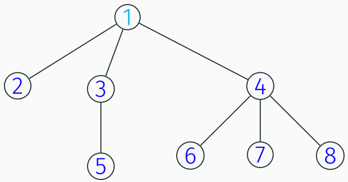
Dualities in Tree Representations

CPM'2018

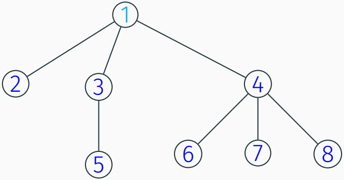
Rayan Chikhi & Alexander Schönhuth

CNRS, University of Lille, France
CWI, Netherlands

Consider an ordinal tree...



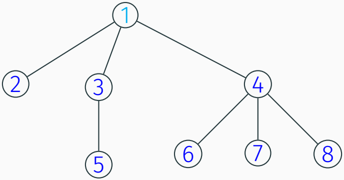
Consider an ordinal tree...



BP representation:
(( ) (( )) (( ) ( ) ( )))
12 35 46 7 8

Balanced Parenthesis:
in DFS order,
init: (
going down: (
going up:)
end:)

Consider an ordinal tree...



DFUDS representation:

((((())())((()))))

1 23 54 678

(Benoit *et al*, 2005)

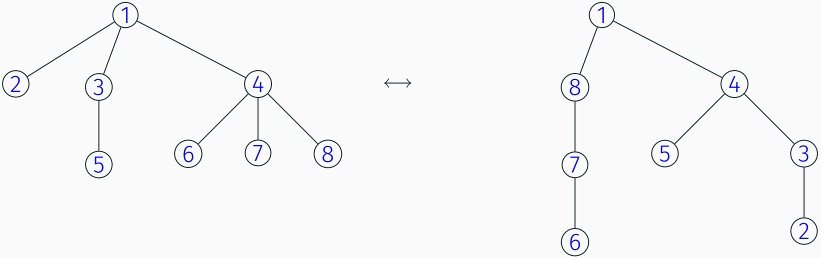
Depth-First Unary Degree Sequence:

in DFS order,

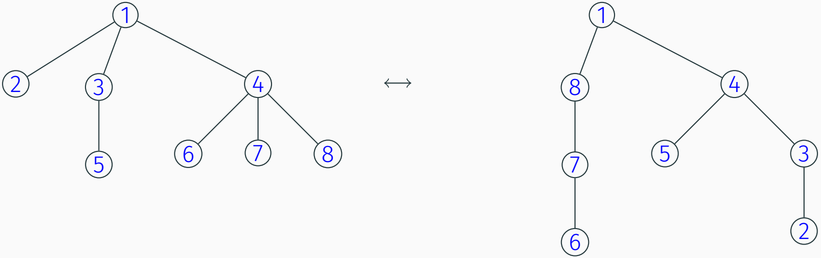
init: (

record # of children as ('s then)

Transformation: [rightmost child] ↔ [left sibling] (except for the root)



Transformation: [rightmost child] ↔ [left sibling] (except for the root)



BP of original tree:

(()(())(()()()))

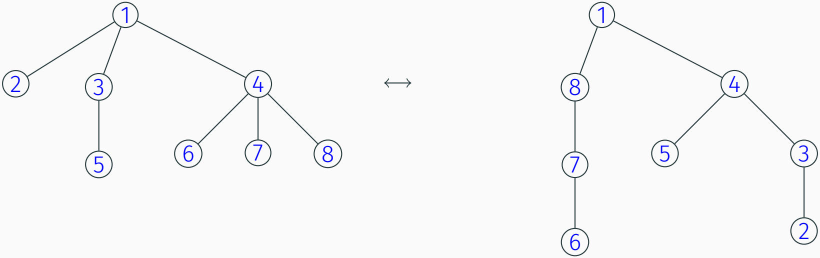
DFUDS of dual tree:

((()()())(())())

DFUDS of dual tree, mirrored:

(()(())(()()()))

Transformation: [rightmost child] ↔ [left sibling] (except for the root)



BP of original tree:

(()(())(()()()))

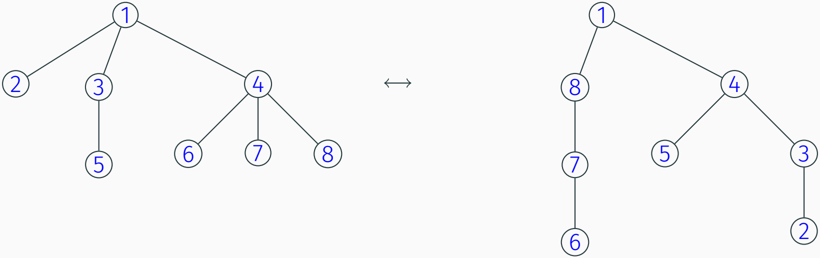
DFUDS of dual tree:

((()()())(())())

DFUDS of dual tree, mirrored:

(()(())(()()()))

Dual trees

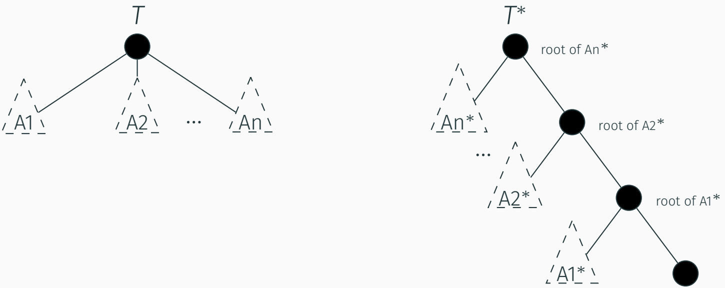


T^* defined from a tree T as:

- Rule 1: root stays the same
- Rule 1b: rightmost child of root stays the same
- Rule 2: rightmost child in T becomes left sibling in T^*
- Rule 3: left sibling in T becomes rightmost child in T^*

Property: $(T^*)^* = T$

$\mathrm{BP}(T^*) = \overleftrightarrow{DFUDS(T)}$, proof sketch



Modulo special handling of subtree roots and their respective rightmost children,

$$\begin{aligned}\mathrm{DFUDS}(T^*) &= (\mathrm{DFUDS}(\mathrm{An}^*)\ \dots\ \mathrm{DFUDS}(A2^*)\ \mathrm{DFUDS}(A1^*))\\ &= (\overleftrightarrow{\mathrm{BP}(\mathrm{An})}\ \dots\ \overleftrightarrow{\mathrm{BP}(A2)}\ \overleftrightarrow{\mathrm{BP}(A1)}), \textit{ by induction}\\ &= \overleftrightarrow{\mathrm{BP}(T)}\end{aligned}$$

Is this novel?

[Farzan *et al*'09]: data structure that emulates BP & DFUDS

[Davoodi *et al*'17]: observed the relation through binary trees, our statement is more direct

Motivation: Range Minimum Queries

Range Minimum Query:

$$\mathrm{rmq}_A(i, j) := \min\{A[k] \mid i \leq k \leq j\}.$$

Motivation: Range Minimum Queries

Range Minimum Query:

$$\mathrm{rmq}_A(i,j) := \min\{A[k] \mid i \leq k \leq j\}.$$

[Fischer & Heun, SICOMP, 2011]

- First structure using $2n + o(n)$ bits, answers queries in $O(1)$ time
- Query runs on $\mathrm{DFUDS}(T[A])$, where $T[A]$ is "2D-Min-Heap" of A

Motivation: Range Minimum Queries

Range Minimum Query:

$$\text{rmq}_A(i, j) := \min\{A[k] \mid i \leq k \leq j\}.$$

[Fischer & Heun, SICOMP, 2011]

- First structure using $2n + o(n)$ bits, answers queries in $O(1)$ time
- Query runs on $\text{DFUDS}(T[A])$, where $T[A]$ is "2D-Min-Heap" of A

[Ferrada & Navarro, JoDA, 2017]

- Construct a different tree $\widehat{T}[A]$
- Improvement: query on $\text{BP}(\widehat{T}[A])$ is shorter

Motivation: Range Minimum Queries

Range Minimum Query:

$$\mathrm{rmq}_A(i,j) := \min\{A[k] \mid i \leq k \leq j\}.$$

[Fischer & Heun, SICOMP, 2011]

- First structure using $2n + o(n)$ bits, answers queries in $O(1)$ time
- Query runs on $\mathrm{DFUDS}(T[A])$, where $T[A]$ is “2D-Min-Heap” of A

[Ferrada & Navarro, JoDA, 2017]

- Construct a different tree $\widehat{T}[A]$
- Improvement: query on $\mathrm{BP}(\widehat{T}[A])$ is shorter

However:

- $\mathrm{DFUDS}(T[A]) = \mathrm{BP}(\widehat{T}[A])$

Motivation: Range Minimum Queries

Range Minimum Query:

$$\mathrm{rmq}_A(i,j) := \min\{A[k] \mid i \leq k \leq j\}.$$

[Fischer & Heun, SICOMP, 2011]

- First structure using $2n + o(n)$ bits, answers queries in $O(1)$ time
- Query runs on $\mathrm{DFUDS}(T[A])$, where $T[A]$ is “2D-Min-Heap” of A

[Ferrada & Navarro, JoDA, 2017]

- Construct a different tree $\widehat{T}[A]$
- Improvement: query on $\mathrm{BP}(\widehat{T}[A])$ is shorter

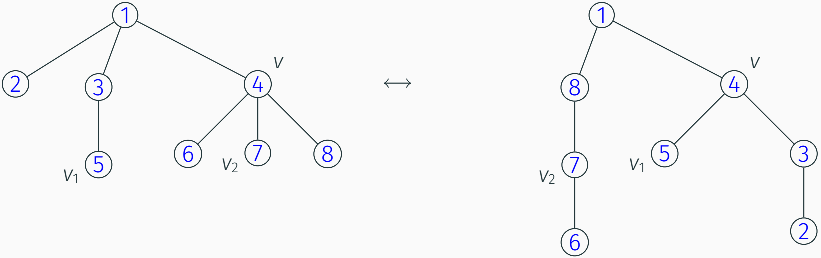
However:

- $\mathrm{DFUDS}(T[A]) = \mathrm{BP}(\widehat{T}[A])$
- What are the underlying principles?

The Primal-Dual Ancestor

Primal-dual ancestor:

Let $v_1 \leq v_2$ in T, $\mathbf{pda}(v_1, v_2) = \mathbf{v}$, s.t. $\begin{cases} v_1 \in T^*[v] \\ v_2 \in T[v] \end{cases}$



- always exists, is unique
- the rightmost (in depth-first traversal order) node between v_1 and v_2 that minimizes the depth.

RMQ reinterpreted

Range Minimum Query:

$$\mathrm{rmq}_A(i,j) := \min\{A[k] \mid i \leq k \leq j\}.$$

[Fischer & Heun, SICOMP, 2011]

- Query on $\mathrm{DFUDS}(T[A])$, where $T[A]$ is “2D-Min-Heap” of A

Re-interpretation using dual trees

$$\mathrm{rmq}_A(i,j) = \mathrm{pda}(i,j) \text{ in } T[A]$$

Motivation: Minimal Length Interval Queries

Let $([a_i, b_i])_{i \in \{1,\ldots,n\}}, a_i, b_i \in \mathbb{N}$ such that $a_i \leq b_i$ for all $i \in \{1, \ldots, n\}$ and $a_i < a_j$ and $b_i < b_j$ for $i < j$.

- **Input**: (a, b) such that $a < b$
- **Output**: The index i_0 such that $[a_{i_0}, b_{i_0}]$ is the shortest interval that contains $[a, b]$, if such an interval exists.

Motivation: Minimal Length Interval Queries

Let $([a_i, b_i])_{i\in\{1,\dots,n\}}, a_i, b_i \in \mathbb{N}$ such that $a_i \leq b_i$ for all $i \in \{1, \dots, n\}$ and $a_i < a_j$ and $b_i < b_j$ for $i < j$.

- **Input**: (a, b) such that $a < b$
- **Output**: The index i_0 such that $[a_{i_0}, b_{i_0}]$ is the shortest interval that contains $[a, b]$, if such an interval exists.

- A solution was presented in [Hu et al., SPIRE 2014] that needs $O(b_n \log b_n)$ space to answer queries in $O(1)$ time.
- Can be immediately improved to $O(n \log(b_n/n)) + o(b_n)$

Motivation: Minimal Length Interval Queries

Let $([a_i, b_i])_{i \in \{1,...,n\}}, a_i, b_i \in \mathbb{N}$ such that $a_i \leq b_i$ for all $i \in \{1, ..., n\}$ and $a_i < a_j$ and $b_i < b_j$ for $i < j$.

- **Input**: (a, b) such that $a < b$
- **Output**: The index i_0 such that $[a_{i_0}, b_{i_0}]$ is the shortest interval that contains $[a, b]$, if such an interval exists.

Further improvement

- If $|a_i - a_{i-1}|, |b_i - b_{i-1}|$ are in $O(\log n)$, further improvements possible.
- Using primal-dual ancestor logic, in combination with techniques presented in [Tsur, arXiv:1312.6039, 2015] (on succinct representatons of weighted trees), we can determine the minimum length interval using
  - two bpselect queries instead of two rank and two select queries
  - $2n \log \log n + o(n)$ space, an improvement over $O(n \log(b_n/n)) + o(b_n)$

Summary

Re-interpretation of Range Minimal Queries

Improvement of Minimal Length Interval Queries

$$\forall T, \exists T^*, BP(T^*) = \overleftrightarrow{DFUDS(T)}$$

Open question: include LOUDS representation

Paper @ LIPIcs & arXiv (full version): *Dualities in tree representations*

Questions?