Smaller Universal Graphs for Caterpillars and Graphs of Bounded Path-Width

Cyril Gavoille and Arnaud Labourel

ICGT ’22 – Montpellier – July 5th, 2022
Representation of Graphs

adjacency lists

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>5</td>
<td>8</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

adjacency matrix

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

1 node = 1 pointer in the data-structure
Implicit Representation

**Idea:** Associate with each node some information and only use the information of any two nodes to determine if they are adjacent.

An **adjacency labeling scheme** for a graph family $\mathcal{F}$ pair of functions $(\ell, f)$ such that $\forall G \in \mathcal{F}, \forall u, v \in V(G)$

- Encoder: $\ell(u, G)$ is a label (binary string)
- Decoder: $f(\ell(u, G), \ell(v, G))$ is true $\iff uv \in E(G)$

**Goal:** minimize label size (in bits)

... and time complexities of $\ell, f$
Ex: Interval Graphs

\( \ell(u, G) := (l_u, r_u) \) with \( l_u, r_u \in \{1, \ldots, 2n\} \)

\[ f((l_u, r_u), (l_v, r_v)) := \text{true} \iff [l_u, r_u] \cap [l_v, r_v] \neq \emptyset \]

\[ \Rightarrow \text{labels of } 2 \lceil \log (2n) \rceil \text{ bits} \]
A graph $U$ is an induced-universal graph for $\mathcal{F}$ if every graph of $\mathcal{F}$ is isomorphic to some induced subgraph of $U$.

$\mathcal{F} = \{\text{trees with 6 nodes}\}$

universal graph for $\mathcal{F}$
\( F = \{ \text{trees with 6 nodes} \} \)

Universal graph for \( F \)

\( k \)-bit labels ⇔ universal graph of \( 2^k \) nodes
<table>
<thead>
<tr>
<th>family</th>
<th>label size</th>
<th>U. graph size</th>
<th>refs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>treewidth $k$</td>
<td>$\log n + O(k \lg \lg n)$</td>
<td>$n \cdot (\log n)^{O(k)}$</td>
<td>[GL07]</td>
</tr>
<tr>
<td>planar, $H \boxtimes P$</td>
<td>$\log n + \tilde{O}(\sqrt{\lg n})$</td>
<td>$n^{1+o(1)}$</td>
<td>[DEGJx21]</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>family</th>
<th>label size</th>
<th>U. graph size</th>
<th>refs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>treewidth $k$</td>
<td>$\log n + O(k \log \log n)$</td>
<td>$n \cdot (\log n)^{O(k)}$</td>
<td>[GL07]</td>
</tr>
<tr>
<td>planar, $H \boxtimes P$</td>
<td>$\log n + \tilde{O}(\sqrt{\log n})$</td>
<td>$n^{1+o(1)}$</td>
<td>[DEGJx21]</td>
</tr>
<tr>
<td>outerplanar $\Delta$</td>
<td>$\log n + O(1)$</td>
<td>$O(n)$</td>
<td>[C90][AR14]</td>
</tr>
<tr>
<td>trees</td>
<td>$\log n + O(1)$</td>
<td>$O(n)$</td>
<td>[ADK17]</td>
</tr>
<tr>
<td>$\Delta \leq 2$</td>
<td>$\log n + O(1)$</td>
<td>$2n$</td>
<td>[AAHKS20]</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>family</th>
<th>label size</th>
<th>U. graph size</th>
<th>refs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>treewidth $k$</td>
<td>$\log n + O(k \lg \lg n)$</td>
<td>$n \cdot (\log n)^{O(k)}$</td>
<td>[GL07]</td>
</tr>
<tr>
<td>planar, $H \boxtimes P$</td>
<td>$\log n + \tilde{O}(\sqrt{\lg n})$</td>
<td>$n^{1+o(1)}$</td>
<td>[DEGJx21]</td>
</tr>
<tr>
<td>outerplanar $\Delta$</td>
<td>$\log n + O(1)$</td>
<td>$O(n)$</td>
<td>[C90][AR14]</td>
</tr>
<tr>
<td>trees</td>
<td>$\log n + O(1)$</td>
<td>$O(n)$</td>
<td>[ADK17]</td>
</tr>
<tr>
<td>$\Delta \leq 2$</td>
<td>$\log n + O(1)$</td>
<td>$2n$</td>
<td>[AAHKS20]</td>
</tr>
<tr>
<td>caterpillars</td>
<td>$\log n + O(1)$</td>
<td>$256n$</td>
<td>[BGL06]</td>
</tr>
<tr>
<td>caterpillars</td>
<td>$\log n + O(1)$</td>
<td>$384n$</td>
<td>[ADK17]</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>family</th>
<th>label size</th>
<th>U. graph size</th>
<th>refs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>treewidth $k$</td>
<td>$\log n + O(k \log \log n)$</td>
<td>$n \cdot (\log n)^O(k)$</td>
<td>[GL07]</td>
</tr>
<tr>
<td>planar, $H \boxtimes P$</td>
<td>$\log n + \tilde{O}(\sqrt{\log n})$</td>
<td>$n^{1+o(1)}$</td>
<td>[DEGJx21]</td>
</tr>
<tr>
<td>outerplanar $\Delta$</td>
<td>$\log n + O(1)$</td>
<td>$O(n)$</td>
<td>[C90][AR14]</td>
</tr>
<tr>
<td>trees</td>
<td>$\log n + O(1)$</td>
<td>$O(n)$</td>
<td>[ADK17]</td>
</tr>
<tr>
<td>$\Delta \leq 2$</td>
<td>$\log n + O(1)$</td>
<td>$2n$</td>
<td>[AAHKS20]</td>
</tr>
<tr>
<td>caterpillars</td>
<td>$\log n + O(1)$</td>
<td>$256n$</td>
<td>[BGL06]</td>
</tr>
<tr>
<td>caterpillars</td>
<td>$\log n + O(1)$</td>
<td>$384n$</td>
<td>[ADK17]</td>
</tr>
<tr>
<td>caterpillars</td>
<td>$\log n + O(1)$</td>
<td>$8n$</td>
<td>new</td>
</tr>
<tr>
<td>bounded pw</td>
<td>$\log n + O(1)$</td>
<td>$O(n)$</td>
<td>new</td>
</tr>
</tbody>
</table>
A caterpillar forest is a forest in which the nodes of degree at least two induce paths.

Theorem 1
Caterpillar forests with $n$ nodes have a universal graph $U_n$ with $8n$ nodes.
Every universal graph for the star and the path with $n$ nodes requires $\lfloor 3n/2 \rfloor$ nodes.
Universal Graph $U_n$ for Caterpillar

Nodes:

- independent set $S$ of $6n$ nodes
- layers $L_i$ of $n/2^i$ nodes for $i$ from 0 to $\lceil \log (n/6) \rceil$

$\Rightarrow 6n + \sum_{i \geq 0} n/2^i < 8n$ nodes in total

Each node $u$ of layer $L_i$ is adjacent to:

- an interval $I(u)$ of $6 \cdot 2^i$ nodes of $S$
- its predecessor and successor in $L_i$ and all nodes vertically below
Universal Graph $\mathcal{U}_n$ for Caterpillar

Nodes:
- independent set $S$ of $6n$ nodes
- layers $L_i$ of $n/2^i$ nodes for $i$ from 0 to $\lceil \log (n/6) \rceil$

$\Rightarrow 6n + \sum_{i \geq 0} n/2^i < 8n$ nodes in total

Each node $u$ of layer $L_i$ is adjacent to:
- an interval $I(u)$ of $6 \cdot 2^i$ nodes of $S$
- its predecessor and successor in $L_i$ and all nodes vertically below

$L_0$: .................................................................

$S$: ........................................................................

$L_0$: .................................................................
Universal Graph $\mathcal{U}_n$ for Caterpillar

Nodes:
- independent set $S$ of $6n$ nodes
- layers $L_i$ of $n/2^i$ nodes for $i$ from $0$ to $\lceil \log (n/6) \rceil$

$\Rightarrow 6n + \sum_{i \geq 0} n/2^i < 8n$ nodes in total

Each node $u$ of layer $L_i$ is adjacent to:
- an interval $I(u)$ of $6 \cdot 2^i$ nodes of $S$
- its predecessor and successor in $L_i$ and all nodes vertically below
Universal Graph $\mathcal{U}_n$ for Caterpillar

Nodes:
- independent set $S$ of $6n$ nodes
- layers $L_i$ of $n/2^i$ nodes for $i$ from 0 to $\lceil \log(n/6) \rceil$

$\Rightarrow 6n + \sum_{i \geq 0} n/2^i < 8n$ nodes in total

Each node $u$ of layer $L_i$ is adjacent to:
- an interval $I(u)$ of $6 \cdot 2^i$ nodes of $S$
- its predecessor and successor in $L_i$ and all nodes vertically below
Universal Graph $U_n$ for Caterpillar

Nodes:
- independent set $S$ of $6n$ nodes
- layers $L_i$ of $n/2^i$ nodes for $i$ from 0 to $\lceil \log (n/6) \rceil$

$\Rightarrow 6n + \sum_{i \geq 0} n/2^i < 8n$ nodes in total

Each node $u$ of layer $L_i$ is adjacent to:
- an interval $I(u)$ of $6 \cdot 2^i$ nodes of $S$
- its predecessor and successor in $L_i$ and all nodes vertically below
Universal Graph $U_n$ for Caterpillar

Nodes:
- independent set $S$ of $6n$ nodes
- layers $L_i$ of $n/2^i$ nodes for $i$ from 0 to $\lceil \log (n/6) \rceil$

$\Rightarrow 6n + \sum_{i \geq 0} n/2^i < 8n$ nodes in total

Each node $u$ of layer $L_i$ is adjacent to:
- an interval $I(u)$ of $6 \cdot 2^i$ nodes of $S$
- its predecessor and successor in $L_i$ and all nodes vertically below
Universal Graph $U_n$ for Caterpillar

Nodes:
- independent set $S$ of $6n$ nodes
- layers $L_i$ of $n/2^i$ nodes for $i$ from 0 to $\lceil \log (n/6) \rceil$

$\Rightarrow 6n + \sum_{i \geq 0} n/2^i < 8n$ nodes in total

Each node $u$ of layer $L_i$ is adjacent to:
- an interval $I(u)$ of $6 \cdot 2^i$ nodes of $S$
- its predecessor and successor in $L_i$ and all nodes vertically below
Embedding Caterpillars Forests in $U_{12}$

Some caterpillar of 12 nodes

Star of 12 nodes

Matching of 6 edges

Path $P_{12}$ of 12 nodes
Adjacency in $U_n$

$\mathcal{U}_n$

$V$

$L_0$

$L_1$

$L_2$

$S$

$I(u)$

$I(v)$
Embedding in $U_n$

$\delta(u) := \text{number of leaves adjacent to internal node } u$

**Basic idea:** Embed each internal node $u_k$ in $u \in V(U_n)$ such that $I(u)$ contains at least $\delta(u)$ nodes.
How to embed $u_k$?
How to embed $u_k$?
How to embed $u_k$?
How to embed $u_k$?
How to embed $u_k$?

In order to move $u_k$, we need to move $u_{k-1}$.

Target coordinate for $u_k$.
How to embed $u_k$?

In order to move $u_{k-1}$, we need to move $u_{k-2}$.
How to embed $u_k$?
How to embed $u_k$?
How to embed $u_k$?

coordinates for $u_k$

coordinates for $u_k-1$

coordinates for $u_{k-2}$

coordinates for $u_{k-1}$

target coordinate for $u_k$
How to embed $u_k$?

now we can place $u_k$ to the desired $x$ coordinate
II. Graphs of Pathwidth $p$

The pathwidth of $G$ is the maximum clique size minus one of an interval graph containing $G$ as subgraph.

\[ p = 1 \iff G \text{ is a caterpillar forest} \]
Theorem 2
There is an adjacency labeling scheme for graphs of $n$ nodes and pathwidth $p$ with $\log n + O(p)$-bit labels. $\Rightarrow$ universal graph of $n \cdot 2^{O(p)}$ nodes.

- previous upper bound: $\log n + O(p \log \log n)$ bits
- matching lower bound: $\log n + \Omega(p)$ bits

$\left[ \geq \frac{1}{n} \log \left( \mathcal{L}_{n,p} \right) \right]$ bits where $\mathcal{L}_{n,p} = n! \cdot 2^{\Theta(np)}$ is the number of pathwidth-$p$ labeled graphs with $n$ nodes
start with the interval with the leftmost left end
at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
Dominating Path Decomposition

- start with the interval with the leftmost left end
- at each step, take as the next interval the neighbor with the rightmost right end
start with the interval with the leftmost left end
at each step, take as the next interval the neighbor with the rightmost right end

⇒ every interval is contained in the union of two consecutive intervals of the blue path
Dominating Path Decomposition

Three kinds of intervals:

- the blue intervals $B$ of the dominating path
- the red intervals $R$ that contain at least one end of a blue interval
- the remaining green intervals
green intervals have thickness strictly less than the original

repeat this process to obtain a decomposition into \((B_1, R_1), (B_2, R_2),..., (B_{p+1}, R_{p+1})\)
Blue Node Encoding

Enlarge the spaces between two consecutive ends of the path such that:

- the spaces between ends are power of two
- two consecutive spaces differ only by a multiplicative factor at most 2
Blue Node Encoding

In the label of each blue interval, we encode:

- the new left end $l$
- the integers $k_0$, $k_1$ and $k_2$
- the layer index (= index $i$ of $B_i$ containing it)
- some more information (more details later)
Red Node Encoding

In the label of each red interval, we encode information relative to the two blue intervals containing it:

- the left end $l$ of the first blue interval
- the integers $k_0, k_1, k_2, k_3$ and $k_4$
- the range $\subseteq [0, 4]$ where the interval falls into
- the layer index
- some more information (more details later)
The node with the smaller layer index is \textit{blue}.

⇒ adjacency is easy since one can compute exactly the ends of the blue interval.

- subcase 1: red interval of the same layer
- subcase 2: interval (blue or red) with greater layer
Adjacency: Hard Case

If the node with the smaller layer index is red

⇒ adjacency is harder to compute since one can only compute some range containing the ends of the interval
Adjacency: Hard Case

Each red interval stores the rank for each of its ends (order by length)

 encode for each layer the number of intersecting red intervals of this layer: 0, 3, 2

⇒ the node (blue or red) can check if the red interval corresponds to an interval intersecting it
Conclusion and Open Problems

Summary

- Universal graph $\mathcal{U}_n$ for caterpillar forests $3n/2 \leq |V(\mathcal{U}_n)| < 8n$ (pathwidth $p = 1$)
- Universal graph for graphs of pathwidth $p$ with $n \cdot 2^{\Theta(p)}$ nodes ($\log n + \Theta(p)$-bit labels)

Open Problems

- Improve bounds for universal graph for caterpillars
- $\log n + \Theta(\log p)$-bit labels for interval graphs of maximum clique size $p$
- $\log n + \Theta(t)$-bit labels for graphs of treewidth $t$