On Reductions of Rooted Plane Trees

Benjamin Hackl

joint work in progress with
Sara Kropf and *Helmut Prodinger*

ALPEN-ADRIA UNIVERSITÄT
KLAGENFURT | WIEN GRAZ

2016, Hagenberg

August 1, 2016

mso DK modeling simulation optimization

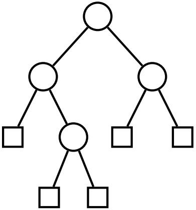
KARL POPPER KOLLEG

FWF Der Wissenschaftsfonds.

Trimming binary trees

Binary trees can be “trimmed” by the following strategy:

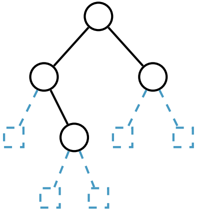
- Remove all leaves
- Merge nodes with only one descendant



Trimming binary trees

Binary trees can be “trimmed” by the following strategy:

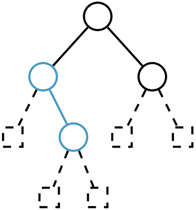
- Remove all leaves
- Merge nodes with only one descendant



Trimming binary trees

Binary trees can be "trimmed" by the following strategy:

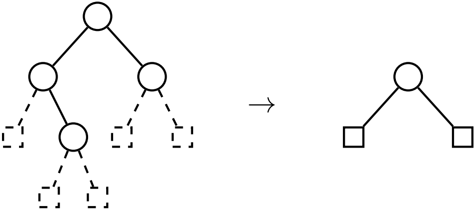
- Remove all leaves
- Merge nodes with only one descendant



Trimming binary trees

Binary trees can be “trimmed” by the following strategy:

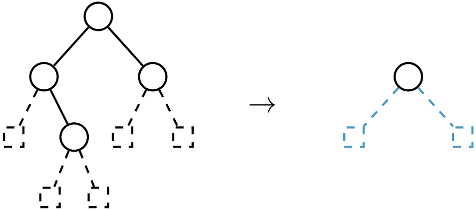
- Remove all leaves
- Merge nodes with only one descendant



Trimming binary trees

Binary trees can be “trimmed” by the following strategy:

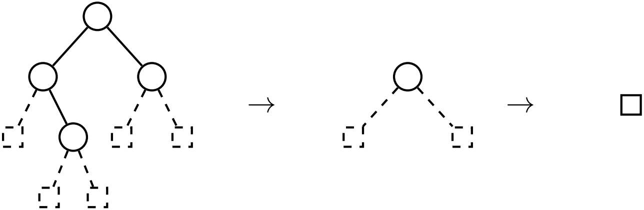
- Remove all leaves
- Merge nodes with only one descendant



Trimming binary trees

Binary trees can be “trimmed” by the following strategy:

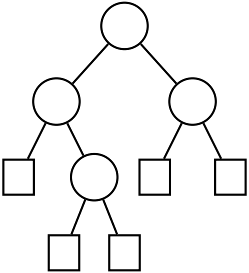
- Remove all leaves
- Merge nodes with only one descendant



“Surviving” nodes

Label all nodes in the tree by the following rules:

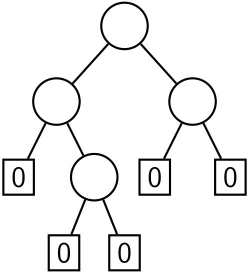
- Leaves $\rightarrow$ 0 (they do not survive a single reduction)
- val(left child) = val(right child) $\rightarrow$ increase by 1
- Otherwise: take the maximum



“Surviving” nodes

Label all nodes in the tree by the following rules:

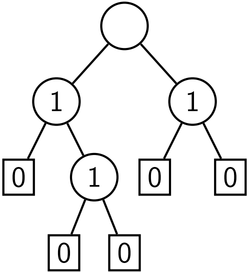
- Leaves → 0 (they do not survive a single reduction)
- val(left child) = val(right child) → increase by 1
- Otherwise: take the maximum



"Surviving" nodes

Label all nodes in the tree by the following rules:

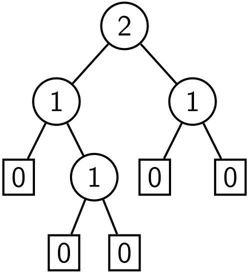
- Leaves $\rightarrow$ 0 (they do not survive a single reduction)
- val(left child) $=$ val(right child) $\rightarrow$ increase by 1
- Otherwise: take the maximum



"Surviving" nodes

Label all nodes in the tree by the following rules:

- Leaves → 0 (they do not survive a single reduction)
- val(left child) = val(right child) → increase by 1
- Otherwise: take the maximum



The register function

Number in the root of the tree: *Register function*, a.k.a. *Horton–Strahler* number

The register function

Number in the root of the tree: *Register function*, a.k.a. *Horton–Strahler* number

- Register function = maximal number of tree trimmings

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

The register function

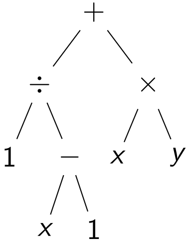
Number in the root of the tree: *Register function*, a.k.a. *Horton–Strahler* number

- Register function = maximal number of tree trimmings
- Applications:

The register function

Number in the root of the tree: *Register function*, a.k.a. *Horton–Strahler* number

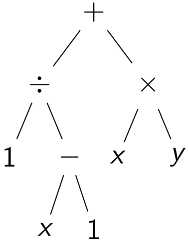
- Register function = maximal number of tree trimmings
- Applications:
  - Required stack size for evaluating an expression



The register function

Number in the root of the tree: *Register function*, a.k.a. *Horton–Strahler* number

- Register function = maximal number of tree trimmings
- Applications:
  - Required stack size for evaluating an expression
  - Branching complexity of river networks (e.g. Danube: 9)



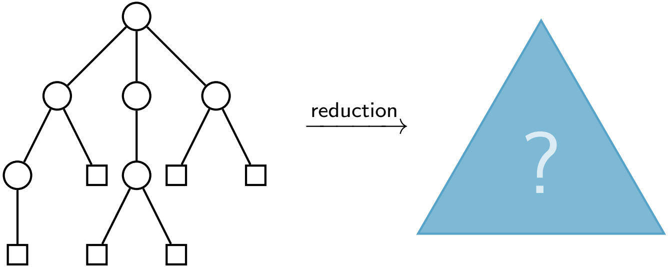
ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

"What?" and "How?"

- Analysis of "surviving nodes" after iterative reduction

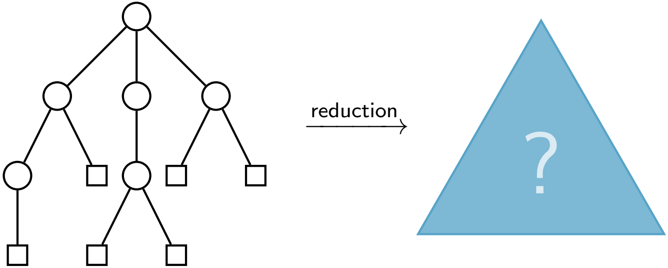
"What?" and "How?"

- Analysis of "surviving nodes" after iterative reduction



"What?" and "How?"

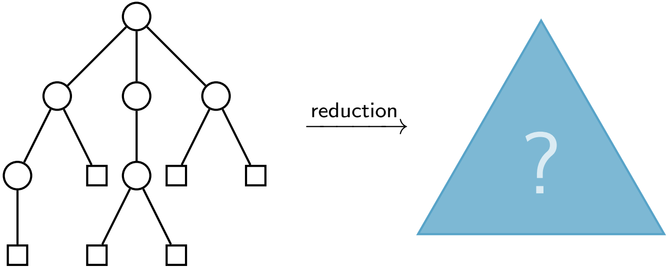
- Analysis of "surviving nodes" after iterative reduction



- Algorithmic description

"What?" and "How?"

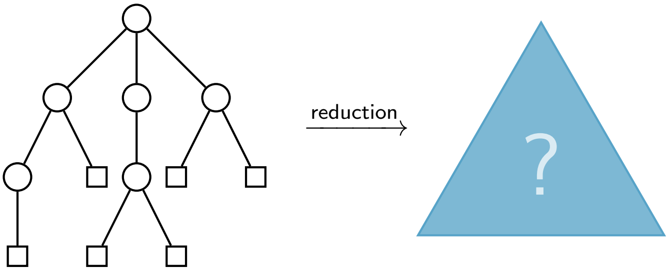
- Analysis of "surviving nodes" after iterative reduction



- Algorithmic description
- Investigation of "tree expansion" $\rightsquigarrow$ GF

"What?" and "How?"

- Analysis of "surviving nodes" after iterative reduction



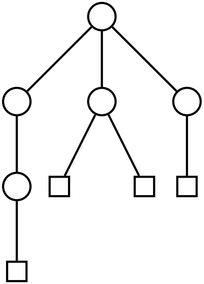
- Algorithmic description
- Investigation of "tree expansion" $\rightsquigarrow$ GF
- Coefficient extraction; Parameter distribution

How do we cut our trees?

- Remove all leaves!

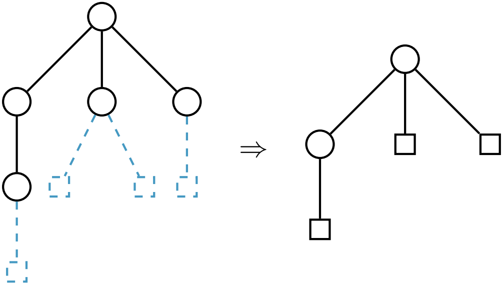
How do we cut our trees?

- Remove all leaves!



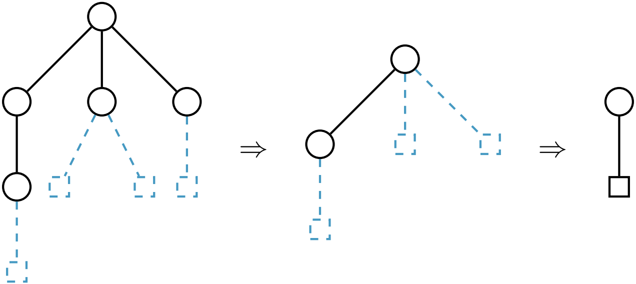
How do we cut our trees?

- Remove all leaves!



How do we cut our trees?

- Remove all leaves!



BGF for rooted plane trees

Proposition

- $\mathcal{T}\dots$ *rooted plane trees*

BGF for rooted plane trees

Proposition

- $\mathcal{T}$ ... *rooted plane trees*
- $T(z, t)$ ... *BGF for* $\mathcal{T}$ *(*$z \rightsquigarrow$ *inner nodes,* $t \rightsquigarrow$ *leaves)*

BGF for rooted plane trees

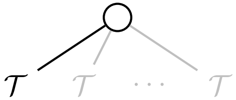
Proposition

- *$\mathcal{T}$ ... rooted plane trees*
- *$T(z,t)$ ... BGF for $\mathcal{T}$ ($z \rightsquigarrow$ inner nodes, $t \rightsquigarrow$ leaves)*

$$\Rightarrow T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}$$

Proof. Symbolic equation

$$\mathcal{T} \quad = \quad \Box \quad +$$



translates into

$$T(z,t) = t + z \cdot \frac{T(z,t)}{1-T(z,t)}$$

which can be solved explicitly.

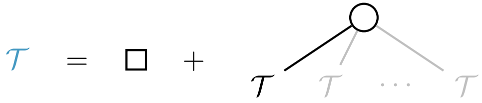
BGF for rooted plane trees

Proposition

- $\mathcal{T}$ ... *rooted plane trees*
- $T(z,t)$... *BGF for* $\mathcal{T}$ *(*$z \rightsquigarrow$ *inner nodes,* $t \rightsquigarrow$ *leaves)*

$$\Rightarrow T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}$$

Proof. Symbolic equation



translates into

$$T(z,t) = t + z \cdot \frac{T(z,t)}{1-T(z,t)}$$

which can be solved explicitly.

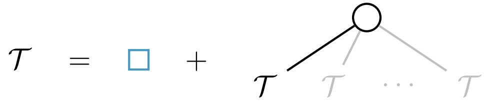
BGF for rooted plane trees

Proposition

- $\mathcal{T}$ ... *rooted plane trees*
- $T(z,t)$ ... *BGF for* $\mathcal{T}$ *(*$z \rightsquigarrow$ *inner nodes,* $t \rightsquigarrow$ *leaves)*

$$\Rightarrow T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}$$

Proof. Symbolic equation



translates into

$$T(z,t) = t + z \cdot \frac{T(z,t)}{1-T(z,t)}$$

which can be solved explicitly.

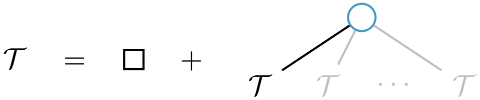
BGF for rooted plane trees

Proposition

- *$\mathcal{T}$ . . . rooted plane trees*
- *$T(z,t)$ . . . BGF for $\mathcal{T}$ ($z \rightsquigarrow$ inner nodes, $t \rightsquigarrow$ leaves)*

$$\Rightarrow T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}$$

Proof. Symbolic equation



translates into

$$T(z,t) = t + z \cdot \frac{T(z,t)}{1-T(z,t)}$$

which can be solved explicitly.

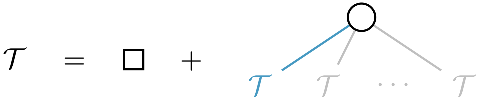
BGF for rooted plane trees

Proposition

- $\mathcal{T}$ ... *rooted plane trees*
- $T(z,t)$... *BGF for* $\mathcal{T}$ *(*$z \rightsquigarrow$ *inner nodes,* $t \rightsquigarrow$ *leaves)*

$$\Rightarrow T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}$$

Proof. Symbolic equation



translates into

$$T(z,t) = t + z \cdot \frac{T(z,t)}{1-T(z,t)}$$

which can be solved explicitly.

BGF for rooted plane trees

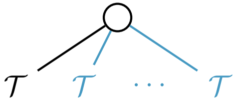
Proposition

- $\mathcal{T}$ ... *rooted plane trees*
- $T(z,t)$ ... *BGF for* $\mathcal{T}$ *(*$z \rightsquigarrow$ *inner nodes,* $t \rightsquigarrow$ *leaves)*

$$\Rightarrow T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}$$

Proof. Symbolic equation

$$\mathcal{T} \quad = \quad \square \quad +$$



translates into

$$T(z,t) = t + z \cdot \frac{T(z,t)}{1-T(z,t)}$$

which can be solved explicitly.

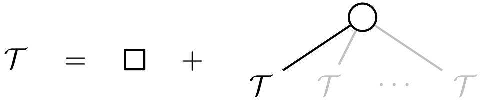
BGF for rooted plane trees

Proposition

- $\mathcal{T}\ldots$ *rooted plane trees*
- $T(z,t)\ldots$ *BGF for* $\mathcal{T}$ *(*$z \rightsquigarrow$ *inner nodes,* $t \rightsquigarrow$ *leaves)*

$$\Rightarrow T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}$$

Proof. Symbolic equation



translates into

$$T(z,t) = t + z \cdot \frac{T(z,t)}{1-T(z,t)}$$

which can be solved explicitly.

Leaf expansion

- $F \ldots$ family of rooted plane trees; BGF $f(z,t)$

Leaf expansion

- F... family of rooted plane trees; BGF $f(z, t)$
- Φ... "expansion operator" $\Rightarrow \Phi(f(z, t))$ is BGF for expanded trees

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Leaf expansion

- $F \ldots$ family of rooted plane trees; BGF $f(z,t)$
- $\Phi \ldots$ "expansion operator" $\Rightarrow \Phi(f(z,t))$ is BGF for expanded trees
- Leaf expansion: inverse operation to leaf reduction

Leaf expansion

- F ... family of rooted plane trees; BGF $f(z,t)$
- Φ... "expansion operator" $\Rightarrow \Phi(f(z,t))$ is BGF for expanded trees
- Leaf expansion: inverse operation to leaf reduction

Proposition

The linear operator

$$\Phi_L(f(z,t)) = (1-t)f\left(\frac{z}{(1-t)^2}, \frac{zt}{(1-t)^2}\right)$$

is the leaf expansion operator.

Leaf expansion

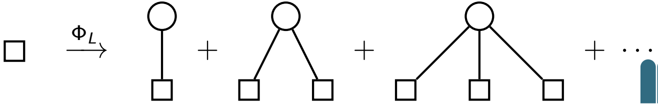
- F... family of rooted plane trees; BGF $f(z,t)$
- Φ... "expansion operator" $\Rightarrow$ $\Phi(f(z,t))$ is BGF for expanded trees
- Leaf expansion: inverse operation to leaf reduction

Proposition

The linear operator

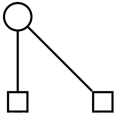
$$\Phi_L(f(z,t)) = (1-t)f\left(\frac{z}{(1-t)^2}, \frac{zt}{(1-t)^2}\right)$$

is the leaf expansion operator.



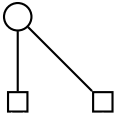
Leaf expansion — Proof

- Tree with n inner nodes and k leaves $\rightsquigarrow z^n t^k$



Leaf expansion — Proof

- Tree with n inner nodes and k leaves $\rightsquigarrow z^n t^k$
- **Expansion:**

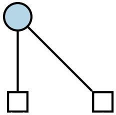


- In total:

$$\Phi_L(z^n t^k) =$$

Leaf expansion — Proof

- Tree with n inner nodes and k leaves $\rightsquigarrow z^n t^k$
- **Expansion:**
  - inner nodes stay inner nodes

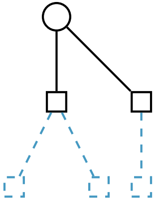


- In total:

$$\Phi_L(z^n t^k) = z^n \cdot$$

Leaf expansion — Proof

- Tree with n inner nodes and k leaves $\rightsquigarrow z^n t^k$
- **Expansion:**
  - inner nodes stay inner nodes
  - attach a non-empty sequence of leaves to all current leaves

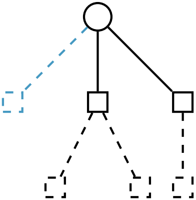


- In total:

$$\Phi_L(z^n t^k) = z^n \cdot \frac{z^k t^k}{(1-t)^k}\,.$$

Leaf expansion — Proof

- Tree with n inner nodes and k leaves $\rightsquigarrow z^n t^k$
- **Expansion:**
  - inner nodes stay inner nodes
  - attach a non-empty sequence of leaves to all current leaves
  - there are $2n + k - 1$ positions where sequences of leaves can be inserted

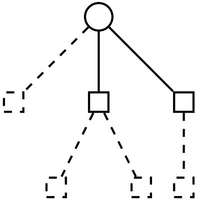


- In total:

$$\Phi_L(z^n t^k) = z^n \cdot \frac{z^k t^k}{(1-t)^k} \cdot \frac{1}{(1-t)^{2n+k-1}}$$

Leaf expansion — Proof

- Tree with n inner nodes and k leaves $\rightsquigarrow z^n t^k$
- **Expansion:**
  - inner nodes stay inner nodes
  - attach a non-empty sequence of leaves to all current leaves
  - there are $2n + k - 1$ positions where sequences of leaves can be inserted



- In total:

$$\Phi_L(z^n t^k) = z^n \cdot \frac{z^k t^k}{(1-t)^k} \cdot \frac{1}{(1-t)^{2n+k-1}}$$

- Linear extension of Φ_L proves the proposition.

Properties of Φ_L

- Functional equation: $T(z,t) = \Phi_L(T(z,t)) + t$

Properties of Φ_L

- Functional equation: $T(z,t) = \Phi_L(T(z,t)) + t$
- With $z = u/(1+u)^2$ and by some manipulations

$$\Phi_L^r(z^n t^k)|_{t=z} = \frac{1-u^{r+2}}{(1-u^{r+1})(1+u)}\left(\frac{u(1-u^{r+1})^2}{(1-u^{r+2})^2}\right)^n\left(\frac{u^{r+1}(1-u)^2}{(1-u^{r+2})^2}\right)^k$$

Properties of Φ_L

- Functional equation: $T(z,t) = \Phi_L(T(z,t)) + t$
- With $z = u/(1+u)^2$ and by some manipulations

$$\Phi_L^r(z^n t^k)|_{t=z} = \frac{1-u^{r+2}}{(1-u^{r+1})(1+u)}\left(\frac{u(1-u^{r+1})^2}{(1-u^{r+2})^2}\right)^n\left(\frac{u^{r+1}(1-u)^2}{(1-u^{r+2})^2}\right)^k$$

- BGF $G_r(z,v)$ for size comparison: z tracks original size, v size of r-fold reduced tree

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Properties of Φ_L

- Functional equation: $T(z,t) = \Phi_L(T(z,t)) + t$
- With $z = u/(1+u)^2$ and by some manipulations

$$\Phi_L^r(z^n t^k)|_{t=z} = \frac{1-u^{r+2}}{(1-u^{r+1})(1+u)}\left(\frac{u(1-u^{r+1})^2}{(1-u^{r+2})^2}\right)^n\left(\frac{u^{r+1}(1-u)^2}{(1-u^{r+2})^2}\right)^k$$

- BGF $G_r(z,v)$ for size comparison: z tracks original size, v size of r-fold reduced tree
- Intuition: v “remembers” size while tree family is expanded

Properties of Φ_L

- Functional equation: $T(z,t) = \Phi_L(T(z,t)) + t$
- With $z = u/(1+u)^2$ and by some manipulations

$$\Phi_L^r(z^n t^k)|_{t=z} = \frac{1-u^{r+2}}{(1-u^{r+1})(1+u)} \Big(\frac{u(1-u^{r+1})^2}{(1-u^{r+2})^2}\Big)^n \Big(\frac{u^{r+1}(1-u)^2}{(1-u^{r+2})^2}\Big)^k$$

- BGF $G_r(z,v)$ for size comparison: z tracks original size, v size of r-fold reduced tree
- Intuition: v “remembers” size while tree family is expanded

$$G_r(z,v) = \frac{1-u^{r+2}}{(1-u^{r+1})(1+u)} T\Big(\frac{u(1-u^{r+1})^2}{(1-u^{r+2})^2} v, \frac{u^{r+1}(1-u)^2}{(1-u^{r+2})^2} v\Big)$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \dots$ number of reductions, fixed*

Cutting leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \dots$ number of reductions, fixed*
- *$X_{n,r} \dots$ RV for size of r-fold leaf-reduced tree with originally n nodes*

Cutting leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *r ... number of reductions, fixed*
- *$X_{n,r}$ ... RV for size of r-fold leaf-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{r+1} - \frac{r(r-1)}{6(r+1)} + O(n^{-1}),$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \ldots$ number of reductions, fixed*
- *$X_{n,r} \ldots$ RV for size of r-fold leaf-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{r+1} - \frac{r(r-1)}{6(r+1)} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \frac{r(r+2)}{6(r+1)^2} n + O(1),$$

Cutting leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *r . . . number of reductions, fixed*
- *$X_{n,r}$ . . . RV for size of r-fold leaf-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{r+1} - \frac{r(r-1)}{6(r+1)} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \frac{r(r+2)}{6(r+1)^2}n + O(1),$$

and $X_{n,r}$ is asymptotically normally distributed.

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting leaves – Some insights

- $\mathbb{E}X_{n,r}$ and $\mathbb{V}X_{n,r}$ follow via singularity analysis

Cutting leaves – Some insights

- $\mathbb{E}X_{n,r}$ and $\mathbb{V}X_{n,r}$ follow via singularity analysis
- We can do even better: all factorial moments:

$$\mathbb{E}X_{n,r}^{\underline{d}} = \frac{1}{(r+1)^d} n^d + O(n^{d-1})$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting leaves – Some insights

- $\mathbb{E}X_{n,r}$ and $\mathbb{V}X_{n,r}$ follow via singularity analysis
- We can do even better: all factorial moments:

$$\mathbb{E}X_{n,r}^{\underline{d}} = \frac{1}{(r+1)^d} n^d + O(n^{d-1})$$

- This requires identities like

$$\sum_{n\geq 1} \binom{n}{d} \frac{u^{n-d}(1-ux)^{2n+d-1}(1-u)^{d-1}}{(1-u^2x)^{2n-1}} \tilde{N}_{n-1}\left(\frac{x(1-u)^2}{(1-ux)^2}\right) = \tilde{N}_{d-1}(x)$$

Cutting leaves – Some insights

- $\mathbb{E}X_{n,r}$ and $\mathbb{V}X_{n,r}$ follow via singularity analysis
- We can do even better: all factorial moments:

$$\mathbb{E}X_{n,r}^{\underline{d}} = \frac{1}{(r+1)^d} n^d + O(n^{d-1})$$

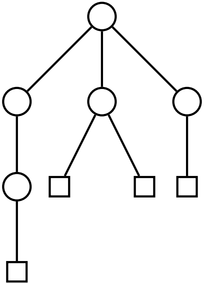
- This requires identities like

$$\sum_{n\geq 1} \binom{n}{d} \frac{u^{n-d}(1-ux)^{2n+d-1}(1-u)^{d-1}}{(1-u^2x)^{2n-1}} \tilde{N}_{n-1}\Big(\frac{x(1-u)^2}{(1-ux)^2}\Big) = \tilde{N}_{d-1}(x)$$

- Asymptotic normality: $X_{n,r}$ is a tree parameter with small toll function, limit law by Wagner (2015)

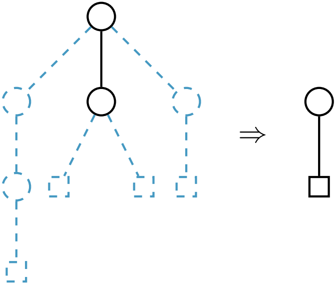
How do we cut our trees? (2)

- Remove all paths that end in a leaf!



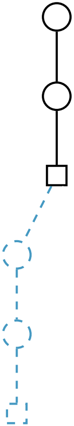
How do we cut our trees? (2)

- Remove all paths that end in a leaf!



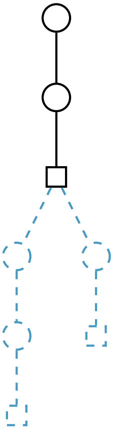
Path expansions

- Append one path to leaf ⇝ longer path ↯



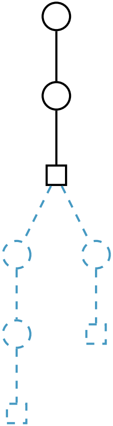
Path expansions

- Append one path to leaf ⇝ longer path ↯
- ⇒ at least two paths need to be appended



Path expansions

- Append one path to leaf $\rightsquigarrow$ longer path $\lightning$
- $\Rightarrow$ at least two paths need to be appended
- Write $p = \frac{t}{1-z}$ ... BGF for paths

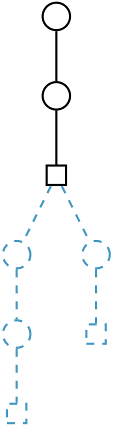


ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Path expansions

- Append one path to leaf $\rightsquigarrow$ longer path ↯
- $\Rightarrow$ at least two paths need to be appended
- Write $p = \frac{t}{1-z}$ … BGF for paths
- Similar to before we obtain

$$\Phi_P(z^n t^k) = z^n \cdot \frac{z^k p^{2k}}{(1-p)^k} \cdot \frac{1}{(1-p)^{2n+k-1}}$$

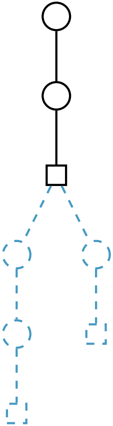


ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Path expansions

- Append one path to leaf $\rightsquigarrow$ longer path ↯
- $\Rightarrow$ at least two paths need to be appended
- Write $p = \frac{t}{1-z}$ ... BGF for paths
- Similar to before we obtain

$$\Phi_P(z^n t^k) = z^n \cdot \frac{z^k p^{2k}}{(1-p)^k} \cdot \frac{1}{(1-p)^{2n+k-1}}$$



Proposition

The linear operator given by

$$\Phi_P(f(z,t)) = (1-p) f\left(\frac{z}{(1-p)^2}, \frac{zp^2}{(1-p)^2}\right)$$

is the path expansion operator.

Generating function for path reductions

Proposition

BGF for size comparison ($z \rightsquigarrow$ original size, $v \rightsquigarrow$ r-fold path reduced size) is

$$\frac{1-u^{2^{r+1}}}{(1-u^{2^{r+1}-1})(1+u)}\,T\Big(\frac{u(1-u^{2^{r+1}-1})^2}{(1-u^{2^{r+1}})^2}v, \frac{u^{2^{r+1}-1}(1-u)^2}{(1-u^{2^{r+1}})^2}v\Big),$$

where $z = u/(1+u)^2$.

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Generating function for path reductions

Proposition

BGF for size comparison ($z \rightsquigarrow$ original size, $v \rightsquigarrow$ r-fold path reduced size) is

$$\frac{1-u^{2^{r+1}}}{(1-u^{2^{r+1}-1})(1+u)} T\Big(\frac{u(1-u^{2^{r+1}-1})^2}{(1-u^{2^{r+1}})^2} v, \frac{u^{2^{r+1}-1}(1-u)^2}{(1-u^{2^{r+1}})^2} v\Big),$$

where $z = u/(1+u)^2$.

Observation. This is the BGF for leaf reductions

$$\frac{1-u^{r+2}}{(1-u^{r+1})(1+u)} T\Big(\frac{u(1-u^{r+1})^2}{(1-u^{r+2})^2} v, \frac{u^{r+1}(1-u)^2}{(1-u^{r+2})^2} v\Big)$$

with $r \mapsto 2^{r+1} - 2$.

Cutting paths – Pruning

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \dots$ number of reductions, fixed*

Cutting paths – Pruning

Theorem (H.–Kropf–Prodinger, 2016)

- *$r\ldots$ number of reductions, fixed*
- *$X_{n,r}\ldots$ RV for size of r-fold path-reduced tree with originally n nodes*

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting paths – Pruning

Theorem (H.–Kropf–Prodinger, 2016)

- *r . . . number of reductions, fixed*
- *$X_{n,r}$ . . . RV for size of r-fold path-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{2^{r+1}-1} - \frac{(2^r-1)(2^{r+1}-3)}{3(2^{r+1}-1)} + O(n^{-1}),$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting paths – Pruning

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \ldots$ number of reductions, fixed*
- *$X_{n,r} \ldots$ RV for size of r-fold path-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{2^{r+1}-1} - \frac{(2^r-1)(2^{r+1}-3)}{3(2^{r+1}-1)} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \frac{2^{r+1}(2^r-1)}{3(2^{r+1}-1)^2}n + O(1).$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting paths – Pruning

Theorem (H.–Kropf–Prodinger, 2016)

- *r . . . number of reductions, fixed*
- *$X_{n,r}$ . . . RV for size of r-fold path-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{2^{r+1}-1} - \frac{(2^r-1)(2^{r+1}-3)}{3(2^{r+1}-1)} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \frac{2^{r+1}(2^r-1)}{3(2^{r+1}-1)^2} n + O(1).$$

Furthermore, $X_{n,r}$ is asymptotically normally distributed.

Cutting paths – Pruning

Theorem (H.–Kropf–Prodinger, 2016)

- *$r\dots$ number of reductions, fixed*
- *$X_{n,r}\dots$ RV for size of r-fold path-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{2^{r+1}-1} - \frac{(2^r-1)(2^{r+1}-3)}{3(2^{r+1}-1)} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \frac{2^{r+1}(2^r-1)}{3(2^{r+1}-1)^2}n + O(1).$$

Furthermore, $X_{n,r}$ is asymptotically normally distributed.

- Factorial moments are known as well

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting paths – Pruning

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \ldots$ number of reductions, fixed*
- *$X_{n,r} \ldots$ RV for size of r-fold path-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{n}{2^{r+1}-1} - \frac{(2^r-1)(2^{r+1}-3)}{3(2^{r+1}-1)} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \frac{2^{r+1}(2^r-1)}{3(2^{r+1}-1)^2}n + O(1).$$

Furthermore, $X_{n,r}$ is asymptotically normally distributed.

- Factorial moments are known as well
- Proof: subsequence of RV's from cutting leaves

How do we cut our trees? (3)

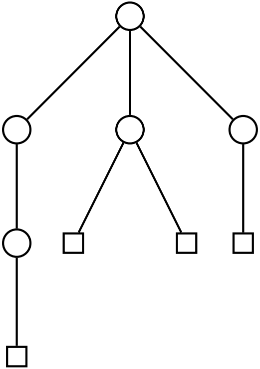
- Introduced by Chen, Deutsch, Elizalde (2006)

How do we cut our trees? (3)

- Introduced by Chen, Deutsch, Elizalde (2006)

Old leaves

- Remove all leaves that are leftmost children

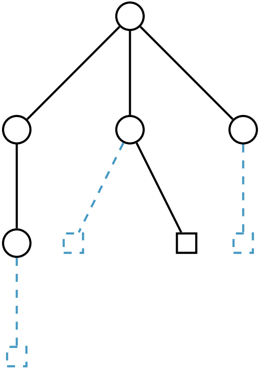


How do we cut our trees? (3)

- Introduced by Chen, Deutsch, Elizalde (2006)

Old leaves

- Remove all leaves that are leftmost children

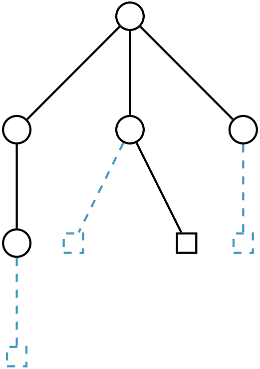


How do we cut our trees? (3)

- Introduced by Chen, Deutsch, Elizalde (2006)

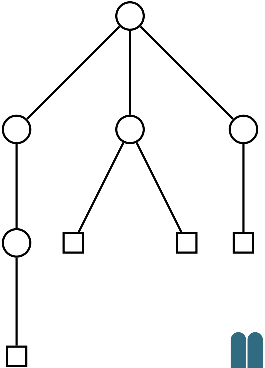
Old leaves

- Remove all leaves that are leftmost children



Old paths

- Remove all paths consisting of leftmost children

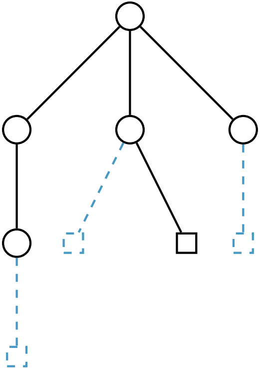


How do we cut our trees? (3)

- Introduced by Chen, Deutsch, Elizalde (2006)

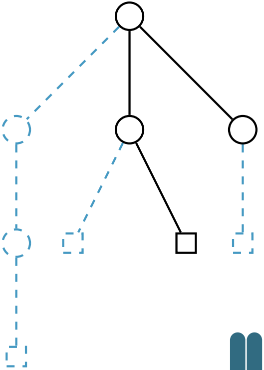
Old leaves

- Remove all leaves that are leftmost children



Old paths

- Remove all paths consisting of leftmost children



Preliminaries

Proposition

- $\mathcal{L} \ldots$ *rooted plane trees*

Preliminaries

Proposition

- *$\mathcal{L}$... rooted plane trees*
- *$L(z, w)$... BGF ($w \rightsquigarrow$ old leaves* **together with parent**, *$z \rightsquigarrow$ all other nodes)*

Preliminaries

Proposition

- *$\mathcal{L}$. . . rooted plane trees*
- *$L(z, w)$. . . BGF ($w \rightsquigarrow$ old leaves* **together with parent**, *$z \rightsquigarrow$ all other nodes)*

Then

$$L(z, w) = \frac{1 - \sqrt{1 - 4z - 4w + 4z^2}}{2}$$

Preliminaries

Proposition

- *$\mathcal{L}$... rooted plane trees*
- *$L(z, w)$... BGF ($w \rightsquigarrow$ old leaves* ***together with parent****, $z \rightsquigarrow$ all other nodes)*

Then

$$L(z, w) = \frac{1 - \sqrt{1 - 4z - 4w + 4z^2}}{2}$$

and there are $C_{k-1}\binom{n-2}{n-2k}2^{n-2k}$ trees of size n with k old leaves.

Proof. Symbolic equation

$$\mathcal{L} = \bigcirc$$

Preliminaries

Proposition

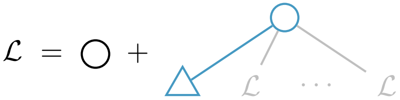
- *$\mathcal{L}$... rooted plane trees*
- *$L(z,w)$... BGF ($w \rightsquigarrow$ old leaves* **together with parent**, *$z \rightsquigarrow$ all other nodes)*

Then

$$L(z,w) = \frac{1 - \sqrt{1 - 4z - 4w + 4z^2}}{2}$$

and there are $C_{k-1}\binom{n-2}{n-2k}2^{n-2k}$ trees of size n with k old leaves.

Proof. Symbolic equation



Preliminaries

Proposition

- *$\mathcal{L}$... rooted plane trees*
- *$L(z,w)$... BGF ($w \rightsquigarrow$ old leaves* **together with parent***, $z \rightsquigarrow$ all other nodes)*

Then

$$L(z,w) = \frac{1-\sqrt{1-4z-4w+4z^2}}{2}$$

and there are $C_{k-1}\binom{n-2}{n-2k}2^{n-2k}$ trees of size n with k old leaves.

Proof. Symbolic equation

translation; Lagrange inversion.

Expansion operators

Proposition

The operators for "old leaf"- and "old path"-expansions are given by

$$\Phi_{OL}(f(z,w)) = f(z+w,(2z+w)w)$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Expansion operators

Proposition

The operators for "old leaf"- and "old path"-expansions are given by

$$\Phi_{OL}(f(z,w)) = f(z+w, (2z+w)w)$$

and

$$\Phi_{OP}(f(z,w)) = f\left(z+\frac{w}{1-z}, \left(z+\frac{w}{1-z}\right)\frac{w}{1-z}\right),$$

respectively.

Expansion operators

Proposition

The operators for "old leaf"- and "old path"-expansions are given by

$$\Phi_{OL}(f(z,w)) = f(z+w,(2z+w)w)$$

and

$$\Phi_{OP}(f(z,w)) = f\left(z+\frac{w}{1-z}, \left(z+\frac{w}{1-z}\right)\frac{w}{1-z}\right),$$

respectively.

Proof for old leaves.

$w \triangleq$ 

Expansion operators

Proposition

The operators for "old leaf"- and "old path"-expansions are given by

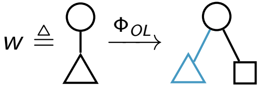
$$\Phi_{OL}(f(z,w)) = f(z+w,(2z+w)w)$$

and

$$\Phi_{OP}(f(z,w)) = f\left(z+\frac{w}{1-z},\left(z+\frac{w}{1-z}\right)\frac{w}{1-z}\right),$$

respectively.

Proof for old leaves.



Expansion operators

Proposition

The operators for "old leaf"- and "old path"-expansions are given by

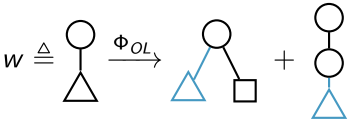
$$\Phi_{OL}(f(z,w)) = f(z+w, (2z+w)w)$$

and

$$\Phi_{OP}(f(z,w)) = f\left(z + \frac{w}{1-z}, \left(z + \frac{w}{1-z}\right)\frac{w}{1-z}\right),$$

respectively.

Proof for old leaves.



Expansion operators

Proposition

The operators for "old leaf"- and "old path"-expansions are given by

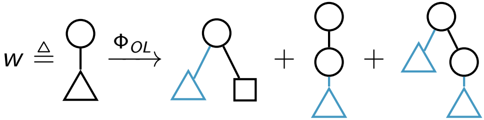
$$\Phi_{OL}(f(z,w)) = f(z+w,(2z+w)w)$$

and

$$\Phi_{OP}(f(z,w)) = f\left(z+\frac{w}{1-z},\left(z+\frac{w}{1-z}\right)\frac{w}{1-z}\right),$$

respectively.

Proof for old leaves.



Expansion operators

Proposition

The operators for “old leaf”- and “old path”-expansions are given by

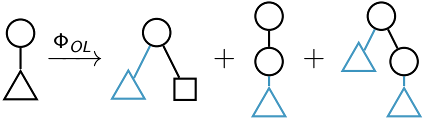
$$\Phi_{OL}(f(z,w)) = f(z+w,(2z+w)w)$$

and

$$\Phi_{OP}(f(z,w)) = f\left(z+\frac{w}{1-z},\left(z+\frac{w}{1-z}\right)\frac{w}{1-z}\right),$$

respectively.

Proof for old leaves.

$w \triangleq$  $\triangleq zw + zw + w^2$

Expansion operators

Proposition

The operators for "old leaf"- and "old path"-expansions are given by

$$\Phi_{OL}(f(z,w)) = f(z+w, (2z+w)w)$$

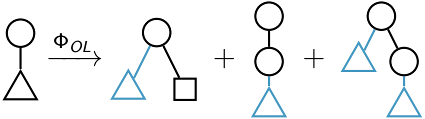
and

$$\Phi_{OP}(f(z,w)) = f\left(z + \frac{w}{1-z}, \left(z + \frac{w}{1-z}\right)\frac{w}{1-z}\right),$$

respectively.

Proof for old leaves.

$w \triangleq$



$\triangleq zw + zw + w^2$

A technical inconvenience

- Φ_{OL} and Φ_{OP} expand $\square$ incorrectly, e.g.

$$z \triangleq \square \xrightarrow{\Phi_{OL}} \square +$$

$$\triangleq z + w$$

A technical inconvenience

- Φ_{OL} and Φ_{OP} expand $\square$ incorrectly, e.g.

$$z \triangleq \square \xrightarrow{\Phi_{OL}} \square +$$  $$\triangleq z + w$$

- Correction term is required for OGF of $\geq r$-fold reducible trees:

$$\Phi_*^r(L(z,w) - z) + \Phi_*^{r-1}(w)$$

A technical inconvenience

- Φ_{OL} and Φ_{OP} expand $\square$ incorrectly, e.g.

$$z \triangleq \square \xrightarrow{\Phi_{OL}} \square +$$

$$\triangleq z + w$$

- Correction term is required for OGF of $\geq r$-fold reducible trees:

$$\Phi_*^r(L(z, w) - z) + \Phi_*^{r-1}(w)$$

- Functional equation degenerates / simplifies:

$$L(z, w) = \Phi_*(L(z, w))$$

Cutting old leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \ldots$ number of reductions, fixed*

Cutting old leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \ldots$ number of reductions, fixed*
- *$B_h(z) \ldots$ polynomial enumerating binary trees of height $\leq h$*

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT I WIEN GRAZ

Cutting old leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \ldots$ number of reductions, fixed*
- *$B_h(z) \ldots$ polynomial enumerating binary trees of height $\leq h$*

Then the expected reduced tree size after r "old leaf"-reductions and the corresponding variance are given by

$$\mathbb{E}X_{n,r} = (2 - B_{r-1}(1/4))n - \frac{B'_{r-1}(1/4)}{8} + O(n^{-1}),$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting old leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *r ... number of reductions, fixed*
- *$B_h(z)$ ... polynomial enumerating binary trees of height $\leq h$*

Then the expected reduced tree size after r "old leaf"-reductions and the corresponding variance are given by

$$\mathbb{E}X_{n,r} = (2 - B_{r-1}(1/4))n - \frac{B'_{r-1}(1/4)}{8} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \Big(B_{r-1}(1/4) - B_{r-1}(1/4)^2 + \frac{(2 - B_{r-1}(1/4))B'_{r-1}(1/4)}{2}\Big)n + O(1).$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting old leaves

Theorem (H.–Kropf–Prodinger, 2016)

- *r . . . number of reductions, fixed*
- *$B_h(z)$ . . . polynomial enumerating binary trees of height $\leq h$*

Then the expected reduced tree size after r "old leaf"-reductions and the corresponding variance are given by

$$\mathbb{E}X_{n,r} = (2 - B_{r-1}(1/4))n - \frac{B'_{r-1}(1/4)}{8} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \Big(B_{r-1}(1/4) - B_{r-1}(1/4)^2 + \frac{(2 - B_{r-1}(1/4))B'_{r-1}(1/4)}{2}\Big)n + O(1).$$

Note. Via Flajolet, Odlyzko (1982):

$$B_r(1/4) = 2 - \frac{4}{r} - \frac{4\log r}{r^2} + O(r^{-3}), \quad r \to \infty$$

Cutting old paths

Theorem (H.–Kropf–Prodinger, 2016)

- *$r\ldots$ number of reductions, fixed*

Cutting old paths

Theorem (H.–Kropf–Prodinger, 2016)

- *r . . . number of reductions, fixed*
- *$X_{n,r}$ . . . RV for size of r-fold "old path"-reduced tree with originally n nodes*

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting old paths

Theorem (H.–Kropf–Prodinger, 2016)

- *$r \dots$ number of reductions, fixed*
- *$X_{n,r} \dots$ RV for size of r-fold "old path"-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{2n}{r+2} - \frac{r(r+1)}{3(r+2)} + O(n^{-1}),$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Cutting old paths

Theorem (H.–Kropf–Prodinger, 2016)

- *r . . . number of reductions, fixed*
- *$X_{n,r}$ . . . RV for size of r-fold "old path"-reduced tree with originally n nodes*

Then the expected size of the reduced tree and the corresponding variance are

$$\mathbb{E}X_{n,r} = \frac{2n}{r+2} - \frac{r(r+1)}{3(r+2)} + O(n^{-1}),$$

$$\mathbb{V}X_{n,r} = \frac{2r(r+1)}{3(r+2)^2} n + O(1).$$

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

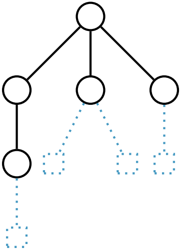
Summary

Leaves

$\mathbb{E} \sim \frac{n}{r+1}$

$\mathbb{V} \sim \frac{r(r+2)}{6(r+1)^2} n$

limit law: ✓



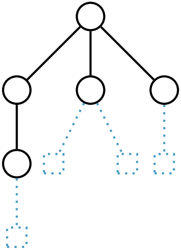
Summary

Leaves

$\mathbb{E} \sim \frac{n}{r+1}$

$\mathbb{V} \sim \frac{r(r+2)}{6(r+1)^2} n$

limit law: ✓

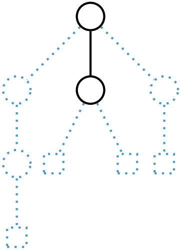


Paths

$\mathbb{E} \sim \frac{n}{2^{r+1}-1}$

$\mathbb{V} \sim \frac{2^{r+1}(2^r-1)}{3(2^{r+1}-1)^2} n$

limit law: ✓



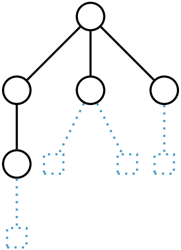
Summary

Leaves

$\mathbb{E} \sim \frac{n}{r+1}$

$\mathbb{V} \sim \frac{r(r+2)}{6(r+1)^2} n$

limit law: ✓

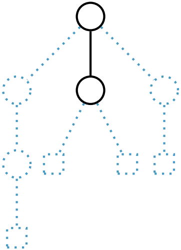


Paths

$\mathbb{E} \sim \frac{n}{2^{r+1}-1}$

$\mathbb{V} \sim \frac{2^{r+1}(2^r-1)}{3(2^{r+1}-1)^2} n$

limit law: ✓

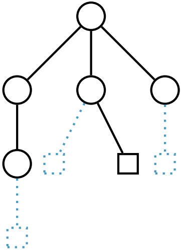


Old leaves

$\mathbb{E} \sim (2 - B_{r-1}(1/4))n$

$\mathbb{V} = \Theta(n)$

limit law: ???



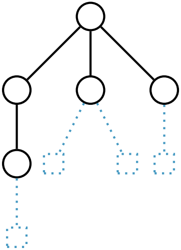
Summary

Leaves

$\mathbb{E} \sim \frac{n}{r+1}$

$\mathbb{V} \sim \frac{r(r+2)}{6(r+1)^2} n$

limit law: ✓

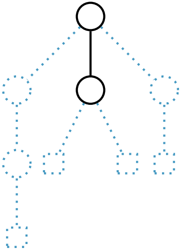


Paths

$\mathbb{E} \sim \frac{n}{2^{r+1}-1}$

$\mathbb{V} \sim \frac{2^{r+1}(2^r-1)}{3(2^{r+1}-1)^2} n$

limit law: ✓

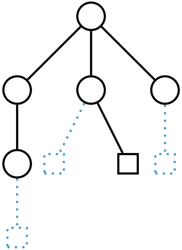


Old leaves

$\mathbb{E} \sim (2 - B_{r-1}(1/4))n$

$\mathbb{V} = \Theta(n)$

limit law: ???

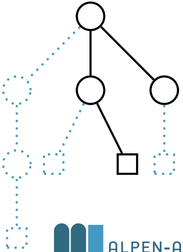


Old paths

$\mathbb{E} \sim \frac{2n}{r+2}$

$\mathbb{V} \sim \frac{2r(r+1)}{3(r+2)^2} n$

limit law: ???



ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ