Probabilistic Proofs of Hooklength Formulas

Bruce Sagan
Department of Mathematics
Michigan State University
East Lansing, MI 48824-1027
sagan@math.msu.edu
www.math.msu.edu/~sagan

October 26, 2009
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set.
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set. An *increasing labeling* of $T$ is a bijection $L : T \rightarrow \{1, 2, \ldots, n\}$ such that if vertex $v$ has a child $w$ then $L(v) < L(w)$. Let

$$f^T = \text{number of increasing labelings of } T.$$
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set. An *increasing labeling* of $T$ is a bijection $L : T \rightarrow \{1, 2, \ldots, n\}$ such that if vertex $v$ has a child $w$ then $L(v) < L(w)$. Let

$$f^T = \text{number of increasing labelings of } T.$$
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set. An *increasing labeling* of $T$ is a bijection $L : T \rightarrow \{1, 2, \ldots, n\}$ such that if vertex $v$ has a child $w$ then $L(v) < L(w)$. Let

$$f^T = \text{number of increasing labelings of } T.$$

Ex.

Let $L : \begin{array}{c}
\text{Ex.} \\
T = \begin{array}{c}
\text{3} \\
\text{4}
\end{array} \\
\begin{array}{c}
\text{1} \\
\text{2} \\
\text{2} \\
\text{4}
\end{array} \\
\begin{array}{c}
\text{1} \\
\text{3} \\
\text{3} \\
\text{4}
\end{array} \\
\begin{array}{c}
\text{1} \\
\text{1} \\
\text{2} \\
\text{3}
\end{array} \\
\begin{array}{c}
\text{4} \\
\text{4} \\
\text{1} \\
\text{4}
\end{array}
\end{array}$$

$$f^T = 3.$$
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set. An *increasing labeling* of $T$ is a bijection $L : T \rightarrow \{1, 2, \ldots, n\}$ such that if vertex $v$ has a child $w$ then $L(v) < L(w)$. Let

$$f^T = \text{number of increasing labelings of } T.$$ 

![Example of an increasing labeling]

The *hooklength* of a vertex $v$ is

$$h_v = \text{number of descendents of } v \text{ (including } v).$$
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set. An increasing labeling of $T$ is a bijection $L : T \rightarrow \{1, 2, \ldots, n\}$ such that if vertex $v$ has a child $w$ then $L(v) < L(w)$. Let

$$f^T = \text{number of increasing labelings of } T.$$  

Ex. $T = L: 3 \quad 1 \quad 2 \quad 2 \quad 1 \quad 4 \quad f^T = 3$

$h_v: 2 \quad 1$

The hooklength of a vertex $v$ is

$$h_v = \text{number of descendents of } v \text{ (including } v).$$
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set. An *increasing labeling* of $T$ is a bijection $L: T \rightarrow \{1, 2, \ldots, n\}$ such that if vertex $v$ has a child $w$ then $L(v) < L(w)$. Let

$$f^T = \text{number of increasing labelings of } T.$$ 

Ex.

$T =
\begin{array}{c}
  \bullet \\
  \bullet \\
  \bullet \\
  \bullet \\
  \bullet \\
\end{array}$

$L: 
\begin{array}{c}
  3 \\
  2 \\
  2 \\
  1 \\
  1 \\
\end{array}$

$f^T = 3$

$h_v: 
\begin{array}{c}
  2 \\
  1 \\
  1 \\
\end{array}$

The *hooklength* of a vertex $v$ is

$$h_v = \text{number of descendents of } v \text{ (including } v).$$

**Theorem (Hooklength Formula)**

*If $T$ has $n$ vertices, then*

$$f^T = \frac{n!}{\prod_{v \in T} h_v}.$$
Let $T$ be a rooted tree with $n$ distinguishable vertices. We also use $T$ for its vertex set. An *increasing labeling* of $T$ is a bijection $L : T \to \{1, 2, \ldots, n\}$ such that if vertex $v$ has a child $w$ then $L(v) < L(w)$. Let

$$f^T = \text{number of increasing labelings of } T.$$ 

**Ex.**

\[
\begin{align*}
T &= \begin{array}{c}
\text{1} \\
\text{3} & \text{4} \\
\end{array} & L : \begin{array}{c}
\text{3} & \text{1} \\
\text{2} & \text{2} \\
\end{array} & f^T = 3 \\

h_v : \begin{array}{c}
\text{2} & \text{4} \\
\text{1} & \text{1} \\
\end{array} & f^T = \frac{4!}{4 \cdot 2 \cdot 1^2} = 3.
\end{align*}
\]

The *hooklength* of a vertex $v$ is

$$h_v = \text{number of descendents of } v \text{ (including } v).$$

**Theorem (Hooklength Formula)**

*If $T$ has $n$ vertices, then*

$$f^T = \frac{n!}{\prod_{v \in T} h_v}.$$
History.

1. There are also hooklength formulas for
   1.1 ordinary Young tableaux (Frame-Robinson-Thrall),
History.

1. There are also hooklength formulas for
   1.1 ordinary Young tableaux (Frame-Robinson-Thrall),
   1.2 shifted Young tableaux (Knuth), and
History.

1. There are also hooklength formulas for
   1.1 ordinary Young tableaux (Frame-Robinson-Thrall),
   1.2 shifted Young tableaux (Knuth), and
   1.3 $d$-complete posets (Proctor).
History.

1. There are also hooklength formulas for
   1.1 ordinary Young tableaux (Frame-Robinson-Thrall),
   1.2 shifted Young tableaux (Knuth), and
   1.3 $d$-complete posets (Proctor).

2. Probabilistic proofs of these formulas were given by
   2.1 Greene-Nijenhuis-Wilf (ordinary tableaux),
History.

1. There are also hooklength formulas for
   1.1 ordinary Young tableaux (Frame-Robinson-Thrall),
   1.2 shifted Young tableaux (Knuth), and
   1.3 $d$-complete posets (Proctor).

2. Probabilistic proofs of these formulas were given by
   2.1 Greene-Nijenhuis-Wilf (ordinary tableaux),
   2.2 S (shifted tableaux),
History.

1. There are also hooklength formulas for
   1.1 ordinary Young tableaux (Frame-Robinson-Thrall),
   1.2 shifted Young tableaux (Knuth), and
   1.3 $d$-complete posets (Proctor).

2. Probabilistic proofs of these formulas were given by
   2.1 Greene-Nijenhuis-Wilf (ordinary tableaux),
   2.2 S (shifted tableaux),
   2.3 S-Yeh (trees),
History.

1. There are also hooklength formulas for
   1.1 ordinary Young tableaux (Frame-Robinson-Thrall),
   1.2 shifted Young tableaux (Knuth), and
   1.3 \(d\)-complete posets (Proctor).

2. Probabilistic proofs of these formulas were given by
   2.1 Greene-Nijenhuis-Wilf (ordinary tableaux),
   2.2 S (shifted tableaux),
   2.3 S-Yeh (trees),
   2.4 Okamura (\(d\)-complete).
Let

\[ B(n) = \text{ set of binary trees, } T, \text{ on } n \text{ vertices.} \]
Let

\[ B(n) = \text{set of binary trees, } T, \text{ on } n \text{ vertices.} \]

Let

\[ \mathcal{L}(n) = \text{set of all increasing labelings, } L, \text{ of trees in } B(n). \]
Let
\[ B(n) = \text{set of binary trees, } T, \text{ on } n \text{ vertices}. \]

Let
\[ \mathcal{L}(n) = \text{set of all increasing labelings, } L, \text{ of trees in } B(n). \]

Ex.
\[ B(3) : \]
\[ \begin{array}{c}
\text{Ex.}\n\mathcal{L}(3) : \\
\end{array} \]
Let

\[ \mathcal{B}(n) = \text{set of binary trees, } T, \text{ on } n \text{ vertices.} \]

Let

\[ \mathcal{L}(n) = \text{set of all increasing labelings, } L, \text{ of trees in } \mathcal{B}(n). \]

Ex.

\[ \mathcal{B}(3) : \]

\[ \mathcal{L}(3) : \]

Theorem (Han, 2008)

For any \( n \geq 0 \),

\[ \sum_{T \in \mathcal{B}(n)} \prod_{v \in T} h_v^2 h_v - 1 = \frac{1}{n!}. \]

Notes.

1. The hooklengths appear as exponents.
2. Han's proof is algebraic. Our proof is probabilistic.
Let
\[ B(n) = \text{set of binary trees, } T, \text{ on } n \text{ vertices.} \]

Let
\[ \mathcal{L}(n) = \text{set of all increasing labelings, } L, \text{ of trees in } B(n). \]

Ex.
\[ B(3) : \]
\[ \mathcal{L}(3) : \]

Theorem (Han, 2008)
For any \( n \geq 0 \),
\[ \sum_{T \in B(n)} \prod_{v \in T} \frac{1}{h_v 2^{h_v - 1}} = \frac{1}{n!}. \]
Let 
\[ B(n) = \text{set of binary trees, } T, \text{ on } n \text{ vertices.} \]

Let 
\[ L(n) = \text{set of all increasing labelings, } L, \text{ of trees in } B(n). \]

Ex.
\[ B(3) : \]
\[ L(3) : \]

Theorem (Han, 2008)
For any \( n \geq 0 \),
\[ \sum_{T \in B(n)} \prod_{v \in T} \frac{1}{h_v 2^{h_v-1}} = \frac{1}{n!}. \]

Notes.
1. The hooklengths appear as exponents.
Let 

\[ B(n) = \text{ set of binary trees, } T, \text{ on } n \text{ vertices.} \]

Let 

\[ \mathcal{L}(n) = \text{ set of all increasing labelings, } L, \text{ of trees in } B(n). \]

Ex. 

\[ B(3) : \]

\[ \mathcal{L}(3) : \]

Theorem (Han, 2008)

For any \( n \geq 0 \),

\[ \sum_{T \in B(n)} \prod_{v \in T} \frac{1}{h_v 2^{h_v-1}} = \frac{1}{n!}. \]

Notes.

1. The hooklengths appear as exponents.
2. Han’s proof is algebraic. Our proof is probabilistic.
Outline
Theorem (Han, 2008)

For any \( n \geq 0 \),

\[
\sum_{T \in B(n)} \prod_{v \in T} \frac{1}{h_v 2^{h_v - 1}} = \frac{1}{n!}.
\]
Theorem (Han, 2008)

For any $n \geq 0$,

$$\sum_{T \in B(n)} \prod_{v \in T} \frac{1}{h_v 2^{h_v-1}} = \frac{1}{n!}.$$  

Proof Multiplying the above equation by $n!$ and using the Hooklength Formula, it suffices to show

$$\sum_{T \in B(n)} f^T \prod_{v \in T} \frac{1}{2^{h_v-1}} = 1.$$  

**Theorem (Han, 2008)**

For any \( n \geq 0 \),

\[
\sum_{T \in B(n)} \prod_{v \in T} \frac{1}{h_v 2^{h_v - 1}} = \frac{1}{n!}.
\]

**Proof** Multiplying the above equation by \( n! \) and using the Hooklength Formula, it suffices to show

\[
\sum_{T \in B(n)} f^T \prod_{v \in T} \frac{1}{2^{h_v - 1}} = 1.
\]

So it suffices to find an algorithm generating each \( L \in \mathcal{L}(n) \) such that

(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and
Theorem (Han, 2008)

*For any* $n \geq 0$,

$$
\sum_{T \in B(n)} \prod_{v \in T} \frac{1}{h_v 2^{h_v - 1}} = \frac{1}{n!}.
$$

**Proof**  Multiplying the above equation by $n!$ and using the Hooklength Formula, it suffices to show

$$
\sum_{T \in B(n)} f^T \prod_{v \in T} \frac{1}{2^{h_v - 1}} = 1.
$$

So it suffices to find an algorithm generating each $L \in \mathcal{L}(n)$ such that

(1) $\text{prob } L = \prod_{v \in T} 1/2^{h_v - 1}$ if $L$ labels $T$, and
(2) $\sum_{L \in \mathcal{L}(n)} \text{prob } L = 1$. 

(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and

(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1. \)
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v-1} \) if \( L \) labels \( T \), and
(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1. \)

For \( w \in T \), consider the \textit{depth} of \( w \):

\[
d_w = \text{length of the unique root-to-}w\text{ path.}
\]
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and
(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1. \)

For \( w \in T \), consider the **depth** of \( w \):

\[
d_w = \text{length of the unique root-to-} w \text{ path.}
\]

**Algorithm.** (a) Let \( L \) consist of a root labeled 1.
\( \textbf{Algorithm.} \) (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob } w = 1/2^{d_w} \).
(I) $\text{prob } L = \prod_{v \in T} 1/2^{h_v} - 1$ if $L$ labels $T$, and
(II) $\sum_{L \in \mathcal{L}(n)} \text{prob } L = 1$.

For $w \in T$, consider the depth of $w$:

$$d_w = \text{length of the unique root-to-}w\text{ path}.$$ 

**Algorithm.** (a) Let $L$ consist of a root labeled 1.
(b) While $|L| < n$, pick a leaf $w$ to be added to $L$ with label $|L| + 1$ and $\text{prob } w = 1/2^{d_w}$.
(c) Output $L$. 
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and
(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1 \).

For \( w \in T \), consider the depth of \( w \):

\[
d_w = \text{length of the unique root-to-}w\text{-path.}
\]

**Algorithm.**
(a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob } w = 1/2^{d_w} \).
(c) Output \( L \).

*Ex. \( n = 3 \)*
(I) \( \text{prob} \ L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and
(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob} \ L = 1. \)

For \( w \in T \), consider the \textit{depth} of \( w \):

\[
d_w = \text{length of the unique root-to-}w\text{-path.}
\]

**Algorithm.** (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob} \ w = 1/2^{d_w} \).
(c) Output \( L \).

Ex. \( n = 3 \)

\[
L: \quad \bullet \quad 1
\]
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and

(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1. \)

For \( w \in T \), consider the **depth** of \( w \):

\[
d_w = \text{length of the unique root-to-}w\text{ path.}
\]

**Algorithm.** (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob } w = 1/2^{d_w} \).
(c) Output \( L \).

**Ex.** \( n = 3 \)

\[
L : \quad 1
\]

\[\text{prob } L = 1\]
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v-1} \) if \( L \) labels \( T \), and

(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1. \)

For \( w \in T \), consider the \textit{depth} of \( w \):

\[ d_w = \text{length of the unique root-to-}w \text{ path.} \]

\textbf{Algorithm.} (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob } w = 1/2^{d_w}. \)
(c) Output \( L \).

\textbf{Ex.} \( n = 3 \)

\[
L : \quad \begin{array}{c}
1 \\
\frac{1}{2} \quad \frac{1}{2}
\end{array}
\]

\( \text{prob } L = 1 \)
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and
(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1 \).

For \( w \in T \), consider the \textit{depth} of \( w \):

\[ d_w = \text{length of the unique root-to-}w \text{ path}. \]

\textbf{Algorithm.} (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob } w = 1/2^{d_w} \).
(c) Output \( L \).

\textbf{Ex.} \( n = 3 \)

\( L : \)

\[ \begin{align*}
\frac{1}{2} & \quad \frac{1}{2} \\
2 & \quad 1
\end{align*} \]

\( \text{prob } L = 1 \)
(I) \( \text{prob} \, L = \prod_{v \in T} 1/2^{h_v-1} \) if \( L \) labels \( T \), and

(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob} \, L = 1 \).

For \( w \in T \), consider the **depth** of \( w \):

\[
d_w = \text{length of the unique root-to-}w \text{ path}.
\]

**Algorithm.**

(a) Let \( L \) consist of a root labeled 1.

(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob} \, w = 1/2^{d_w} \).

(c) Output \( L \).

**Ex.** \( n = 3 \)

\[
L:\quad \begin{array}{c}
1 \\
\circ \quad \circ \quad \circ \\
\frac{1}{2} \quad \frac{1}{2} \quad 2
\end{array}
\]

\[
\text{prob} \, L = 1 \quad \cdot \quad \frac{1}{2}
\]
(I) \( \text{prob} \ L = \prod_{v \in T} 1/2^{h_v-1} \) if \( L \) labels \( T \), and

(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob} \ L = 1 \).

For \( w \in T \), consider the \textit{depth} of \( w \):

\[
d_w = \text{length of the unique root-to-}w \text{ path}.
\]

**Algorithm.** (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob} \ w = 1/2^{d_w} \).
(c) Output \( L \).

**Ex.** \( n = 3 \)

\[
\text{prob} \ L = 1 \cdot \frac{1}{2} \cdot \frac{1}{2} = \prod_{v \in T} 1/2^{h_v-1}.
\]
(I) $\text{prob} \ L = \prod_{v \in T} 1/2^{h_v - 1}$ if $L$ labels $T$, and

(II) $\sum_{L \in \mathcal{L}(n)} \text{prob} \ L = 1$.

For $w \in T$, consider the depth of $w$:

$$d_w = \text{length of the unique root-to-w path}.$$

**Algorithm.** (a) Let $L$ consist of a root labeled 1.
(b) While $|L| < n$, pick a leaf $w$ to be added to $L$ with label $|L| + 1$ and $\text{prob} \ w = 1/2^{d_w}$.
(c) Output $L$.

**Ex.** $n = 3$

**L:**

:\begin{align*}
1 & \quad \frac{1}{2} \quad \frac{1}{2} \\
\frac{1}{2} & \quad \frac{1}{2} \quad 2 \\
\frac{1}{2^2} & \quad \frac{1}{2^2} \\
& \quad 2 \quad 3
\end{align*}:

$\text{prob} \ L = 1 \cdot \frac{1}{2}$
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \), and

(II) \( \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1 \).

For \( w \in T \), consider the depth of \( w \):

\[
d_w = \text{length of the unique root-to-} w \text{ path}.\]

Algorithm. (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob } w = 1/2^{d_w} \).
(c) Output \( L \).

Ex. \( n = 3 \)

\[
L: \quad \begin{array}{c}
1 \quad 1 \\
\frac{1}{2} \quad \frac{1}{2}
\end{array}
\]

\[
\text{prob } L = 1 \quad . \quad \frac{1}{2} \quad . \quad \frac{1}{2^2}
\]
(I) \[ \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \] if \( L \) labels \( T \), and

(II) \[ \sum_{L \in \mathcal{L}(n)} \text{prob } L = 1. \]

For \( w \in T \), consider the **depth** of \( w \):

\[ d_w = \text{length of the unique root-to-}w \text{ path.} \]

**Algorithm.** (a) Let \( L \) consist of a root labeled 1.
(b) While \( |L| < n \), pick a leaf \( w \) to be added to \( L \) with label \( |L| + 1 \) and \( \text{prob } w = 1/2^{d_w} \).
(c) Output \( L \).

**Ex.** \( n = 3 \)

\[ \begin{align*}
L: & \quad \begin{array}{c}
\begin{array}{c}
1 \quad \frac{1}{2} \quad \frac{1}{2}
\end{array}
\end{array} \\
& \quad \begin{array}{c}
\begin{array}{c}
2 \quad \frac{1}{2^2} \quad \frac{1}{2^2}
\end{array}
\end{array} \\
& \quad \begin{array}{c}
\begin{array}{c}
3 \quad \frac{1}{2} \quad \frac{1}{2^2}
\end{array}
\end{array}
\end{align*} \]

\[ h_v = \begin{array}{c}
\begin{array}{c}
3 \quad 2 \quad 1
\end{array}
\end{array} \]

\[ \text{prob } L = 1 \]

\[ \frac{1}{2} \quad \frac{1}{2^2} \]
(I) $\text{prob } L = \prod_{v \in T} 1/2^{h_v-1}$ if $L$ labels $T$, and

(II) $\sum_{L \in \mathcal{L}(n)} \text{prob } L = 1$.

For $w \in T$, consider the \textit{depth} of $w$:

$$d_w = \text{length of the unique root-to-}w\text{ path.}$$

\textbf{Algorithm.} (a) Let $L$ consist of a root labeled 1.
(b) While $|L| < n$, pick a leaf $w$ to be added to $L$ with label $|L| + 1$ and $\text{prob } w = 1/2^{d_w}$.
(c) Output $L$.

\textbf{Ex.} $n = 3$

\begin{align*}
L: \\
1 \quad \frac{1}{2} \quad \frac{1}{2} \\
\frac{1}{2^2} \quad 2 \quad \frac{1}{2^2} \\
\frac{1}{2^2} \quad \frac{1}{2^2} \quad \frac{1}{2^2} \\
\frac{1}{2^2} \quad 3 \quad 1 \\
\frac{1}{2^2} = \prod_{v \in T} \frac{1}{2^{h_v-1}}.
\end{align*}
(I) $\text{prob } L = \prod_{v \in T} 1/2^{h_v - 1}$ if $L$ labels $T$. 

(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \).

**Proof**  
Let \( w \) be the node labeled \( n \) in \( L \) and let \( L' = L - w \).
(I) \( \text{prob} \ L = \prod_{v \in T} 1/2^{h_v-1} \) if \( L \) labels \( T \).

**Proof**  Let \( w \) be the node labeled \( n \) in \( L \) and let \( L' = L - w \).

The hooklengths in \( L \) and \( L' \) are related by

\[
h_v = \begin{cases} 
  h'_v + 1 & \text{if } v \text{ is on the unique root-to-}w \text{-path } P, \\
  h'_v & \text{else.}
\end{cases}
\]
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \).

**Proof**  Let \( w \) be the node labeled \( n \) in \( L \) and let \( L' = L - w \).

![Diagram of two labeled trees](image)

The hooklengths in \( L \) and \( L' \) are related by

\[
h_v = \begin{cases} 
  h'_v + 1 & \text{if } v \text{ is on the unique root-to-}w \text{ path } P, \\
  h'_v & \text{else.}
\end{cases}
\]

Note that there are \( d_w \) vertices on \( P \cap L' \).
(l) \( \text{prob} \ L = \prod_{v \in T} 1/2^{h_v - 1} \) if \( L \) labels \( T \).

**Proof** Let \( w \) be the node labeled \( n \) in \( L \) and let \( L' = L - w \).

The hooklengths in \( L \) and \( L' \) are related by

\[
 h_v = \begin{cases} 
 h'_v + 1 & \text{if } v \text{ is on the unique root-to-} w \text{ path } P, \\
 h'_v & \text{else.}
\end{cases}
\]

Note that there are \( d_w \) vertices on \( P \cap L' \). So

\[
 \text{prob} \ L = \text{prob} \ w \cdot \text{prob} \ L'
\]
(l) \text{prob } L = \prod_{v \in T} 1/2^{h_v-1} \text{ if } L \text{ labels } T.

\textbf{Proof} \text{ Let } w \text{ be the node labeled } n \text{ in } L \text{ and let } L' = L - w.

The hooklengths in } L \text{ and } L' \text{ are related by

\[ h_v = \begin{cases} 
  h'_v + 1 & \text{if } v \text{ is on the unique root-to-} w \text{ path } P, \\
  h'_v & \text{else.}
\end{cases} \]

Note that there are } d_w \text{ vertices on } P \cap L'. \text{ So

\[ \text{prob } L = \text{prob } w \cdot \text{prob } L' = \frac{1}{2^{d_w}} \prod_{v \in L'} \frac{1}{2^{h'_v-1}} \]
(I) \( \text{prob } L = \prod_{v \in T} 1/2^{h_v-1} \) if \( L \) labels \( T \).

**Proof**  Let \( w \) be the node labeled \( n \) in \( L \) and let \( L' = L - w \).

The hooklengths in \( L \) and \( L' \) are related by

\[
h_v = \begin{cases} 
  h'_v + 1 & \text{if } v \text{ is on the unique root-to-} w \text{ path } P, \\
  h'_v & \text{else.}
\end{cases}
\]

Note that there are \( d_w \) vertices on \( P \cap L' \). So

\[
\text{prob } L = \text{prob } w \cdot \text{prob } L' = \frac{1}{2^{d_w}} \prod_{v \in L'} 2^{h'_v-1} = \prod_{v \in L} 2^{h_v-1}.
\]

\( \square \)
(a) Yang generalized Han’s formula to weighted ordered trees and a similar probabilistic proof works.
(a) Yang generalized Han’s formula to weighted ordered trees and a similar probabilistic proof works. Let

\[ \mathcal{O}(n) = \text{set of ordered trees on } n \text{ vertices.} \]
(a) Yang generalized Han’s formula to weighted ordered trees and a similar probabilistic proof works. Let

\[ \mathcal{O}(n) = \text{set of ordered trees on } n \text{ vertices.} \]

If \( m \) is a variable and \( c_v \) is the number of children of \( v \) in \( T \), let

\[ \text{wt}(T) = \prod_{v \in T} \binom{m}{c_v}. \]
(a) Yang generalized Han’s formula to weighted ordered trees and a similar probabilistic proof works. Let

$$\mathcal{O}(n) = \text{set of ordered trees on } n \text{ vertices.}$$

If \( m \) is a variable and \( c_v \) is the number of children of \( v \) in \( T \), let

$$\text{wt}(T) = \prod_{v \in T} \binom{m}{c_v}. $$

**Theorem (Yang, 2008)**

For any \( n \)

$$\sum_{T \in \mathcal{O}(n)} \text{wt}(T) \prod_{v \in T} \frac{1}{h_v m^{h_v-1}} = \frac{1}{n!}. \quad \square$$
(a) Yang generalized Han’s formula to weighted ordered trees and a similar probabilistic proof works. Let

\[ \mathcal{O}(n) = \text{set of ordered trees on n vertices}. \]

If \( m \) is a variable and \( c_v \) is the number of children of \( v \) in \( T \), let

\[ \text{wt}(T) = \prod_{v \in T} \binom{m}{c_v}. \]

**Theorem (Yang, 2008)**

*For any* \( n \)

\[
\sum_{T \in \mathcal{O}(n)} \text{wt}(T) \prod_{v \in T} \frac{1}{h_v m^{h_v-1}} = \frac{1}{n!}. \]

Note that if \( m = 2 \) then

\[
\binom{m}{c_v} = \binom{2}{c_v} = \text{# of ways to make the children of } v \text{ binary}. \]
(a) Yang generalized Han’s formula to weighted ordered trees and a similar probabilistic proof works. Let

$$\mathcal{O}(n) = \text{ set of ordered trees on } n \text{ vertices.}$$

If $m$ is a variable and $c_v$ is the number of children of $v$ in $T$, let

$$\text{wt}(T) = \prod_{v \in T} \binom{m}{c_v}.$$ 

**Theorem (Yang, 2008)**

*For any $n$*

$$\sum_{T \in \mathcal{O}(n)} \text{wt}(T) \prod_{v \in T} \frac{1}{h_v m^{h_v - 1}} = \frac{1}{n!}.$$ 

Note that if $m = 2$ then

$$\binom{m}{c_v} = \binom{2}{c_v} = \text{ # of ways to make the children of } v \text{ binary.}$$

So $\text{wt}(T)$ becomes the number of ways to make $T$ binary and Yang’s result implies Han’s.
(b) One can also generalize Han’s formula and the probabilistic proof by considering \( n \)-vertex subtrees of a given infinite tree.

\[ \sum_{T \in B(n)} \prod_{v \in T} \left( 2h_v + 1 \right) = \left( \frac{2n + 1}{2} \right)! \]

Note that if \( \hat{T} \) is the completion of \( T \), i.e., \( T \) with all possible leaves added, then

\[ f_{\hat{T}} = \left( \frac{2n + 1}{2} \right)! \prod_{v \in T} \left( 2h_v + 1 \right) \]

(e) What is the analogue for tableaux of Han’s formulas?
(b) One can also generalize Han’s formula and the probabilistic proof by considering $n$-vertex subtrees of a given infinite tree.

(c) With Carla Savage, we are considering probabilistic proofs of $q$-hooklength formulas of Björner and Wachs and $q$, $t$-analogues of Novelli and Thibon.

(d) Han also proved the following result.

**Theorem (Han, 2008)**

$$\sum_{T \in B(n)} \prod_{v \in T} \left(2h_v + 1\right)^2 = \left(2n + 1\right)!.$$
(b) One can also generalize Han’s formula and the probabilistic proof by considering $n$-vertex subtrees of a given infinite tree.

(c) With Carla Savage, we are considering probabilistic proofs of $q$-hooklength formulas of Björner and Wachs and $q, t$-analogues of Novelli and Thibon.

(d) Han also proved the following result.

**Theorem (Han, 2008)**

*For any $n$,*

\[
\sum_{T \in \mathcal{B}(n)} \prod_{v \in T} \frac{1}{(2h_v + 1)2^{2h_v-1}} = \frac{1}{(2n+1)!}.
\]
(b) One can also generalize Han’s formula and the probabilistic proof by considering \( n \)-vertex subtrees of a given infinite tree.

(c) With Carla Savage, we are considering probabilistic proofs of \( q \)-hooklength formulas of Björner and Wachs and \( q, t \)-analogues of Novelli and Thibon.

(d) Han also proved the following result.

**Theorem (Han, 2008)**

For any \( n \),

\[
\sum_{T \in \mathcal{B}(n)} \prod_{v \in T} \frac{1}{(2h_v + 1)2^{2h_v-1}} = \frac{1}{(2n+1)!}.
\]

Is there a probabilistic proof?
(b) One can also generalize Han’s formula and the probabilistic proof by considering \( n \)-vertex subtrees of a given infinite tree.

(c) With Carla Savage, we are considering probabilistic proofs of \( q \)-hooklength formulas of Björner and Wachs and \( q, t \)-analogues of Novelli and Thibon.

(d) Han also proved the following result.

**Theorem (Han, 2008)**

For any \( n \),

\[
\sum_{T \in \mathcal{B}(n)} \prod_{v \in T} \frac{1}{(2h_v + 1)2^{2h_v-1}} = \frac{1}{(2n+1)!}.
\]

Is there a probabilistic proof? Note that if \( \hat{T} \) is the completion of \( T \), i.e., \( T \) with all possible leaves added, then

\[
f_{\hat{T}} = \frac{(2n + 1)!}{\prod_{v \in T} (2h_v + 1)}.
\]
(b) One can also generalize Han’s formula and the probabilistic proof by considering \( n \)-vertex subtrees of a given infinite tree.

(c) With Carla Savage, we are considering probabilistic proofs of \( q \)-hooklength formulas of Björner and Wachs and \( q, t \)-analogues of Novelli and Thibon.

(d) Han also proved the following result.

**Theorem (Han, 2008)**

For any \( n \),

\[
\sum_{T \in B(n)} \prod_{v \in T} \frac{1}{(2h_v + 1)2^{h_v-1}} = \frac{1}{(2n + 1)!}.
\]

Is there a probabilistic proof? Note that if \( \hat{T} \) is the completion of \( T \), i.e., \( T \) with all possible leaves added, then

\[
f(\hat{T}) = \frac{(2n + 1)!}{\prod_{v \in T}(2h_v + 1)}.
\]

(e) What is the analogue for tableaux of Han’s formulas?