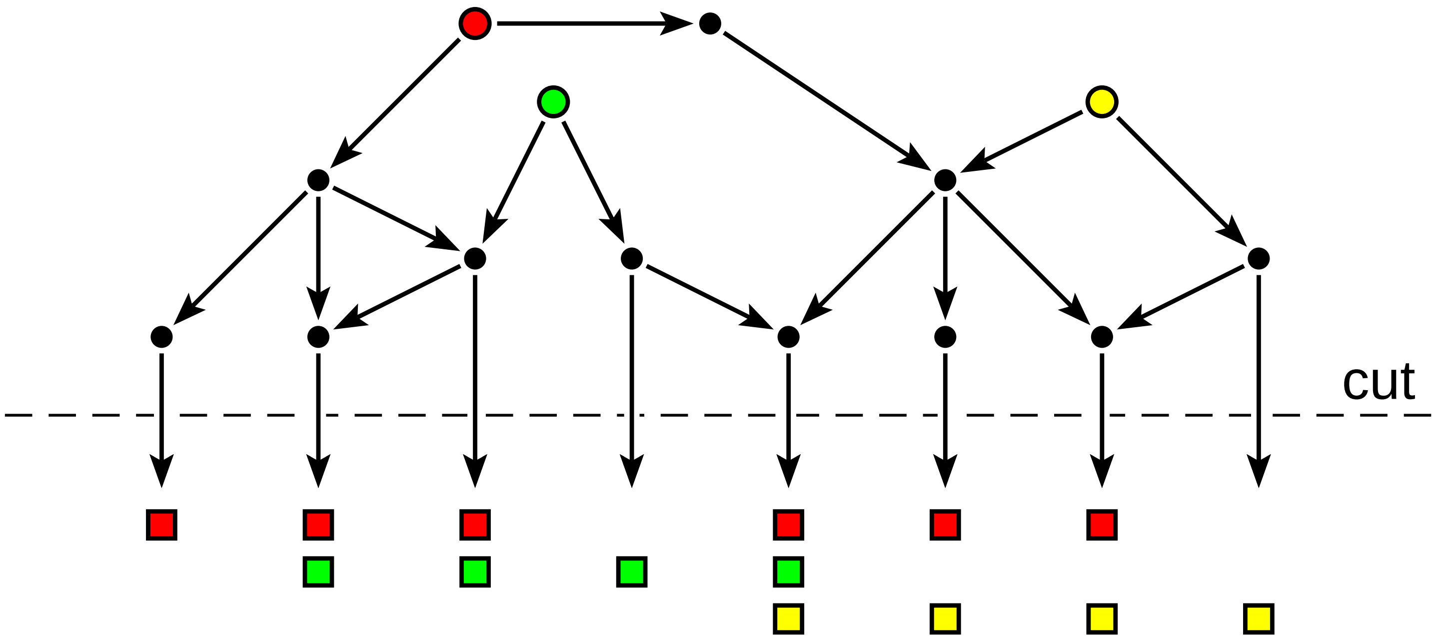


Types Along the Cut



Lemma. (The Type Bound) The number of different types (and also of type changes!) is linear in the number of colors.

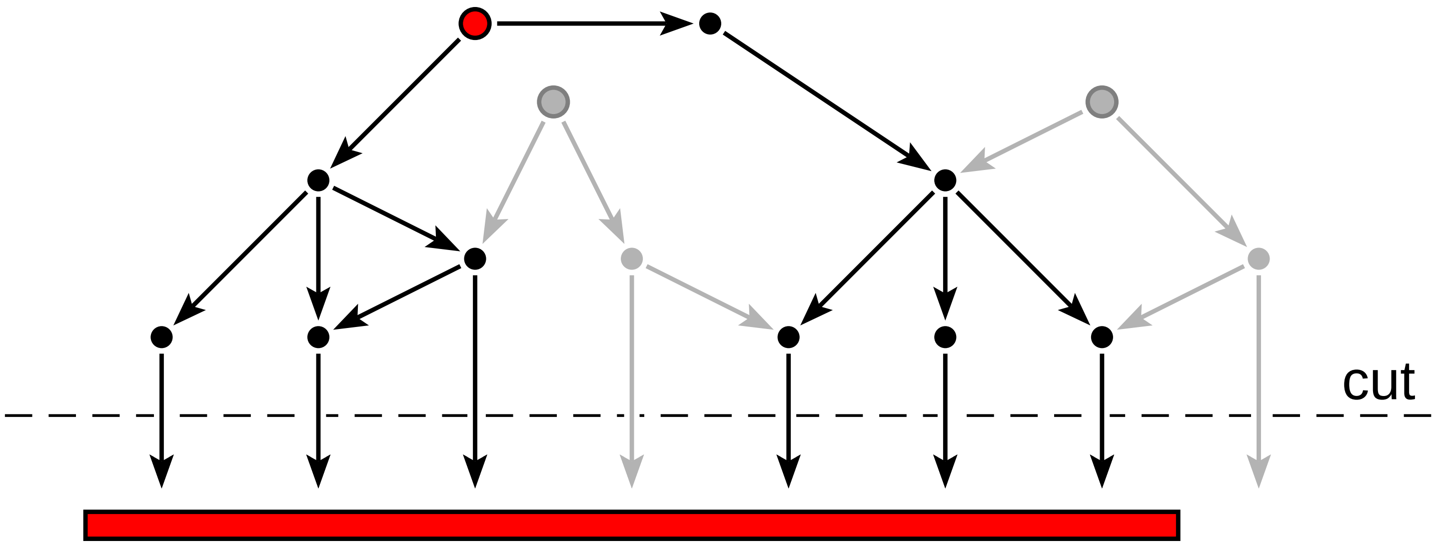
Types Along the Cut



Lemma. **(The Type Bound)** The number of different types (and also of type changes!) is linear in the number of colors.

Lemma. There exists a dag of size $O(k \log k)$ that encodes all reachabilities from the k colors down to the cut line.

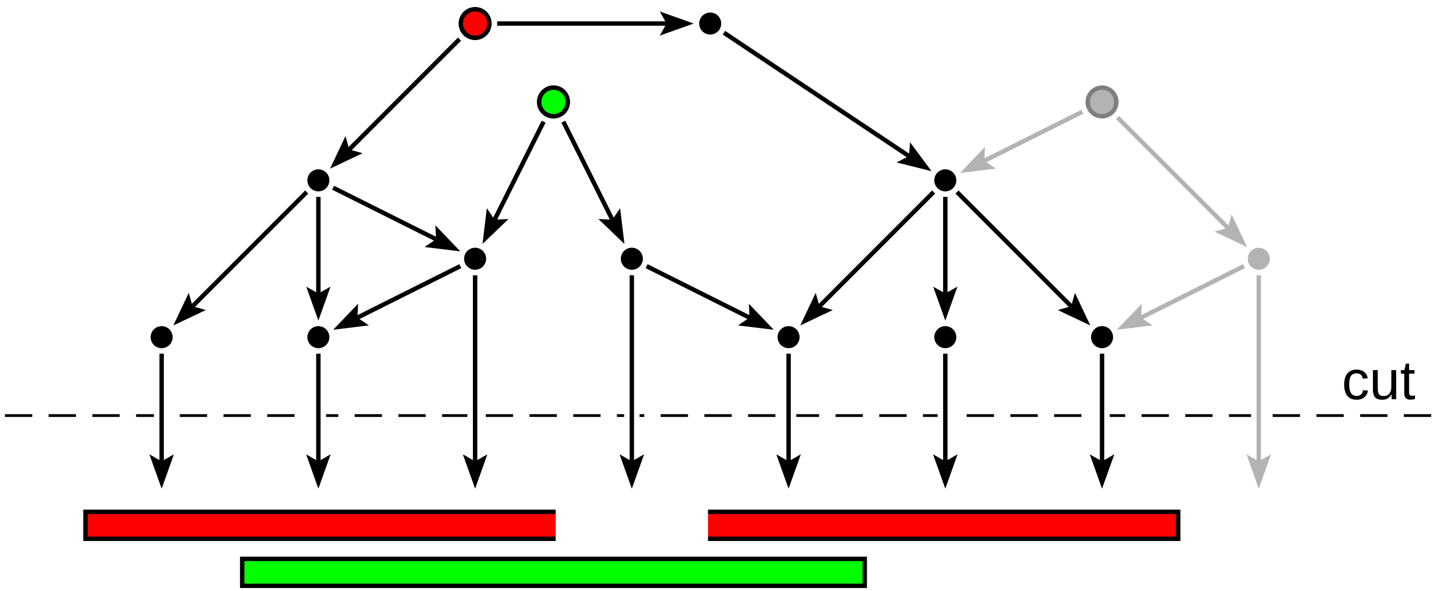
Types Along the Cut



proof idea: *nested intervals*

- insert one interesting vertex after another, each together with all vertices reachable from it
- every interesting vertex must appear before all interesting vertices in its “shadow”

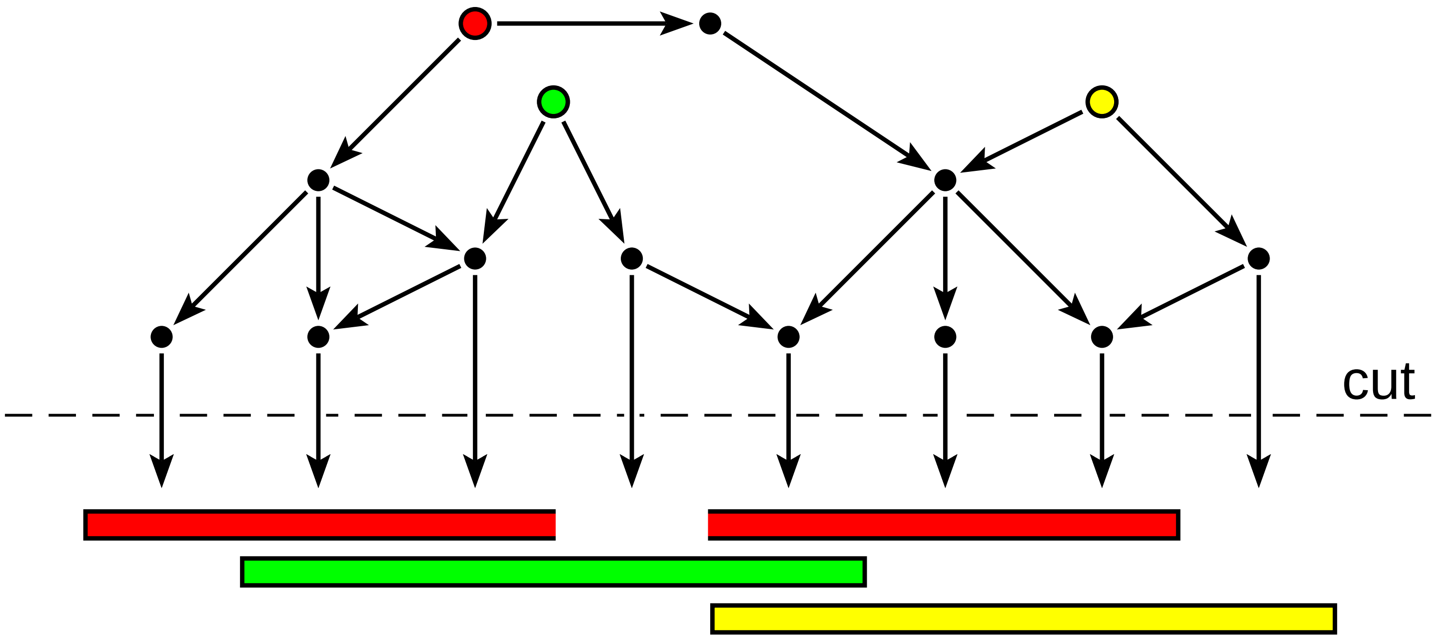
Types Along the Cut



proof idea: *nested intervals*

- insert one interesting vertex after another, each together with all vertices reachable from it
- every interesting vertex must appear before all interesting vertices in its “shadow”

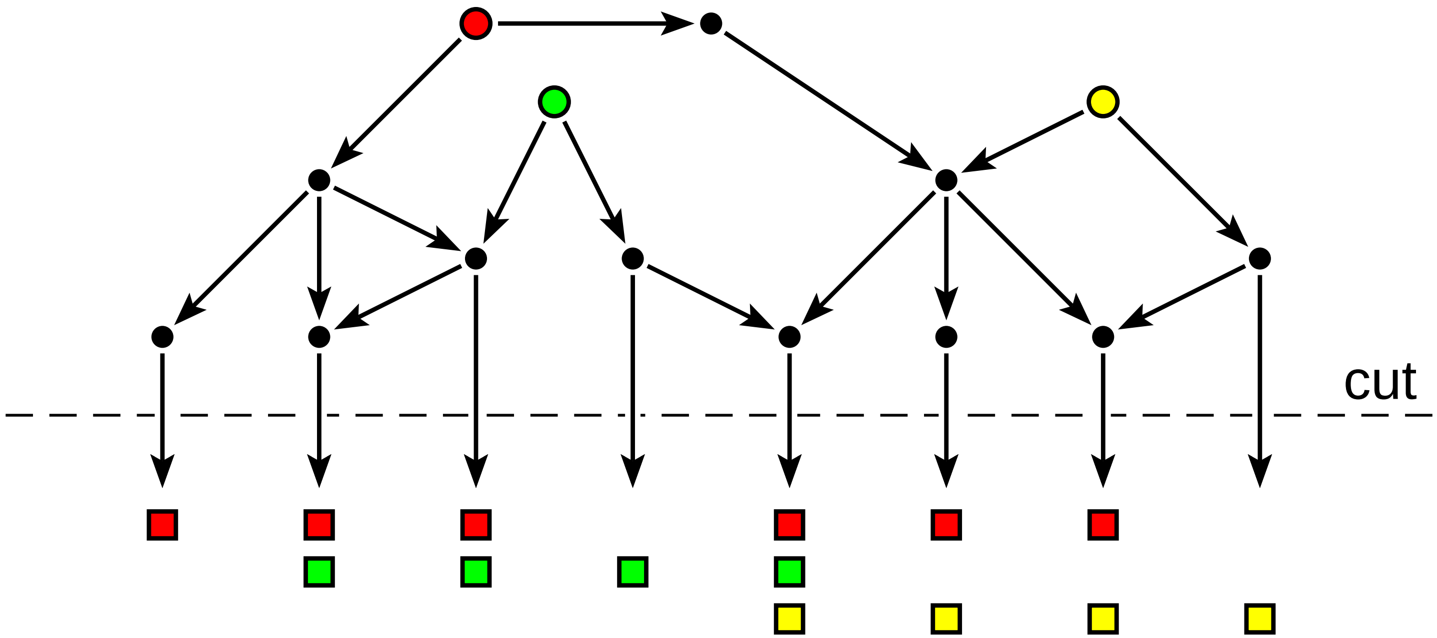
Types Along the Cut



proof idea: *nested intervals*

- insert one interesting vertex after another, each together with all vertices reachable from it
- every interesting vertex must appear before all interesting vertices in its "shadow"

Types Along the Cut



Lemma. (The Type Bound) The number of different types (and also of type changes!) is linear in the number of colors.

Lemma. There exists a dag of size $O(k \log k)$ that encodes all reachabilities from the k colors down to the cut line.

Balanced Directed Cuts

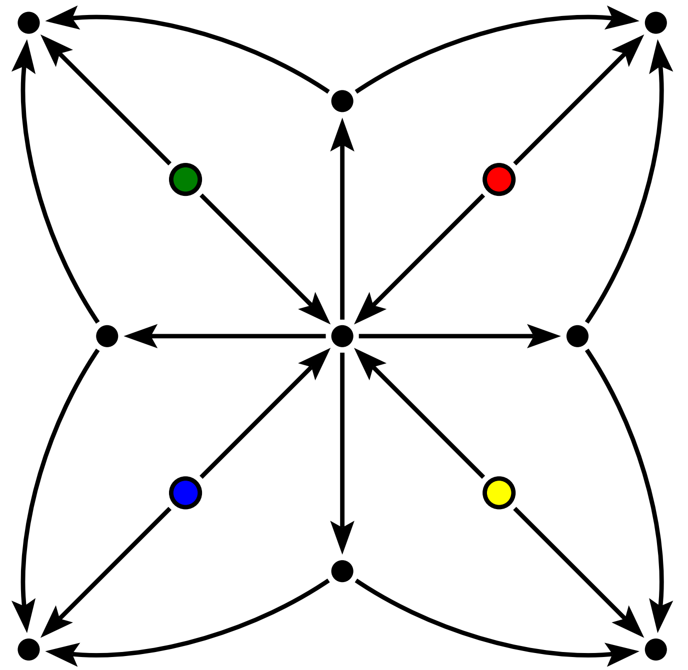
- wanted: two simply-connected regions separated by a closed Jordan curve
- *directed* needed for cross-cut interval structure to work
- *balanced* (w.r.t. $\mathcal{U}$) required for recursion depth $O(\log |\mathcal{U}|)$

Balanced Directed Cuts

- wanted: two simply-connected regions separated by a closed Jordan curve
- *directed* needed for cross-cut interval structure to work
- *balanced* (w.r.t. $\mathcal{U}$) required for recursion depth $O(\log|\mathcal{U}|)$

BUT — in general we cannot have both!

example:



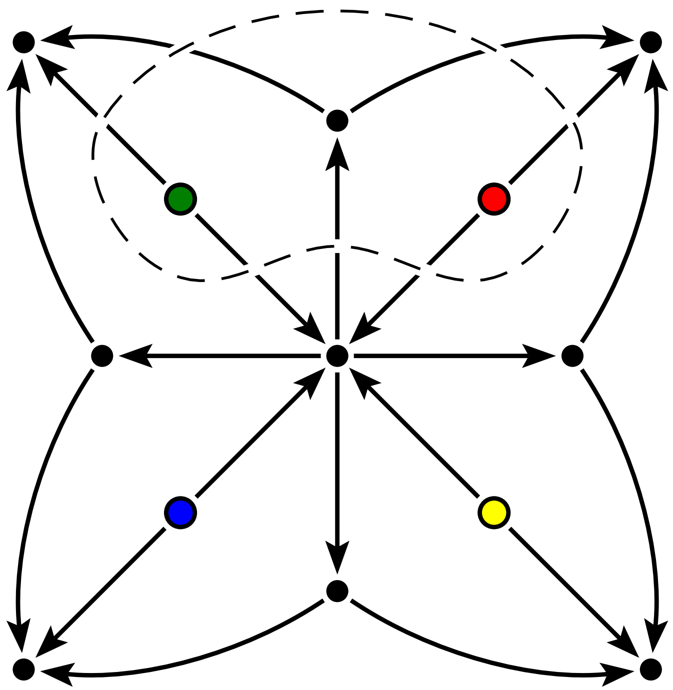
Balanced Directed Cuts

- wanted: two simply-connected regions separated by a closed Jordan curve
- *directed* needed for cross-cut interval structure to work
- *balanced* (w.r.t. U) required for recursion depth $O(\log|U|)$

BUT — in general we cannot have both!

example:

- choose two colors, red and green, for the simply connected "out set" A
- draw the cut line around them



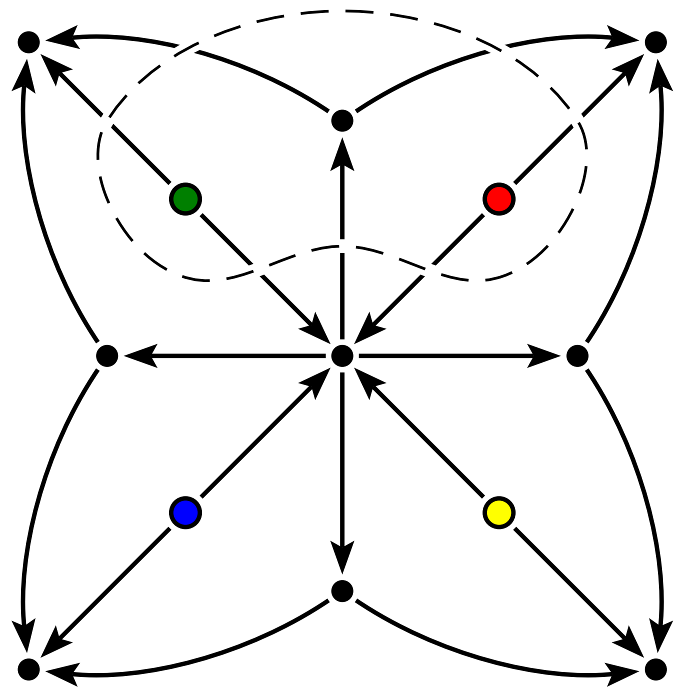
Balanced Directed Cuts

- wanted: two simply-connected regions separated by a closed Jordan curve
- *directed* needed for cross-cut interval structure to work
- *balanced* (w.r.t. $\mathcal{U}$) required for recursion depth $O(\log |\mathcal{U}|)$

BUT — in general we cannot have both!

example:

- choose two colors, red and green, for the simply connected “out set” A
- draw the cut line around them
- this collects at least one • vertex



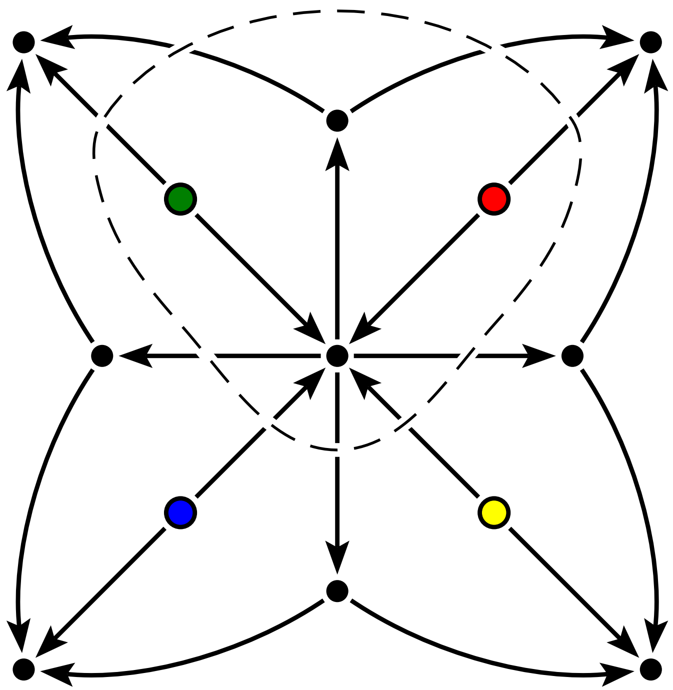
Balanced Directed Cuts

- wanted: two simply-connected regions separated by a closed Jordan curve
- *directed* needed for cross-cut interval structure to work
- *balanced* (w.r.t. $\mathcal{U}$) required for recursion depth $O(\log|\mathcal{U}|)$

BUT — in general we cannot have both!

example:

- choose two colors, red and green, for the simply connected “out set” A
- draw the cut line around them
- this collects at least one • vertex
- then by transitivity, the other two colors must sit in A, too

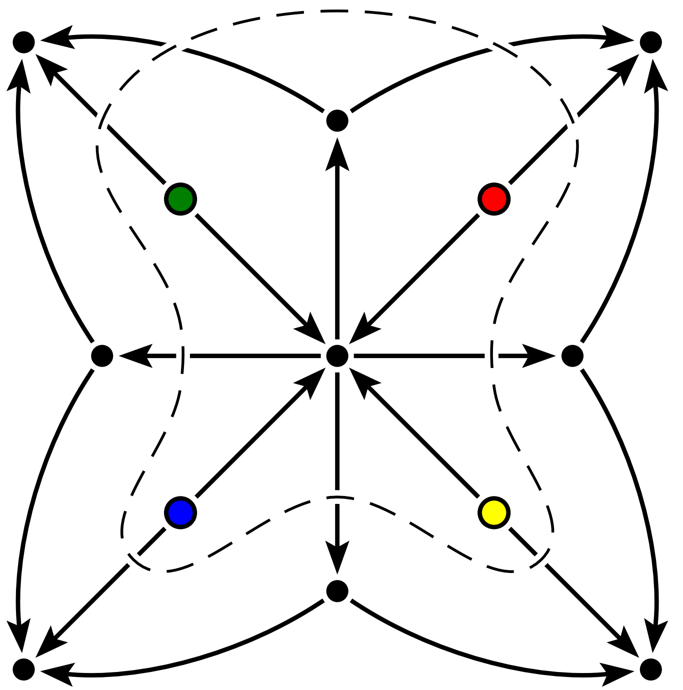


Balanced Directed Cuts

- wanted: two simply-connected regions separated by a closed Jordan curve
- *directed* needed for cross-cut interval structure to work
- *balanced* (w.r.t. $\mathcal{U}$) required for recursion depth $O(\log|\mathcal{U}|)$

BUT — in general we cannot have both!

example:



- choose two colors, red and green, for the simply connected “out set” A
- draw the cut line around them
- this collects at least one • vertex
- then by transitivity, the other two colors must sit in A, too

Almost-Directed Cuts

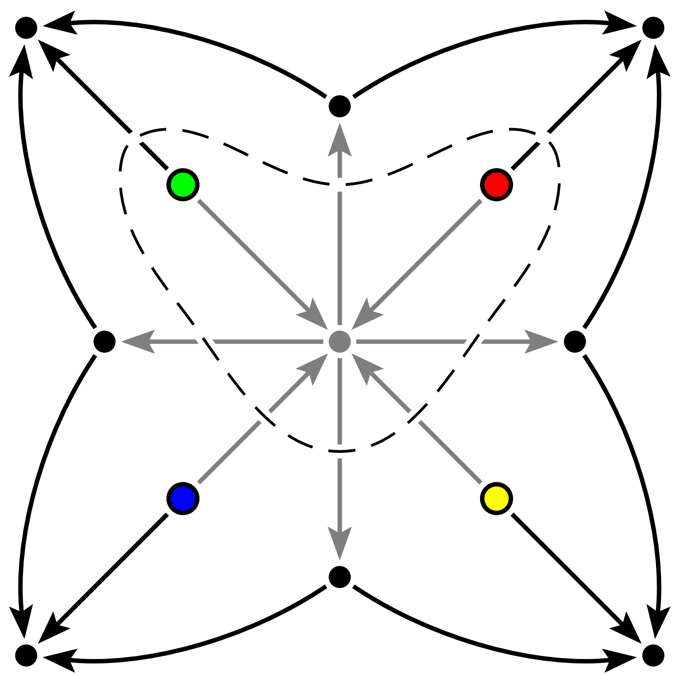
Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.



Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order

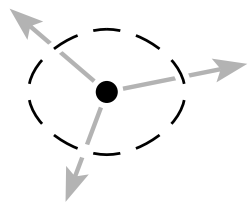
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex



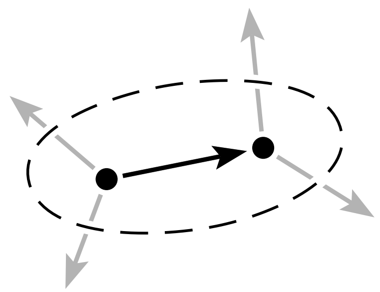
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex



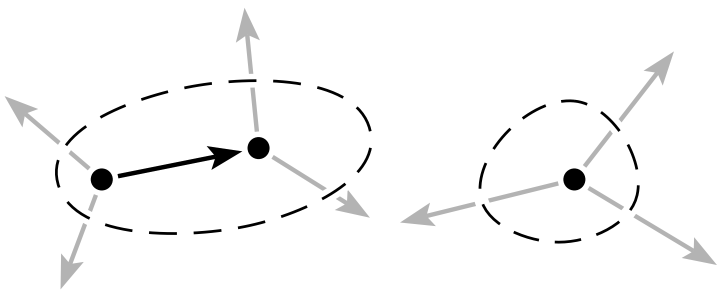
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex



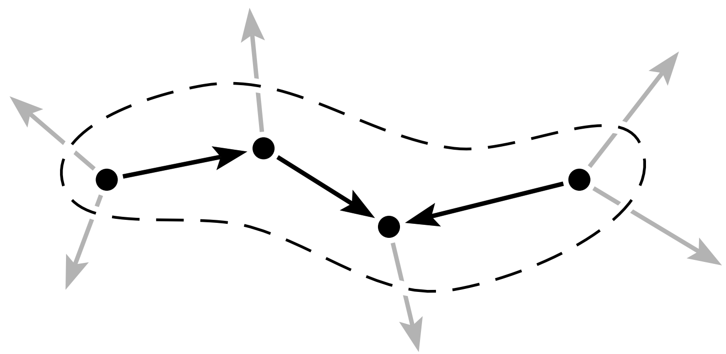
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex
- merge regions when necessary



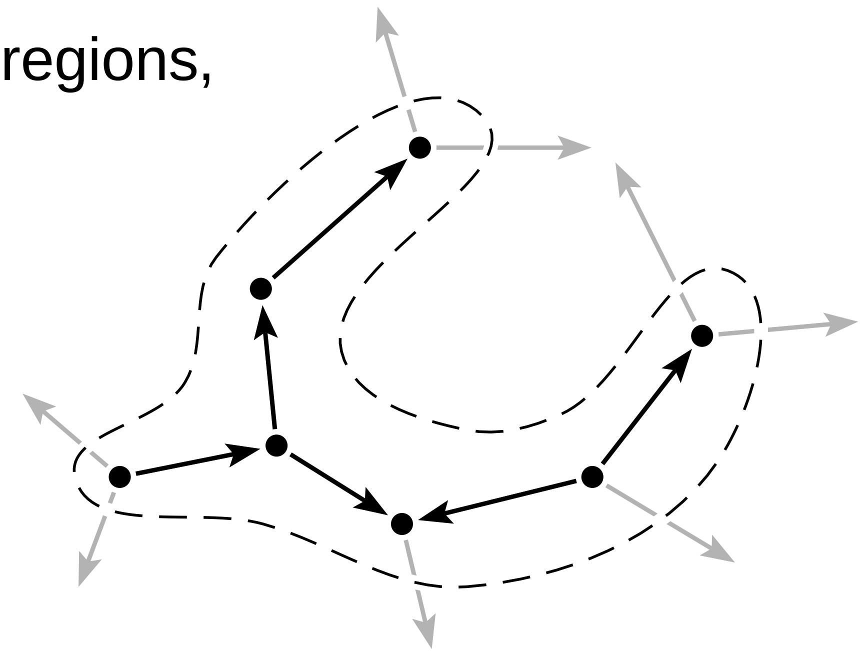
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex
- merge regions when necessary



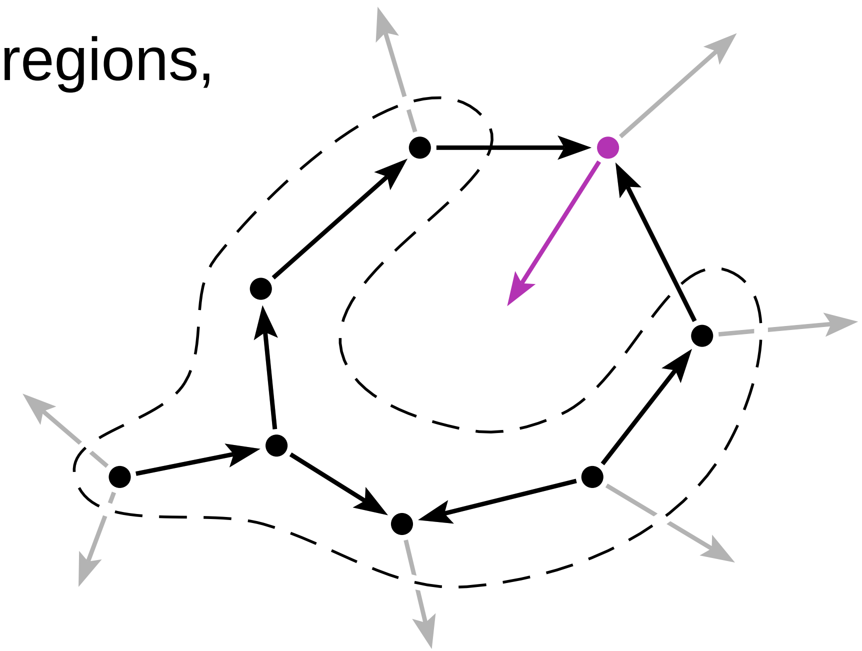
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex
- merge regions when necessary
- saddle points are critical



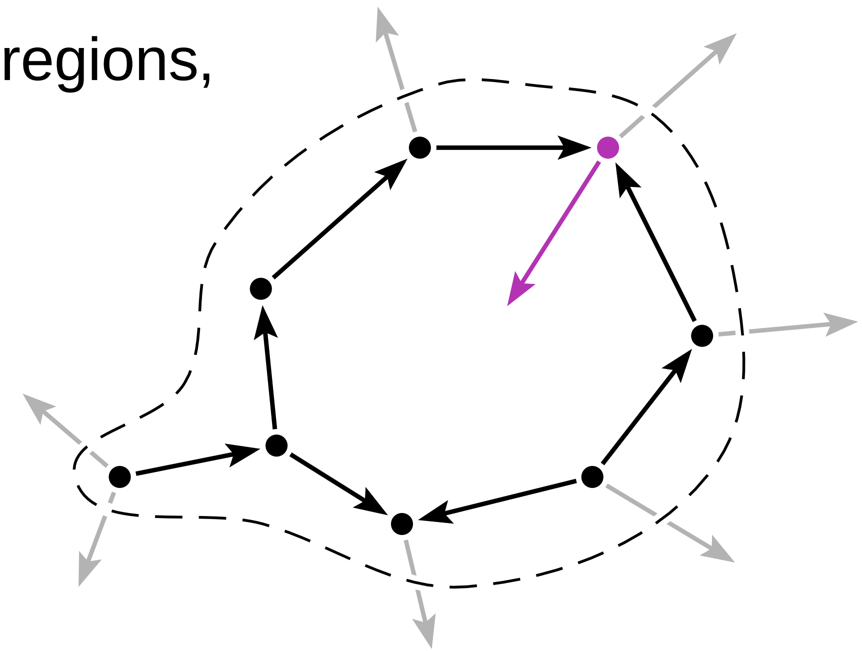
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex
- merge regions when necessary
- saddle points are critical



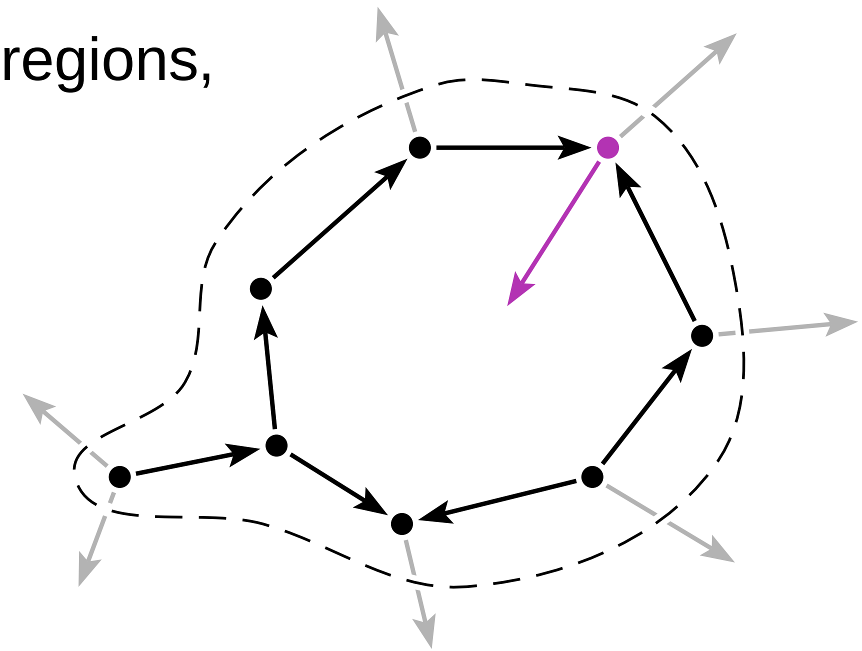
Almost-Directed Cuts

Def. A plane cut (Jordan curve) is *almost-directed* if deletion of just one vertex makes it directed.

Theorem. Every plane dag has an almost-directed 1:3-cut.

proof sketch:

- process the dag in some fixed topological order
- grow simply-connected out-directed regions, vertex by vertex
- merge regions when necessary
- saddle points are critical
- but only one will persist



The Big Picture

- cut the given dag in half

The Big Picture

- cut the given dag in half
- represent cross-cut reachabilities:
  - size $O(k \log k)$ interval structure
    (takes care of all single-cross paths)

The Big Picture

- cut the given dag in half
- represent cross-cut reachabilities:
  - size $O(k \log k)$ interval structure
    (takes care of all single-cross paths)
  - $O(k)$ arcs for the *“almost”*-outlier
    (takes care of zig-zag paths)

The Big Picture

- cut the given dag in half
- represent cross-cut reachabilities:
  - size $O(k \log k)$ interval structure (takes care of all single-cross paths)
  - $O(k)$ arcs for the *“almost”*-outlier (takes care of zig-zag paths)
- recurse $O(\log |U|)$ times

The Big Picture

- cut the given dag in half
- represent cross-cut reachabilities:
  - size $O(k \log k)$ interval structure
    (takes care of all single-cross paths)
  - $O(k)$ arcs for the *“almost”*-outlier
    (takes care of zig-zag paths)
- recurse $O(\log |U|)$ times

- directed cycles can be taken care of separately in advance
  (they cut the plane into well-separated areas)

Outlook / Open Problems

Reachability substitutes might shine a new combinatorial but “non-Kuratowskian” light on planarity.

A charakterization of digraphs with planar reachability substitutes in terms of forbidden directed minors might be possilbe.

Outlook / Open Problems

Reachability substitutes might shine a new combinatorial but "non-Kuratowskian" light on planarity.

A charakterization of digraphs with planar reachability substitutes in terms of forbidden directed minors might be possilbe.

- How difficult is it to decide whether a given digraph has a planar reachability substitute?

Outlook / Open Problems

Reachability substitutes might shine a new combinatorial but "non-Kuratowskian" light on planarity.

A charakterization of digraphs with planar reachability substitutes in terms of forbidden directed minors might be possilbe.

- How difficult is it to decide whether a given digraph has a planar reachability substitute?
- Find a general bound on the size of such a substitute.
  (Some instances known to require $\Omega(|U|^2)$ size.)

Outlook / Open Problems

Reachability substitutes might shine a new combinatorial but "non-Kuratowskian" light on planarity.

A charakterization of digraphs with planar reachability substitutes in terms of forbidden directed minors might be possilbe.

- How difficult is it to decide whether a given digraph has a planar reachability substitute?
- Find a general bound on the size of such a substitute.
  (Some instances known to require $\Omega(|U|^2)$ size.)

General Problems:

- Prove a super-linear lower bound on the size of general (non-planar) reachability substitutes.
- Are the two log-factors in our construction really necessary?