On graphs coverable with k shortest paths

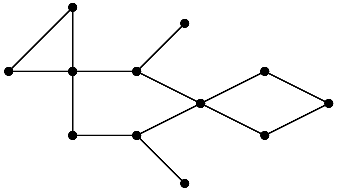
Maël Dumas[1], Florent Foucaud[2], Anthony Perez[1], Ioan Todinca[1]

[1]LIFO, Université d'Orléans, France
[2]LIMOS, Université Clermont Auvergne, France

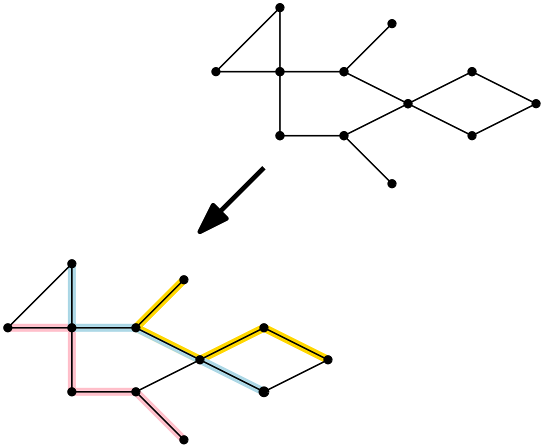
1. Graphs coverable by k shortest paths
   - Graphs edge-coverable by k shortest paths
   - Graphs vertex-coverable by k shortest paths
2. Algorithmic Consequences
3. Conclusion

Graphs coverable by k shortest paths

Graphs coverable by k shortest paths

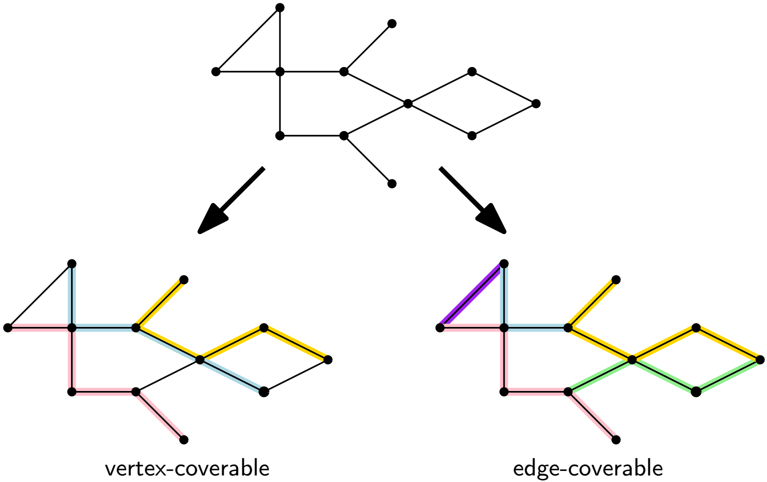


Graphs coverable by k shortest paths



vertex-coverable

Graphs coverable by k shortest paths



Combinatorial results

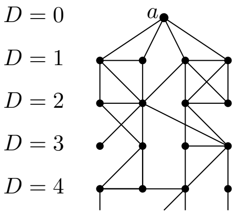
Theorem 1

If G is **coverable by k shortest paths** then, for any vertex a and fixed distance D, the number of vertices at distance exactly D from a is upper bounded by some function $g(k)$.

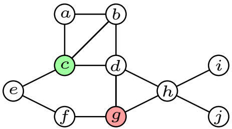
- Edge-coverable : $g(k) = O(3^k)$.
- Vertex-coverable : $g(k) = O(k \cdot 3^k)$.

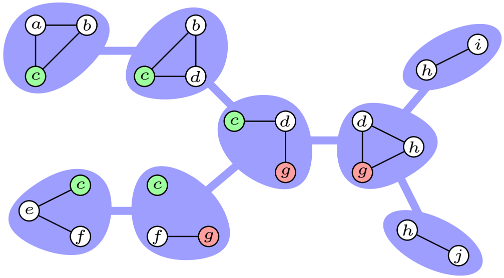
Corollary 1

G is of **treewidth** at most $2 \cdot g(k) - 1$.



- **Tree decomposition** :
  - Each vertex and each edge is in at least on bag
  - The bags containing a vertex v induce a connected subtree





- Width of a decomposition : size of the biggest bag -1
- **Treewidth** : smallest width among all tree decomposition

Combinatorial results

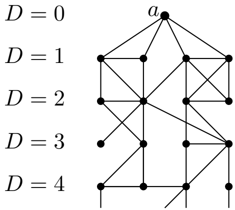
Theorem 1

If G is **coverable by** k **shortest paths** then, for any vertex a and fixed distance D, the number of vertices at distance exactly D from a is upper bounded by some function $g(k)$.

- Edge-coverable : $g(k) = O(3^k)$.
- Vertex-coverable : $g(k) = O(k \cdot 3^k)$.

Corollary 1

G is of **treewidth** at most $2 \cdot g(k) - 1$.



Tree decomposition :

Combinatorial results

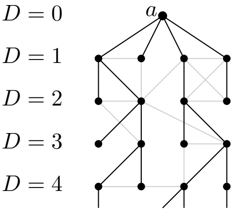
Theorem 1

If G is **coverable by k shortest paths** then, for any vertex a and fixed distance D, the number of vertices at distance exactly D from a is upper bounded by some function $g(k)$.

- Edge-coverable : $g(k) = O(3^k)$.
- Vertex-coverable : $g(k) = O(k \cdot 3^k)$.

Corollary 1

G is of **treewidth** at most $2 \cdot g(k) - 1$.



Tree decomposition :

- Do a a breadth-first search (BFS) from a vertex a.

Combinatorial results

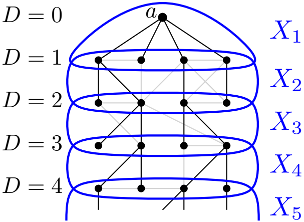
Theorem 1

If G is **coverable by k shortest paths** then, for any vertex a and fixed distance D, the number of vertices at distance exactly D from a is upper bounded by some function $g(k)$.

- Edge-coverable : $g(k) = O(3^k)$.
- Vertex-coverable : $g(k) = O(k \cdot 3^k)$.

Corollary 1

G is of **treewidth** at most $2 \cdot g(k) - 1$.



Tree decomposition :

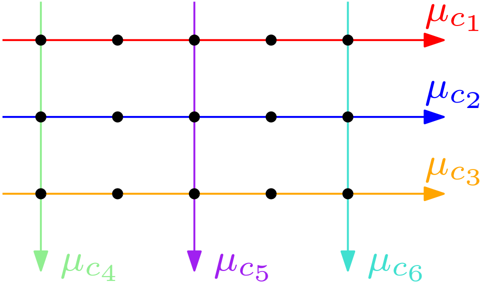
- Do a a breadth-first search (BFS) from a vertex a.
- Each bag : two consecutive layers.

Graphs edge-coverable by k shortest paths

Base paths colouring

Base paths : k shortest paths $\mu_1, \ldots, \mu_k$ that cover the graph
To each base path μ_c we give :

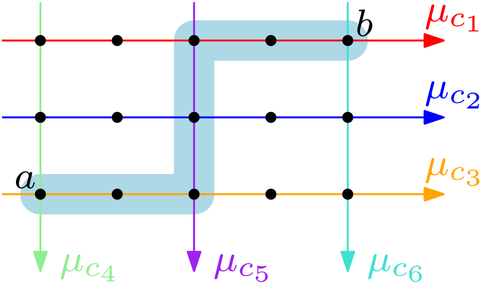
- A **colour** c, $1 \leq c \leq k$,
- An arbitrary **direction**.



For an edge e of the graph : colours(e) $= \{i \mid e \in \mu_i\}$.

Colouring of a path

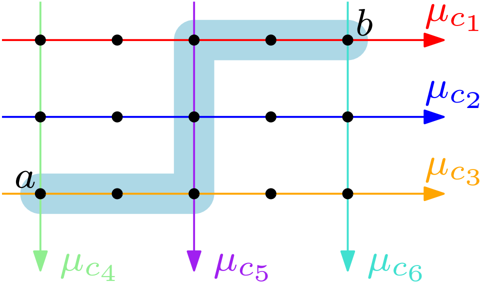
Colouring col of P : $\forall e \in E(P)$, $\mathrm{col}(e) \in \mathrm{colours}(e)$
Coloured path : (P, col)



Colours-signs word

For a coloured path (P, col) :

- divide it in monochromatic subpath of colour c_i,
- each subpath induce a "+" sign or a "−" sign w.r.t. the direction of μ_{c_i}.



$$\omega = ((c_3, +), (c_5, -), (c_1, +))$$

Good colouring of a path

Well-coloured : the edges using a colour c form a **connected subpath** of P.

A path well-coloured :

A path not well-coloured :

Good colouring of a path

Good colouring Lemma

For every pair of vertices a, b, there exists a **shortest well-coloured** a-b path.

Good colouring of a path

Good colouring Lemma

For every pair of vertices a, b, there exists a **shortest well-coloured** a-b path.

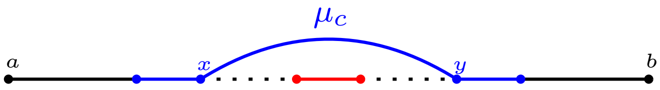
If a a-b shortest path (P, col) isn't well-coloured :

Good colouring of a path

Good colouring Lemma

For every pair of vertices a, b, there exists a **shortest well-coloured** a-b path.

If a a-b shortest path (P, col) isn't well-coloured :



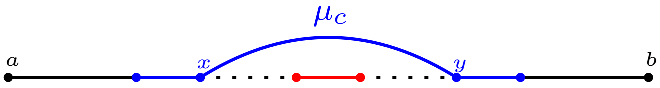
$\Rightarrow$ Replace $P[x, y]$ by $\mu_c[x, y]$.

Good colouring of a path

Good colouring Lemma

For every pair of vertices a, b, there exists a **shortest well-coloured** a-b path.

If a a-b shortest path (P, col) isn't well-coloured :



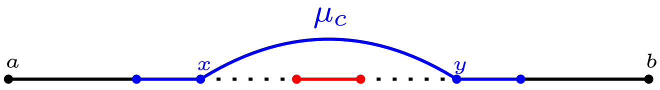
$\Rightarrow$ Replace $P[x, y]$ by $\mu_c[x, y]$.
The constructed path is still a shortest path ($|\mu_c[x, y]| \leq |P[x, y]|$).

Good colouring of a path

Good colouring Lemma

For every pair of vertices a, b, there exists a **shortest well-coloured** a-b path.

If a a-b shortest path (P, col) isn't well-coloured :

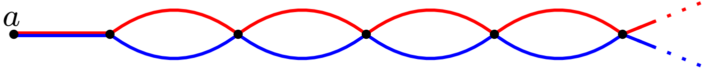


$\Rightarrow$ Replace $P[x, y]$ by $\mu_c[x, y]$.
The constructed path is still a shortest path ($|\mu_c[x, y]| \leq |P[x, y]|$).

The number of colours-signs words possible on every well-coloured paths is upper bounded by $\sum_{l=1}^{k} 2^l \frac{k!}{(k-l)!} = O(k^k)$.

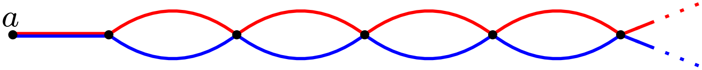
A first bound

Multiple shortest paths of same length may have the same colours-signs word :



A first bound

Multiple shortest paths of same length may have the same colours-signs word :

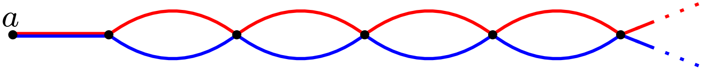


Colours-signs word Lemma

The shortest paths starting at a vertex a, of length D and colours-signs word ω all ends at the same vertex b.

A first bound

Multiple shortest paths of same length may have the same colours-signs word :



Colours-signs word Lemma

The shortest paths starting at a vertex a, of length D and colours-signs word ω all ends at the same vertex b.

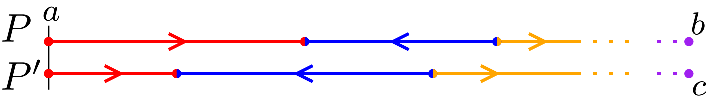
(weak) Theorem 1

For any vertex a and any fixed distance D, the number of vertices at distance exactly D from a is upper bounded by $O(k^k)$ (number of colours-signs words).

Proof of the Colours-signs word Lemma

Let b and c be vertices at same distance from a vertex a of G.
Let (P, col), (P', col') be a well-coloured shortest a-b and a-c paths.

Claim : If they have the same colours-signs word, then $b = c$.

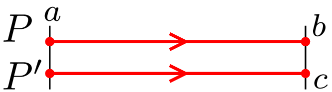


Proof of the Colours-signs word Lemma

Let b and c be vertices at same distance from a vertex a of G.
Let (P, col), (P', col') be a well-coloured shortest a-b and a-c paths.

Claim : If they have the same colours-signs word, then $b = c$.

Proof by induction on the number of letters ℓ of the colours-signs word.
If $\ell = 1$



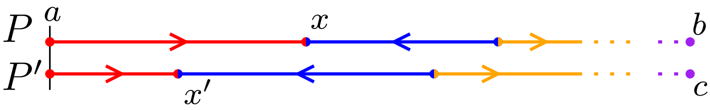
$dist(a, b) = dist(a, c)$ thus $b = c$.

Proof of the Colours-signs word Lemma

Let b and c be vertices at same distance from a vertex a of G.
Let (P, col), (P', col') be a well-coloured shortest a-b and a-c paths.

Claim : If they have the same colours-signs word, then $b = c$.

Proof by induction on the number of letters ℓ of the colours-signs word.
If $\ell > 1$



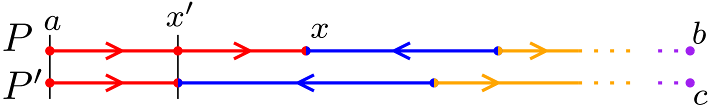
Take P the path with the longest subpath of the colour c_1.

Proof of the Colours-signs word Lemma

Let b and c be vertices at same distance from a vertex a of G.
Let (P, col), (P', col') be a well-coloured shortest a-b and a-c paths.

Claim : If they have the same colours-signs word, then $b = c$.

Proof by induction on the number of letters ℓ of the colours-signs word.
If $\ell > 1$



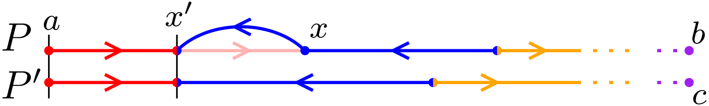
The vertex x' is in the path $\mu_{c_1}[a, x]$, thus in the path P

Proof of the Colours-signs word Lemma

Let b and c be vertices at same distance from a vertex a of G.
Let (P, col), (P', col') be a well-coloured shortest a-b and a-c paths.

Claim : If they have the same colours-signs word, then $b = c$.

Proof by induction on the number of letters ℓ of the colours-signs word.
If $\ell > 1$

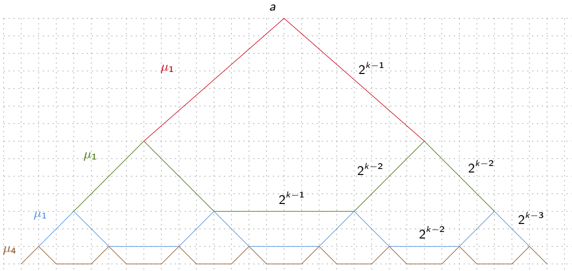


Replace $P[x', x]$ by $\mu_{c_2}[x', x]$.
$P[x', b]$ and $P'[x', c]$ have $\ell - 1$ colours, by the induction hypothesis, $b = c$.

A better bound ?

We shown an the upper bound : $O(k^k)$

Lower bound : $O(2^k)$

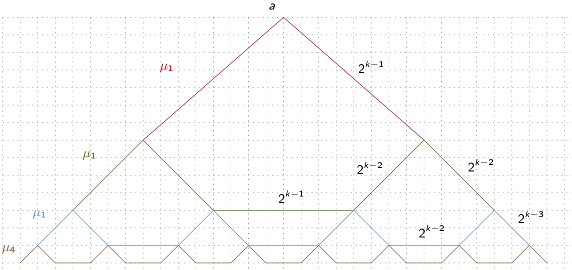


Goal : Single exponential bound.

A better bound ?

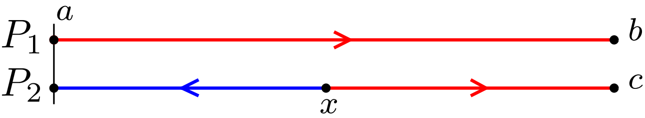
We shown an the upper bound : $O(k^k)$

Lower bound : $O(2^k)$



Goal : Single exponential bound.

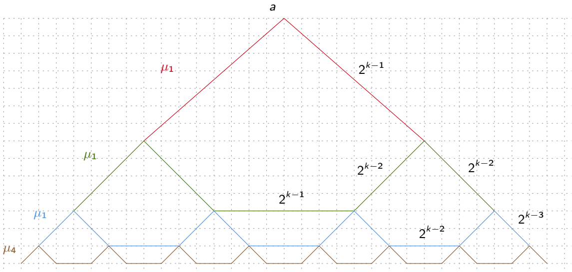
Observation : two colours-signs word may define the same vertex.



A better bound ?

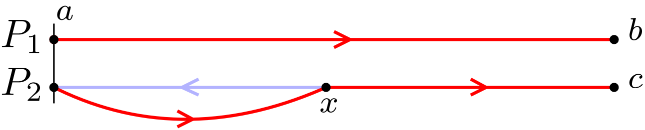
We shown an the upper bound : $O(k^k)$

Lower bound : $O(2^k)$



Goal : Single exponential bound.

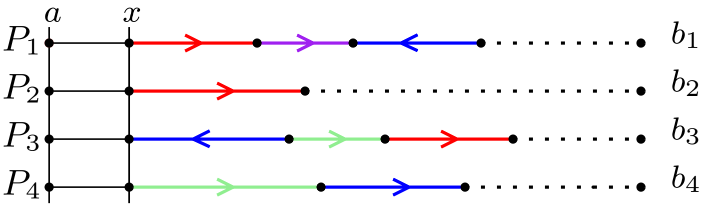
Observation : two colours-signs word may define the same vertex.



Branched colouring

Idea : Generalize the previous observation recursively.

Take a set of paths from *a* to the vertices at distance *D* from *a*.

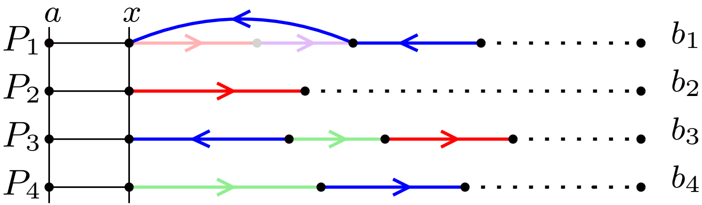


The colours red, blue and green does not appear in the dotted subpaths.

Branched colouring

Idea : Generalize the previous observation recursively.

Take a set of paths from *a* to the vertices at distance *D* from *a*.

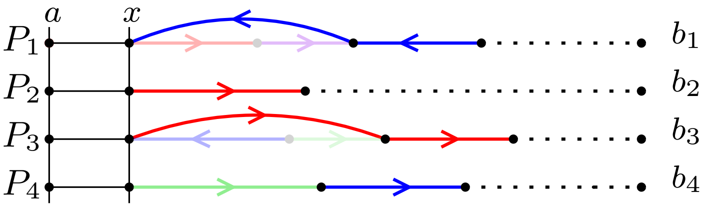


The colours red, blue and green does not appear in the dotted subpaths.

Branched colouring

Idea : Generalize the previous observation recursively.

Take a set of paths from a to the vertices at distance D from a.

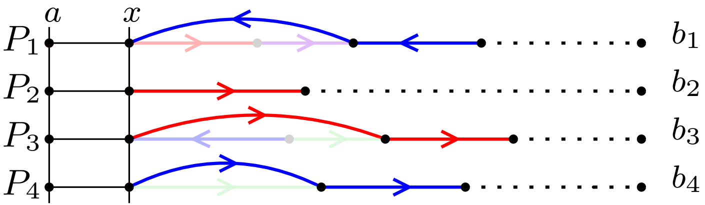


The colours red, blue and green does not appear in the dotted subpaths.

Branched colouring

Idea : Generalize the previous observation recursively.

Take a set of paths from a to the vertices at distance D from a.

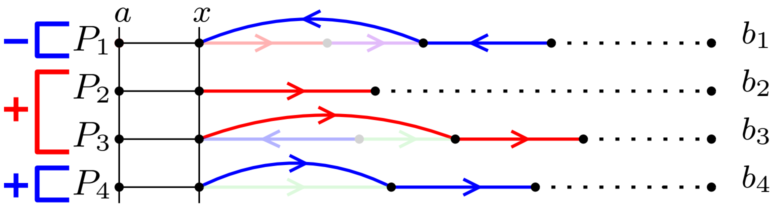


The colours red, blue and green does not appear in the dotted subpaths.

Branched colouring

Idea : Generalize the previous observation recursively.

Take a set of paths from *a* to the vertices at distance *D* from *a*.

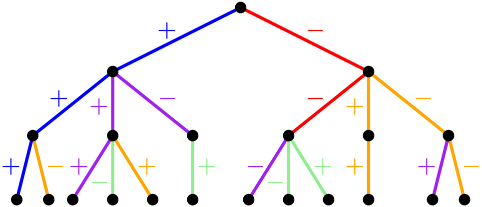


The colours red, blue and green does not appear in the dotted subpaths.

$\Rightarrow$ Apply recursively this process on each subset of paths independently.

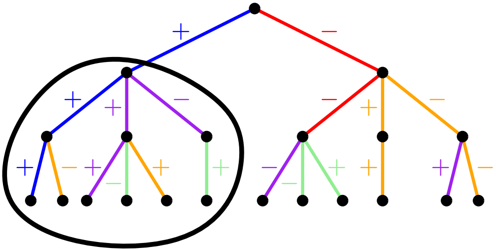
Branched colouring

Structure of the paths at the end of the recursive process :



Branched colouring

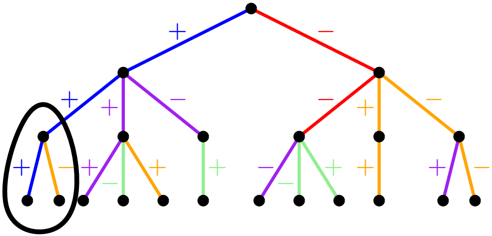
Structure of the paths at the end of the recursive process :



No : • red
• blue -

Branched colouring

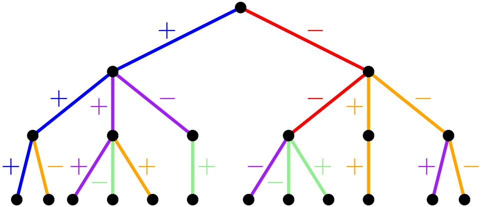
Structure of the paths at the end of the recursive process :



No :
- red
- blue -
- purple

Branched colouring

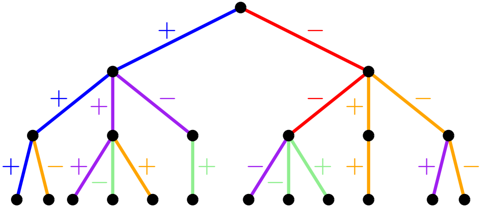
Structure of the paths at the end of the recursive process :



- k colours, 2 signs $\Rightarrow$ $O(4^k)$ leaves ($O(3^k)$ with a more precise analysis)

Branched colouring

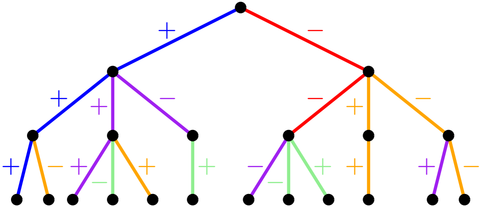
Structure of the paths at the end of the recursive process :



- k colours, 2 signs $\Rightarrow$ $O(4^k)$ leaves ($O(3^k)$ with a more precise analysis)
- bijection between leaves and vertices at distance D

Branched colouring

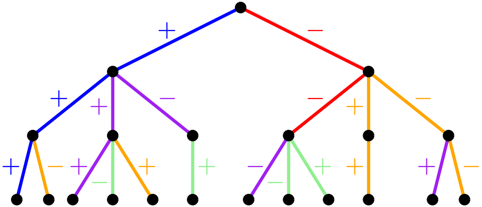
Structure of the paths at the end of the recursive process :



- k colours, 2 signs $\Rightarrow$ $O(4^k)$ leaves ($O(3^k)$ with a more precise analysis)
- bijection between leaves and vertices at distance D
- **Theorem 1 :** $O(3^k)$ vertices at a given distance of an arbitrary vertex

Branched colouring

Structure of the paths at the end of the recursive process :

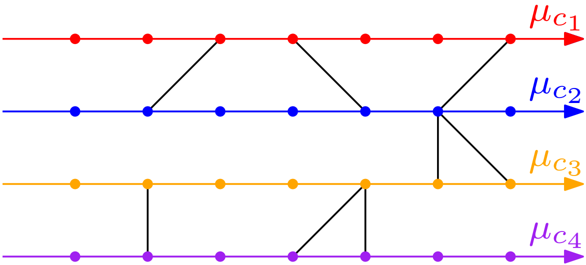


- k colours, 2 signs $\Rightarrow$ $O(4^k)$ leaves ($O(3^k)$ with a more precise analysis)
- bijection between leaves and vertices at distance D
- **Theorem 1 :** $O(3^k)$ vertices at a given distance of an arbitrary vertex

Graphs vertex-coverable by k shortest paths

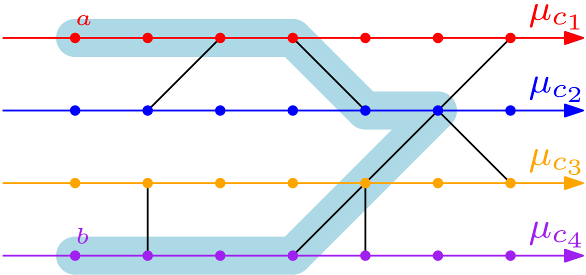
Colouring of a path

A colour and a direction given to each base path.



Colouring of a path

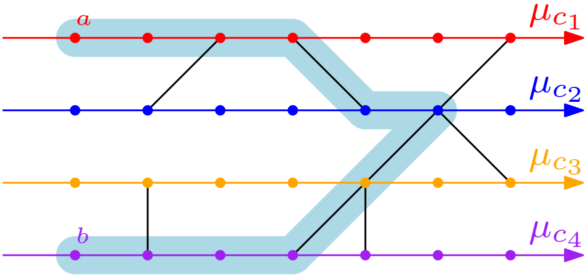
A colour and a direction given to each base path.



Colouring col of P : $\forall v \in V(P)$, $\mathrm{col}(v) \in \mathrm{colours}(v)$

Colouring of a path

A colour and a direction given to each base path.



Colouring col of P : $\forall v \in V(P)$, $\mathrm{col}(v) \in \mathrm{colours}(v)$

Colours-signs words are defined the same way as in the edge case.
Here : $\omega = ((c_1, +), (c_2, +), (c_3, +), (c_4, -),)$

Bound for the vertex case

- Good colouring Lemma works the same way.

Bound for the vertex case

- Good colouring Lemma works the same way.

Colours-signs word Lemma

The shortest paths starting in a vertex a, of length D and colours-signs word ω all ends in the same vertex b.

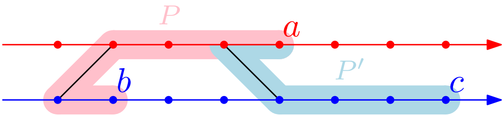
Bound for the vertex case

- Good colouring Lemma works the same way.

Colours-signs word Lemma

The shortest paths starting in a vertex a, of length D and colours-signs word ω all ends in the same vertex b.

$\Rightarrow$ FALSE in the vertex case



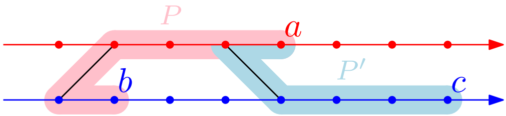
Bound for the vertex case

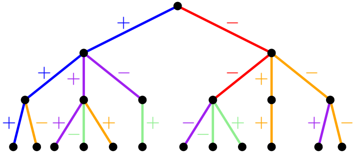
- Good colouring Lemma works the same way.

Colours-signs word Lemma

The shortest paths starting in a vertex a, of length D and colours-signs word ω all ends in the same vertex b.

$\Rightarrow$ FALSE in the vertex case





- **Branched colouring** can be adapted to the vertex case, but $\cancel{1}$ $O(k)$ paths may share the same colours-signs word.

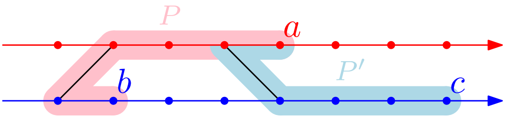
Bound for the vertex case

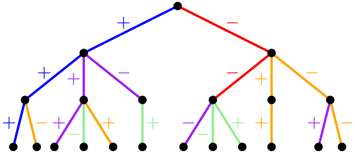
- Good colouring Lemma works the same way.

Colours-signs word Lemma

The shortest paths starting in a vertex a, of length D and colours-signs word ω all ends in the same vertex b.

$\Rightarrow$ FALSE in the vertex case

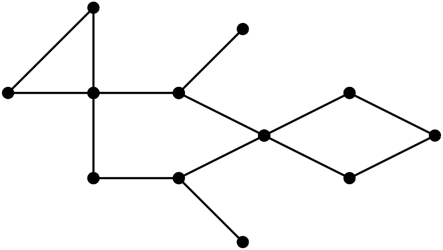




- **Branched colouring** can be adapted to the vertex case, but $\cancel{1}$ $O(k)$ paths may share the same colours-signs word.
- There is at most $g(k) = O(k \cdot 3^k)$ vertices at a given distance of a.

Algorithmic Consequences

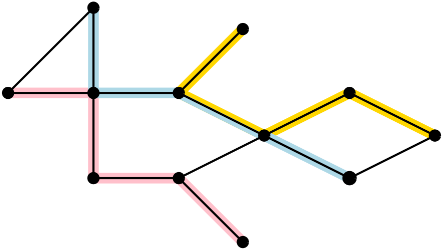
Problems



Isometric Path Cover (IPC)

Input : A graph G and an integer k.
Question : Does there exists a set of k shortest paths of G, such that each vertex of G belongs to at least one of the shortest paths ?

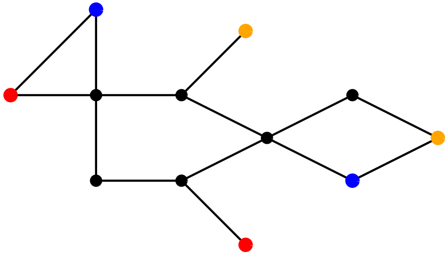
Problems



Isometric Path Cover (IPC)

Input : A graph G and an integer k.
Question : Does there exists a set of k shortest paths of G, such that each vertex of G belongs to at least one of the shortest paths ?

with terminals



Isometric Path Cover with Terminals (IPC-WT)

Input : A graph G, and k pairs of vertices $(s_1, t_1), \ldots, (s_k, t_k)$, the **terminals**.
Question : Does there exists a set of k shortest paths of G, the ith path being an s_i-t_i shortest path, such that each vertex of G belongs to at least one of the shortest paths ?

Context

What is known ?

- IPC is NP-Complete even on chordal graphs

  [Chakraborty, Dailly, Das, Foucaud, Gahlawat, and Ghosh, 2022]
- IPC is polynomial on block graphs [Pan and Chang, 2005]
- IPC is approximable by a factor $log(d)$ on graphs of diameter d

  [Thiessen and Gaertner, 2021]

Context

What is known ?

- IPC is NP-Complete even on chordal graphs
  [Chakraborty, Dailly, Das, Foucaud, Gahlawat, and Ghosh, 2022]
- IPC is polynomial on block graphs [Pan and Chang, 2005]
- IPC is approximable by a factor $log(d)$ on graphs of diameter d
  [Thiessen and Gaertner, 2021]

We have shown that IPC-WT is NP-Complete

Context

What is known ?

- IPC is NP-Complete even on chordal graphs [Chakraborty, Dailly, Das, Foucaud, Gahlawat, and Ghosh, 2022]
- IPC is polynomial on block graphs [Pan and Chang, 2005]
- IPC is approximable by a factor $log(d)$ on graphs of diameter d [Thiessen and Gaertner, 2021]

We have shown that IPC-WT is NP-Complete

Question

Are problems IPC and IPC-WT **FPT** ? Or at least **XP** ?

FPT : running time $f(k) \cdot n^{O(1)}$

XP : running time $n^{f(k)}$

Algorithmic consequences

Theorem 2

Problem IPC-WT is **FPT**, with running time $O(f(k) \cdot n)$.

Algorithmic consequences

Theorem 2

Problem IPC-WT is **FPT**, with running time $O(f(k) \cdot n)$.

- Yes-instances have bounded treewidth by Corollary 1,
- Courcelle's theorem to solve this problem on bounded treewidth graphs.

Algorithmic consequences

Theorem 2

Problem IPC-WT is **FPT**, with running time $O(f(k) \cdot n)$.

- Yes-instances have bounded treewidth by Corollary 1,
- Courcelle's theorem to solve this problem on bounded treewidth graphs.

Corollary 2

Problem IPC is **XP** for parameter k.

- Brute force : try all possible pairs of k terminals + FPT algorithm

Courcelle's Theorem

Theorem [Courcelle. 1990]

Every problem expressible in **monadic second order logic** (MSO_2) can be solved in $f(w) \cdot n$ time on graphs of treewidth at most w.

Courcelle's Theorem

Theorem [Courcelle. 1990]

Every problem expressible in **monadic second order logic** (MSO_2) can be solved in $f(w) \cdot n$ time on graphs of treewidth at most w.

Extended MSO_2 problem :

- MSO_2 formula $\varphi(X_1, \ldots, X_l)$ and an linear function $g(|X_1|, \ldots, |X_l|)$
- Find an assignation of $X_1, \ldots, X_l$ that satisfies $\varphi(X_1, \ldots, X_l)$ and maximize/minimize $g(|X_1|, \ldots, |X_l|)$

Theorem [Arnborg, Lagergren, Seese. 1991]

Every problem expressible as an $EMSO_2$ problem can be solved in $f(w) \cdot n$ time on graphs of treewidth at most w.

IPC-WT is FPT

Corollary 1

Graphs coverable with k shortest paths have treewidth bounded by $2g(k)$.

IPC-WT is FPT

Corollary 1

Graphs coverable with k shortest paths have treewidth bounded by $2g(k)$.

FPT algortihm :

1. Compute a tree decomposition by BFS. If width $> 2g(k)$ return `false`.

IPC-WT is FPT

Corollary 1

Graphs coverable with k shortest paths have treewidth bounded by $2g(k)$.

FPT algortihm :

1. Compute a tree decomposition by BFS. If width $> 2g(k)$ return `false`.
2. Find $E_1, \ldots, E_k$ minimizing $|E_1| + \cdots + |E_k|$ and satisfying the MSO_2 formula :

$$\varphi(E_1, \ldots, E_k) \quad = \quad \exists\ V_1, \ldots, V_k, \mathrm{Cover}(V_1, \ldots, V_k) \bigwedge_{1 \leq i \leq k} \mathrm{Path}(V_i, E_i, s_i, t_i)$$

IPC-WT is FPT

Corollary 1

Graphs coverable with k shortest paths have treewidth bounded by $2g(k)$.

FPT algortihm :

1. Compute a tree decomposition by BFS. If width $> 2g(k)$ return `false`.
2. Find $E_1, \dots, E_k$ minimizing $|E_1| + \dots + |E_k|$ and satisfying the MSO_2 formula :

$$\varphi(E_1, \dots, E_k) \quad = \quad \exists\ V_1, \dots, V_k, \mathrm{Cover}(V_1, \dots, V_k) \bigwedge_{1 \leq i \leq k} \mathrm{Path}(V_i, E_i, s_i, t_i)$$

3. **If** $\forall i, |E_i| = dist(s_i, t_i)$ **then** answer `true`, **else** answer `false`.

IPC-WT is FPT

Corollary 1

Graphs coverable with k shortest paths have treewidth bounded by $2g(k)$.

FPT algortihm :

1. Compute a tree decomposition by BFS. If width $> 2g(k)$ return `false`.
2. Find $E_1, \ldots, E_k$ minimizing $|E_1| + \cdots + |E_k|$ and satisfying the MSO_2 formula :

$$\varphi(E_1, \ldots, E_k) \quad = \quad \exists\ V_1, \ldots, V_k, \mathrm{Cover}(V_1, \ldots, V_k) \bigwedge_{1 \leq i \leq k} \mathrm{Path}(V_i, E_i, s_i, t_i)$$

3. **If** $\forall i, |E_i| = dist(s_i, t_i)$ **then** answer `true`, **else** answer `false`.

$\Rightarrow$ Can be easily generalized to edge covering & edge/vertex partitioning.

Conclusion

Conclusion

- In graphs vertex/edge-coverable by k shortest paths, the number of vertices at same distance of a source is upper bounded by $g(k) = O^*(3^k)$.
- Implies a $O^*(3^k)$ upper bound on treewidth.
- Problem IPC-WT is FPT.
- Problem IPC is XP.

Questions

- Polynomial bound on the treewidth/pathwidth ?
- Is IPC (without terminals) FPT ? W-hard ?