Random Generation of Binary Trees

Maryam Bahrani
August 4, 2016
Trees
Trees

- **root**
- **internal node**
- **leaf**
**Binary Trees**

**Binary:** each internal node has exactly 2 children
Binary Trees

**Binary:** each internal node has exactly 2 children

**Size:** # number of internal nodes (\(\square\))

Size = 6
**Binary Trees**

**Binary:** each internal node has exactly 2 children

**Size:** # number of internal nodes (6)

Size = 6
Goal: Random Generation of Binary Trees

demo: http://www.cecm.sfu.ca/~jjh13/
Uniform Generation

**Example:** All trees of size 3 (3 internal nodes)
Example: All trees of size 3 (3 internal nodes)

Uniform = All trees equally likely to be picked
From Random Numbers to Trees

**Symbol Table:** Map from indices to trees

- **Con:** Requires a lot of memory

**Generation on the fly:**

- **Pro:** No extra memory!

  But tricky to maintain uniformity (stay tuned)
Building Trees

**Key Idea:** Big trees made up of: root + two smaller trees
Building Trees

**Key Idea:** Big trees made up of: root + two smaller trees

Recursivity!!!
Building Trees

- \text{tree}(0)
- \text{tree}(1)
- \text{tree}(2)
- \text{tree}(3)
Building Trees

tree(0)

root

tree(0)  tree(0)

tree(1)

tree(2)

tree(3)

root
Building Trees
Building Trees
Building Trees: An Algorithm

\[ \text{tree}(n) := \text{tree}(k) \cdot \text{root} \cdot \text{tree}(n-k-1) \]

Need to pull \( k \) out of randomness to ensure uniform generation

\[
\begin{align*}
\text{tree}(3) &:= \\
\text{With probability } 2/5: & \quad k = 0 \\
\text{With probability } 1/5: & \quad k = 1 \\
\text{With probability } 2/5: & \quad k = 2 \\
\text{return } & \left( \text{tree}(k), \text{root}, \text{tree}(n-k-1) \right)
\end{align*}
\]
Building Trees: An Algorithm

What about general $n$?

tree(3) :=
  With probability $2/5$:
    $k = 0$
  With probability $1/5$:
    $k = 1$
  With probability $2/5$:
    $k = 2$

return (tree(k), root, tree(n-k-1))

tree(n) :=
  With probability ???:
    $k = 0$
  With probability ???:
    $k = 1$
  ...  
  With probability ???:
    $k = n-1$

return (tree(k), root, tree(n-k-1))
Tree Enumeration

tree(n) :=

With probability ???:
  k = 0
With probability ???:
  k = 1
...
With probability ???:
  k = n-1

return (tree(k), root, tree(n-k-1))

\[ p_k := \frac{b_k \cdot b_{n-k-1}}{b_n} \]

Enumeration:
The problem of finding \( b_k \)

Known for binary trees:
(Catalan Numbers)
\[ \frac{(2n)}{n + 1} \]

More challenging in other contexts:
The topic of my research
(Graph Enumeration)
Graph Enumeration

A 3-leaf power graph

A distance-hereditary graph

A cactus graph

Split-decomposition: a technique for converting graphs into trees
Thank you!

And thanks to my advisor Dr. Jérémie Lumbroso for all of his invaluable guidance.
Appendix: Extending to Other Trees

**Binary trees:** each internal node has exactly 2 children

**Ternary trees:** each internal node has exactly 3 children

A Binary tree of size 6

A ternary tree of size 8