Online coloring blowups of a known graph

Kevin G. Milans (milans@math.wvu.edu)
Michael C. Wigal (mcwigal@mix.wvu.edu)

West Virginia University

AMS Spring Central Sectional Meeting
Indiana University
Bloomington, IN
April 2, 2017

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

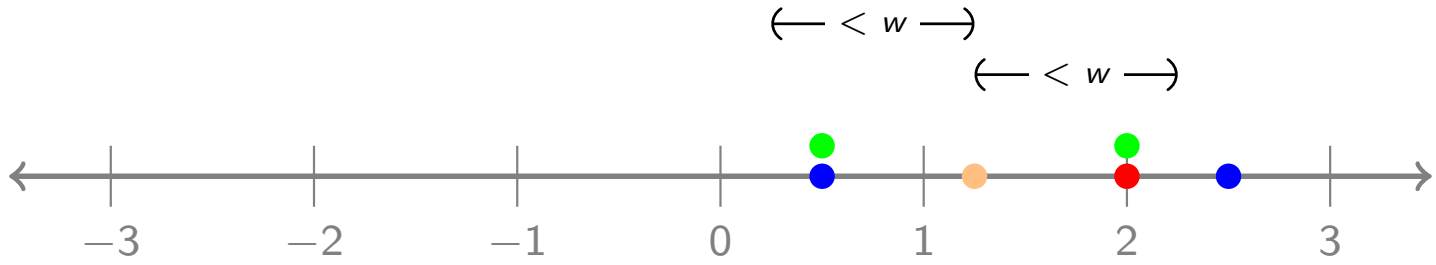
A motivating problem

- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.

A motivating problem

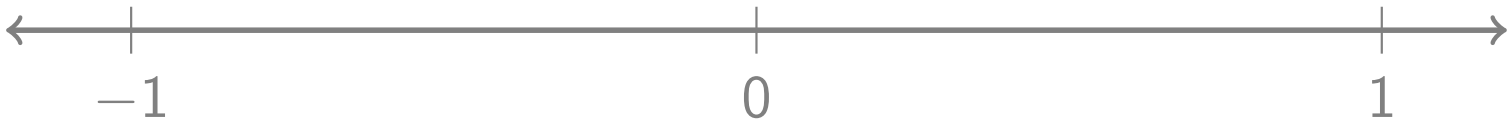
- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.
- Open: How many colors does $\mathrm{Algorithm}$ need?

A motivating problem



- Consider a game between $\mathrm{Algorithm}$ and $\mathrm{Spoiler}$.
- $\mathrm{Spoiler}$ selects a point $x \in \mathbb{R}$. Each open unit interval can have at most w selected points; we call w the width of the game.
- $\mathrm{Algorithm}$ assigns x a color. Colors must be distinct on open unit intervals.
- Open: How many colors does $\mathrm{Algorithm}$ need?
- The greedy algorithm uses at most $2w - 1$ colors.

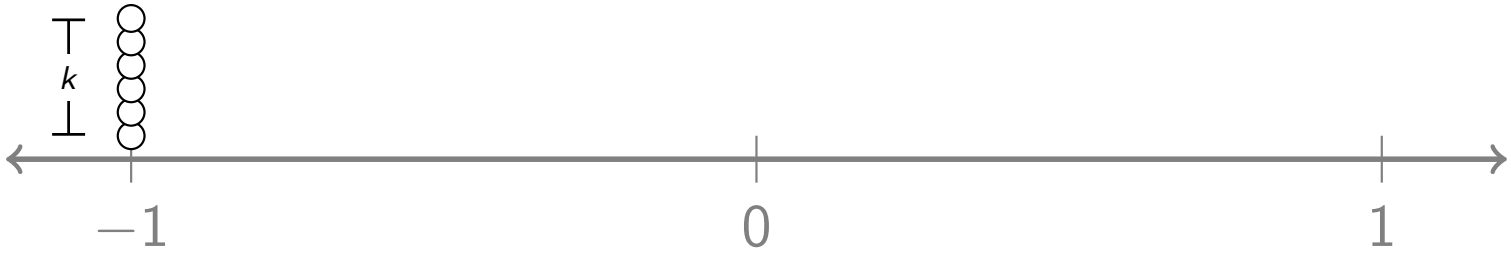
A motivating problem



Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\lfloor \frac{3}{2} w \rfloor$ colors.

A motivating problem

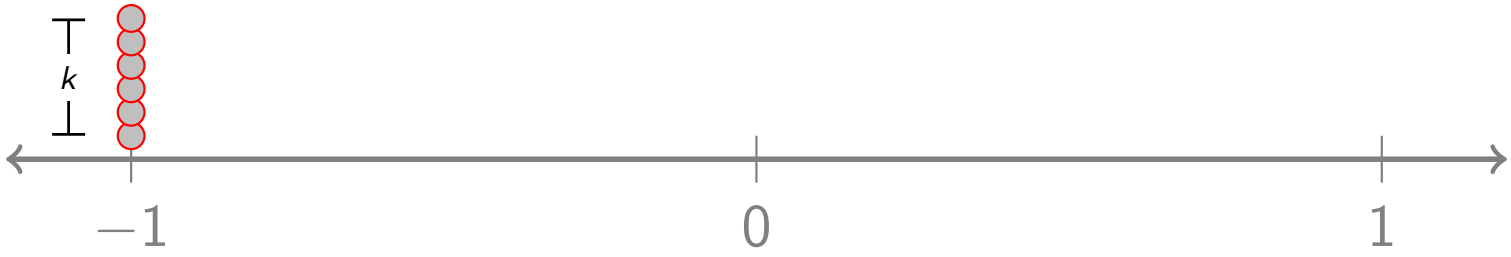


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.

A motivating problem

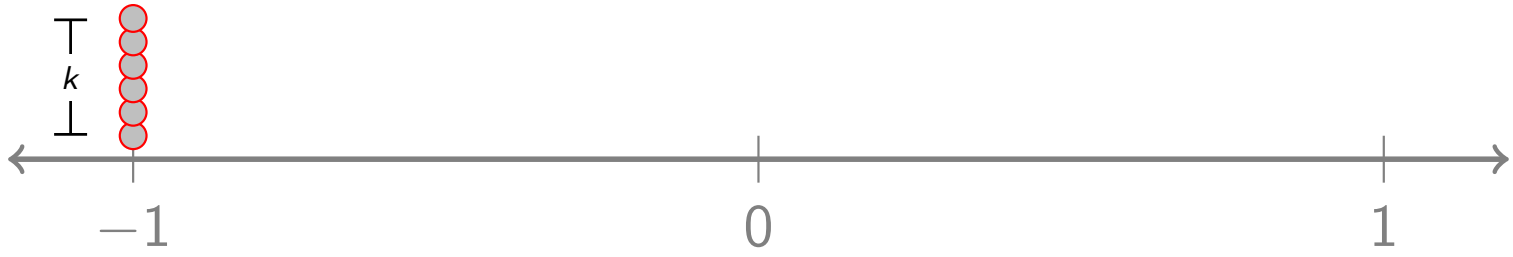


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.

A motivating problem

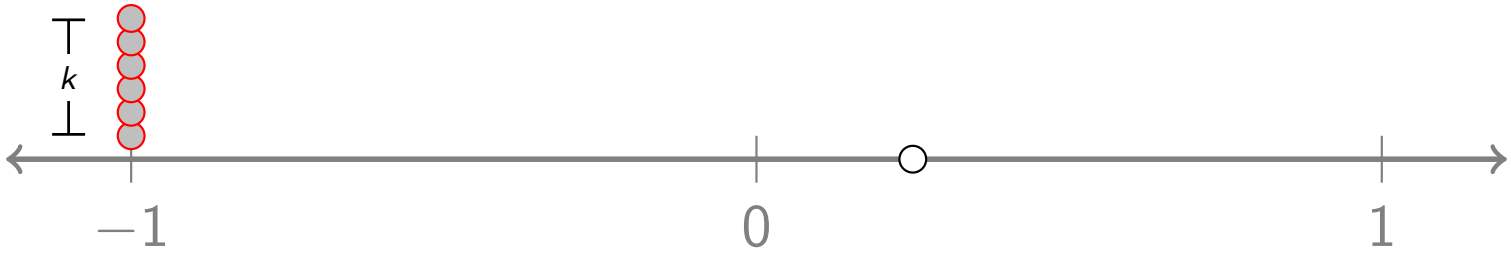


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

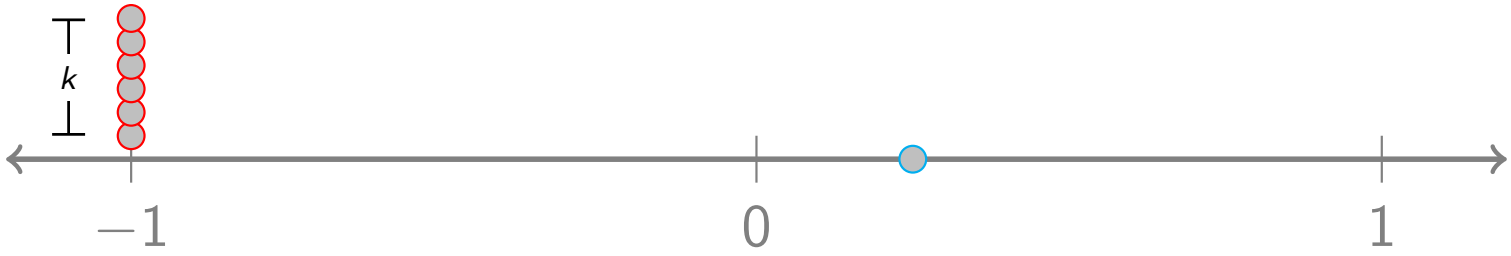


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

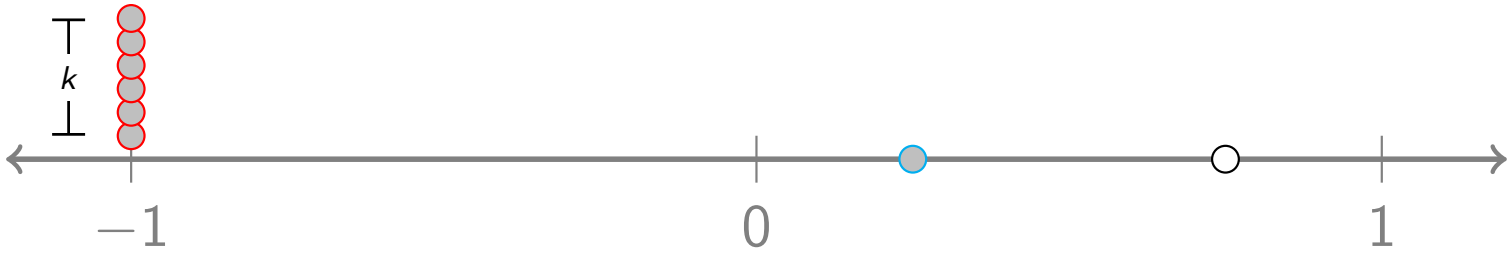


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for Spoiler that forces Algorithm to use at least $\lfloor \frac{3}{2} w \rfloor$ colors.

- Spoiler plays k times at -1, forcing k "old" colors.
- Spoiler plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

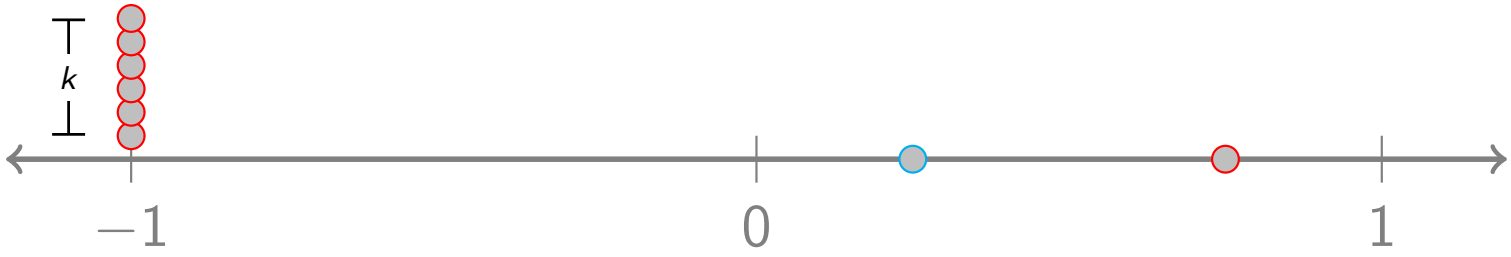


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

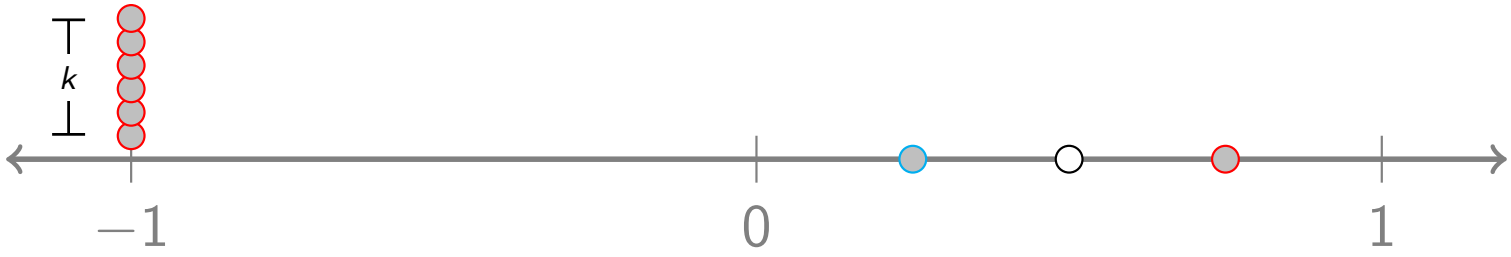


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for Spoiler that forces Algorithm to use at least $\lfloor \frac{3}{2} w \rfloor$ colors.

- Spoiler plays k times at -1, forcing k "old" colors.
- Spoiler plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

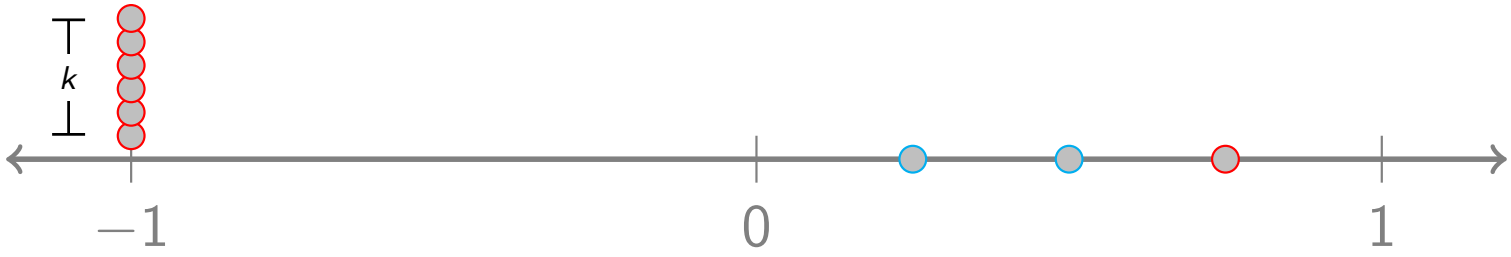


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

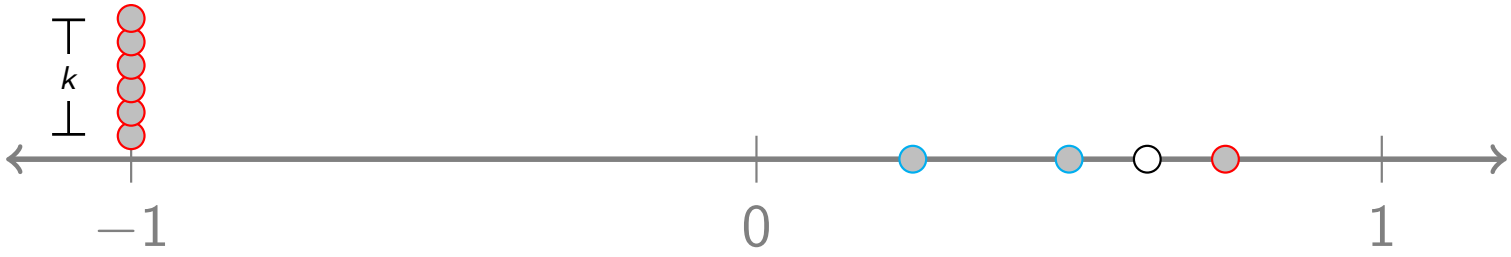


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for Spoiler that forces Algorithm to use at least $\lfloor \frac{3}{2} w \rfloor$ colors.

- Spoiler plays k times at -1, forcing k "old" colors.
- Spoiler plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

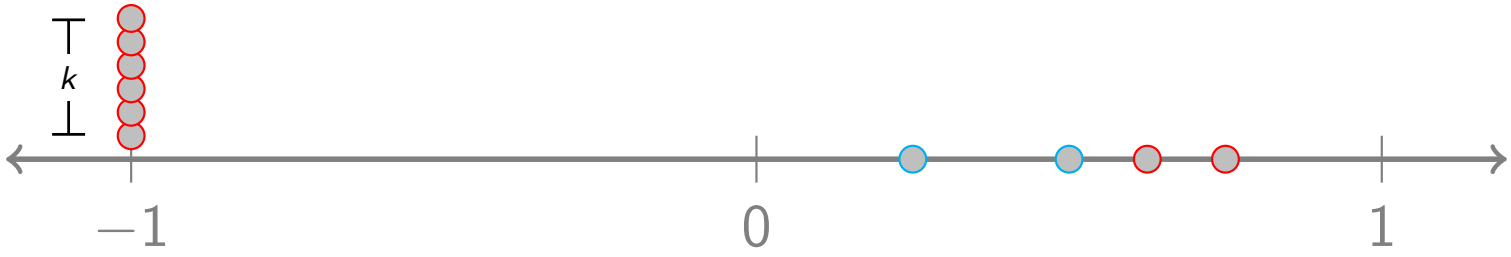


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

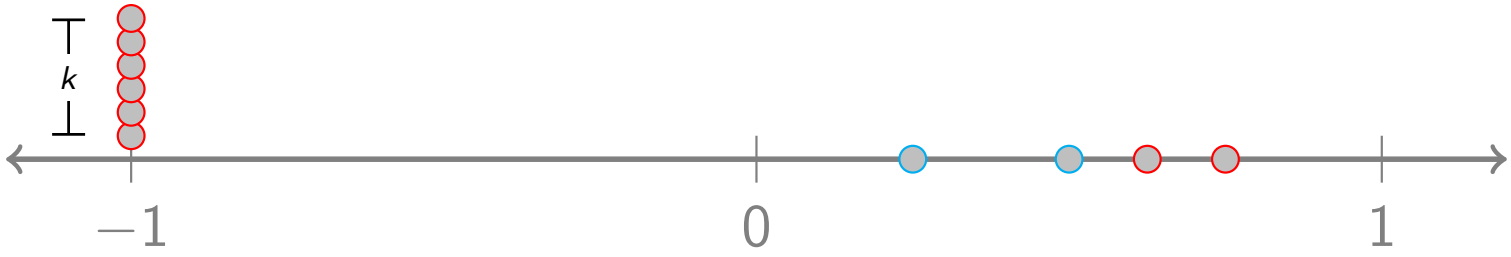


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.

A motivating problem

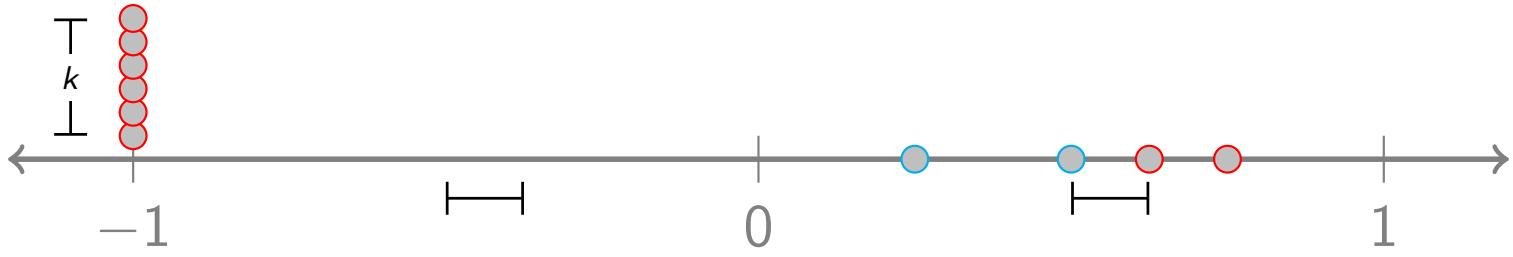


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for Spoiler that forces Algorithm to use at least $\lfloor \frac{3}{2} w \rfloor$ colors.

- Spoiler plays k times at -1, forcing k "old" colors.
- Spoiler plays in $[0, 1]$ so that new colors are left of old colors.
- After at most $2k$ rounds, Spoiler forces at least k new colors.

A motivating problem

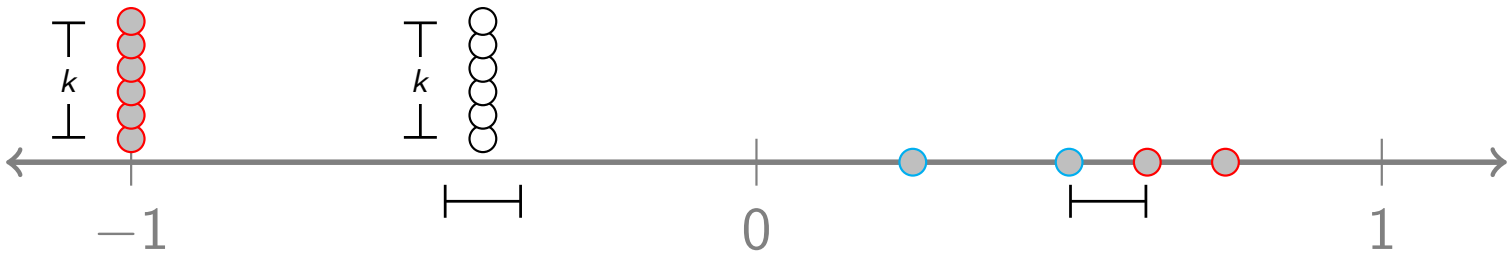


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.
- After at most $2k$ rounds, $\mathrm{Spoiler}$ forces at least k new colors.
- $\mathrm{Spoiler}$ plays k times at a point in $[-1, 0]$ that conflicts with new colors in $[0, 1]$ but is far away from old colors.

A motivating problem

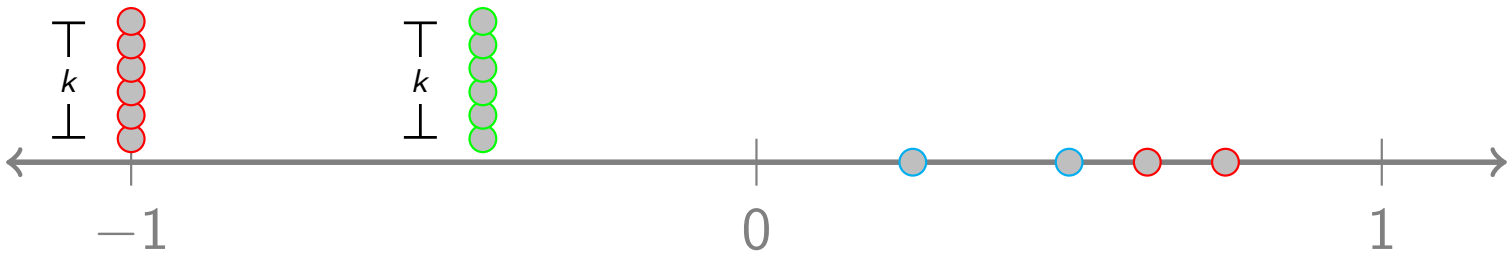


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for Spoiler that forces Algorithm to use at least $\lfloor \frac{3}{2} w \rfloor$ colors.

- Spoiler plays k times at -1, forcing k "old" colors.
- Spoiler plays in $[0, 1]$ so that new colors are left of old colors.
- After at most $2k$ rounds, Spoiler forces at least k new colors.
- Spoiler plays k times at a point in $[-1, 0]$ that conflicts with new colors in $[0, 1]$ but is far away from old colors.

A motivating problem

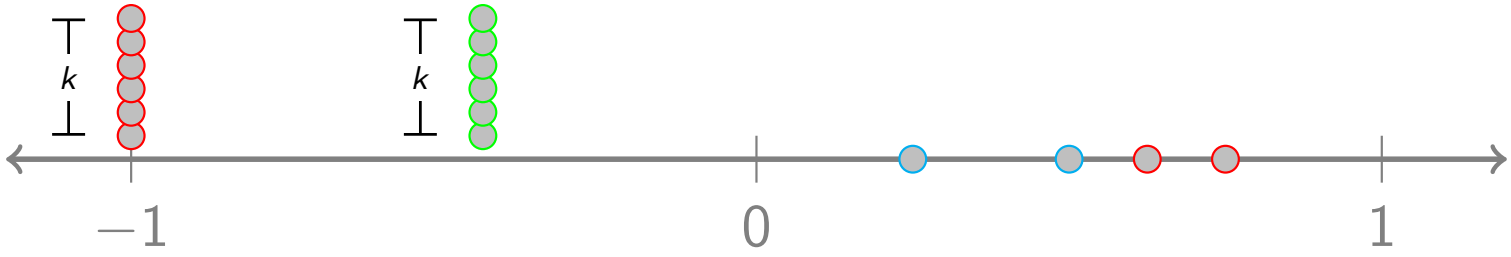


Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- $\mathrm{Spoiler}$ plays k times at -1, forcing k "old" colors.
- $\mathrm{Spoiler}$ plays in $[0, 1]$ so that new colors are left of old colors.
- After at most $2k$ rounds, $\mathrm{Spoiler}$ forces at least k new colors.
- $\mathrm{Spoiler}$ plays k times at a point in $[-1, 0]$ that conflicts with new colors in $[0, 1]$ but is far away from old colors.

A motivating problem



Prop. (Bosek, Felsner, Kloch, Krawczyk, Matecki, Micek)

There is a strategy for $\mathrm{Spoiler}$ that forces $\mathrm{Algorithm}$ to use at least $\left\lfloor \frac{3}{2} w \right\rfloor$ colors.

- This forces $3k$ colors in a game of width $2k$.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.
  - $\mathrm{Algorithm}$ assigns x a color.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.
  - $\mathrm{Algorithm}$ assigns x a color.
- The associated token graph H is obtained from G by replacing each vertex v with the complete graph on the tokens at v.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.
  - $\mathrm{Algorithm}$ assigns x a color.
- The associated token graph H is obtained from G by replacing each vertex v with the complete graph on the tokens at v.
- $\mathrm{Algorithm}$ must give a proper coloring of H and wants to minimize the number of colors used.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.
  - $\mathrm{Algorithm}$ assigns x a color.
- The associated token graph H is obtained from G by replacing each vertex v with the complete graph on the tokens at v.
- $\mathrm{Algorithm}$ must give a proper coloring of H and wants to minimize the number of colors used.
- $\mathrm{Spoiler}$ must ensure that $\chi(H) \leq w$ and wants to force many colors.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.
  - $\mathrm{Algorithm}$ assigns x a color.
- The associated token graph H is obtained from G by replacing each vertex v with the complete graph on the tokens at v.
- $\mathrm{Algorithm}$ must give a proper coloring of H and wants to minimize the number of colors used.
- $\mathrm{Spoiler}$ must ensure that $\chi(H) \leq w$ and wants to force many colors.
- The value of the game, denoted $f(G; w)$, is the number of colors needed by an optimal strategy for $\mathrm{Algorithm}$.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.
  - $\mathrm{Algorithm}$ assigns x a color.
- The associated token graph H is obtained from G by replacing each vertex v with the complete graph on the tokens at v.
- $\mathrm{Algorithm}$ must give a proper coloring of H and wants to minimize the number of colors used.
- $\mathrm{Spoiler}$ must ensure that $\chi(H) \leq w$ and wants to force many colors.
- The value of the game, denoted $f(G; w)$, is the number of colors needed by an optimal strategy for $\mathrm{Algorithm}$.
- Let $\mathbb{G}$ be the graph on $\mathbb{R}$ with $uv \in E(\mathbb{G})$ if and only if $|u - v| < 1$.

A generalization to graphs

- For a graph G, the G-coloring game of width w is played in rounds between $\mathrm{Spoiler}$ and $\mathrm{Algorithm}$:
  - $\mathrm{Spoiler}$ chooses a vertex $v \in V(G)$ and plays a token x at v.
  - $\mathrm{Algorithm}$ assigns x a color.
- The associated token graph H is obtained from G by replacing each vertex v with the complete graph on the tokens at v.
- $\mathrm{Algorithm}$ must give a proper coloring of H and wants to minimize the number of colors used.
- $\mathrm{Spoiler}$ must ensure that $\chi(H) \leq w$ and wants to force many colors.
- The value of the game, denoted $f(G; w)$, is the number of colors needed by an optimal strategy for $\mathrm{Algorithm}$.
- Let $\mathbb{G}$ be the graph on $\mathbb{R}$ with $uv \in E(\mathbb{G})$ if and only if $|u - v| < 1$.
- $\left\lfloor \frac{3}{2} w \right\rfloor \leq f(\mathbb{G}; w) \leq 2w - 1$.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.

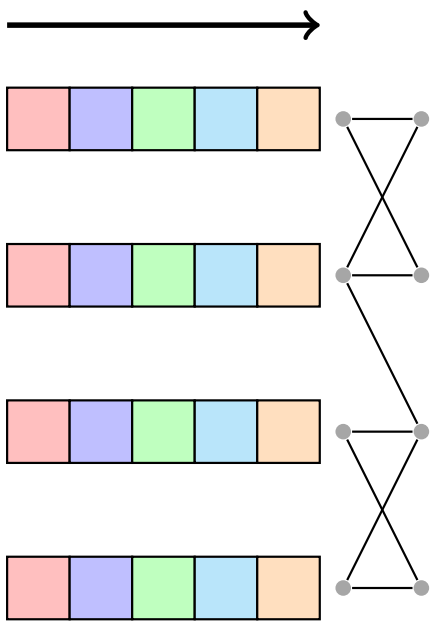
- Choose a linear ordering on a set of w colors.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



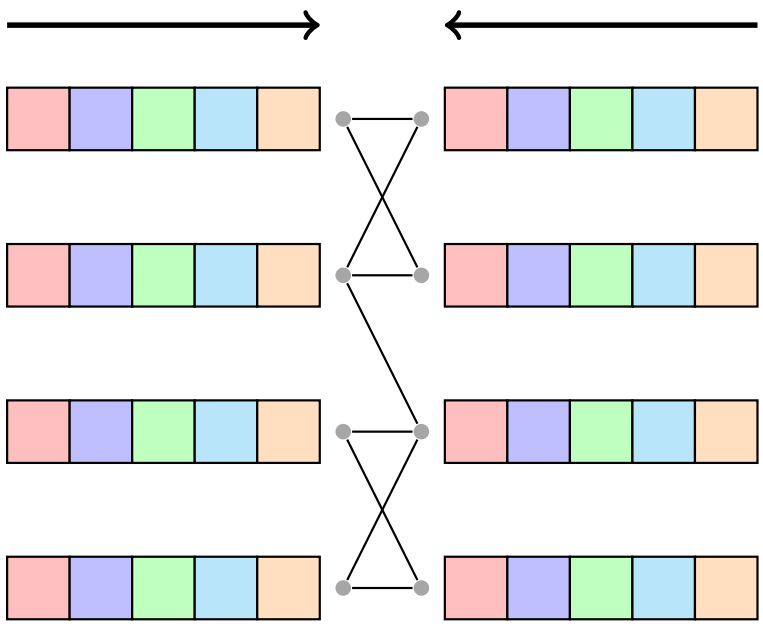
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



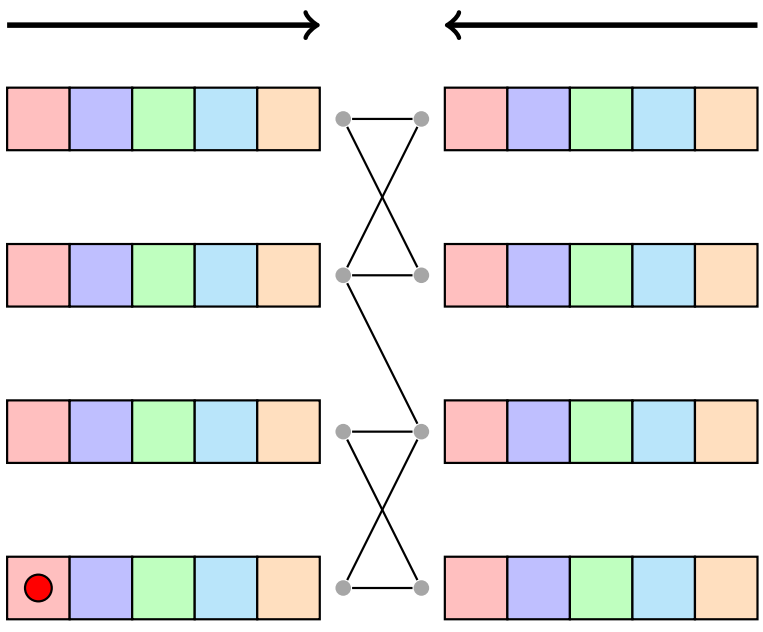
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



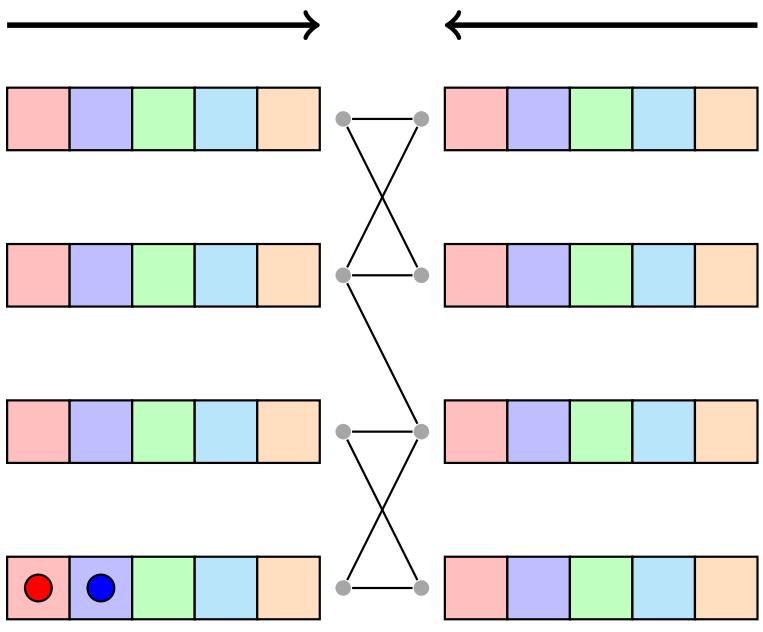
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



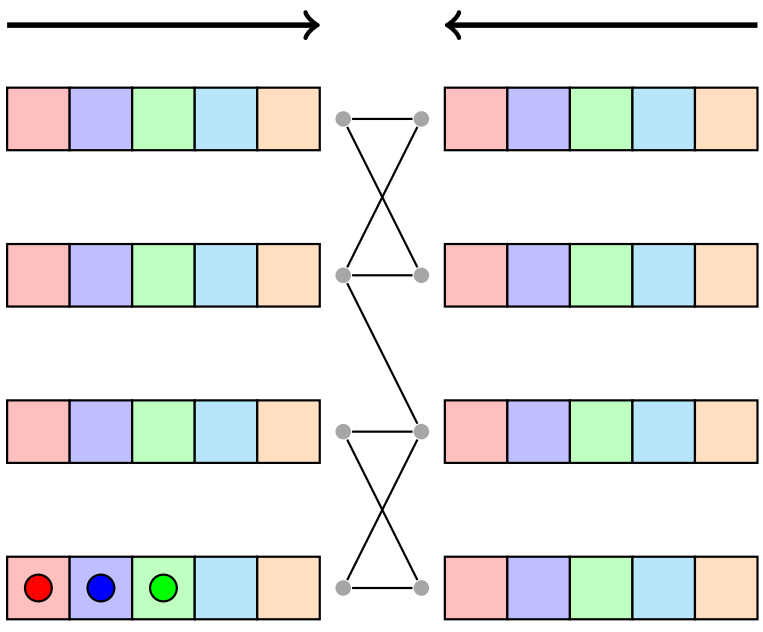
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



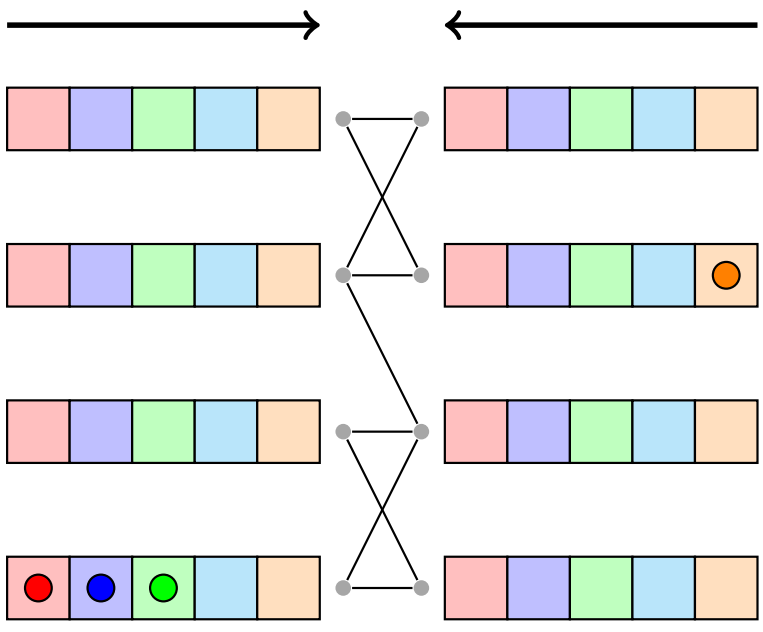
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



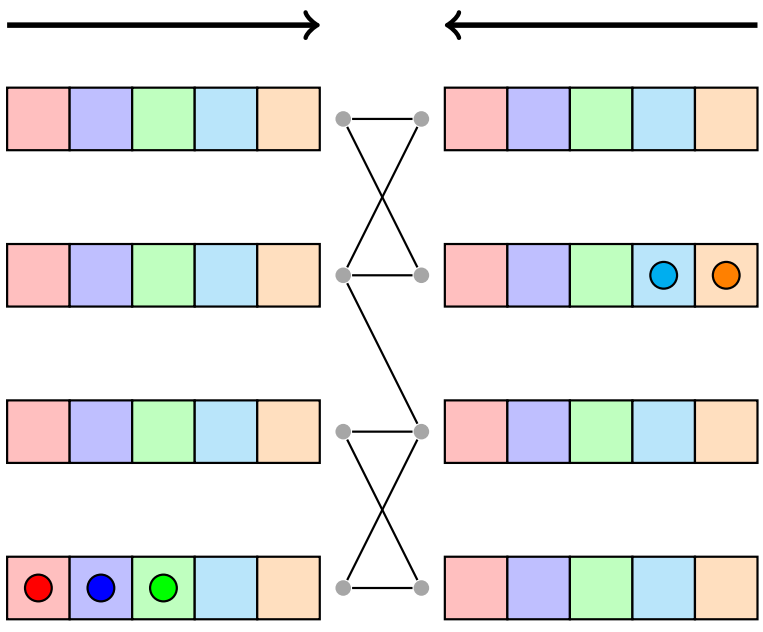
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



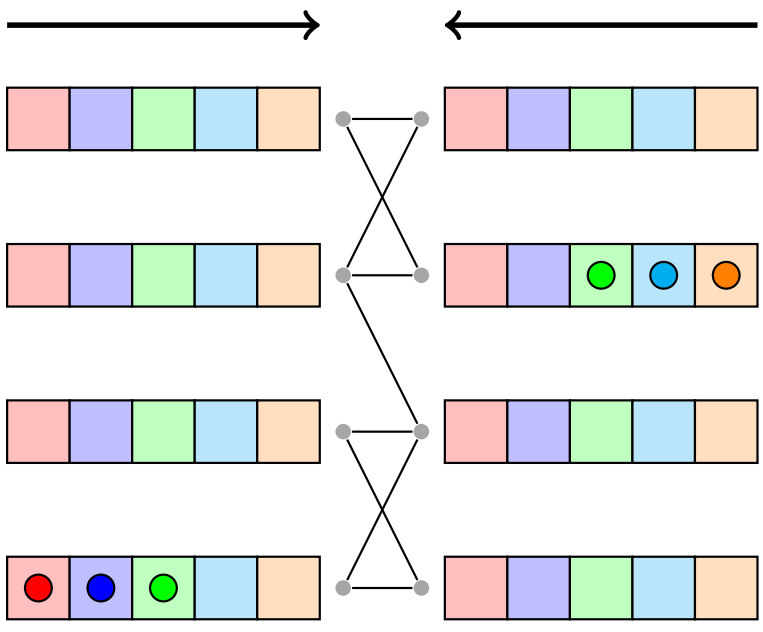
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



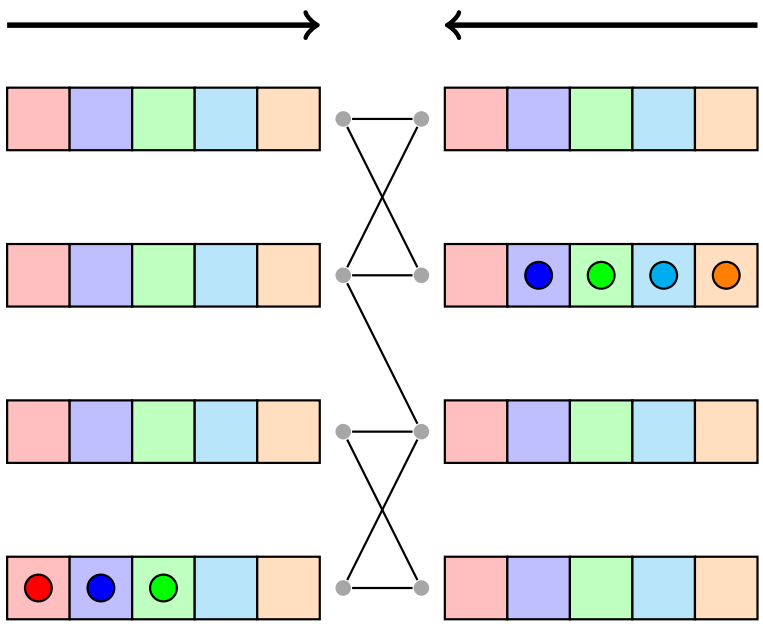
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



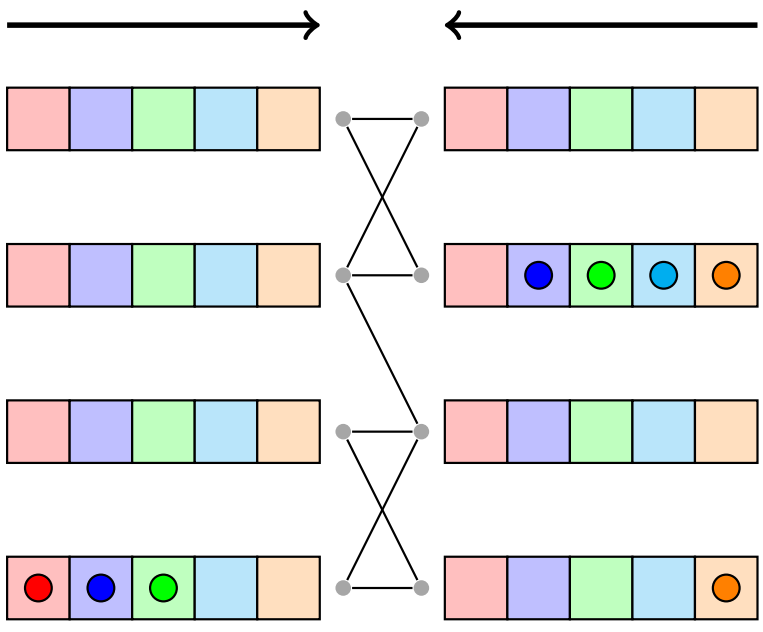
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



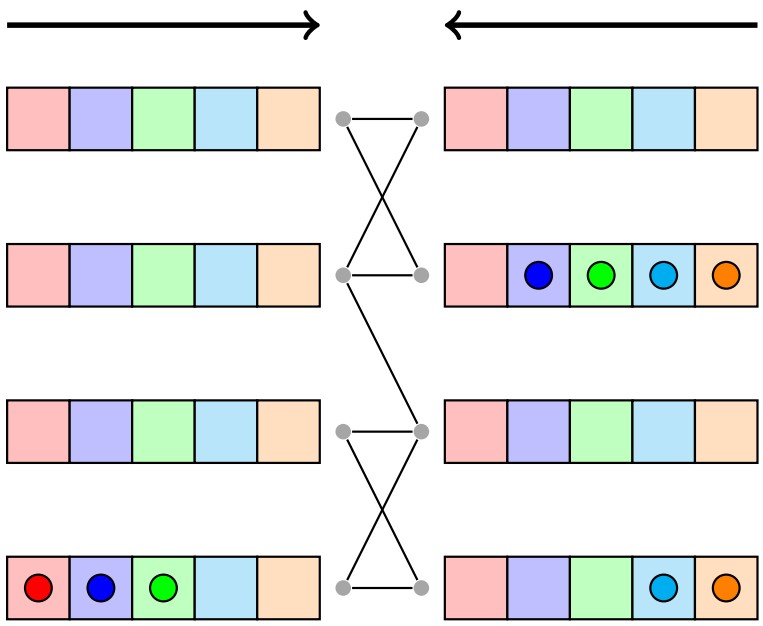
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



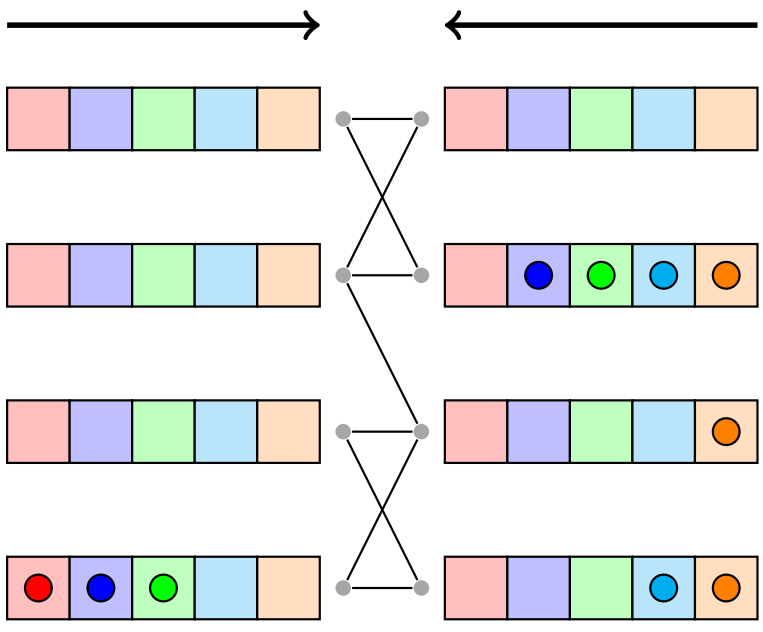
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



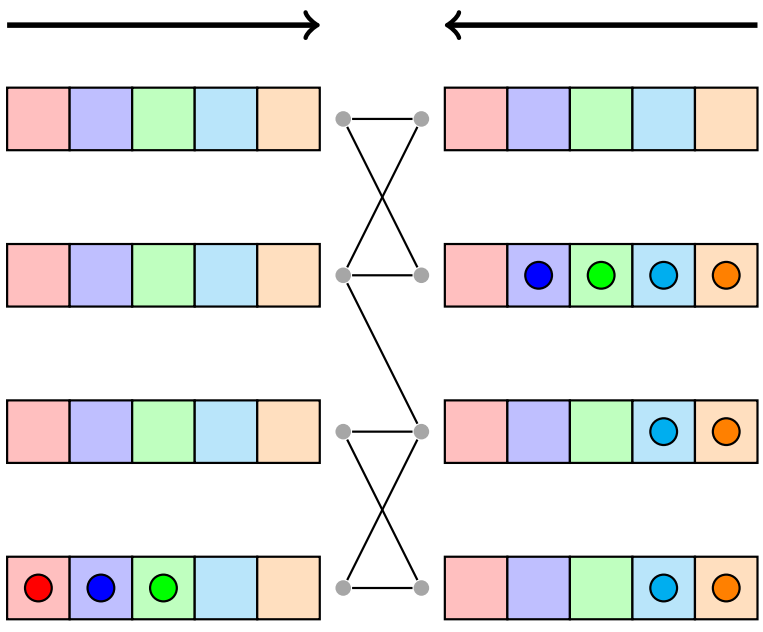
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

Proposition

Every bipartite graph is online-perfect.



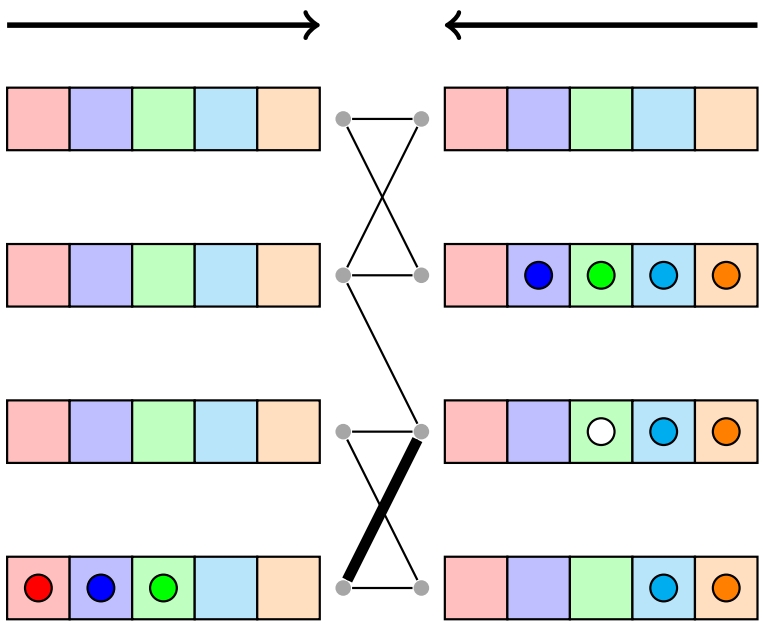
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.

Bipartite graphs

- A graph G is online-perfect if $f(G; w) = w$.

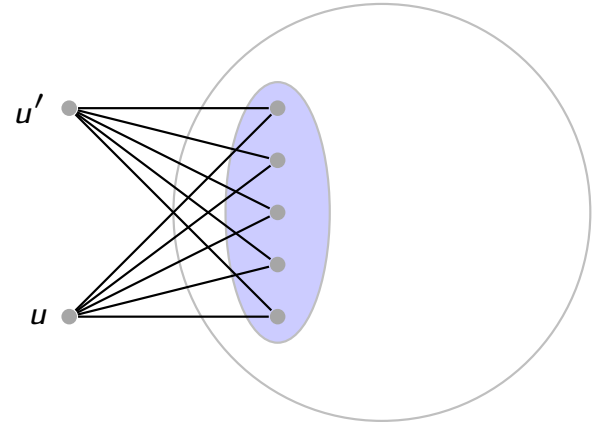
Proposition

Every bipartite graph is online-perfect.



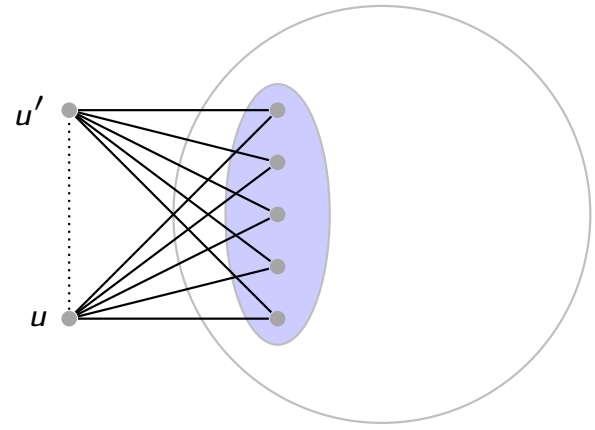
- Choose a linear ordering on a set of w colors.
- Tokens played at the left part are assigned colors greedily in order.
- Tokens played at the right part are assigned colors greedily in reverse order.
- A conflict would imply the token graph has a clique on more than w vertices.

Twins



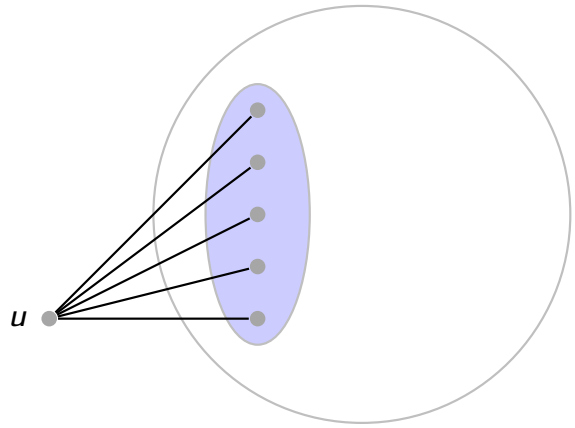
- Vertices u and u' are twins in G if they have the same neighborhood in $G - \{u, u'\}$.

Twins



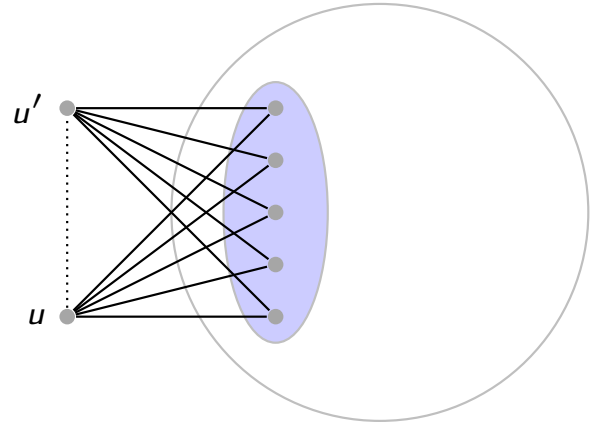
- Vertices u and u' are twins in G if they have the same neighborhood in $G - \{u, u'\}$.
- Both $uu' \in E(G)$ and $uu' \notin E(G)$ are possible.

Twins



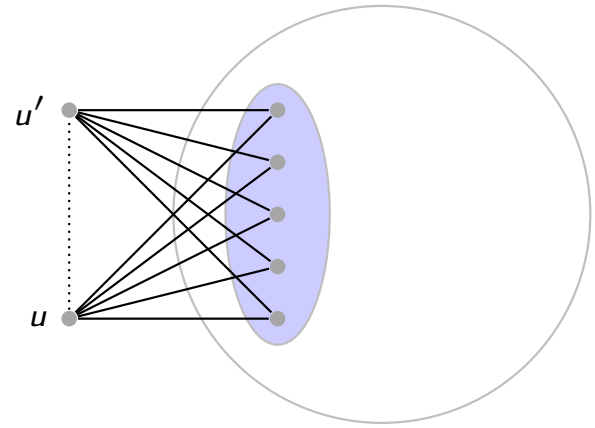
- Vertices u and u' are twins in G if they have the same neighborhood in $G - \{u, u'\}$.
- Both $uu' \in E(G)$ and $uu' \notin E(G)$ are possible.
- Given $u \in V(G)$, we may clone u to produce a new graph G' with an additional vertex u' that is a twin of G.

Twins



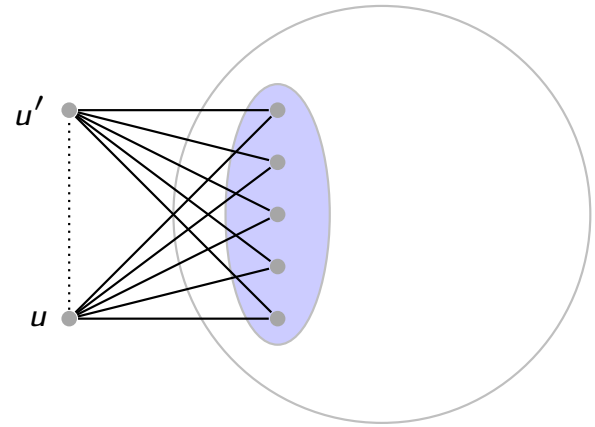
- Vertices u and u' are twins in G if they have the same neighborhood in $G - \{u, u'\}$.
- Both $uu' \in E(G)$ and $uu' \notin E(G)$ are possible.
- Given $u \in V(G)$, we may clone u to produce a new graph G' with an additional vertex u' that is a twin of G.

Twins



- Vertices u and u' are twins in G if they have the same neighborhood in $G - \{u, u'\}$.
- Both $uu' \in E(G)$ and $uu' \notin E(G)$ are possible.
- Given $u \in V(G)$, we may clone u to produce a new graph G' with an additional vertex u' that is a twin of G.
- Fact: a graph G is P_4-free if and only if G is obtainable from a single vertex by cloning.

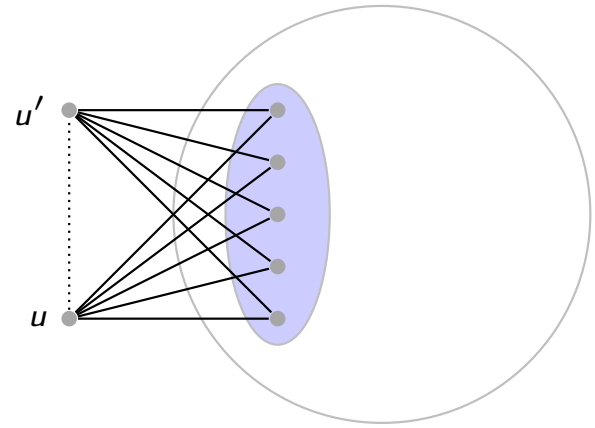
Twins



Proposition

If G' is obtained from G by cloning u, then $f(G'; w) = f(G; w)$.

Twins

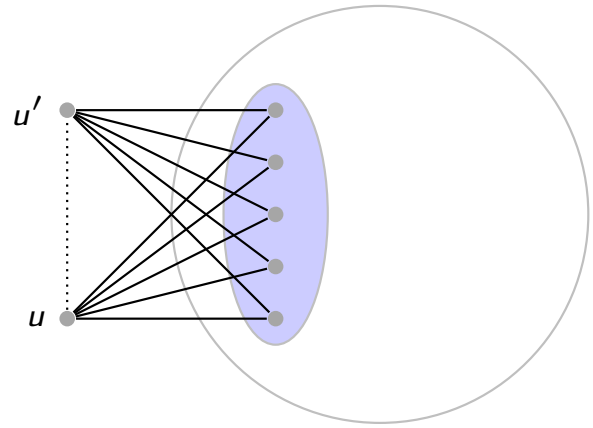


Proposition

If G' is obtained from G by cloning u, then $f(G'; w) = f(G; w)$.

- $f(G; w) \leq f(G'; w)$: clear since G' has an induced copy of G.

Twins

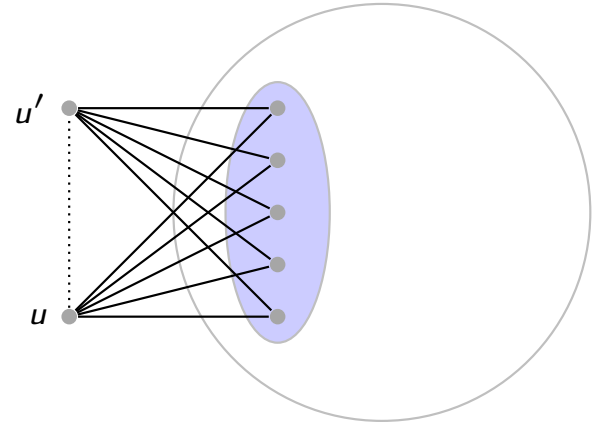


Proposition

If G' is obtained from G by cloning u, then $f(G'; w) = f(G; w)$.

- $f(G; w) \leq f(G'; w)$: clear since G' has an induced copy of G.
- $f(G'; w) \leq f(G; w)$: adapt an optimal strategy for G.

Twins



Proposition

If G' is obtained from G by cloning u, then $f(G'; w) = f(G; w)$.

- $f(G; w) \leq f(G'; w)$: clear since G' has an induced copy of G.
- $f(G'; w) \leq f(G; w)$: adapt an optimal strategy for G.

Corollary

If G is obtainable from a bipartite graph by cloning, then G is online-perfect.

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

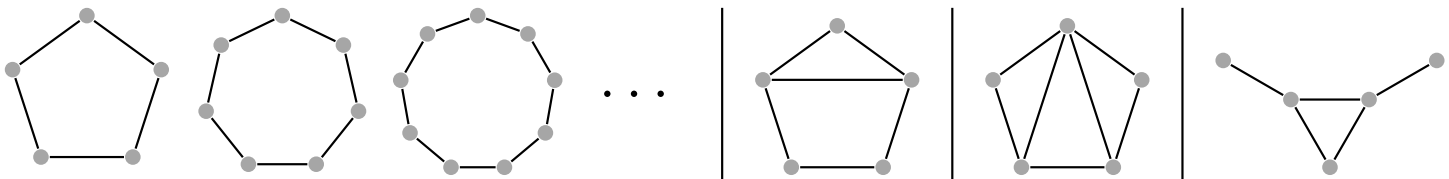
1. *G is online-perfect.*
2. $f(G;2) = 2$.

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



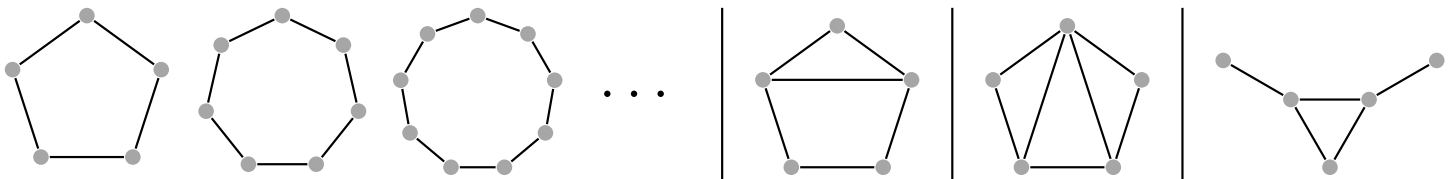
C_n for odd n at least 5 | C_5^+ | P_5^2 | *The bull*

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



C_n *for odd* n *at least* 5	C_5^+	P_5^2	*The bull*

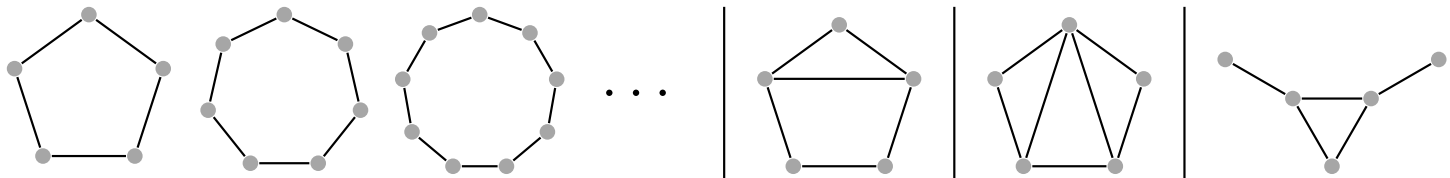
4. *G is obtainable from a bipartite graph by cloning vertices.*

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



C_n *for odd* n *at least* 5	C_5^+	P_5^2	*The bull*

4. *G is obtainable from a bipartite graph by cloning vertices.*

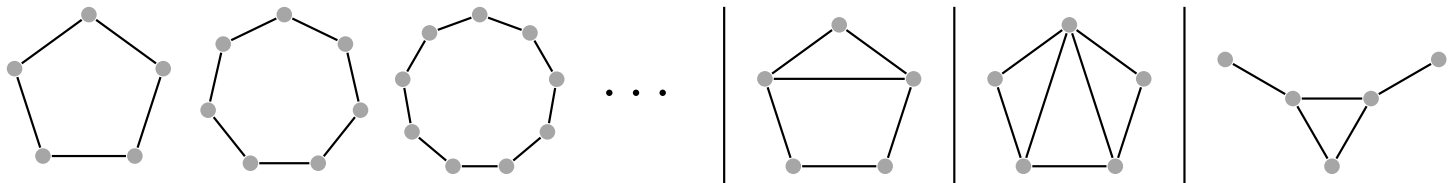
- (1) $\rightarrow$ (2): clear

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



C_n for odd n at least 5 | *C_5^+* | *P_5^2* | *The bull*

4. *G is obtainable from a bipartite graph by cloning vertices.*

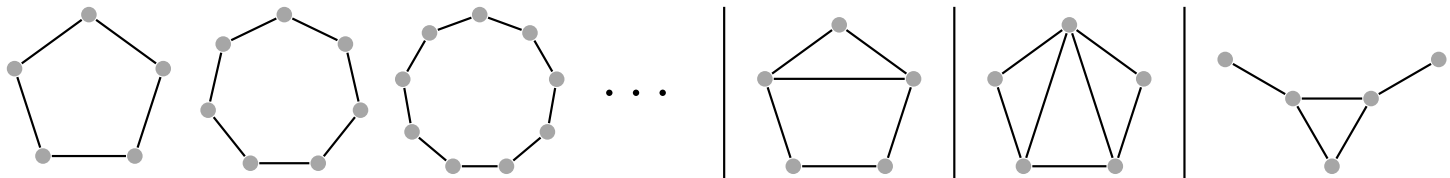
- (1) $\to$ (2): clear
- (2) $\to$ (3): Spoiler Lemma

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



C_n *for odd* n *at least* 5	C_5^+	P_5^2	*The bull*

4. *G is obtainable from a bipartite graph by cloning vertices.*

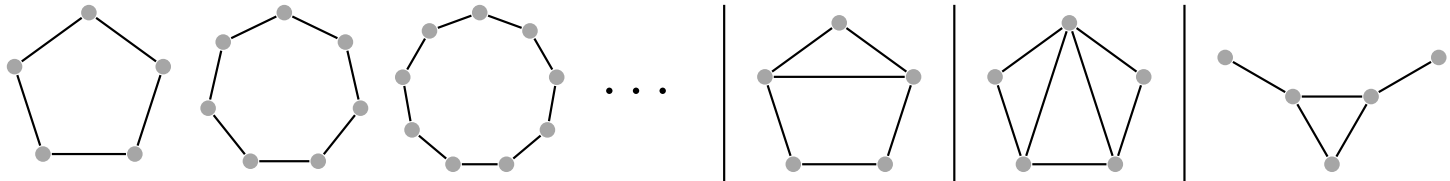
- (1) $\to$ (2): clear
- (2) $\to$ (3): Spoiler Lemma
- (3) $\to$ (4): roughly a page of structural graph theory.

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



C_n *for odd n at least* 5	C_5^+	P_5^2	*The bull*

4. *G is obtainable from a bipartite graph by cloning vertices.*

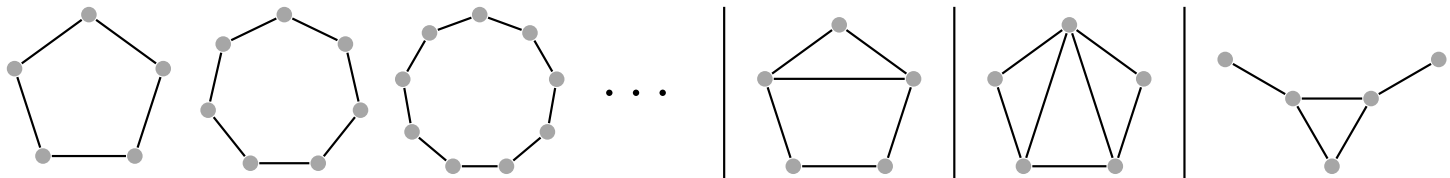
- (1) $\to$ (2): clear
- (2) $\to$ (3): Spoiler Lemma
- (3) $\to$ (4): roughly a page of structural graph theory.
- (4) $\to$ (1): previous corollary

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



C_n for odd n at least 5 | C_5^+ | P_5^2 | *The bull*

4. *G is obtainable from a bipartite graph by cloning vertices.*

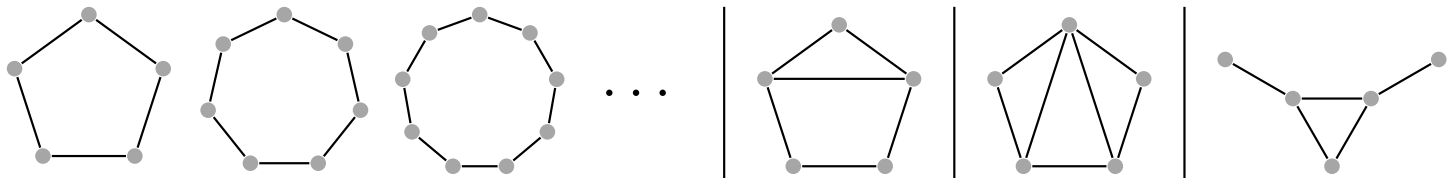
- Cor: P_4-free graphs $\subsetneq$ online-perfect graphs $\subsetneq$ perfect graphs

Characterization of online-perfect graphs

Theorem

Let G be a graph. The following are equivalent.

1. *G is online-perfect.*
2. *$f(G;2) = 2$.*
3. *G does not have an induced copy of any of the following:*



C_n for odd n at least 5 | *C_5^+* | *P_5^2* | *The bull*

4. *G is obtainable from a bipartite graph by cloning vertices.*

- Cor: P_4-free graphs $\subsetneq$ online-perfect graphs $\subsetneq$ perfect graphs
- 4': G is online-perfect if and only if it the result of replacing each vertex in a bipartite graph with a P_4-free graph.

A Spoiler strategy

Lemma (Spoiler Lemma)

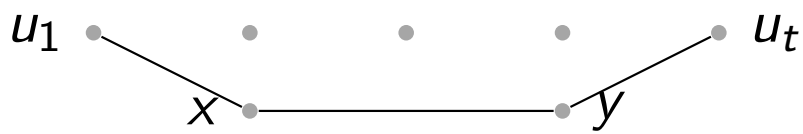
Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

A Spoiler strategy

Lemma (Spoiler Lemma)

Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*

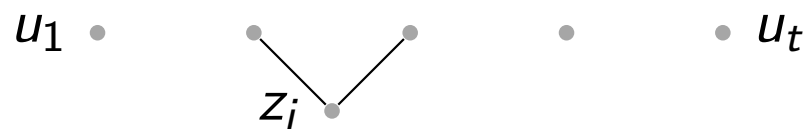


A Spoiler strategy

Lemma (Spoiler Lemma)

Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*



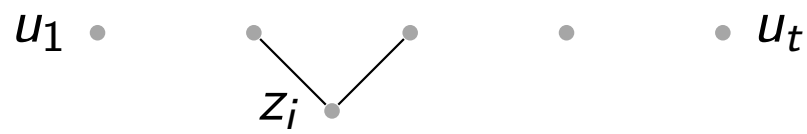
A Spoiler strategy

Lemma (Spoiler Lemma)

Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*

If w is even, then $f(G; w) \geq (1 + \frac{1}{2t})w$.



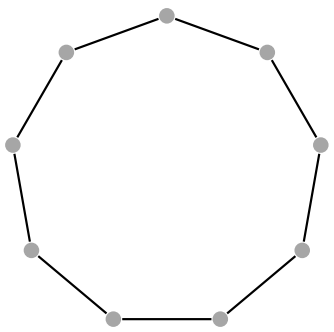
Minimal non-online-perfect graphs: lower bounds

Lemma (Spoiler Lemma)

Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*

If w is even, then $f(G; w) \geq (1 + \frac{1}{2t})w$.



C_n

Corollary

If n is odd and $n \geq 5$ and w is even, then $f(C_n; w) \geq \frac{n}{n-1}w$.

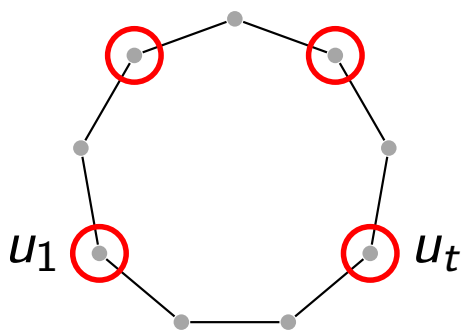
Minimal non-online-perfect graphs: lower bounds

Lemma (Spoiler Lemma)

Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*

If w is even, then $f(G; w) \geq (1 + \frac{1}{2t})w$.



C_n

Corollary

If n is odd and $n \geq 5$ and w is even, then $f(C_n; w) \geq \frac{n}{n-1}w$.

- Apply Spoiler Lemma with $t = (n-1)/2$.

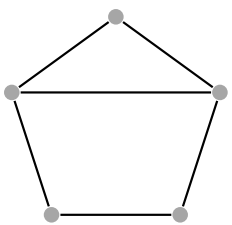
Minimal non-online-perfect graphs: lower bounds

Lemma (Spoiler Lemma)

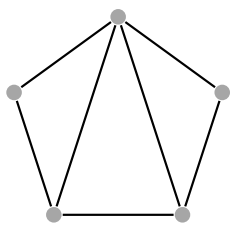
Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*

If w is even, then $f(G; w) \geq (1 + \frac{1}{2t})w$.



C_5^+



P_5^2

Corollary

If $G \in \{C_5^+, P_5^2\}$ and w is even, then $f(G; w) \geq \frac{5}{4}w$.

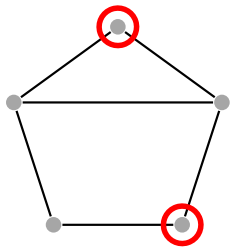
Minimal non-online-perfect graphs: lower bounds

Lemma (Spoiler Lemma)

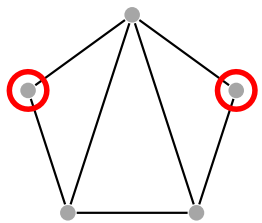
Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*

If w is even, then $f(G; w) \geq (1 + \frac{1}{2t})w$.



C_5^+



P_5^2

Corollary

If $G \in \{C_5^+, P_5^2\}$ and w is even, then $f(G; w) \geq \frac{5}{4}w$.

- Apply Spoiler Lemma with $t = 2$.

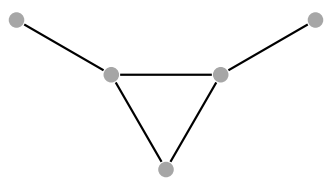
Minimal non-online-perfect graphs: lower bounds

Lemma (Spoiler Lemma)

Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*

If w is even, then $f(G; w) \geq (1 + \frac{1}{2t})w$.



Corollary

If G is the bull graph and w is even, then $f(G; w) \geq \frac{7}{6}w$.

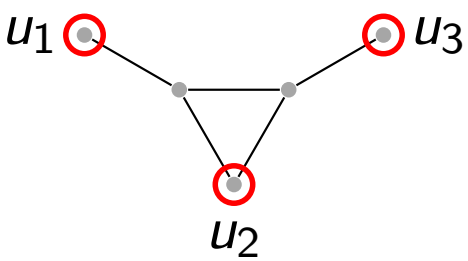
Minimal non-online-perfect graphs: lower bounds

Lemma (Spoiler Lemma)

Let $U \subseteq V(G)$, where $U = \{u_1, \ldots, u_t\}$, and suppose that:

1. *There are vertices x and y such that u_1xyu_t is a path and $G[\{u_1, x, y, u_t\}]$ is bipartite, and*
2. *For each i, there is a common neighbor z_i of u_i and u_{i+1} such that $G[U \cup \{z_i\}]$ is bipartite.*

If w is even, then $f(G; w) \geq (1 + \frac{1}{2t})w$.



Corollary

If G is the bull graph and w is even, then $f(G; w) \geq \frac{7}{6}w$.

- Apply Spoiler Lemma with $t = 3$.

Minimal non-online-perfect graphs: upper bounds

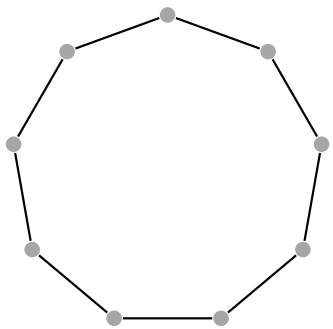
Proposition (Fractional Coloring Strategy)

If G has a (p, q)-coloring, then $f(G; w) \leq p \left\lceil \frac{w}{2q} \right\rceil$. In particular, $f(G; w) \leq (\frac{1}{2}\chi_f(G) + o(1))w$.

Minimal non-online-perfect graphs: upper bounds

Proposition (Fractional Coloring Strategy)

If G has a (p, q)-coloring, then $f(G; w) \leq p \left\lceil \frac{w}{2q} \right\rceil$. In particular, $f(G; w) \leq (\frac{1}{2}\chi_f(G) + o(1))w$.



C_n

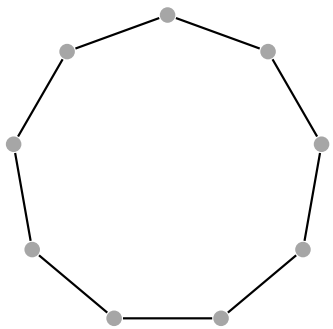
Corollary

If n is odd, then $f(C_n; w) \leq (\frac{n}{n-1} + o(1))w$.

Minimal non-online-perfect graphs: upper bounds

Proposition (Fractional Coloring Strategy)

If G has a (p, q)-coloring, then $f(G; w) \leq p \left\lceil \frac{w}{2q} \right\rceil$. In particular, $f(G; w) \leq (\frac{1}{2}\chi_f(G) + o(1))w$.



C_n

Corollary

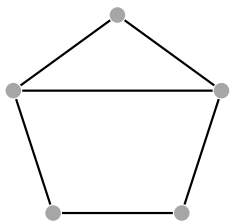
If n is odd, then $f(C_n; w) \leq (\frac{n}{n-1} + o(1))w$.

- Apply Prop. with $\chi_f(C_n) = \frac{2n}{n-1}$.

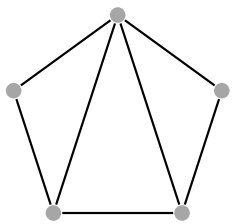
Minimal non-online-perfect graphs: upper bounds

Proposition (Fractional Coloring Strategy)

If G has a (p, q)-coloring, then $f(G; w) \leq p \left\lceil \frac{w}{2q} \right\rceil$. In particular, $f(G; w) \leq (\frac{1}{2}\chi_f(G) + o(1))w$.



C_5^+



P_5^2

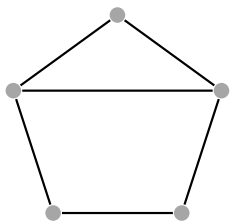
Corollary

If $G \in \{C_5^+, P_5^2\}$, *then* $f(G; w) \leq (\frac{3}{2} + o(1))w$.

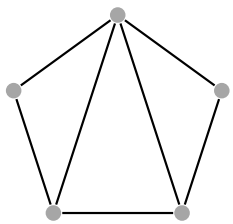
Minimal non-online-perfect graphs: upper bounds

Proposition (Fractional Coloring Strategy)

If G has a (p, q)-coloring, then $f(G; w) \leq p \left\lceil \frac{w}{2q} \right\rceil$. In particular, $f(G; w) \leq (\frac{1}{2}\chi_f(G) + o(1))w$.



C_5^+



P_5^2

Corollary

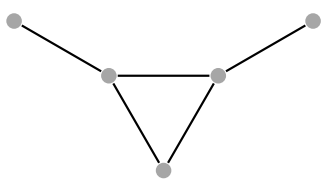
If $G \in \{C_5^+, P_5^2\}$, *then* $f(G; w) \leq (\frac{3}{2} + o(1))w$.

- Apply Prop. with $\chi_f(G) = \chi(G) = 3$.

Minimal non-online-perfect graphs: upper bounds

Proposition (Fractional Coloring Strategy)

If G has a (p, q)-coloring, then $f(G; w) \leq p \left\lceil \frac{w}{2q} \right\rceil$. In particular, $f(G; w) \leq (\frac{1}{2}\chi_f(G) + o(1))w$.



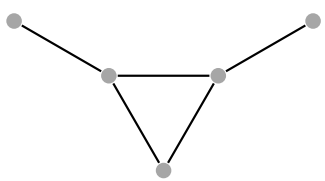
Corollary

If G is the bull graph and w is even, then $f(G; w) \leq (\frac{3}{2} + o(1))w$.

Minimal non-online-perfect graphs: upper bounds

Proposition (Fractional Coloring Strategy)

If G has a (p, q)-coloring, then $f(G; w) \leq p \left\lceil \frac{w}{2q} \right\rceil$. In particular, $f(G; w) \leq (\frac{1}{2}\chi_f(G) + o(1))w$.



Corollary

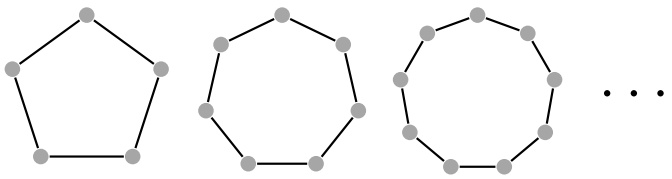
If G is the bull graph and w is even, then $f(G; w) \leq (\frac{3}{2} + o(1))w$.

- Apply Prop. with $\chi_f(G) = \chi(G) = 3$.

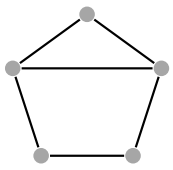
Summary

Theorem

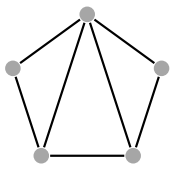
A graph is online-perfect if and only if it does not have an induced copy of any of the following:



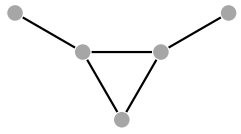
C_n *for odd* n *at least* 5



C_5^+



P_5^2

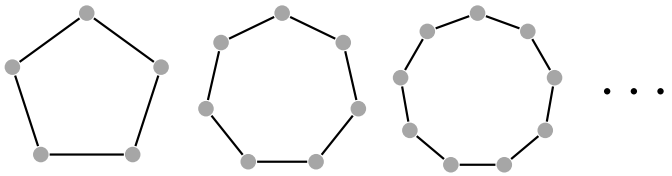


The bull

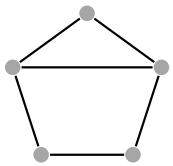
Summary

Theorem

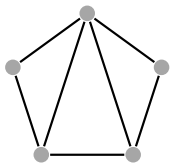
A graph is online-perfect if and only if it does not have an induced copy of any of the following:



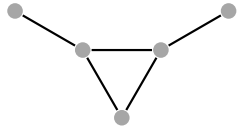
C_n *for odd* n *at least* 5



C_5^+



P_5^2



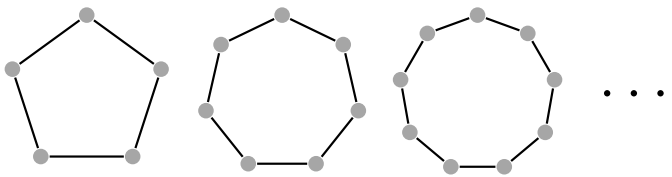
The bull

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.

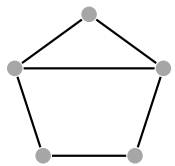
Summary

Theorem

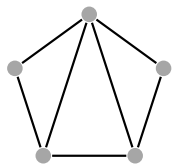
A graph is online-perfect if and only if it does not have an induced copy of any of the following:



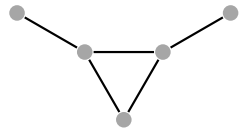
C_n *for odd* n *at least* 5



C_5^+



P_5^2



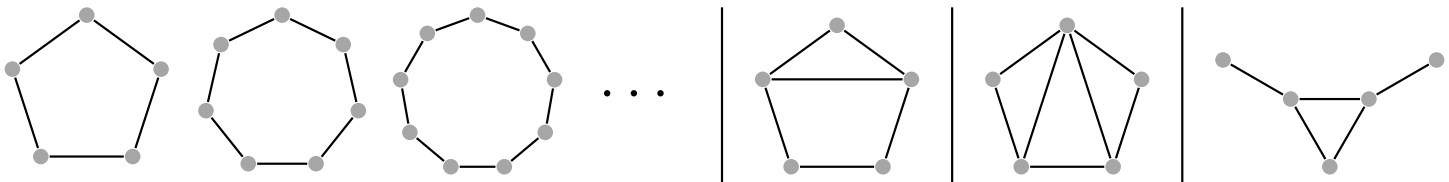
The bull

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.
- $(\frac{5}{4} - o(1))w \leq f(C_5^+; w) \leq (\frac{3}{2} + o(1))w$.

Summary

Theorem

A graph is online-perfect if and only if it does not have an induced copy of any of the following:



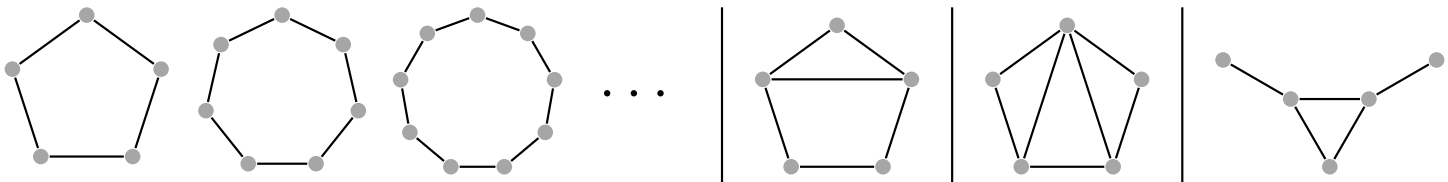
C_n *for odd* n *at least* 5 | C_5^+ | P_5^2 | *The bull*

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.
- $(\frac{5}{4} - o(1))w \leq f(C_5^+; w) \leq (\frac{3}{2} + o(1))w$.
- Thm: $f(P_5^2; w) = (\frac{5}{4} + o(1))w$.

Summary

Theorem

A graph is online-perfect if and only if it does not have an induced copy of any of the following:



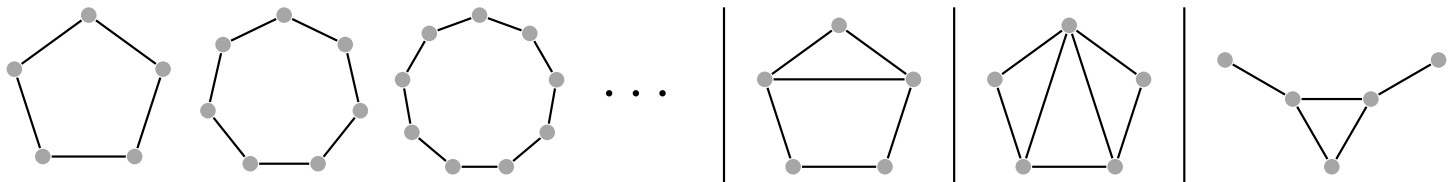
C_n *for odd* n *at least* 5 | C_5^+ | P_5^2 | *The bull*

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.
- $(\frac{5}{4} - o(1))w \leq f(C_5^+; w) \leq (\frac{3}{2} + o(1))w$.
- Thm: $f(P_5^2; w) = (\frac{5}{4} + o(1))w$.
- For the bull B: $(\frac{7}{6} - o(1))w \leq f(B; w) \leq (\frac{3}{2} + o(1))w$.

Summary

Theorem

A graph is online-perfect if and only if it does not have an induced copy of any of the following:



C_n *for odd* n *at least* 5 | C_5^+ | P_5^2 | *The bull*

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.
- $(\frac{5}{4} - o(1))w \leq f(C_5^+; w) \leq (\frac{3}{2} + o(1))w$.
- Thm: $f(P_5^2; w) = (\frac{5}{4} + o(1))w$.
- For the bull B: $(\frac{7}{6} - o(1))w \leq f(B; w) \leq (\frac{3}{2} + o(1))w$.

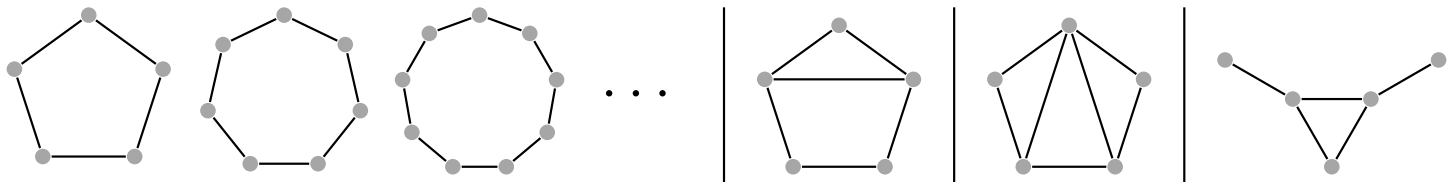
Open Problems

Determine the asymptotics of $f(C_5^+; w)$,

Summary

Theorem

A graph is online-perfect if and only if it does not have an induced copy of any of the following:



C_n *for odd* n *at least* 5 | C_5^+ | P_5^2 | *The bull*

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.
- $(\frac{5}{4} - o(1))w \leq f(C_5^+; w) \leq (\frac{3}{2} + o(1))w$.
- Thm: $f(P_5^2; w) = (\frac{5}{4} + o(1))w$.
- For the bull B: $(\frac{7}{6} - o(1))w \leq f(B; w) \leq (\frac{3}{2} + o(1))w$.

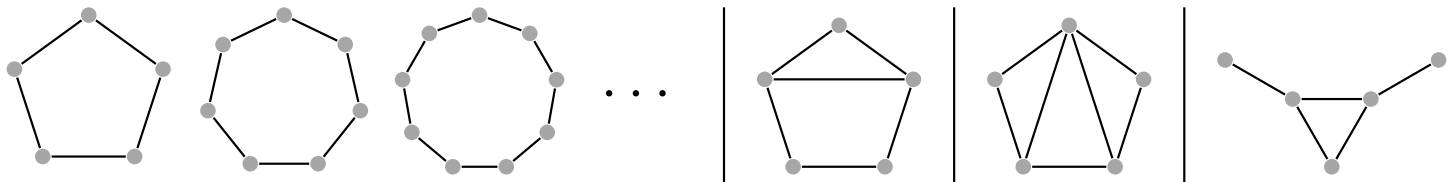
Open Problems

Determine the asymptotics of $f(C_5^+; w)$, $f(B; w)$,

Summary

Theorem

A graph is online-perfect if and only if it does not have an induced copy of any of the following:



C_n *for odd* n *at least* 5 | C_5^+ | P_5^2 | *The bull*

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.
- $(\frac{5}{4} - o(1))w \leq f(C_5^+; w) \leq (\frac{3}{2} + o(1))w$.
- Thm: $f(P_5^2; w) = (\frac{5}{4} + o(1))w$.
- For the bull B: $(\frac{7}{6} - o(1))w \leq f(B; w) \leq (\frac{3}{2} + o(1))w$.

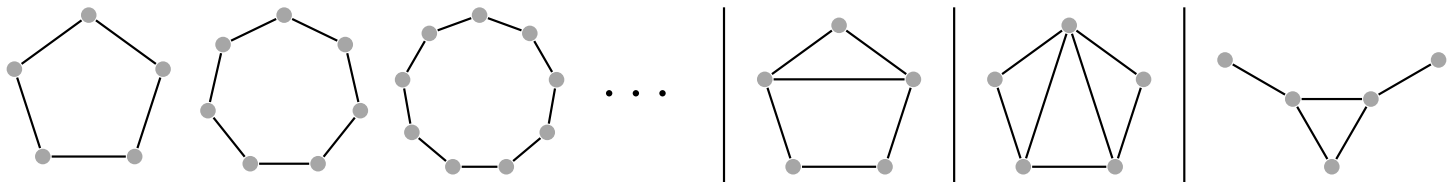
Open Problems

Determine the asymptotics of $f(C_5^+; w)$, $f(B; w)$, $f(\mathbb{G}; w)$.

Summary

Theorem

A graph is online-perfect if and only if it does not have an induced copy of any of the following:



C_n *for odd* n *at least* 5 | C_5^+ | P_5^2 | *The bull*

- For odd n and $n \geq 5$, we have $f(C_n; w) = (\frac{n}{n-1} + o(1))w$.
- $(\frac{5}{4} - o(1))w \leq f(C_5^+; w) \leq (\frac{3}{2} + o(1))w$.
- Thm: $f(P_5^2; w) = (\frac{5}{4} + o(1))w$.
- For the bull B: $(\frac{7}{6} - o(1))w \leq f(B; w) \leq (\frac{3}{2} + o(1))w$.

Open Problems

Determine the asymptotics of $f(C_5^+; w)$, $f(B; w)$, $f(\mathbb{G}; w)$.

Thank You.