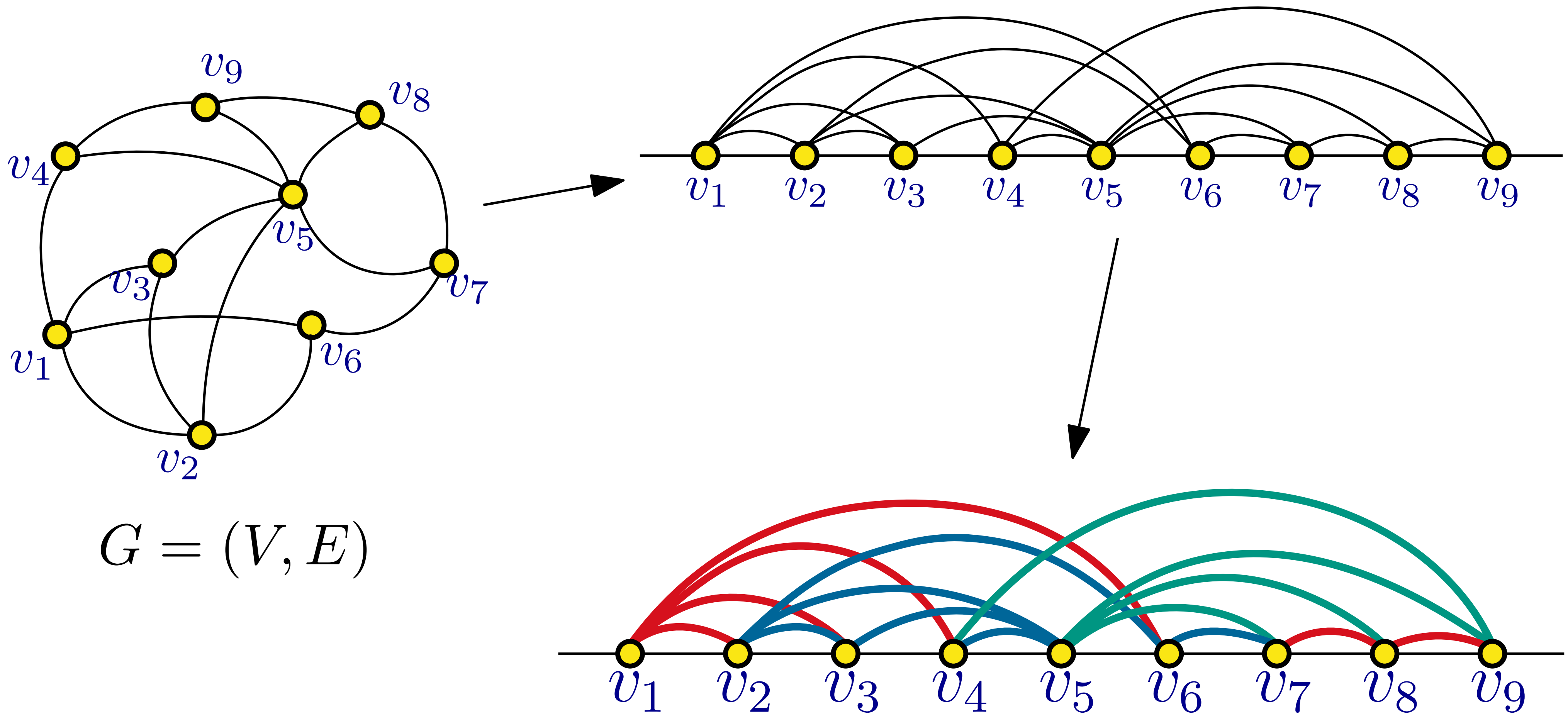
Parameterized Algorithms for Book-Embedding Problems

Sujoy Bhore, Robert Ganian, Fabrizio Montecchiani, Martin Nöllenburg

Graph Drawing · September 19, 2019

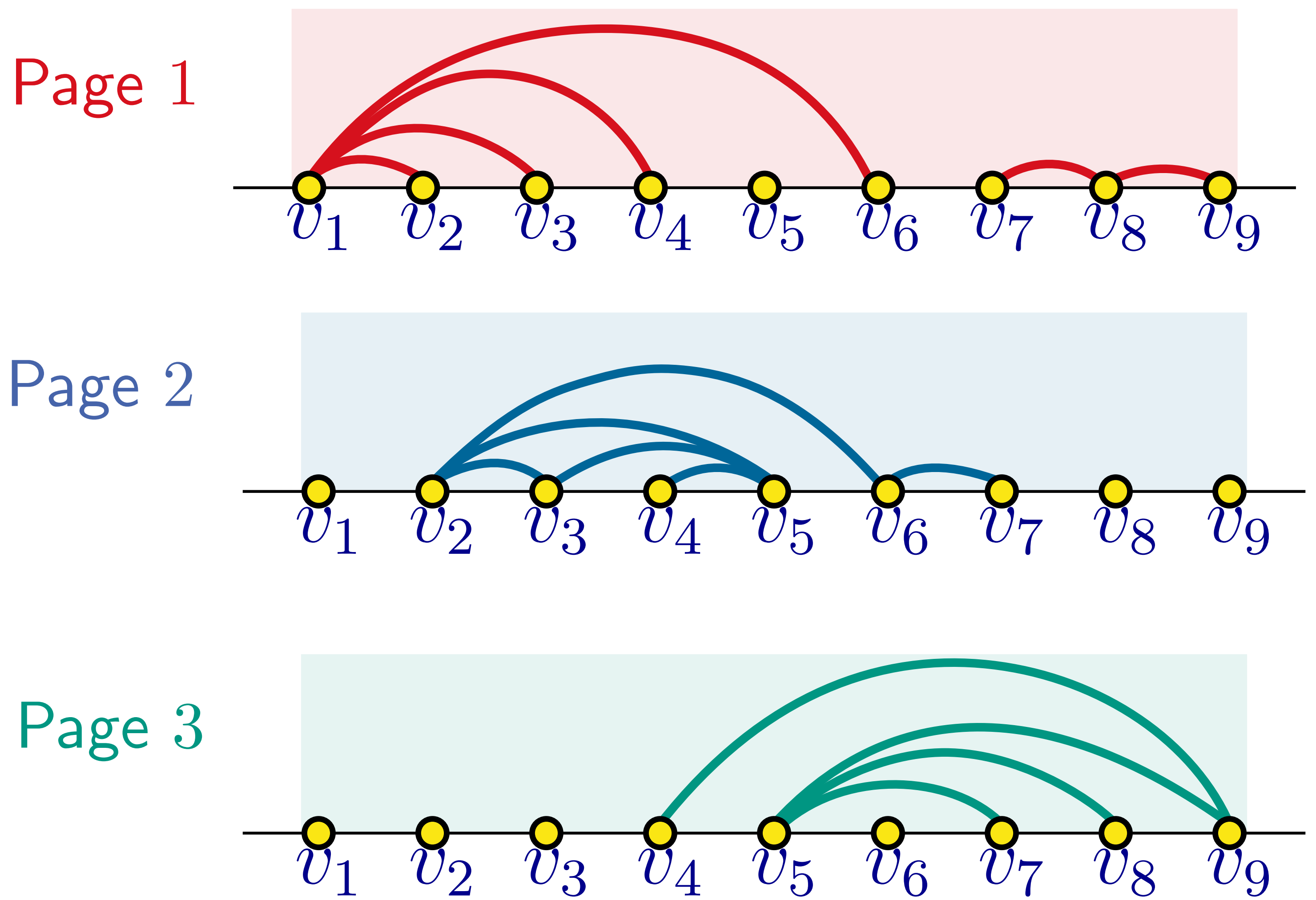
ALGORITHMS AND COMPLEXITY GROUP

The Problem



$G = (V, E)$

The Problem



G has 3-Page Book-Embedding

Book Thickness

- Book Thickness ($bt(G)$): the minimum k such that G admits a k-page book-embedding.
- Alternatively, known as Stack Number.

Applications:

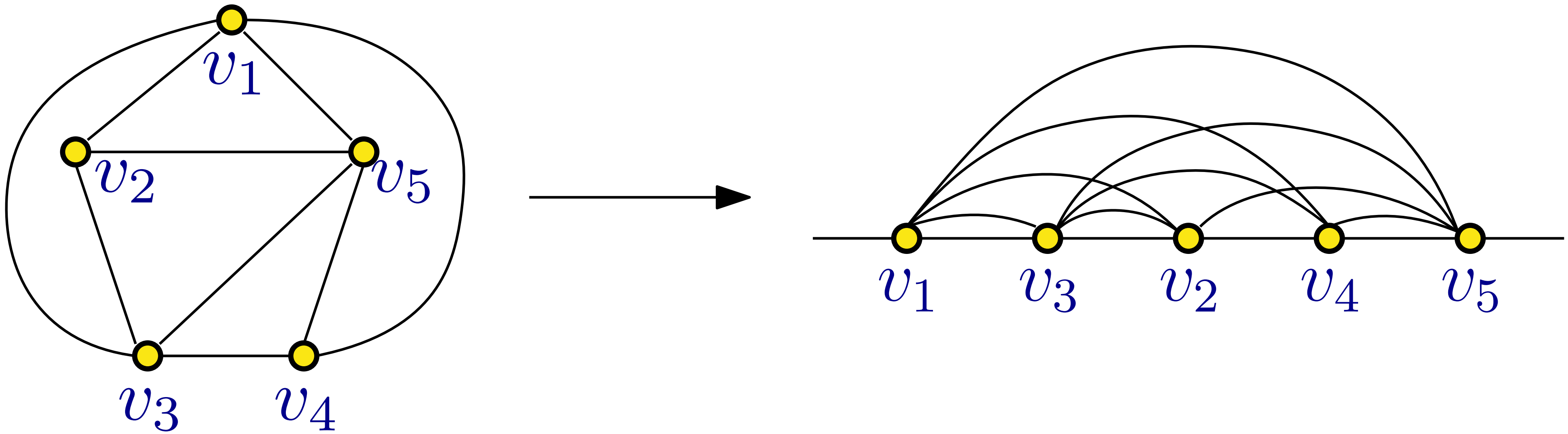
- Bioinformatics
- VLSI
- Parallel Computing

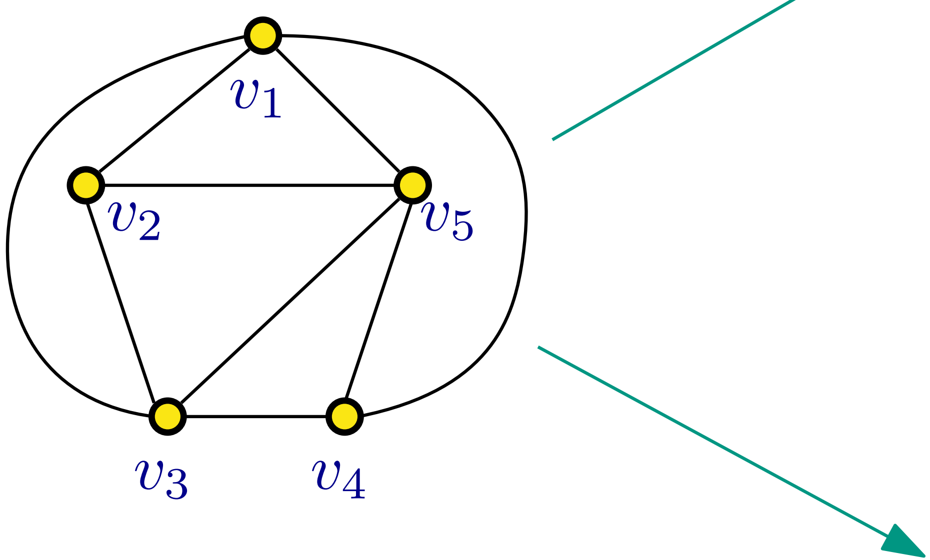
What we know ...

- Every planar graph has book thickness at most four.
  [Yannakakis – J. Comput. Syst. Sci., 89]

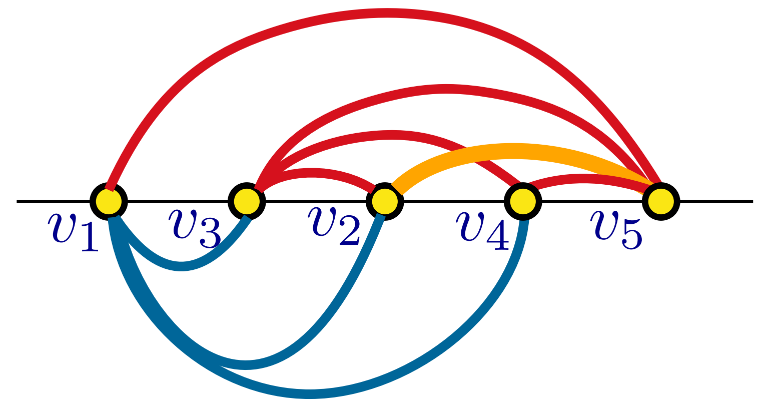
- Given a graph G and a positive integer k, determining whether $bt(G) \leq k$ is NP-complete (even for $k \geq 2$).
  [Bernhart et al. – J. Comb. Theory, Ser. B, 79]

What happens if the linear order $\prec$ of the vertices is fixed?

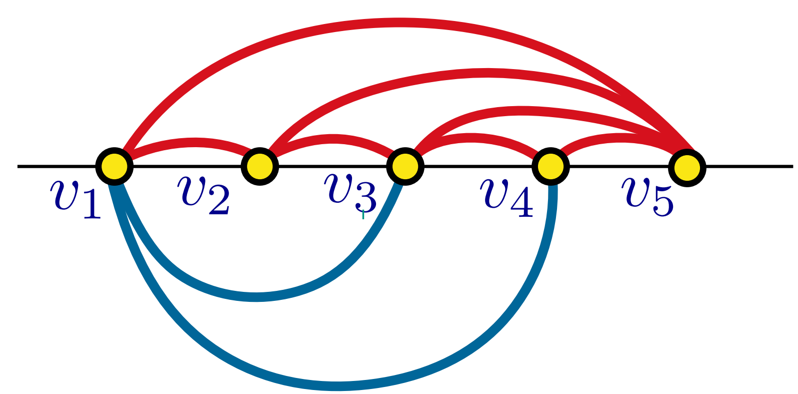




Fixed-order book-thickness (fo-bt$(G) = 3$):



Book-thickness (bt$(G) = 2$):



What we know ...

- Deciding whether fo-bt$(G, \prec) \leq 2$ is Polynomial, since equivalent to testing the bipartiteness of a suitable conflict graph.

- Deciding if fo-bt$(G, \prec) \leq 4$ is NP-Complete, since equivalent to finding a 4-coloring of a circle graph which is NP-complete [W. Unger – STACS 1992].

- Problem + Parameter

A problem is fixed-parameter tractable (FPT) with respect to parameter k if there exists a solution running in $f(k) \cdot n^{O(1)}$ time, where f is a computable function of k which is independent of n.

Results:

FPT-algorithms :

- Fixed-Order Book Thickness **parameterized by the vertex cover number of the graph**

- Fixed-Order Book Thickness parameterized by the pathwidth of the graph w.r.t the vertex order

- Book Thickness parameterized by the vertex cover number of the graph

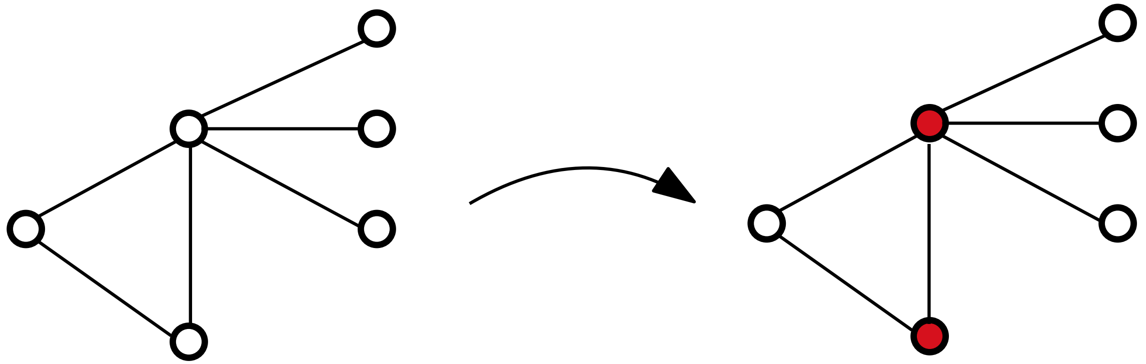
Algorithms for Fixed-Order Book-Thickness ...

- **Input:** Graph $G = (V, E)$, a linear order $\prec$ of V, and a positive integer k.
- **Task:** Decide if there is a page assignment $\sigma\colon E \to [k]$ such that $\langle \prec, \sigma \rangle$ is a k-page book embedding of G, that is whether $\mathrm{fo\text{-}bt}(G, \prec) \leq k$.

If the answer is 'YES' we shall return a corresponding k-page book embedding as a witness.

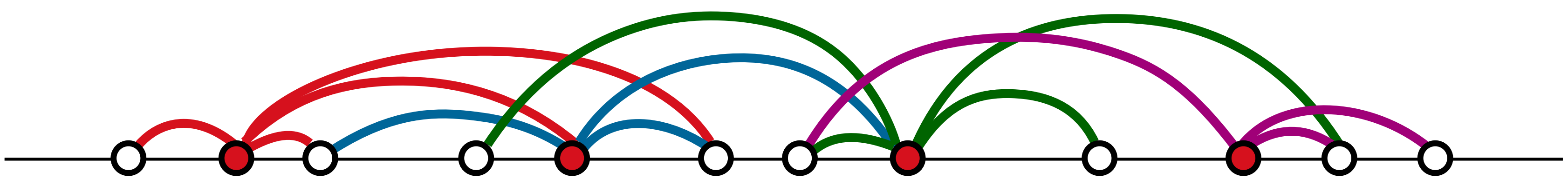
Parameterization by the Vertex Cover number (τ)...

Vertex Cover



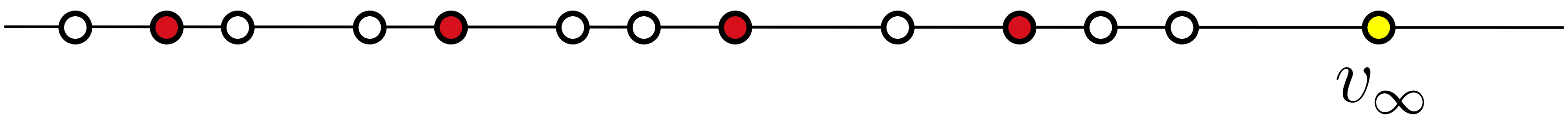
- Vertex Cover C of a graph G can be computed in time $\mathcal{O}(2^\tau + \tau \cdot n)$ [TCS, 10 - Chen et al.]

Observation 1 *Every graph G with a vertex cover C of size τ admits a τ-page book embedding with any vertex order $\prec$.*

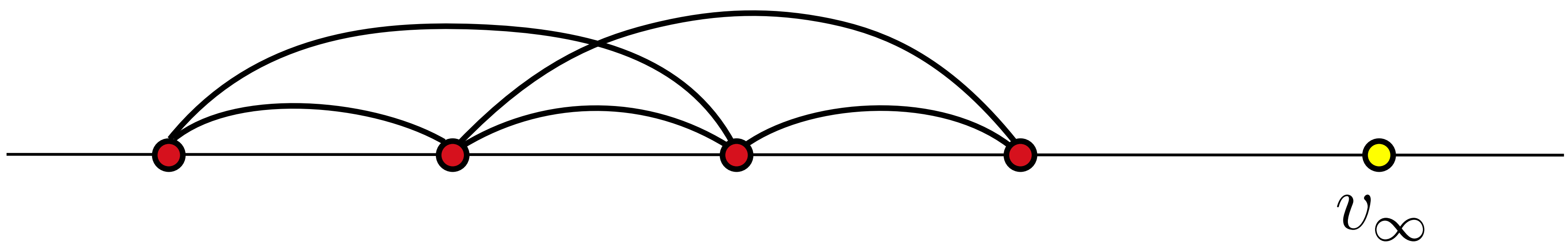


4-page book embedding ...

Set-up for the Algorithm

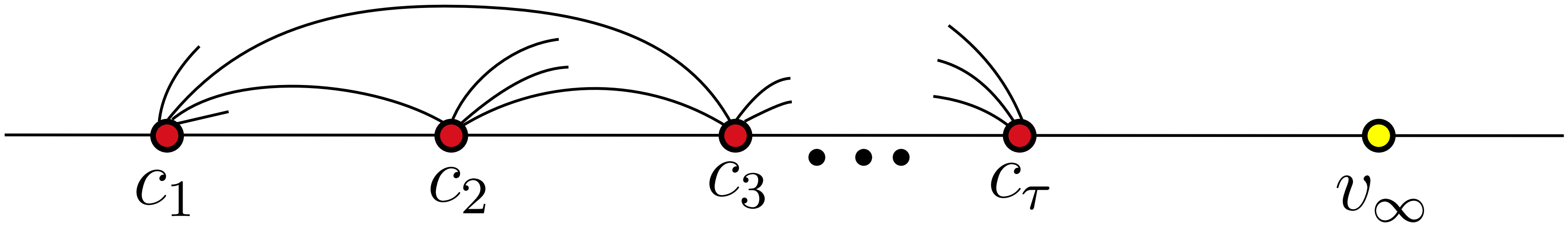


- If $\tau \leq k$ - Yes!
- Else ...

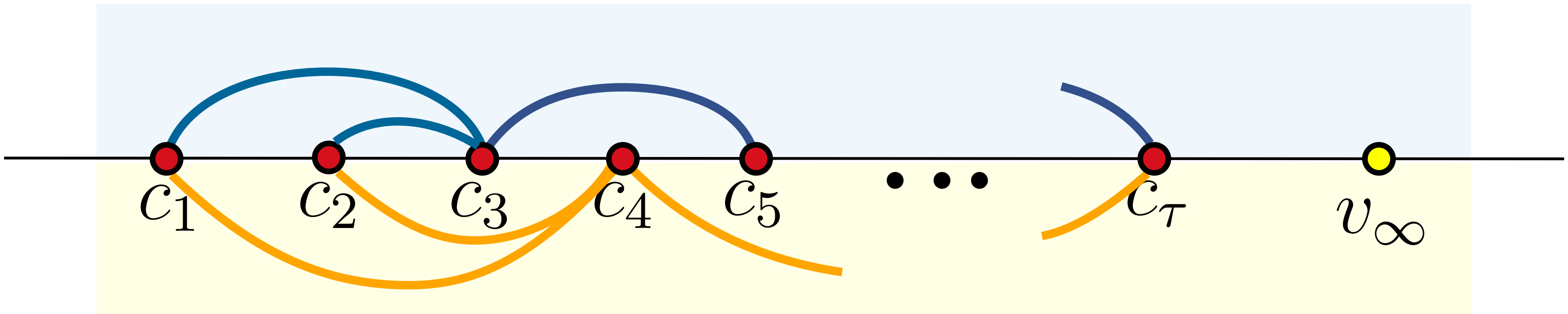


- Compute set of all valid page assignments S of $G[C]$
- $|S| < \tau^{\tau^2}$

Towards the dynamic program ...



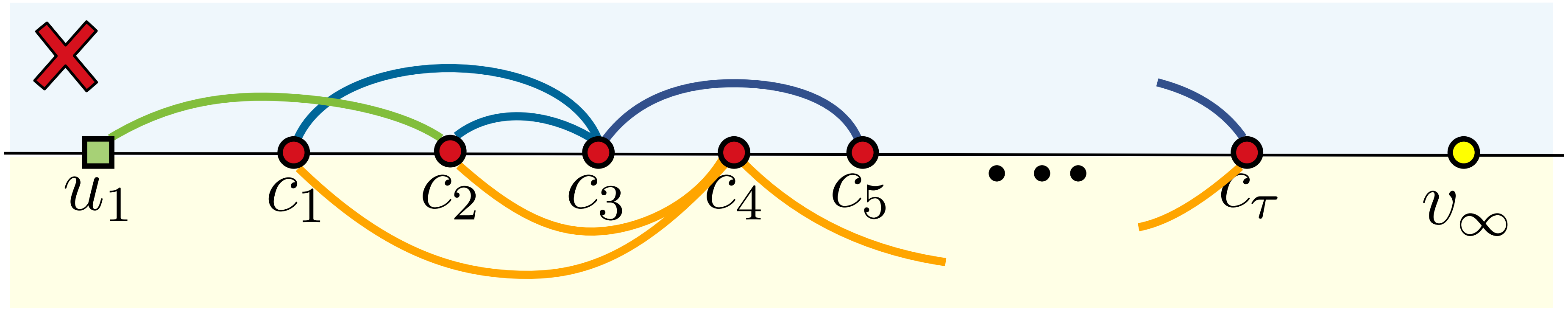
Consider an assignment $s \in S$

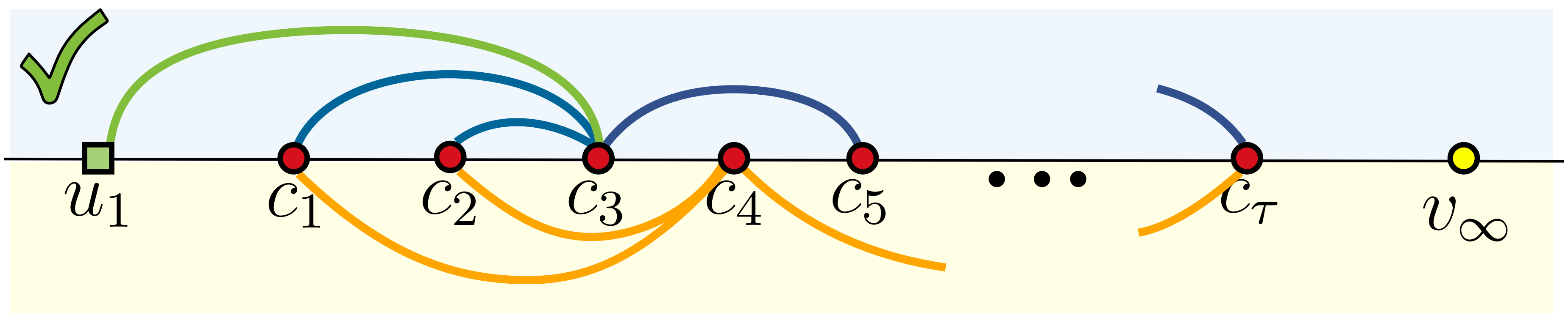


2-page assignment

Notion of Visibility ...

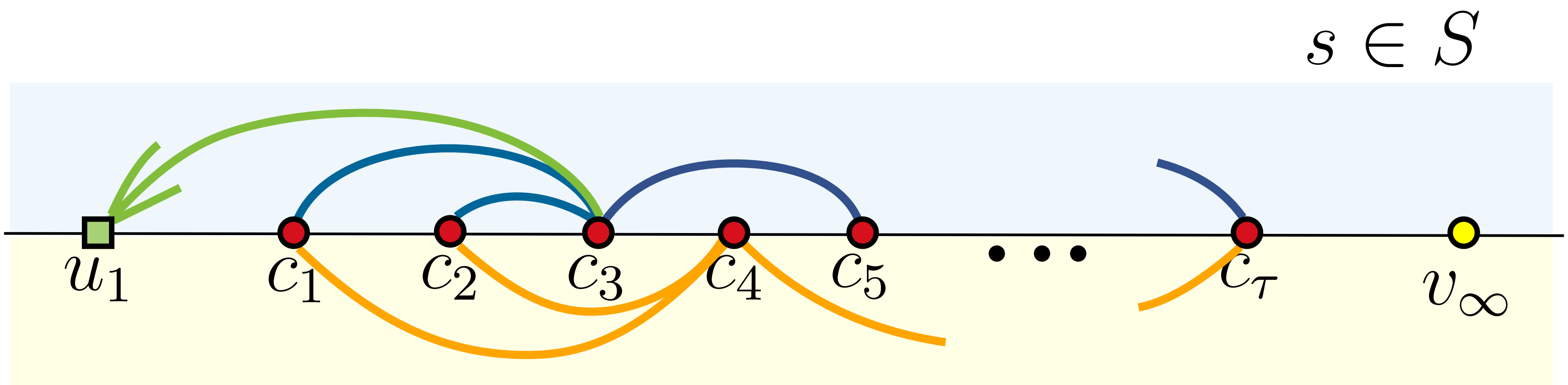
$s \in S$



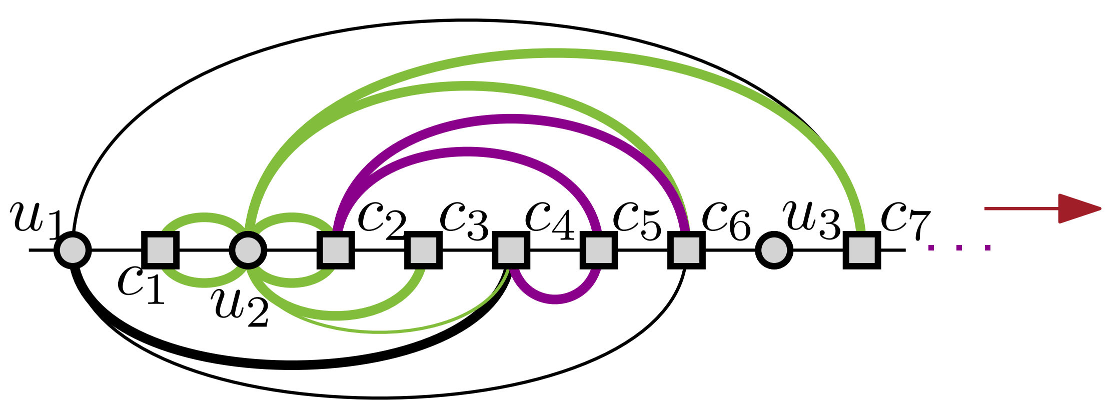


valid page assignment...

Building visibility matrix ...



for an index $a \in [n - \tau]$, a $k \times \tau$ **visibility matrix** $M_i(a, \alpha, s)$...



$$M_2(2, \alpha, s) = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

High-level Idea ...

- Dynamically process the vertices in U(non vertex cover vertices) from left to right ...

- For each vertex,

  a bounded size **snapshot** of its visibility vertices ...

- Store one (arbitarily) chosen valid partial edge assignment ...

All valid partial page assignments lead to the same visibility matrices are interchangeable ...

Record set ...

For a vertex $u_i \in U$,

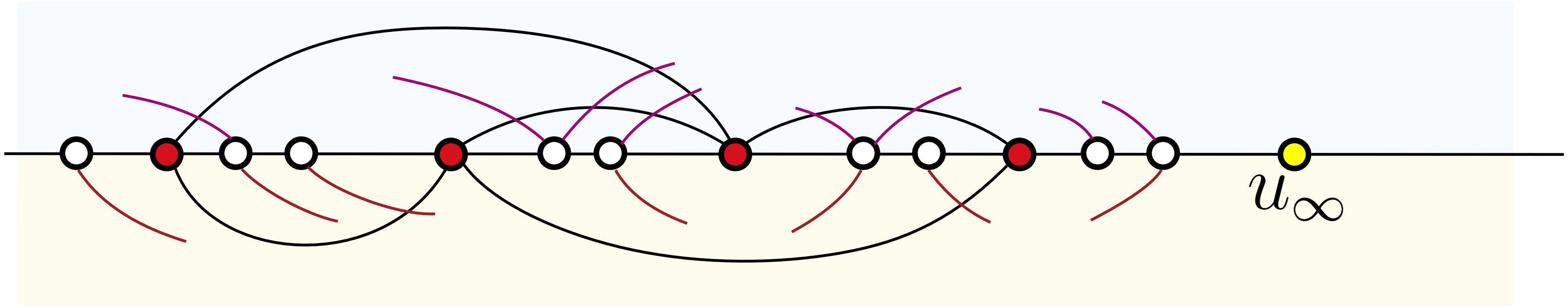
$$\mathcal{R}_i(s) = \{\big(M_i(i,\alpha,s), M_i(x_1,\alpha,s), \ldots, M_i(x_z,\alpha,s)\big) \mid \exists \text{ valid partial page assignment } \alpha\colon E_i \to [k]\}$$

Some Observations ...

- $|\mathcal{R}_i(s)| \leq 2^{\tau^3+\tau^2}$
- If $\mathcal{R}_{n-\tau}(s) \neq \emptyset$ for some s ($u_{n-\tau}$ is a dummy vertex)

  then there is a valid partial page assignment $\alpha\colon E_{n-\tau} \to [k]$ s.t. $s \cup \alpha$ is a non-crossing page assignment of *all* edges in G.

Observation 2 *If for all $s \in S$ it holds that $\mathcal{R}_{n-\tau}(s) = \emptyset$, then $(G, \prec, k)$ is a NO-instance of* FIXED-ORDER BOOK THICKNESS.



- It suffices to compute $\mathcal{R}_{n-\tau}(s)$ for each $s \in S$.

Dynamic Step ...

- Compute $\mathcal{R}_1(s)$ ...
- Assume we have computed $\mathcal{R}_{i-1}(s)$ ...
- Branch over each page assignment β of the edges($\leq \tau$) incident to u_{i-1}, and each tuple $\rho \in \mathcal{R}_{i-1}(s)$ ...
- If it is **NOT** a valid partial page assignment - discard!
- Else, compute the visibility matrices add the corresponding tuple into $\mathcal{R}_i(s)$.

Lemma 1 *The procedure correctly computes $\mathcal{R}_i(s)$ from $\mathcal{R}_{i-1}(s)$.*

Runtime is upper-bounded by -

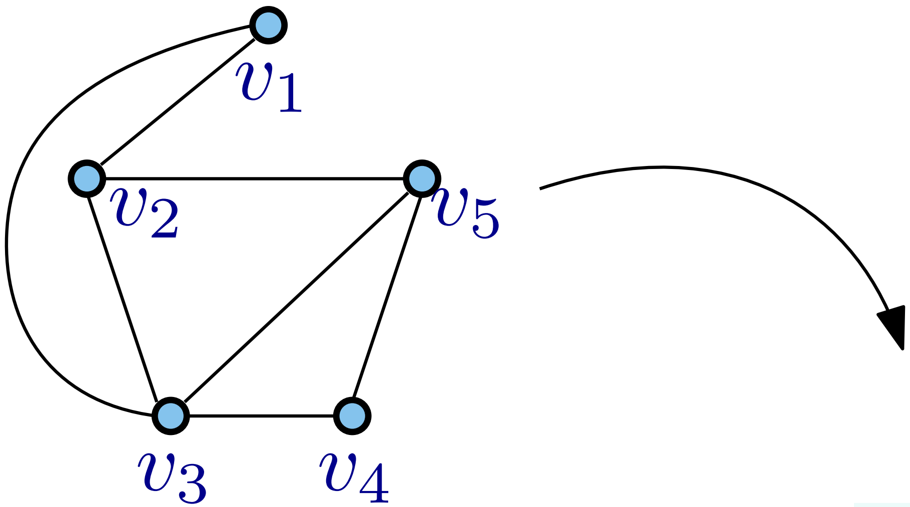
$$(\tau^{\tau^2}) \cdot n \cdot (2^{\tau^3+\tau^2} \tau^{\tau})$$

Theorem 1 *There is an algorithm which takes as input an n-vertex graph G with a vertex order $\prec$, runs in time $2^{\mathcal{O}(\tau^3)} \cdot n$ where τ is the vertex cover number of G, and computes a page assignment σ such that $(\prec, \sigma)$ is a $(\text{fo-bt}(G, \prec))$-page book embedding of G.*

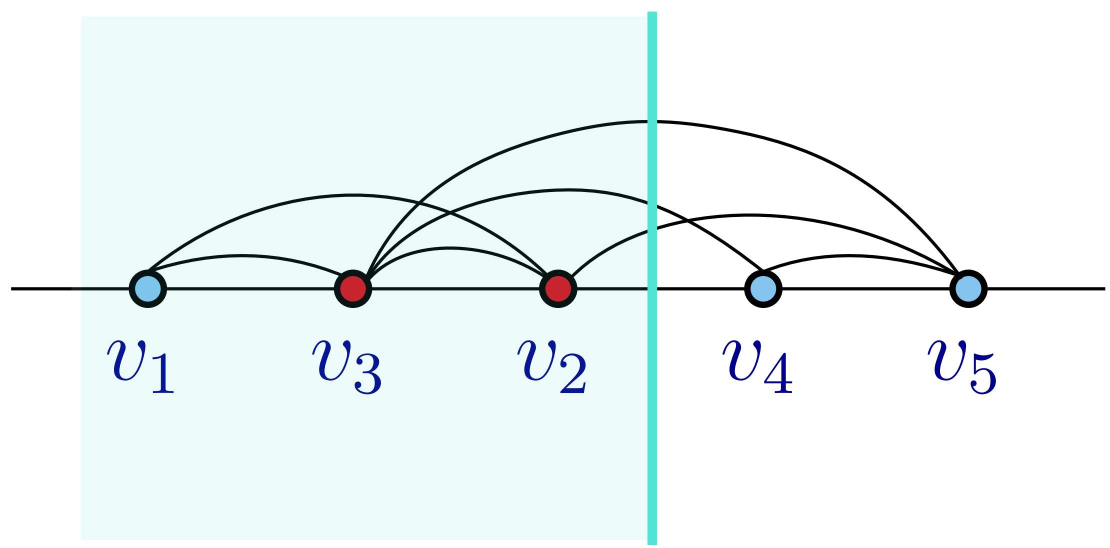
FPT-algorithms :

- Fixed-Order Book Thickness parameterized by the vertex cover number of the graph
- Fixed-Order Book Thickness **parameterized by the pathwidth of the graph w.r.t a vertex order**
- Book Thickness parameterized by the vertex cover number of the graph.

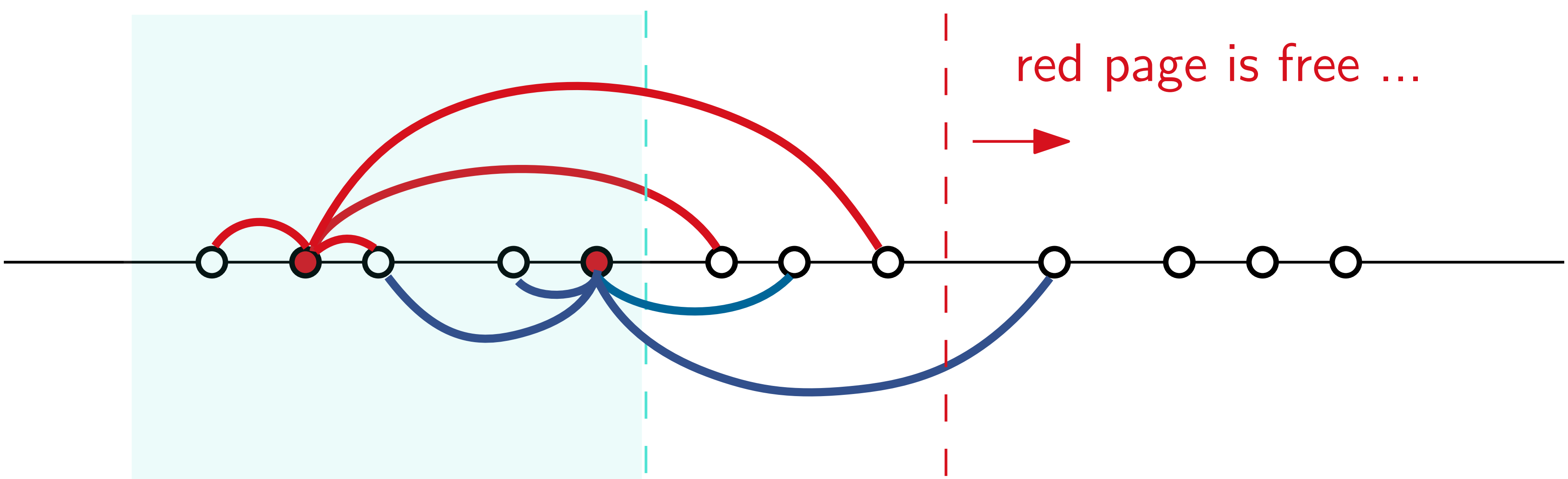
Parameterization by the pathwidth ...



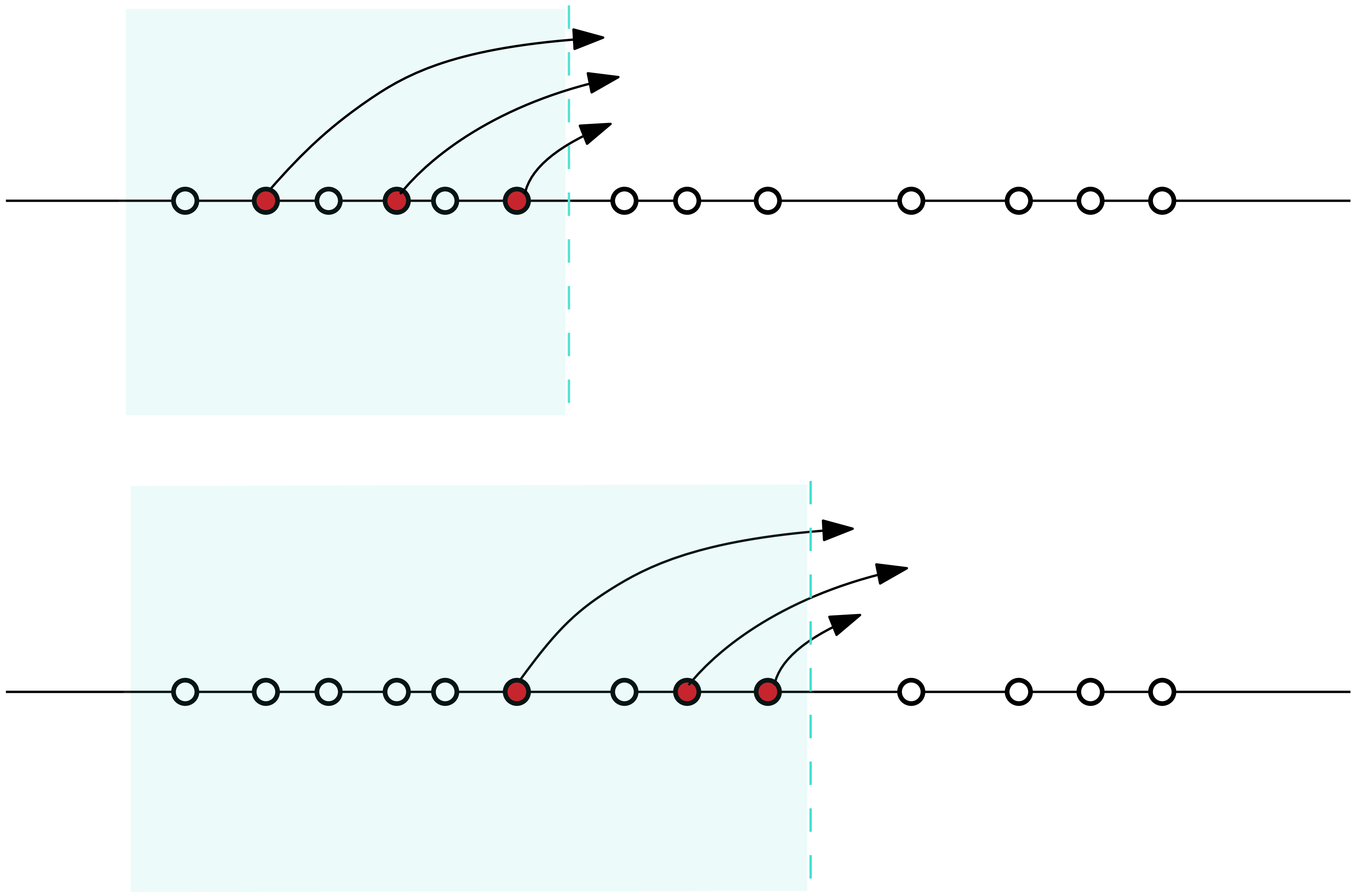
$$i/p : G = (V, E), \prec$$



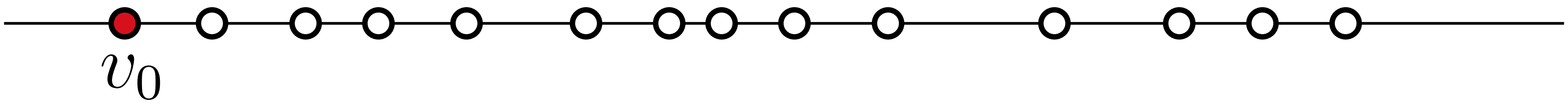
Lemma 2 *Every graph $G = (V, E)$ with a linear order $\prec$ of V such that $(G, \prec)$ has pathwidth k admits a k-page book embedding $\langle \prec, \sigma \rangle$, which can be computed in $\mathcal{O}(n + k \cdot n)$ time.*



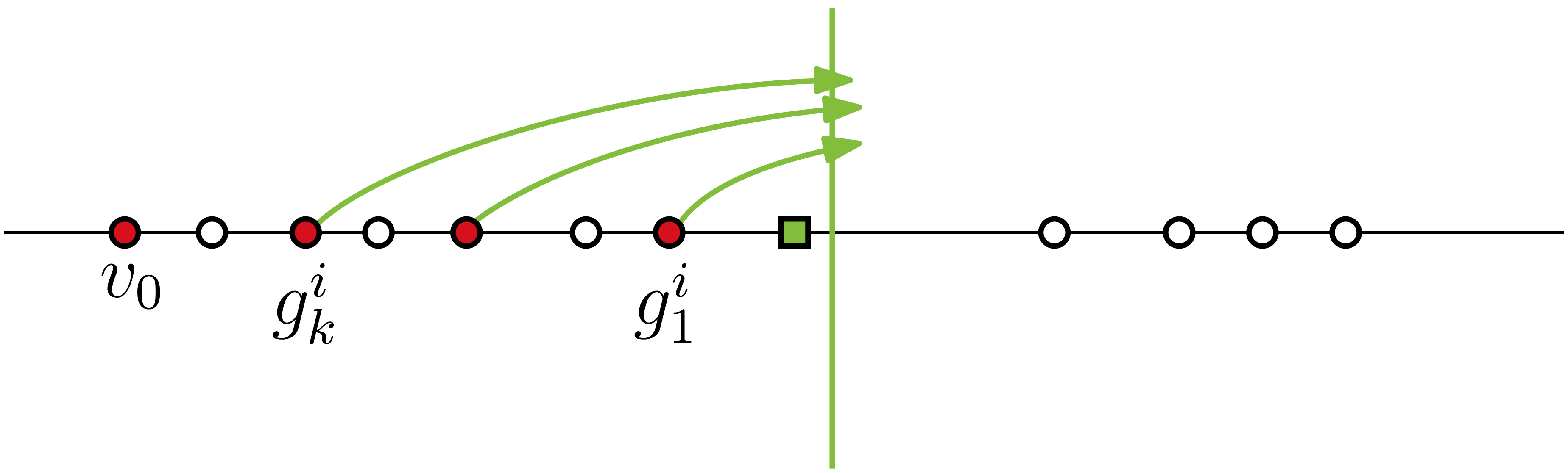
Dynamic guard sets ...



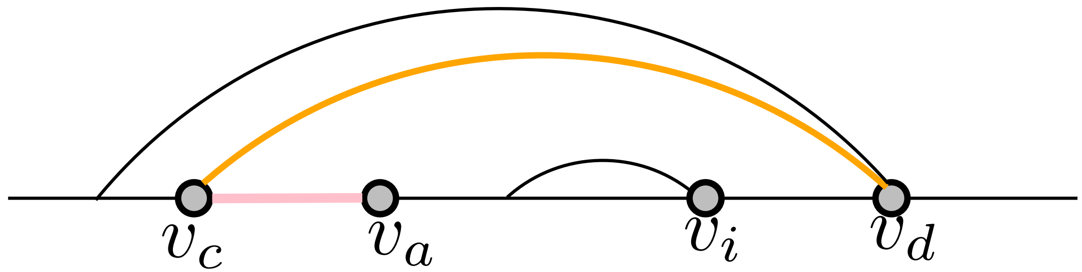
Set-up for the algorithm ...



concept of guards, **BUT**, in reverse order ...



Concept of (α, i, p) important edge ...



Observation 3 *If v_a has no (α, i, p)-important edge, then every vertex v_x with $x < a$ is α-visible to v_a. If the (α, i, p)-important guard of v_a is v_c, then v_x $(x < a)$ is α-visible to v_a if and only if $x \geq c$.*

Lemma 3 *The procedure correctly computes* Q_{i-1} *from* Q_i.

Runtime is upper bounded by
$\mathcal{O}(n \cdot (\kappa+2)^{\kappa^2} \cdot \kappa^\kappa)$

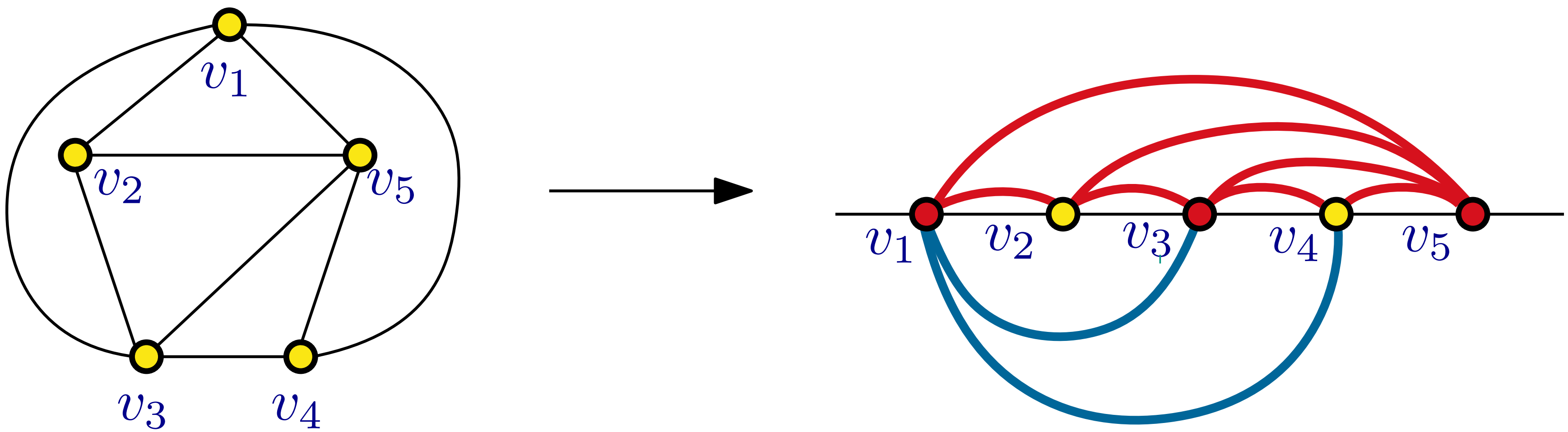
Theorem 2 *There is an algorithm which takes as input an* n*-vertex graph* $G = (V, E)$ *with a vertex ordering* $\prec$ *and computes a page assignment* σ *of* E *such that* $(\prec, \sigma)$ *is a* $(\text{fo-bt}(G, \prec))$*-page book embedding of* G. *The algorithm runs in* $n \cdot \kappa^{\mathcal{O}(\kappa^2)}$ *time where* κ *is the pathwidth of* $(G, \prec)$.

FPT-algorithms :

- FIXED-ORDER BOOK THICKNESS parameterized by the vertex cover number of the graph
- FIXED-ORDER BOOK THICKNESS parameterized by the pathwidth of the graph w.r.t a vertex order
- BOOK THICKNESS **parameterized by the vertex cover number of the graph**

Algorithm for Book-Thickness

Book-thickness ($bt(G)$):



Theorem 3 *Given a graph graph $G = (V, E)$ with vertex cover number τ and a positive integer k, there is an algorithm that runs in time $\mathcal{O}(\tau^{\tau^{\mathcal{O}(\tau)}} + 2^{\tau} \cdot n)$ ($\tau = \tau(G)$ is the vertex cover number of G), and decides whether $\mathrm{bt}(G) \leq k$.*

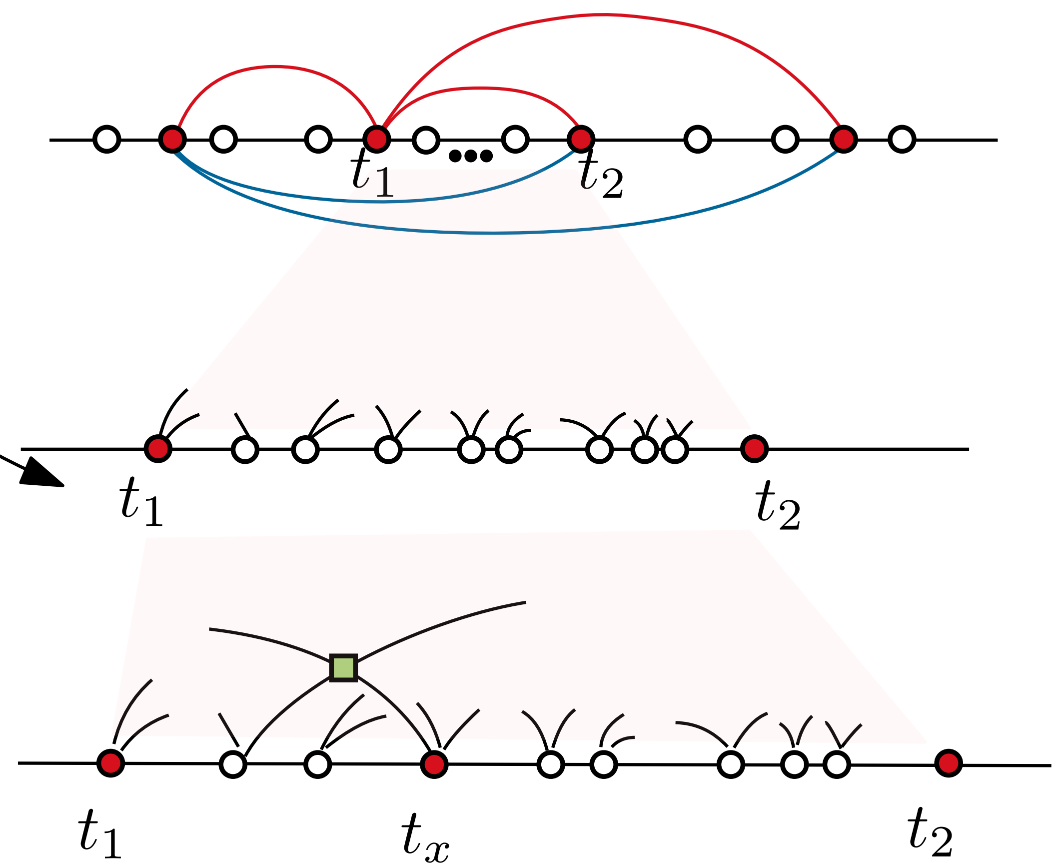
Conclusion ...

FPT algorithms

- for fixed-order book-thickness (parameter: vertex cover, pathwidth)
- for book-thickness (parameter: vertex cover)

Issues with Treewidth ...

what if we allow constant number of crossings ??? ...



Thank You