Local limit of Mallows trees

TU/e

Benoît Corsini

Table of Content

Table of Content

Construction

Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.

Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

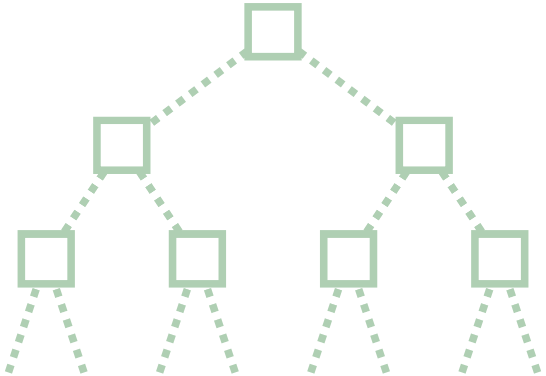
→ $x = (4, 1, 8, 6, 9)$

Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

→ $x = (4, 1, 8, 6, 9)$

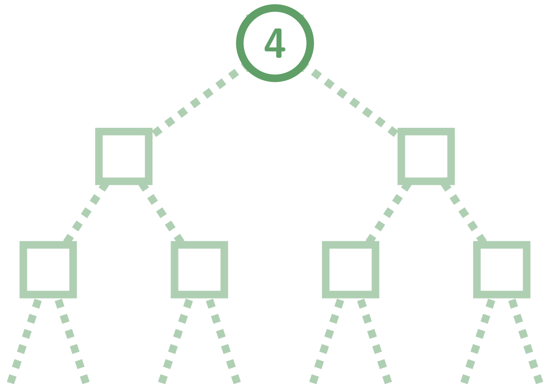


Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

→ $x = (4, 1, 8, 6, 9)$

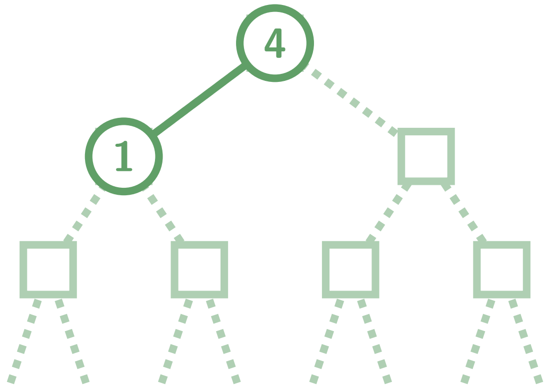


Construction

Given a sequence of distinct integers $x = (x_1, \dots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

→ $x = (4, 1, 8, 6, 9)$

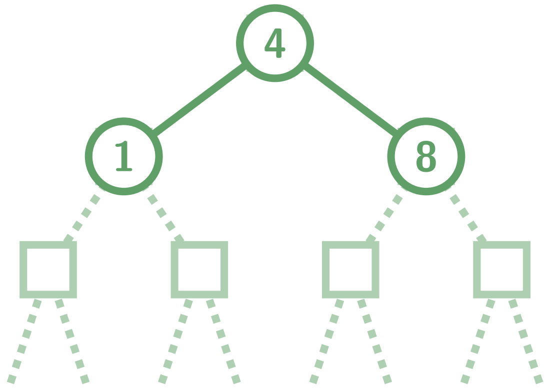


Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

→ $x = (4, 1, 8, 6, 9)$

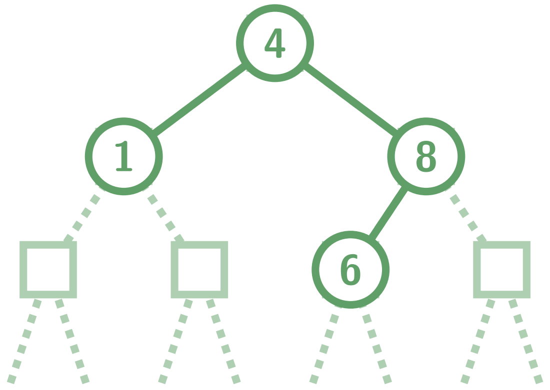


Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

→ $x = (4, 1, 8, 6, 9)$

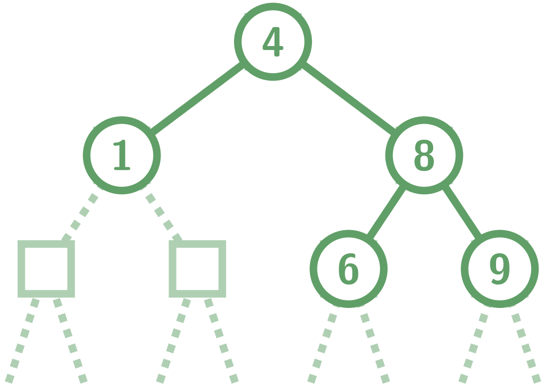


Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

→ $x = (4, 1, 8, 6, 9)$

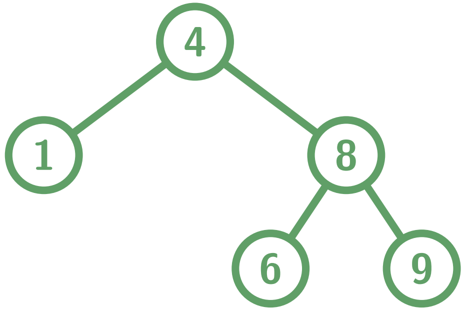


Construction

Given a sequence of distinct integers $x = (x_1, \ldots, x_n)$, the corresponding binary search tree is inductively constructed as follows:

- Insert x_1 at the root.
- Insert x_i down the tree by going to the left (respectively right) if x_i is smaller (respectively larger) than the current node value.

→ $x = (4, 1, 8, 6, 9)$



Properties

Properties

Binary search trees have a few noteworthy properties:

Properties

Binary search trees have a few noteworthy properties:

- The can be defined on infinite sequences: $x = (x_1, \ldots, x_n, \ldots) = (x_i)_{i \geq 1}$.

Properties

Binary search trees have a few noteworthy properties:

- The can be defined on infinite sequences: $x = (x_1, \ldots, x_n, \ldots) = (x_i)_{i \geq 1}$.
- Their rightmost branch corresponds to the records of the sequence: $\{i : \forall j < i, x_i > x_j\}$.

Properties

Binary search trees have a few noteworthy properties:

- The can be defined on infinite sequences: $x = (x_1, \ldots, x_n, \ldots) = (x_i)_{i \geq 1}$.
- Their rightmost branch corresponds to the records of the sequence: $\{i : \forall j < i, x_i > x_j\}$.
- When x is an infinite sequence of integers, it has an infinite rightmost branch.

Properties

Binary search trees have a few noteworthy properties:

- The can be defined on infinite sequences: $x = (x_1, \ldots, x_n, \ldots) = (x_i)_{i \geq 1}$.
- Their rightmost branch corresponds to the records of the sequence: $\{i : \forall j < i, x_i > x_j\}$.
- When x is an infinite sequence of integers, it has an infinite rightmost branch.

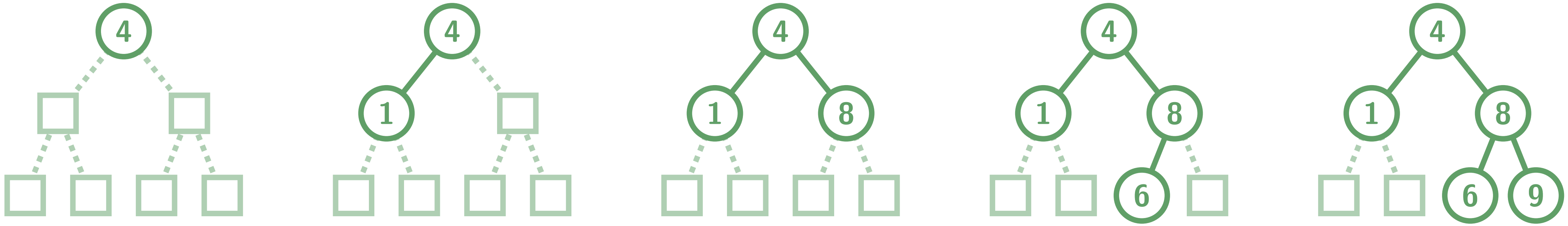
→ $x = (4, 1, 8, 6, 9)$

Properties

Binary search trees have a few noteworthy properties:

- The can be defined on infinite sequences: $x = (x_1, \ldots, x_n, \ldots) = (x_i)_{i \geq 1}$.
- Their rightmost branch corresponds to the records of the sequence: $\{i : \forall j < i, x_i > x_j\}$.
- When x is an infinite sequence of integers, it has an infinite rightmost branch.

→ $x = (4, 1, 8, 6, 9)$

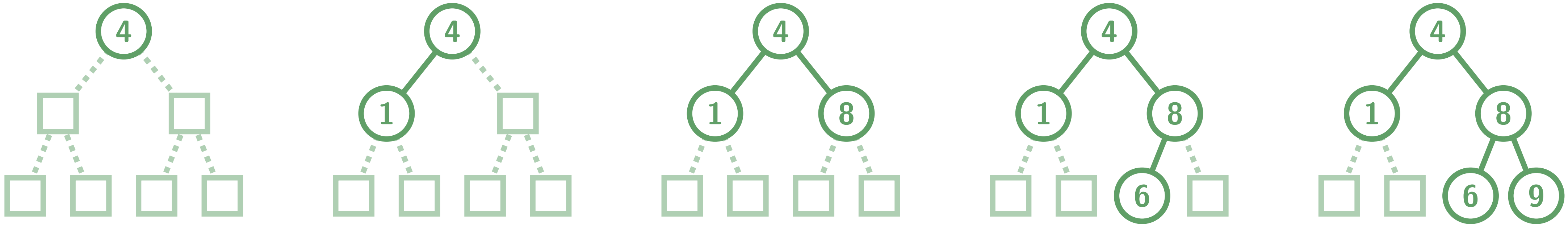


Properties

Binary search trees have a few noteworthy properties:

- The can be defined on infinite sequences: $x = (x_1, \ldots, x_n, \ldots) = (x_i)_{i\geq 1}$.
- Their rightmost branch corresponds to the records of the sequence: $\{i : \forall j < i, x_i > x_j\}$.
- When x is an infinite sequence of integers, it has an infinite rightmost branch.

$\rightarrow$ $x = (4, 1, 8, 6, 9, \ldots)$



Properties

Binary search trees have a few noteworthy properties:

- The can be defined on infinite sequences: $x = (x_1, \dots, x_n, \dots) = (x_i)_{i \geq 1}$.
- Their rightmost branch corresponds to the records of the sequence: $\{i : \forall j < i, x_i > x_j\}$.
- When x is an infinite sequence of integers, it has an infinite rightmost branch.

$\rightarrow$ $x = (4, 1, 8, 6, 9, \dots)$

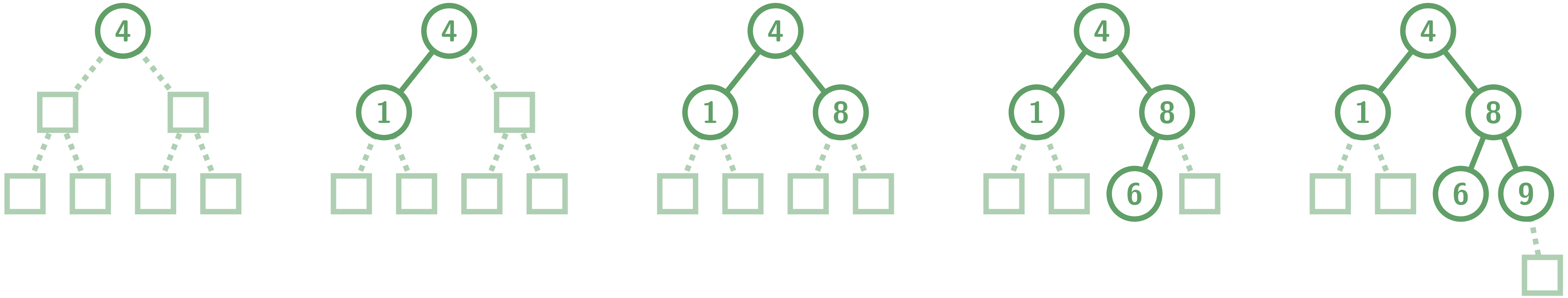


Table of Content

Mallows permutations

Mallows permutations

Definition

Mallows permutations

Definition

A Mallows permutation $X_{n,q}$ with parameters $n \in \mathbb{N}$ and $q \in [0, \infty)$ with a random permutation with distribution

$$\mathbb{P}(X_{n,q} = \sigma) = \frac{q^{\mathrm{Inv}(\sigma)}}{\prod_{k=1}^{n}(1 + q + \ldots + q^{k-1})} \propto q^{\mathrm{Inv}(\sigma)}$$

where $\mathrm{Inv}(\sigma) = |\{i < j : \sigma(i) > \sigma(j)\}|$ is the number of inversions of σ.

Mallows permutations

Definition

A Mallows permutation $X_{n,q}$ with parameters $n \in \mathbb{N}$ and $q \in [0, \infty)$ with a random permutation with distribution

$$\mathbb{P}(X_{n,q} = \sigma) = \frac{q^{\mathrm{Inv}(\sigma)}}{\prod_{k=1}^{n}(1 + q + \ldots + q^{k-1})} \propto q^{\mathrm{Inv}(\sigma)}$$

where $\mathrm{Inv}(\sigma) = |\{i < j : \sigma(i) > \sigma(j)\}|$ is the number of inversions of σ.

→ For $q = 1$, $X_{n,q}$ is a uniform permutation of size n.

Mallows permutations

Definition

A Mallows permutation $X_{n,q}$ with parameters $n \in \mathbb{N}$ and $q \in [0, \infty)$ with a random permutation with distribution

$$\mathbb{P}(X_{n,q} = \sigma) = \frac{q^{\mathrm{Inv}(\sigma)}}{\prod_{k=1}^{n}(1 + q + \ldots + q^{k-1})} \propto q^{\mathrm{Inv}(\sigma)}$$

where $\mathrm{Inv}(\sigma) = |\{i < j : \sigma(i) > \sigma(j)\}|$ is the number of inversions of σ.

→ For $q = 1$, $X_{n,q}$ is a uniform permutation of size n.

→ For $q = 0$, $X_{n,q}$ is the identity permutation.

Definition

A Mallows permutation $X_{n,q}$ with parameters $n \in \mathbb{N}$ and $q \in [0,\infty)$ with a random permutation with distribution

$$\mathbb{P}(X_{n,q} = \sigma) = \frac{q^{\mathrm{Inv}(\sigma)}}{\prod_{k=1}^{n}(1+q+\ldots+q^{k-1})} \propto q^{\mathrm{Inv}(\sigma)}$$

where $\mathrm{Inv}(\sigma) = |\{i<j : \sigma(i) > \sigma(j)\}|$ is the number of inversions of σ.

→ For $q=1$, $X_{n,q}$ is a uniform permutation of size n.
→ For $q=0$, $X_{n,q}$ is the identity permutation.
→ For $q \in (0,1)$, $X_{n,q}$ tends to be "ordered".

Definition

A Mallows permutation $X_{n,q}$ with parameters $n \in \mathbb{N}$ and $q \in [0, \infty)$ with a random permutation with distribution

$$\mathbb{P}(X_{n,q} = \sigma) = \frac{q^{\mathrm{Inv}(\sigma)}}{\prod_{k=1}^{n}(1 + q + \ldots + q^{k-1})} \propto q^{\mathrm{Inv}(\sigma)}$$

where $\mathrm{Inv}(\sigma) = |\{i < j : \sigma(i) > \sigma(j)\}|$ is the number of inversions of σ.

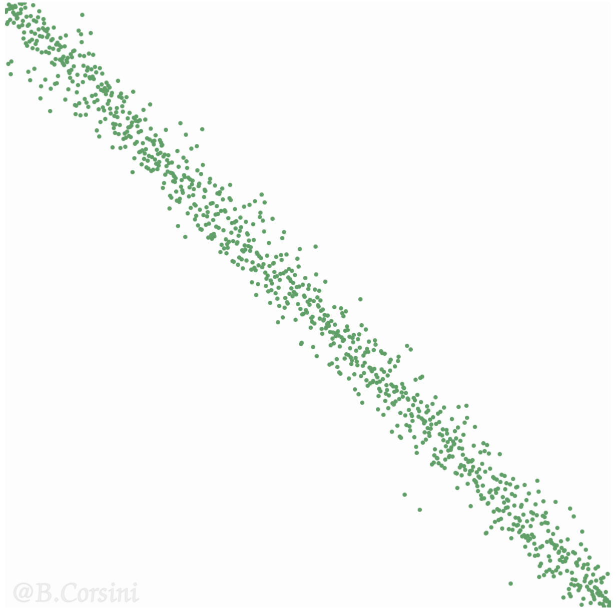
- → For $q = 1$, $X_{n,q}$ is a uniform permutation of size n.
- → For $q = 0$, $X_{n,q}$ is the identity permutation.
- → For $q \in (0, 1)$, $X_{n,q}$ tends to be "ordered".
- → We restrict ourselves to the case $q \in [0, 1)$.

Mallows trees

Mallows trees

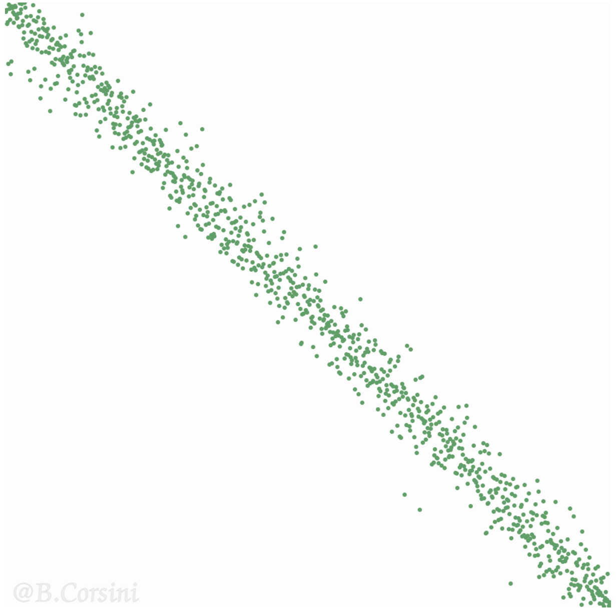
Q: What happens when we consider the binary search tree of a Mallows permutation $X_{n,q}$?

Q: What happens when we consider the binary search tree of a Mallows permutation $X_{n,q}$?

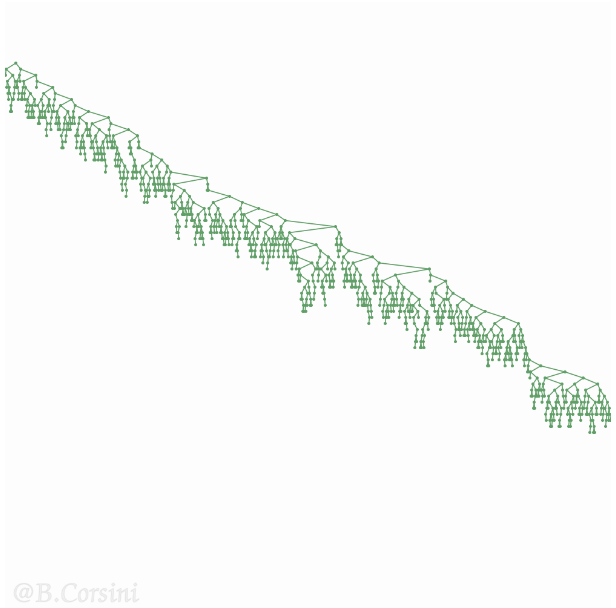


Mallows permutation

Q: What happens when we consider the binary search tree of a Mallows permutation $X_{n,q}$?



Mallows permutation



Mallows tree

Construction

Construction

In spite of their complicated definition, Mallows trees can easily be constructed inductively:

Construction

In spite of their complicated definition, Mallows trees can easily be constructed inductively:

- The size S of the left subtree of the root is a geometric conditioned to be in $[0, n-1]$:

$$\mathbb{P}(S = k) = \frac{q^k(1-q)}{1-q^n}\,.$$

Construction

In spite of their complicated definition, Mallows trees can easily be constructed inductively:

- The size S of the left subtree of the root is a geometric conditioned to be in $[0, n-1]$:

$$\mathbb{P}(S = k) = \frac{q^k(1-q)}{1-q^n}\,.$$

- The right subtree of the root has size $n - 1 - S$.

Construction

In spite of their complicated definition, Mallows trees can easily be constructed inductively:

- The size S of the left subtree of the root is a geometric conditioned to be in $[0, n-1]$:

$$\mathbb{P}(S = k) = \frac{q^k(1-q)}{1-q^n}\,.$$

- The right subtree of the root has size $n - 1 - S$.
- Both left and right subtree are Mallows trees.

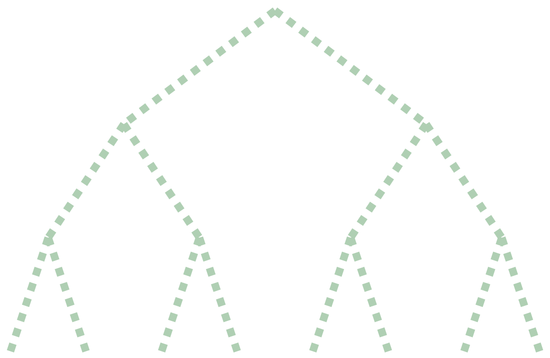
Construction

In spite of their complicated definition, Mallows trees can easily be constructed inductively:

- The size S of the left subtree of the root is a geometric conditioned to be in $[0, n-1]$:

$$\mathbb{P}(S = k) = \frac{q^k(1-q)}{1-q^n}\,.$$

- The right subtree of the root has size $n - 1 - S$.
- Both left and right subtree are Mallows trees.

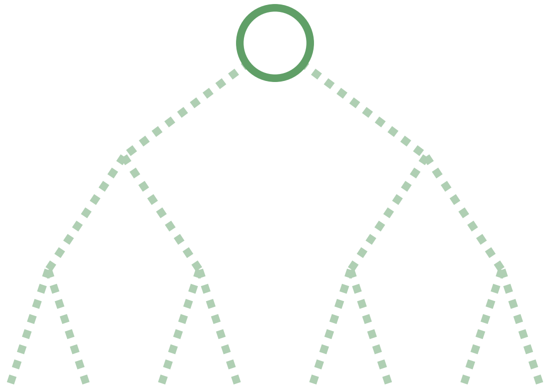


In spite of their complicated definition, Mallows trees can easily be constructed inductively:

- The size S of the left subtree of the root is a geometric conditioned to be in $[0, n-1]$:

$$\mathbb{P}(S = k) = \frac{q^k(1-q)}{1-q^n}\,.$$

- The right subtree of the root has size $n - 1 - S$.
- Both left and right subtree are Mallows trees.



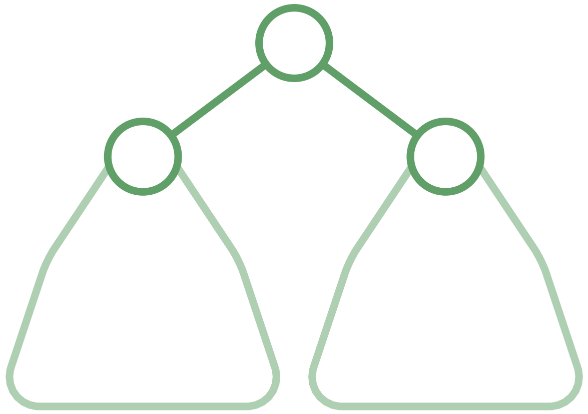
Construction

In spite of their complicated definition, Mallows trees can easily be constructed inductively:

- The size S of the left subtree of the root is a geometric conditioned to be in $[0, n-1]$:

$$\mathbb{P}(S = k) = \frac{q^k(1-q)}{1-q^n} .$$

- The right subtree of the root has size $n - 1 - S$.
- Both left and right subtree are Mallows trees.



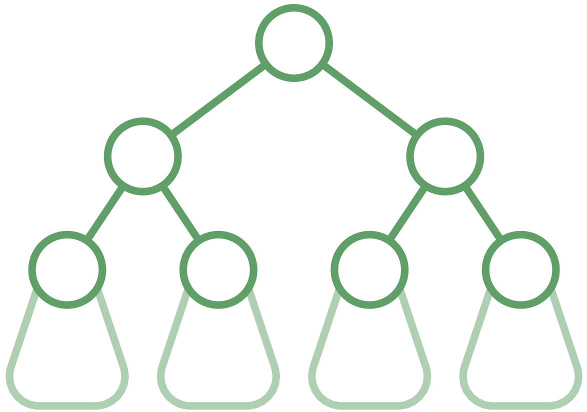
Construction

In spite of their complicated definition, Mallows trees can easily be constructed inductively:

- The size S of the left subtree of the root is a geometric conditioned to be in $[0, n-1]$:

$$\mathbb{P}(S = k) = \frac{q^k(1-q)}{1-q^n}\,.$$

- The right subtree of the root has size $n - 1 - S$.
- Both left and right subtree are Mallows trees.



Construction (approximative)

Construction (approximative)

A Mallows tree is approximately constructed as follows:

Construction (approximative)

A Mallows tree is approximately constructed as follows:

- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1-q)$.

Construction (approximative)

A Mallows tree is approximately constructed as follows:

- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1-q)$.
- Generate a random Mallows tree of size G with parameter q.

Construction (approximative)

A Mallows tree is approximately constructed as follows:

- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1 - q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.

Construction (approximative)

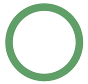
A Mallows tree is approximately constructed as follows:

- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1 - q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.

Construction (approximative)

A Mallows tree is approximately constructed as follows:

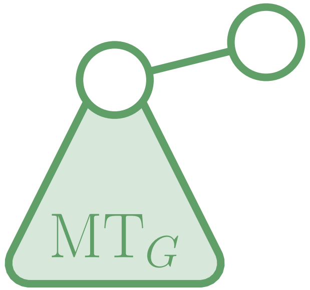
- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1 - q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.



Construction (approximative)

A Mallows tree is approximately constructed as follows:

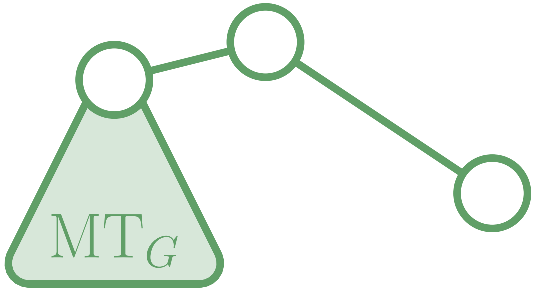
- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1 - q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.



Construction (approximative)

A Mallows tree is approximately constructed as follows:

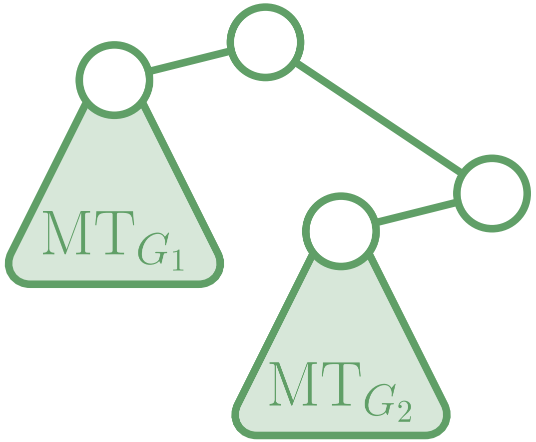
- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1 - q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.



Construction (approximative)

A Mallows tree is approximately constructed as follows:

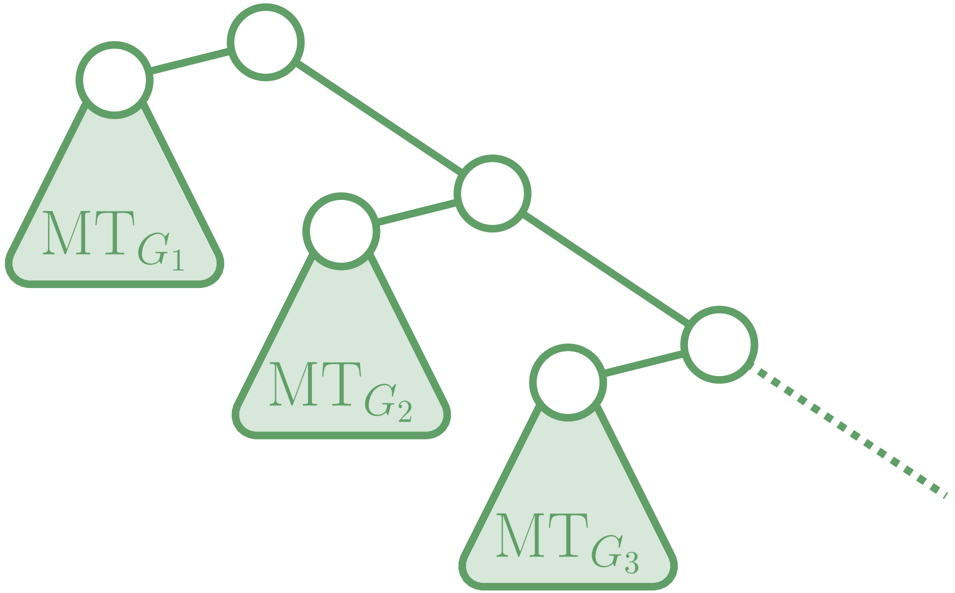
- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1 - q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.



Construction (approximative)

A Mallows tree is approximately constructed as follows:

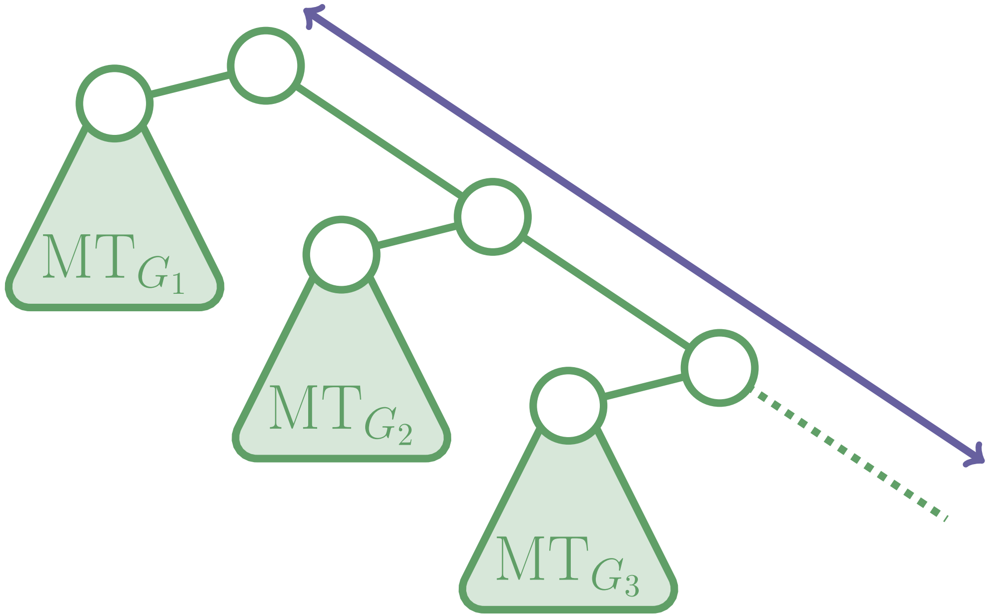
- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1-q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.



Construction (approximative)

A Mallows tree is approximately constructed as follows:

- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1-q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.



Construction (approximative)

A Mallows tree is approximately constructed as follows:

- Generate a random geometric random variable G with $\mathbb{P}(G = k) = q^k(1-q)$.
- Generate a random Mallows tree of size G with parameter q.
- Attach it to the left of the root.
- Go down to the right child of the root and repeat until the tree has n nodes.

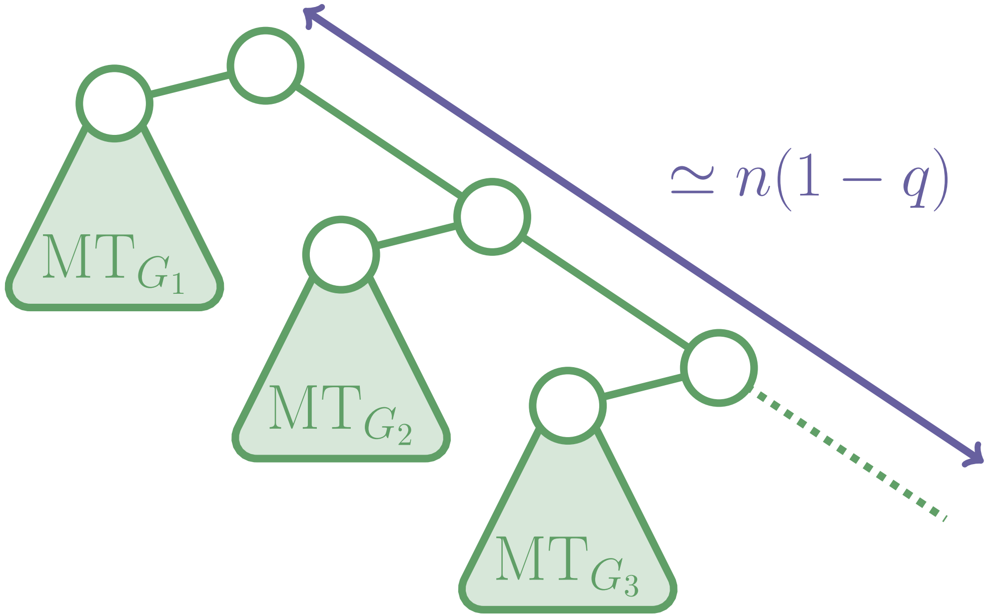


Image (exact)

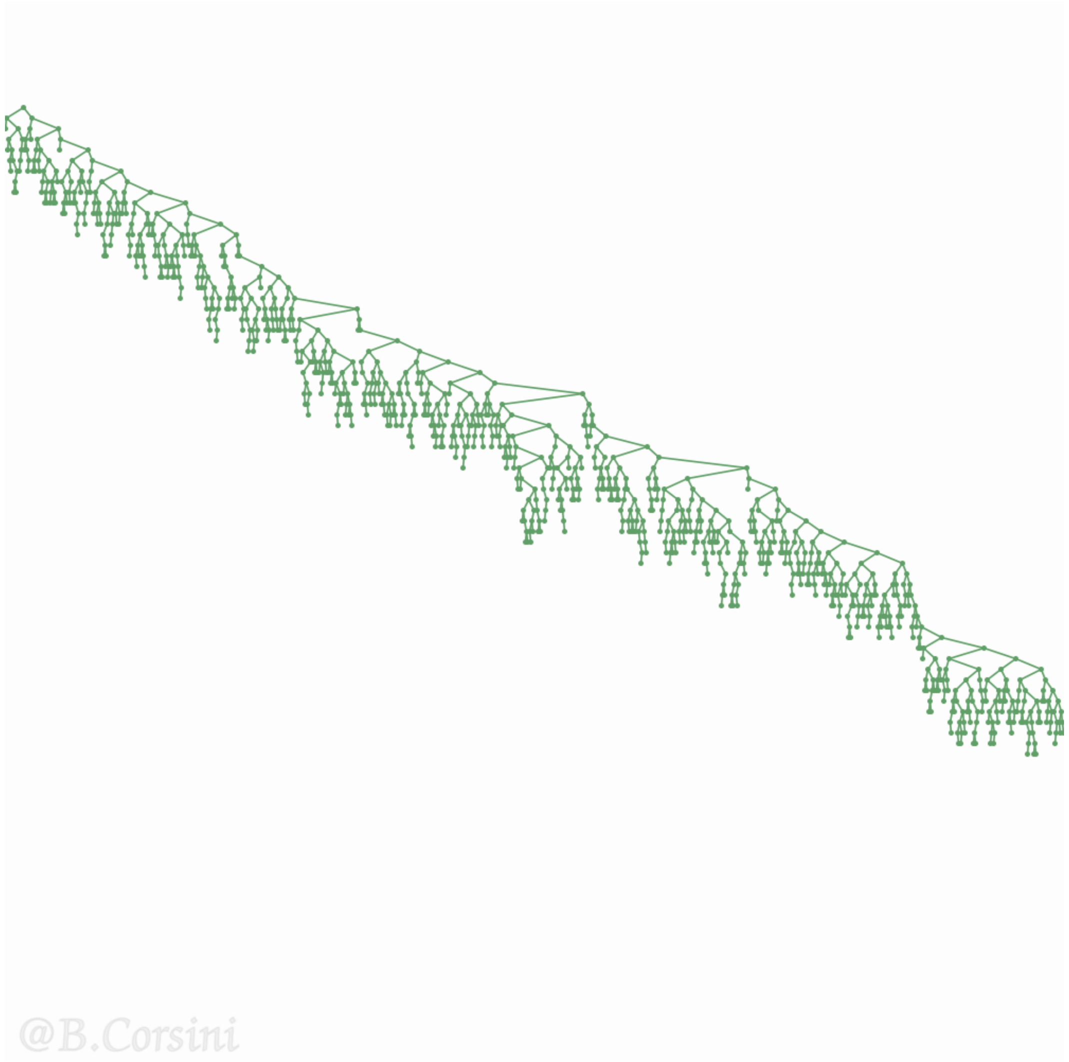
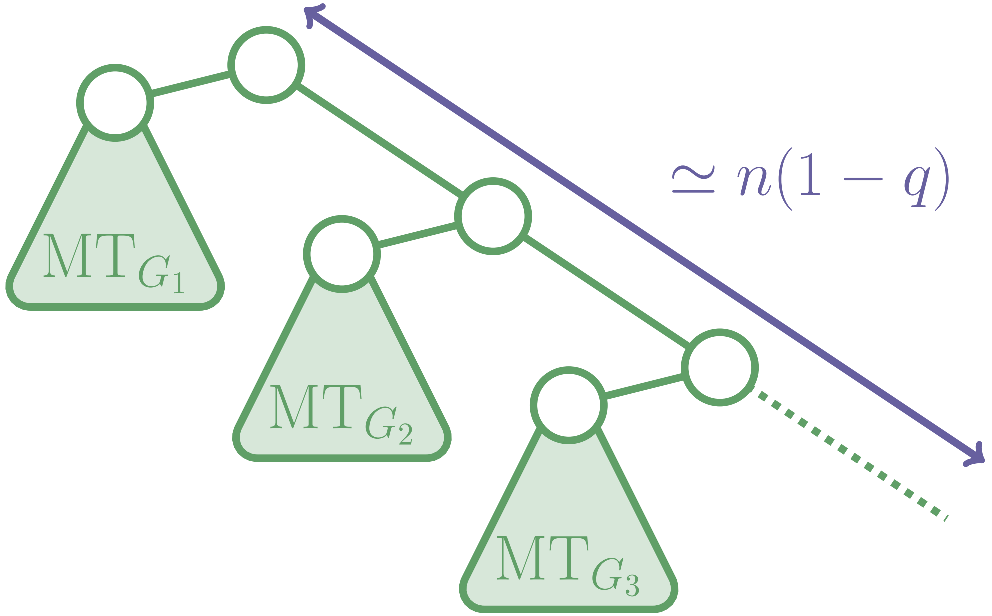


Table of Content

Guessing the local limit

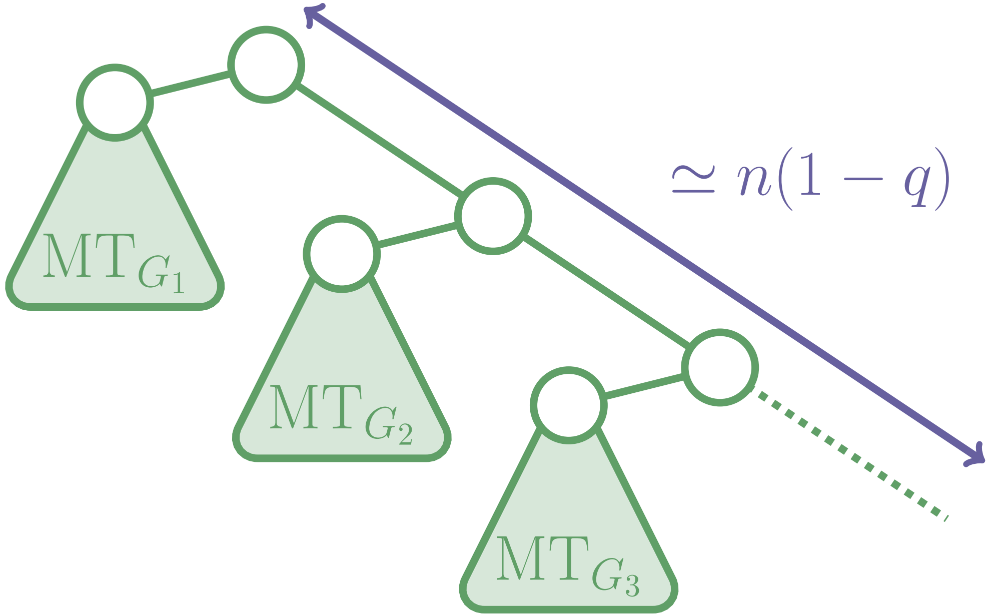
Guessing the local limit

Mallows tree: :



Guessing the local limit

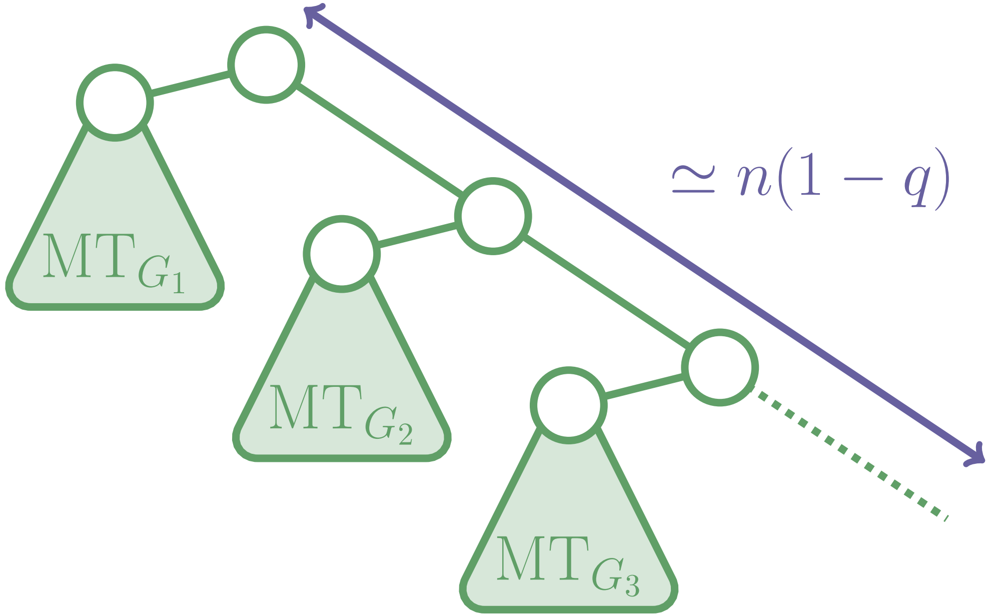
Mallows tree:



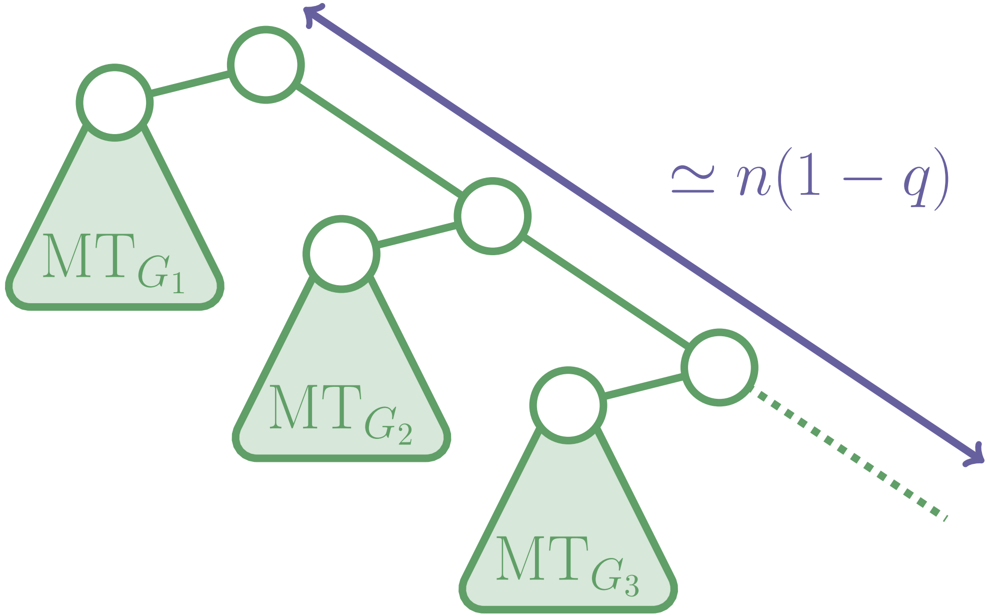
Local limit:

Guessing the local limit

Mallows tree:

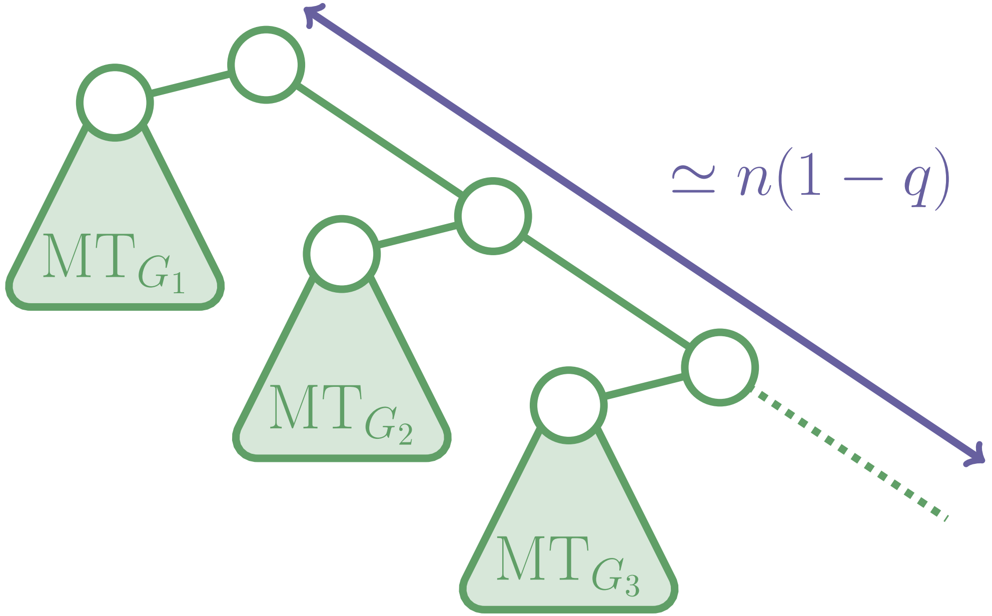


Local limit:

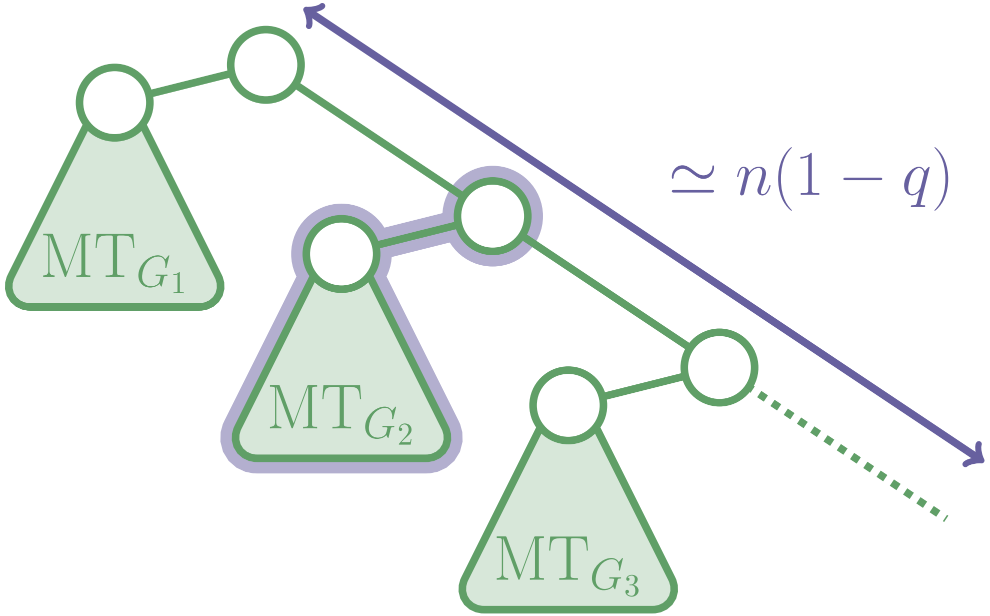


Guessing the local limit

Mallows tree:

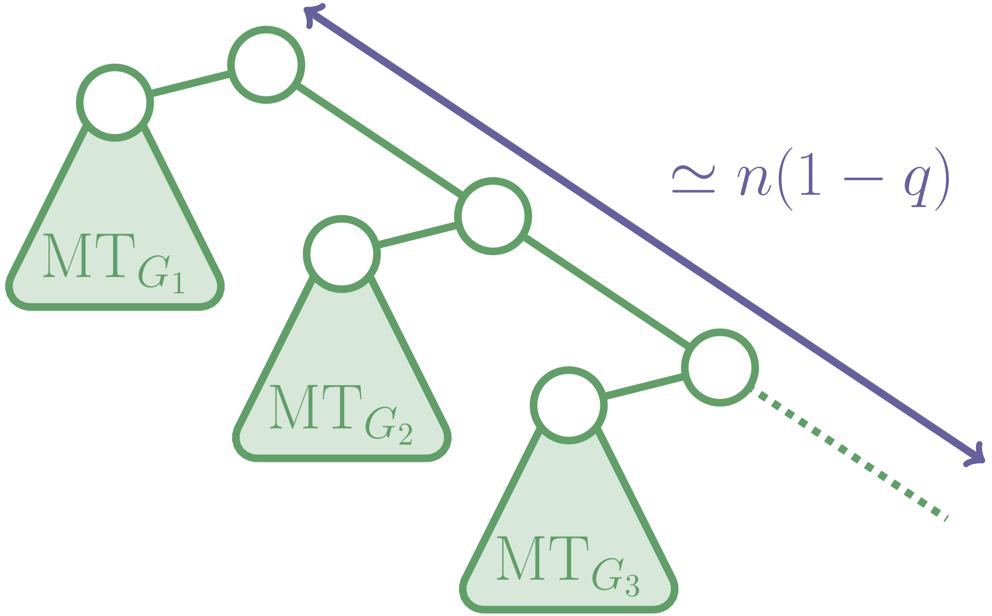


Local limit:

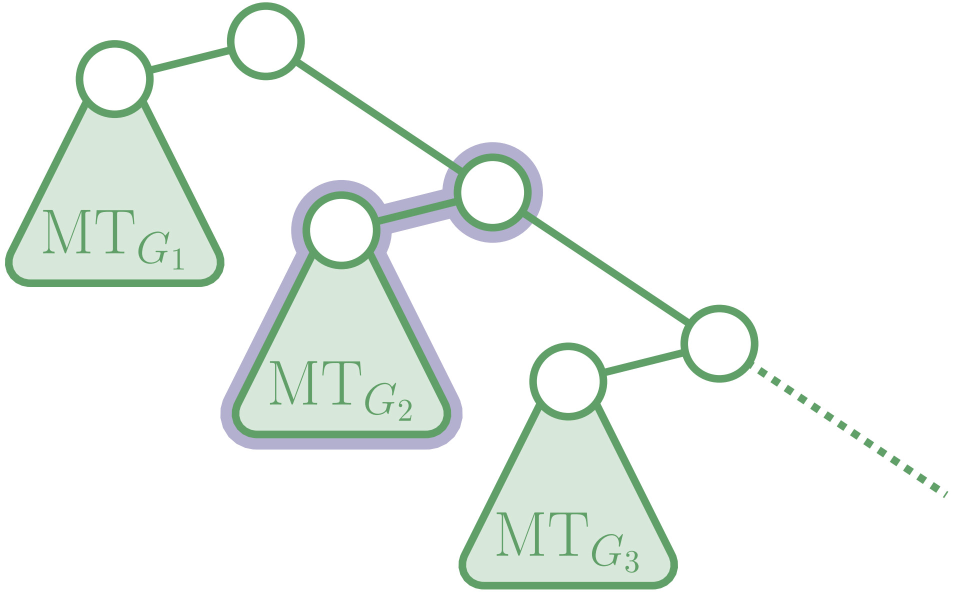


Guessing the local limit

Mallows tree:

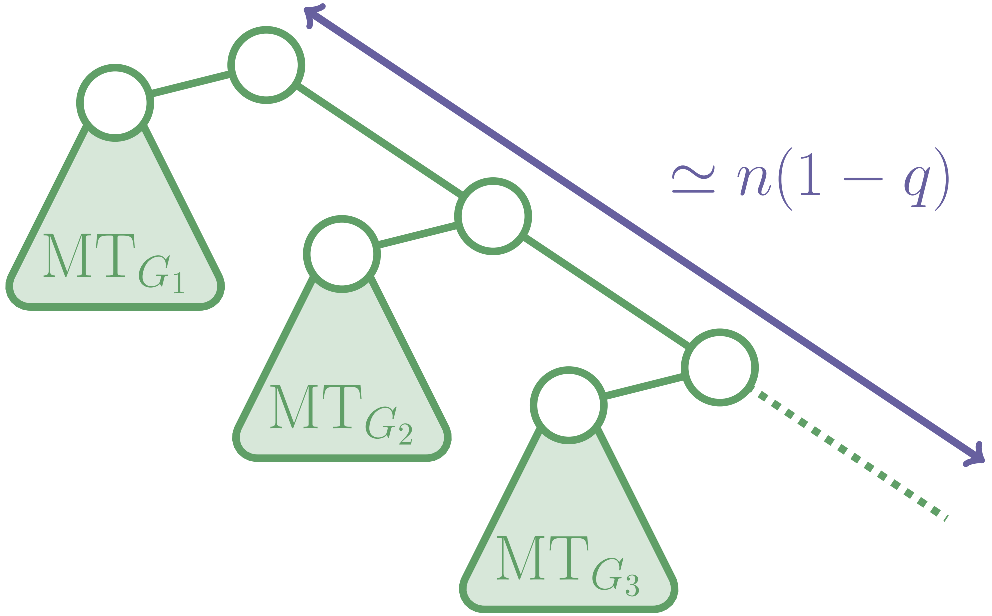


Local limit:

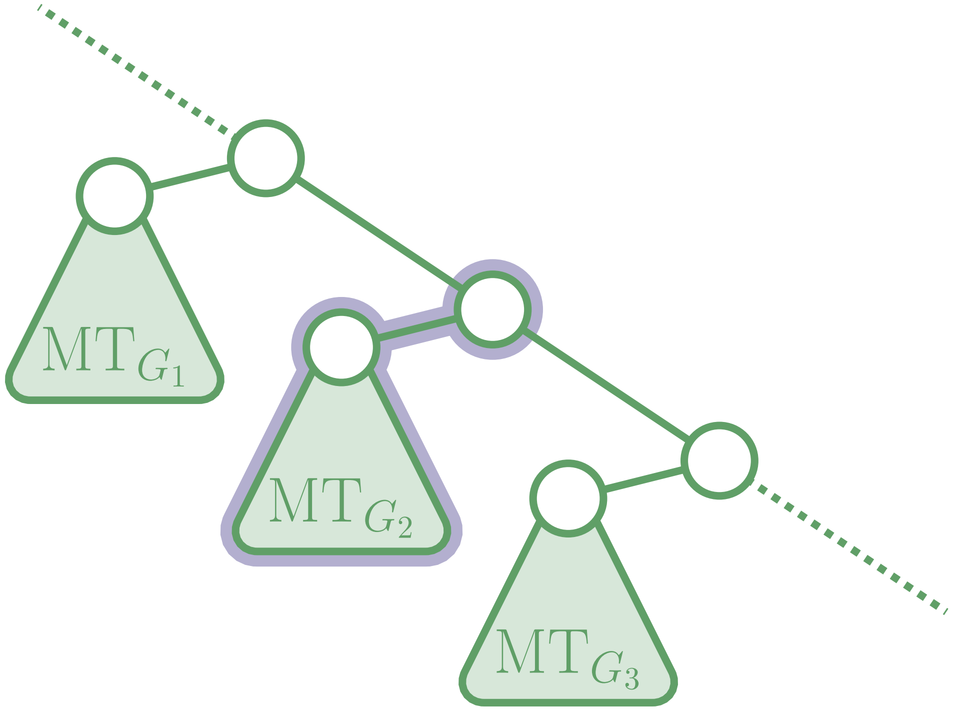


Guessing the local limit

Mallows tree:

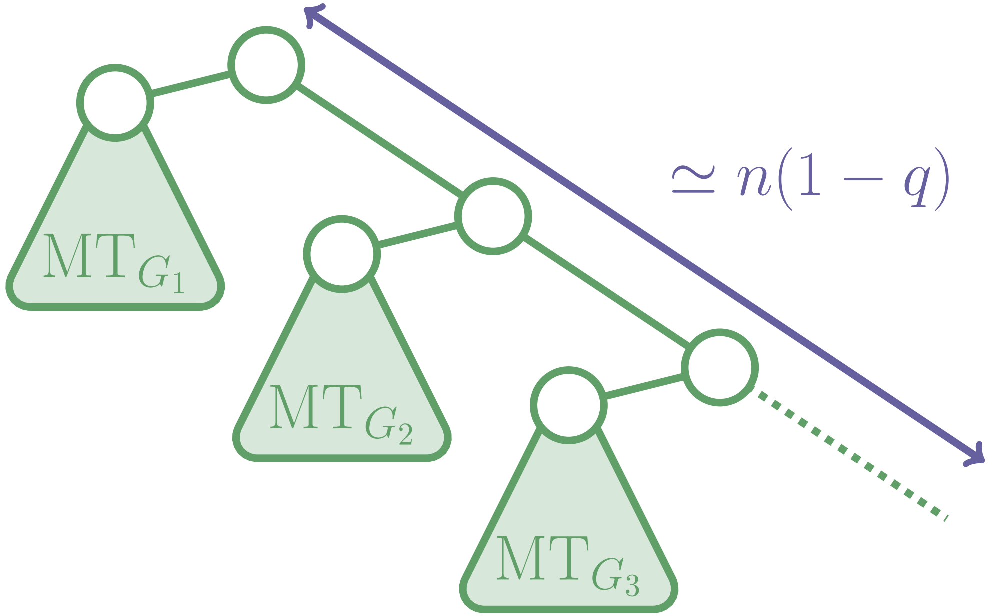


Local limit:

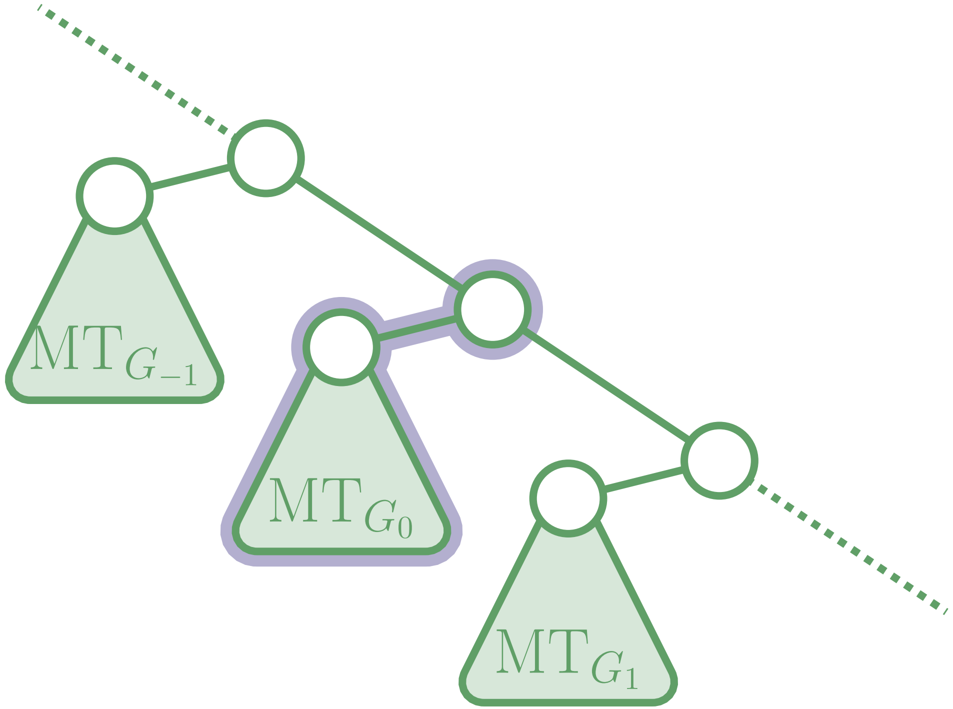


Guessing the local limit

Mallows tree:

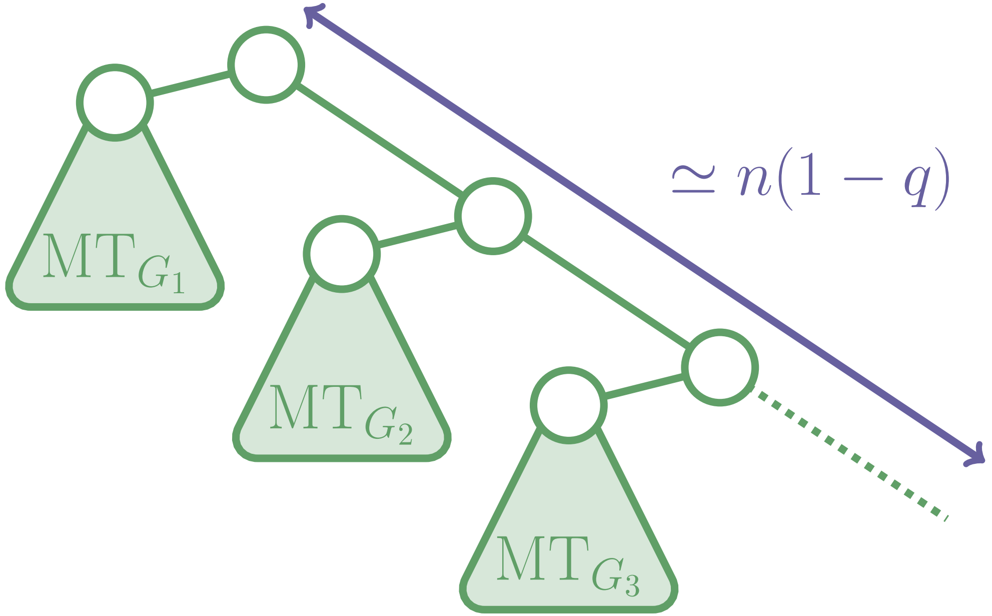


Local limit:

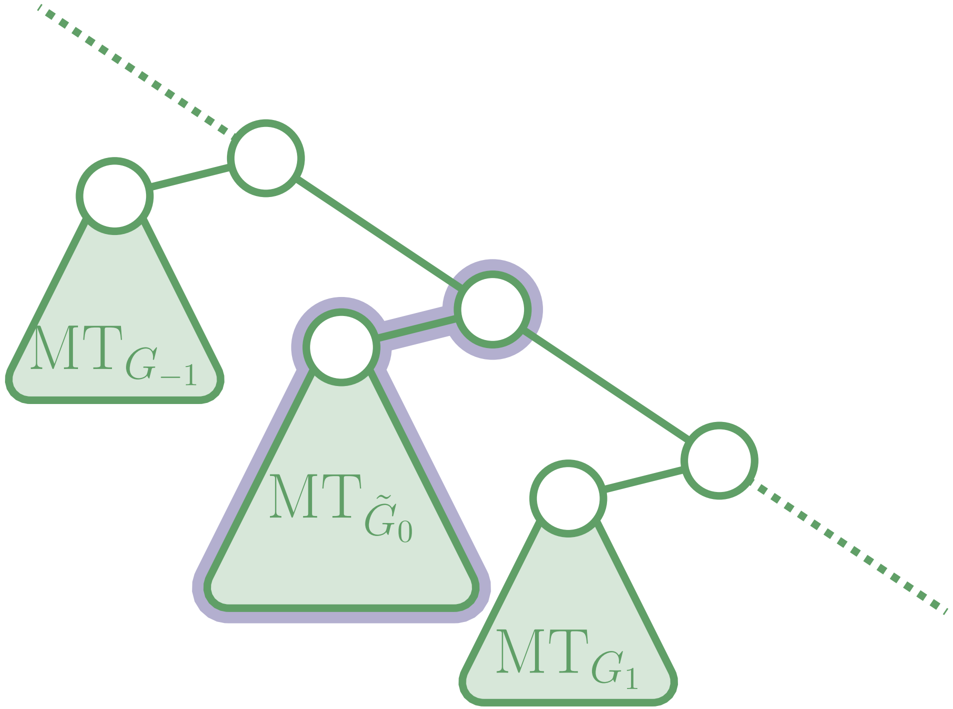


Guessing the local limit

Mallows tree:



Local limit:



Local limit of a Mallows tree

Local limit of a Mallows tree

Theorem (2023)

Theorem (🧒 2023)

Any sequence of Mallows trees with increasing size and fixed $q \in [0,1)$ converges locally almost-surely to $(o, \mathcal{T})$ constructed as follows.

Theorem (🧒 2023)

Any sequence of Mallows trees with increasing size and fixed $q \in [0,1)$ converges locally almost-surely to $(o, \mathcal{T})$ constructed as follows.

- Let $(G_i)_{i\in\mathbb{Z}}$ be a sequence of independent random variables, all having the geometric distribution with parameter q, except for G_0 being size-biased (note that $\mathbb{P}(G_i = 0) > 0$ for all $i \in \mathbb{Z}$).

Theorem (🧒 2023)

Any sequence of Mallows trees with increasing size and fixed $q \in [0,1)$ converges locally almost-surely to $(o, \mathcal{T})$ constructed as follows.

- Let $(G_i)_{i\in\mathbb{Z}}$ be a sequence of independent random variables, all having the geometric distribution with parameter q, except for G_0 being size-biased (note that $\mathbb{P}(G_i = 0) > 0$ for all $i \in \mathbb{Z}$).
- Let $(T_i)_{i\in\mathbb{Z}}$ be a sequence of Mallows trees with respective parameters G_i and q.

Theorem (🧒 2023)

Any sequence of Mallows trees with increasing size and fixed $q \in [0,1)$ converges locally almost-surely to $(o, \mathcal{T})$ constructed as follows.

- Let $(G_i)_{i\in\mathbb{Z}}$ be a sequence of independent random variables, all having the geometric distribution with parameter q, except for G_0 being size-biased (note that $\mathbb{P}(G_i = 0) > 0$ for all $i \in \mathbb{Z}$).
- Let $(T_i)_{i\in\mathbb{Z}}$ be a sequence of Mallows trees with respective parameters G_i and q.
- Set $\mathcal{T}$ to be the infinite line on $\mathbb{Z}$ with the root of each T_i attached via an edge to i.

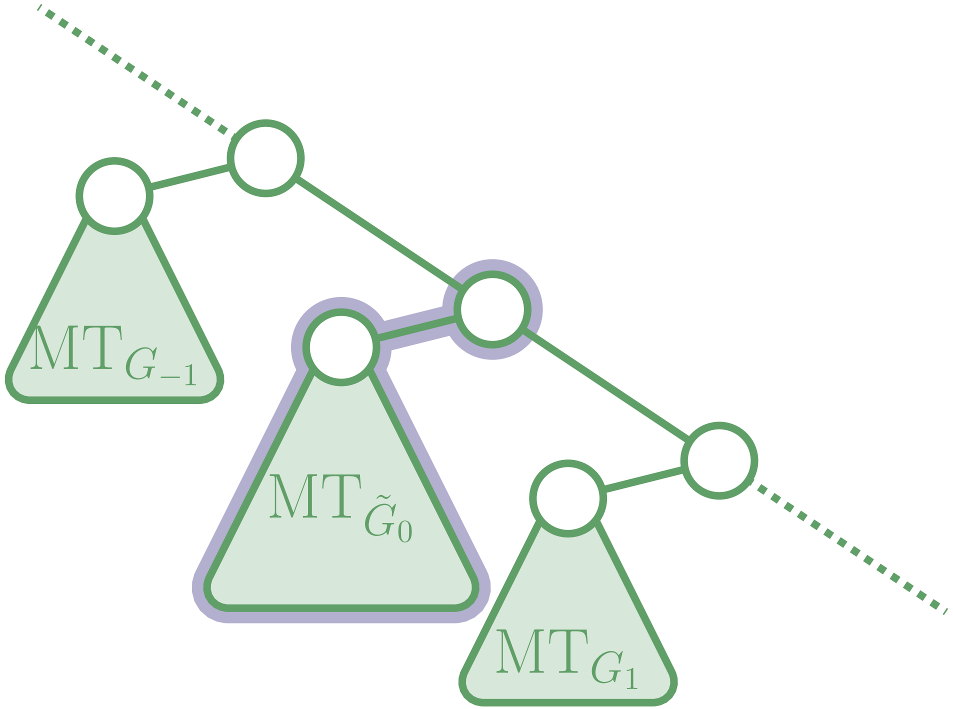
Theorem (🧒 2023)

Any sequence of Mallows trees with increasing size and fixed $q \in [0,1)$ converges locally almost-surely to $(o, \mathcal{T})$ constructed as follows.

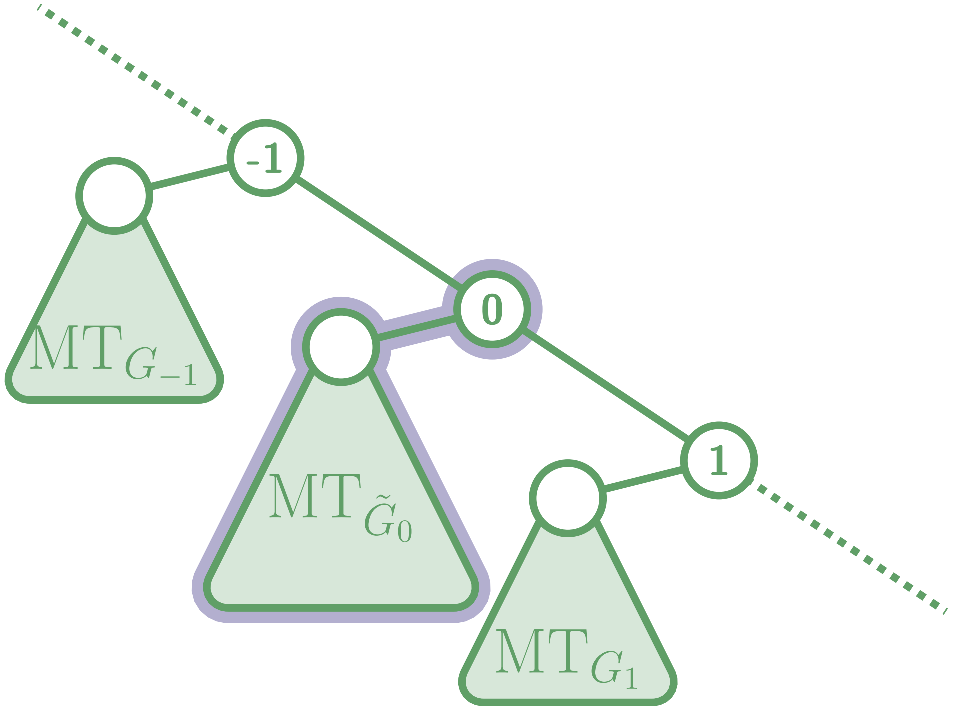
- Let $(G_i)_{i\in\mathbb{Z}}$ be a sequence of independent random variables, all having the geometric distribution with parameter q, except for G_0 being size-biased (note that $\mathbb{P}(G_i = 0) > 0$ for all $i \in \mathbb{Z}$).
- Let $(T_i)_{i\in\mathbb{Z}}$ be a sequence of Mallows trees with respective parameters G_i and q.
- Set $\mathcal{T}$ to be the infinite line on $\mathbb{Z}$ with the root of each T_i attached via an edge to i.
- Choose o uniformly over $\{0\} \cup V(T_0)$.

Local limit of a Mallows tree

Local limit of a Mallows tree



Local limit of a Mallows tree



Local limit of a Mallows tree

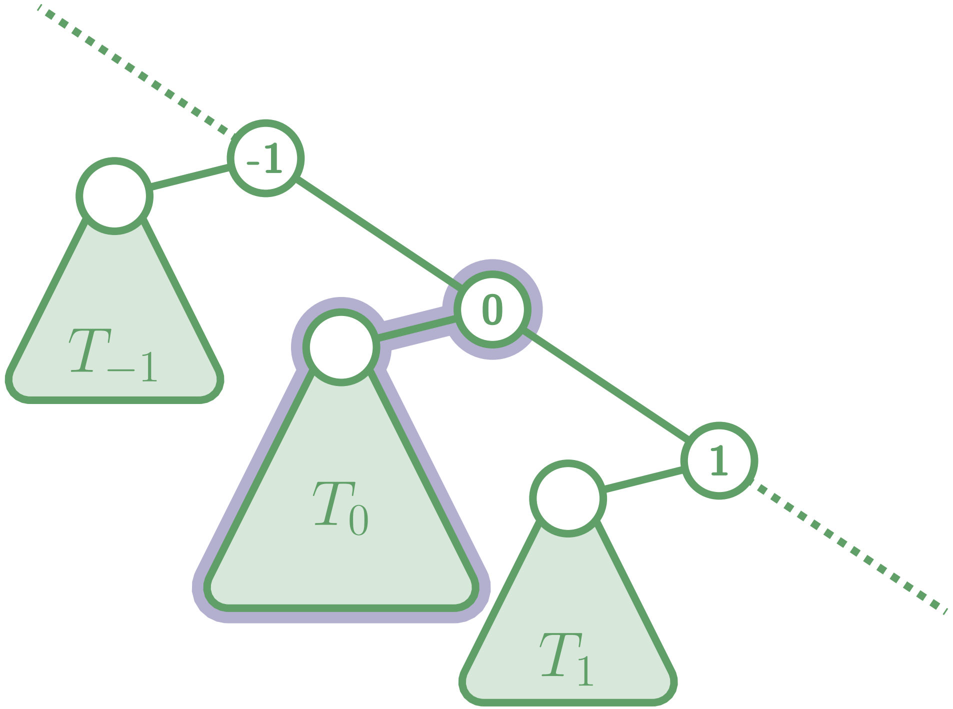


Table of Content

Summary

Summary

Here is what we did so far:

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.
- We took the local limit of this tree as $n \to \infty$.

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.
- We took the local limit of this tree as $n \to \infty$.

$\rightarrow$ The local limit is composed of many sub-structures distributed as Mallows trees.

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.
- We took the local limit of this tree as $n \to \infty$.

$\rightarrow$ The local limit is composed of many sub-structures distributed as Mallows trees.

$\rightarrow$ These substructures were already present when considering finite Mallows permutations.

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.
- We took the local limit of this tree as $n \to \infty$.

→ The local limit is composed of many sub-structures distributed as Mallows trees.

→ These substructures were already present when considering finite Mallows permutations.

Q: Can we swap the local limit and the binary search tree structure to first construct an infinite Mallows permutation which we then insert into a binary search tree?

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.
- We took the local limit of this tree as $n \to \infty$.

→ The local limit is composed of many sub-structures distributed as Mallows trees.

→ These substructures were already present when considering finite Mallows permutations.

Q: Can we swap the local limit and the binary search tree structure to first construct an infinite Mallows permutation which we then insert into a binary search tree?

→ Infinite Mallows permutations are already defined, both on $\mathbb{N}$ and $\mathbb{Z}$!

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.
- We took the local limit of this tree as $n \to \infty$.

→ The local limit is composed of many sub-structures distributed as Mallows trees.

→ These substructures were already present when considering finite Mallows permutations.

Q: Can we swap the local limit and the binary search tree structure to first construct an infinite Mallows permutation which we then insert into a binary search tree?

→ Infinite Mallows permutations are already defined, both on $\mathbb{N}$ and $\mathbb{Z}$!
So we are good, **right**?

Summary

Here is what we did so far:

- We took a random Mallows permutation $X_{n,q}$ of size n and parameter q.
- We inserted this permutation into a binary search tree structure to obtain a Mallows tree.
- We took the local limit of this tree as $n \to \infty$.

→ The local limit is composed of many sub-structures distributed as Mallows trees.

→ These substructures were already present when considering finite Mallows permutations.

Q: Can we swap the local limit and the binary search tree structure to first construct an infinite Mallows permutation which we then insert into a binary search tree?

→ Infinite Mallows permutations are already defined, both on $\mathbb{N}$ and $\mathbb{Z}$!
So we are good, **right**? Well, **not exactly**...

The problem

The problem

The natural “local limit” of a Mallows permutation in that setting is the two-sided infinite Mallows permutation, defined on $\mathbb{Z}$.

The problem

The natural “local limit” of a Mallows permutation in that setting is the two-sided infinite Mallows permutation, defined on $\mathbb{Z}$.

- Such a permutation can be seen as a **two-sided** sequence of distinct and signed integers.

The problem

The natural "local limit" of a Mallows permutation in that setting is the two-sided infinite Mallows permutation, defined on $\mathbb{Z}$.

- Such a permutation can be seen as a **two-sided** sequence of distinct and signed integers.
- Binary search trees are only defined on **one-sided** sequences of distinct integers.

The problem

The natural "local limit" of a Mallows permutation in that setting is the two-sided infinite Mallows permutation, defined on $\mathbb{Z}$.

- Such a permutation can be seen as a **two-sided** sequence of distinct and signed integers.
- Binary search trees are only defined on **one-sided** sequences of distinct integers.

→ We need to extend binary search trees to two-sided sequences $x = (\ldots, x_{-1}, x_0, x_1, \ldots)$.

The problem

The natural "local limit" of a Mallows permutation in that setting is the two-sided infinite Mallows permutation, defined on $\mathbb{Z}$.

- Such a permutation can be seen as a **two-sided** sequence of distinct and signed integers.
- Binary search trees are only defined on **one-sided** sequences of distinct integers.

→ We need to extend binary search trees to two-sided sequences $x = (\ldots, x_{-1}, x_0, x_1, \ldots)$.

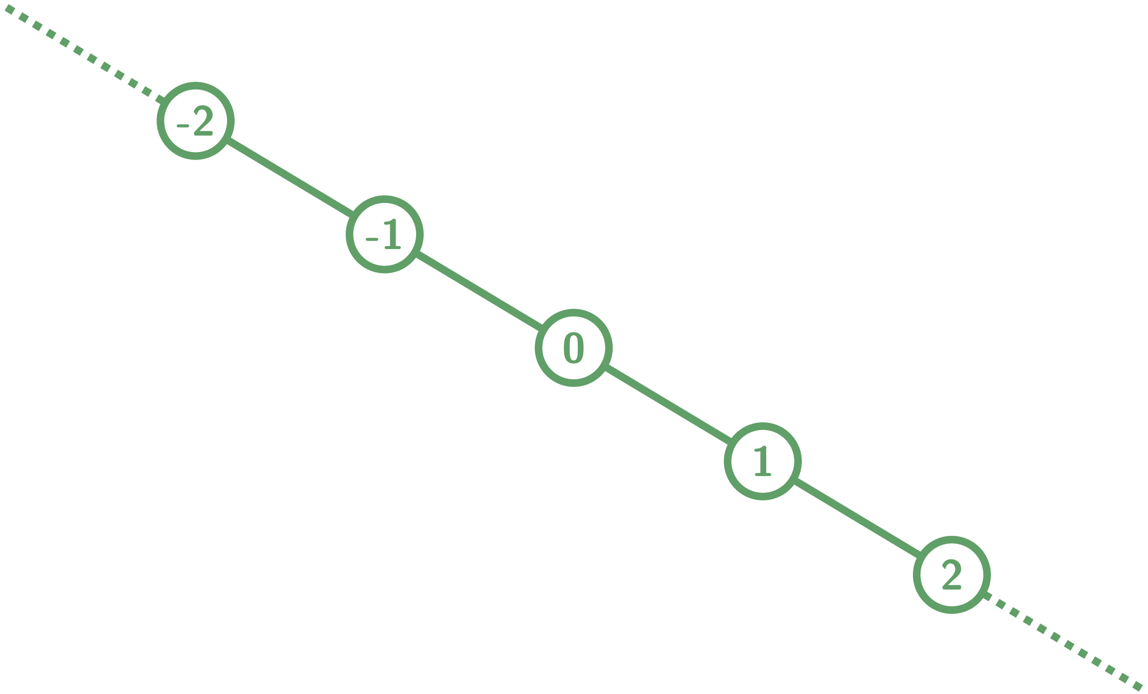
→ Let us try some cases to understand how they work.

Examples

Examples

What is the tree for $x = (\ldots, -2 - 1, 0, 1, 2, \ldots)$?

What is the tree for $x = (\ldots, -2 - 1, 0, 1, 2, \ldots)$?



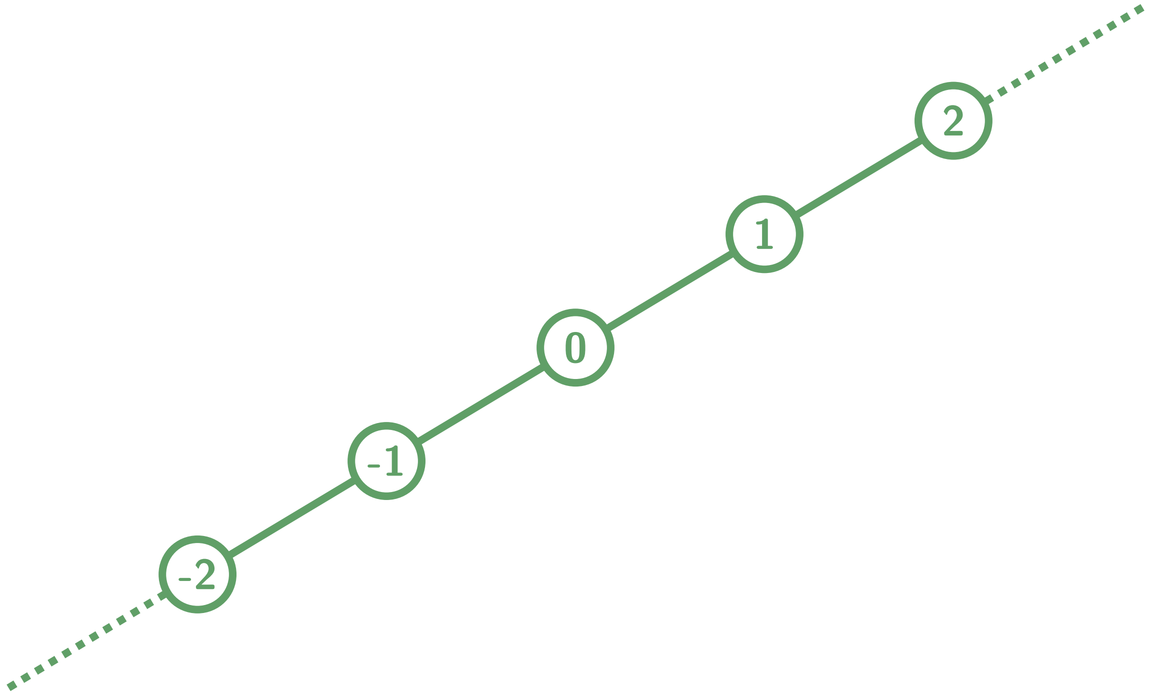
Examples

Examples

What is the tree for $x = (\ldots, 2, 1, 0, -1, -2, \ldots)$?

Examples

What is the tree for $x = (\ldots, 2, 1, 0, -1, -2, \ldots)$?

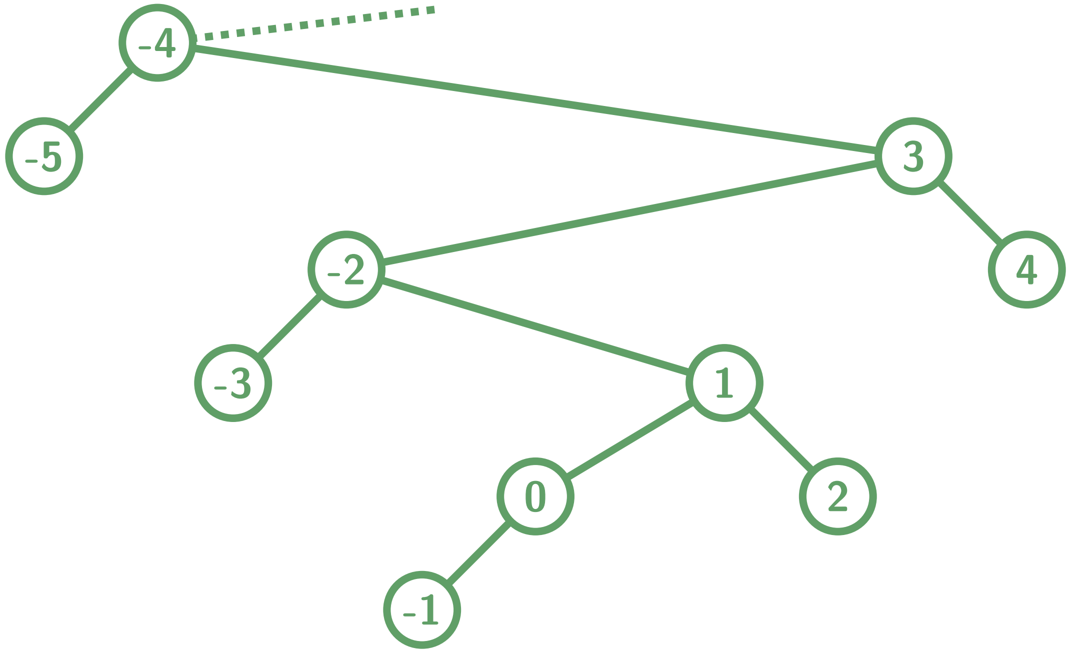


Examples

Examples

What is the tree for $x = (\ldots, 3, -2, 1, 0, -1, 2, -3, 4, -5, 6, \ldots)$?

What is the tree for $x = (\ldots, 3, -2, 1, 0, -1, 2, -3, 4, -5, 6, \ldots)$?



Generalizing binary search trees

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

- Infinite rightward path (from top-left to bottom-right).

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

- Infinite rightward path (from top-left to bottom-right).
- Infinite leftward path (from top-right to bottom-left).

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

- Infinite rightward path (from top-left to bottom-right).
- Infinite leftward path (from top-right to bottom-left).
- Zigzagging tree.

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

- Infinite rightward path (from top-left to bottom-right).
- Infinite leftward path (from top-right to bottom-left).
- Zigzagging tree.

$\rightarrow$ This makes the formal definition of "general binary search trees" more difficult.

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

- Infinite rightward path (from top-left to bottom-right).
- Infinite leftward path (from top-right to bottom-left).
- Zigzagging tree.

→ This makes the formal definition of "general binary search trees" more difficult.

Luckily for us, infinite Mallows permutations (with $q \in [0, 1)$) tend to be "ordered". In particular, they can always be put in a tree with a single infinite path to the right.

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

- Infinite rightward path (from top-left to bottom-right).
- Infinite leftward path (from top-right to bottom-left).
- Zigzagging tree.

→ This makes the formal definition of "general binary search trees" more difficult.

Luckily for us, infinite Mallows permutations (with $q \in [0, 1)$) tend to be "ordered". In particular, they can always be put in a tree with a single infinite path to the right.

→ I refer to these trees as *redwood trees*.

Generalizing binary search trees

As we can see from the previous three examples, different types of behaviour might arise:

- Infinite rightward path (from top-left to bottom-right).
- Infinite leftward path (from top-right to bottom-left).
- Zigzagging tree.

→ This makes the formal definition of “general binary search trees” more difficult.

Luckily for us, infinite Mallows permutations (with $q \in [0, 1)$) tend to be “ordered”. In particular, they can always be put in a tree with a single infinite path to the right.

→ I refer to these trees as *redwood trees*.

→ For more details, check out the paper or come talk to me. 👍

References

- Addario-Berry, L., & Corsini, B. (2021). **The height of Mallows trees.** *The Annals of Probability, 49(5), 2220-2271.*
- Corsini, B. (2023). **Limits of Mallows trees.** *arXiv preprint arXiv:2312.13817.*
- Evans, S. N., Grübel, R., & Wakolbinger, A. (2012). **Trickle-down processes and their boundaries.** *Electron. J. Probab, 17(1), 1-58.*
- Gnedin, A., & Olshanski, G. (2012). **The two-sided infinite extension of the Mallows model for random permutations.** *Advances in Applied Mathematics, 48(5), 615-639.*
- Mallows, C. L. (1957). **Non-null ranking models. I.** *Biometrika, 44(1/2), 114-130.*

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie Grant Agreement No. 101034253.

Thank you!

Thank you!
Thank you!
Thank you!