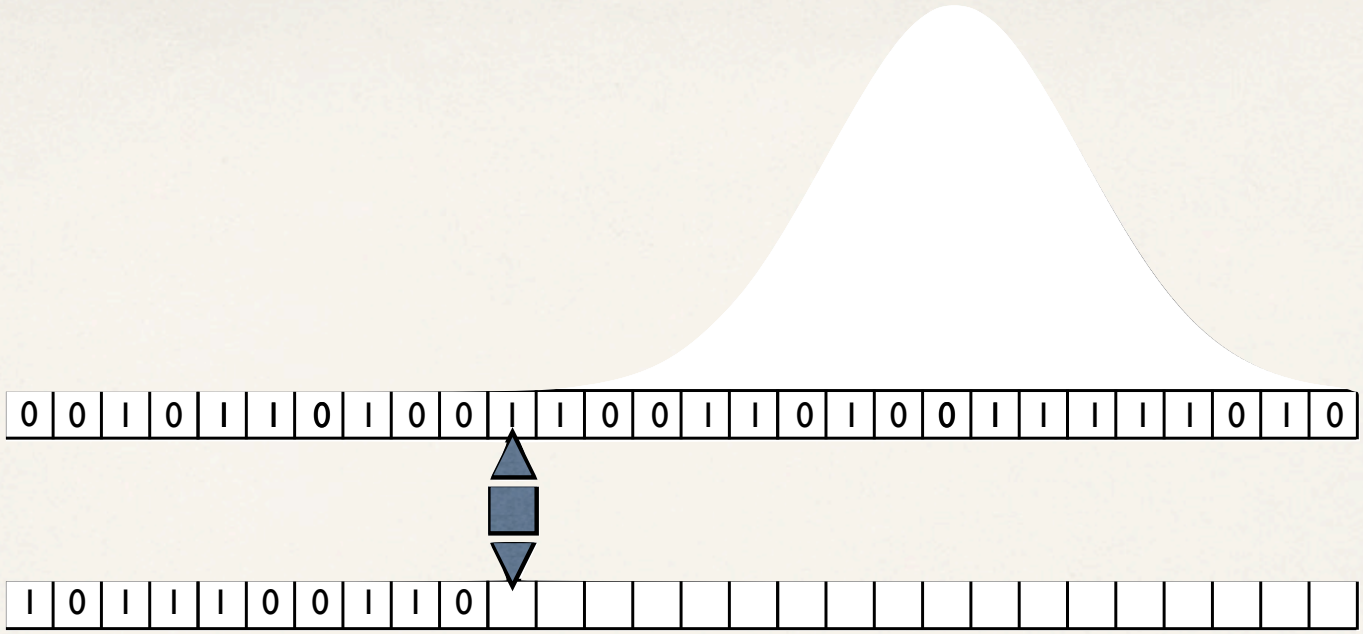


Randomness, Probability, and Computation

Christopher Porter, LIAFA, Université Paris 7

Joint Mathematics Meetings, Special Session On Logic And Probability *January 15, 2014*

Motivation

In computability theory, one major concern is to determine which problems are solvable by Turing machines and which ones are not.

- For a given set of natural numbers S, is there an effective procedure for determining membership in S?

If a given problem is shown to be effectively unsolvable, we can further ask: Just how unsolvable is it?

In fact, there are a number of hierarchies for classifying the difficulty of solving various problems.

An alternative approach

In these investigations of the solvability of problems, the computations are carried out by Turing machines (often equipped with an oracle).

What picture would emerge if instead we were to work with some model of probabilistic Turing machine?

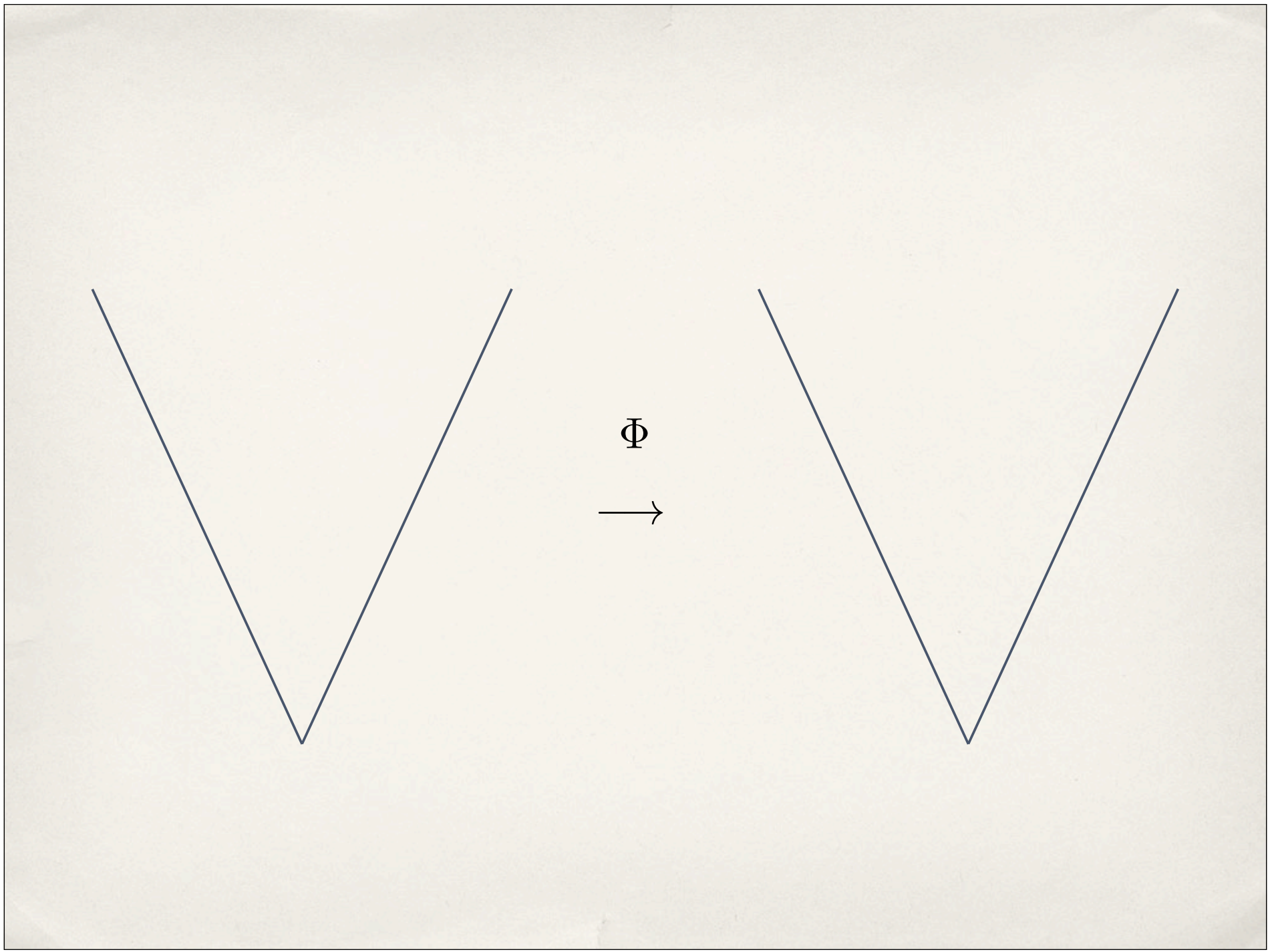
In this talk, I will discuss recent work with Laurent Bienvenu and Antoine Taveneaux on two kinds of effectively closed classes whose members are difficult to compute probabilistically, namely ***negligible*** classes and ***deep*** classes.

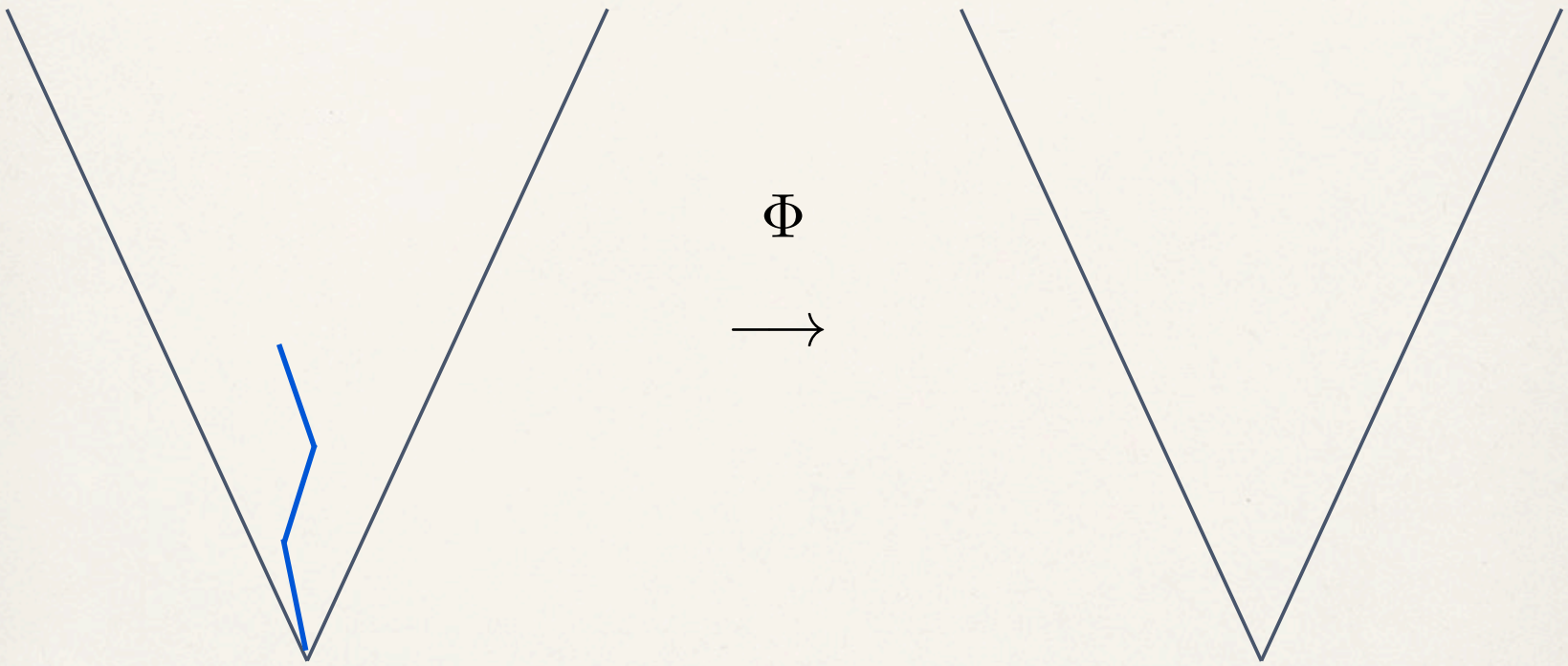
Turing computation

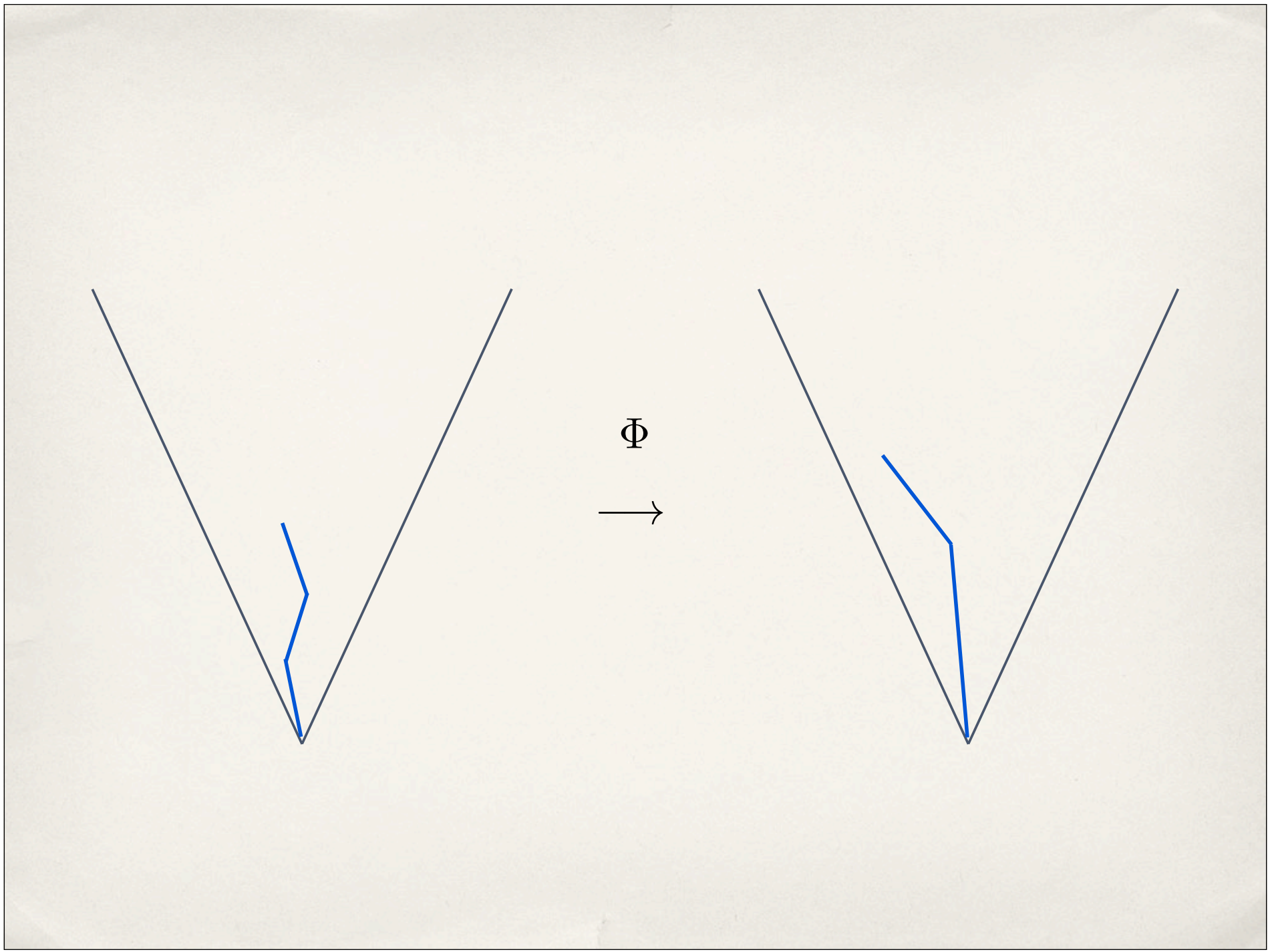
As a first step towards describing the model of probabilistic computation under consideration, we need to discuss ***Turing functionals.***

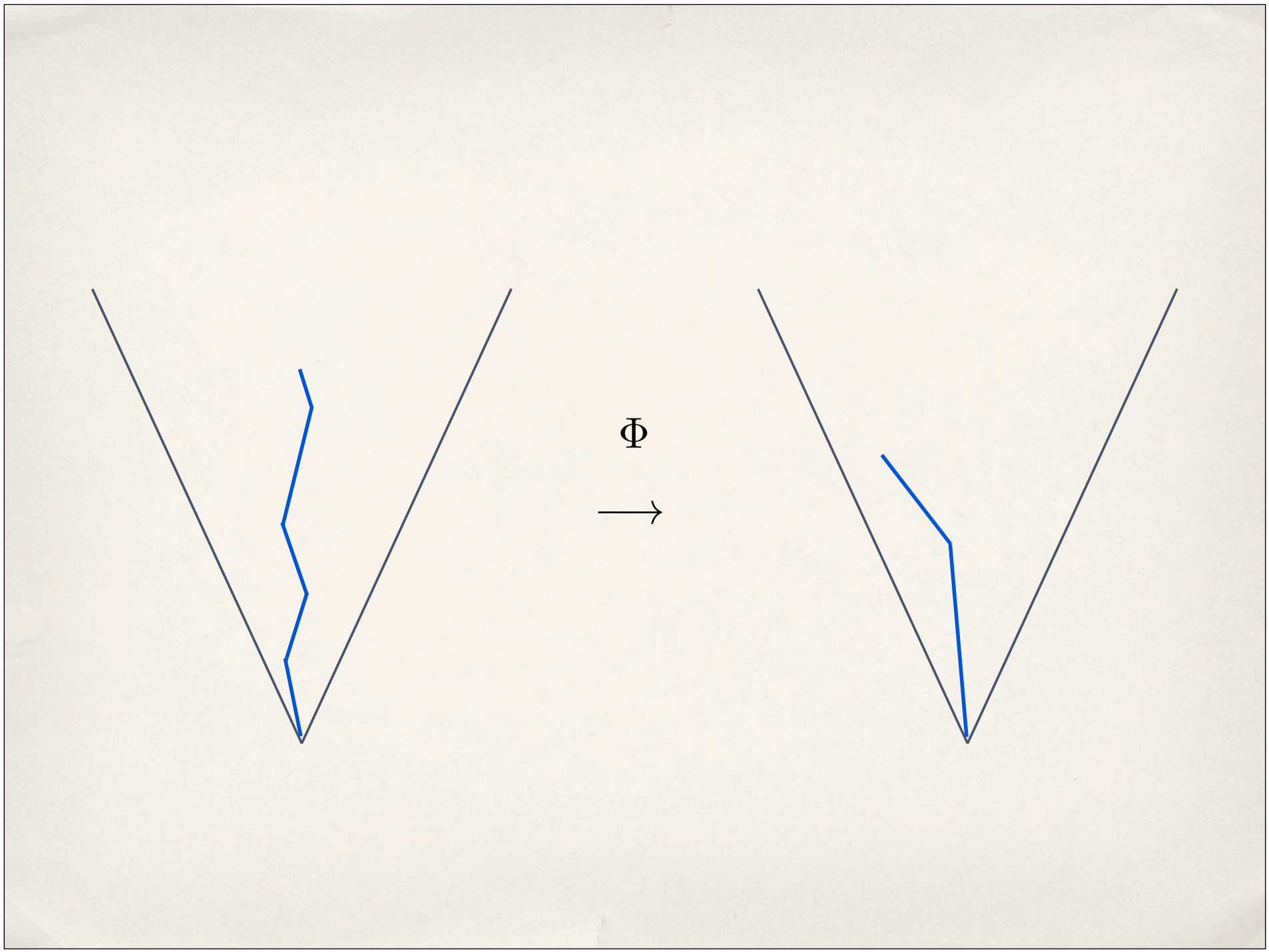
Here we view a Turing functional as an effective map from 2^ω to 2^ω.

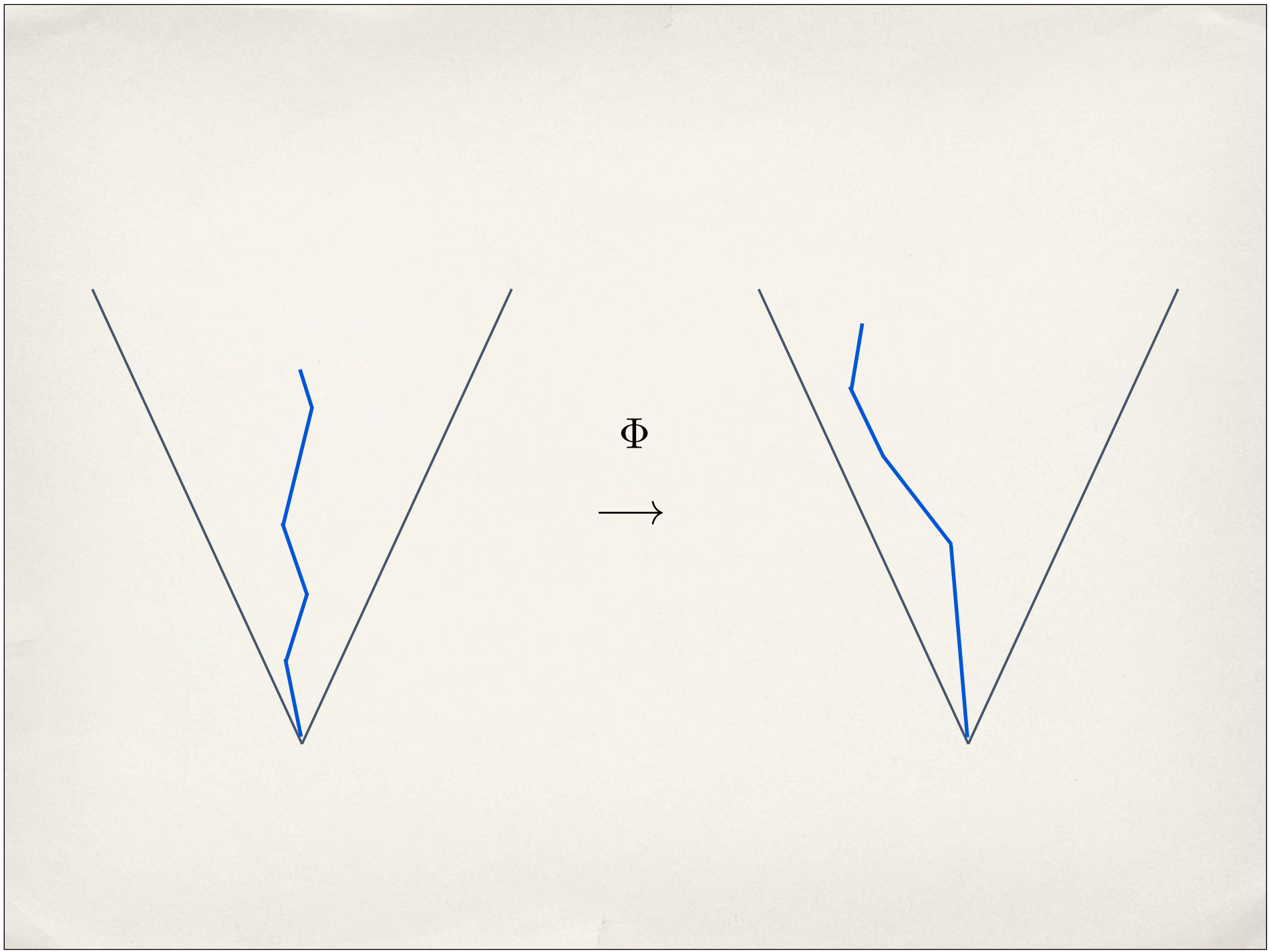
More precisely, a Turing functional is given in terms of a computably enumerable set of pairs of strings satisfying a certain consistency condition (as illustrated by the next set of slides).

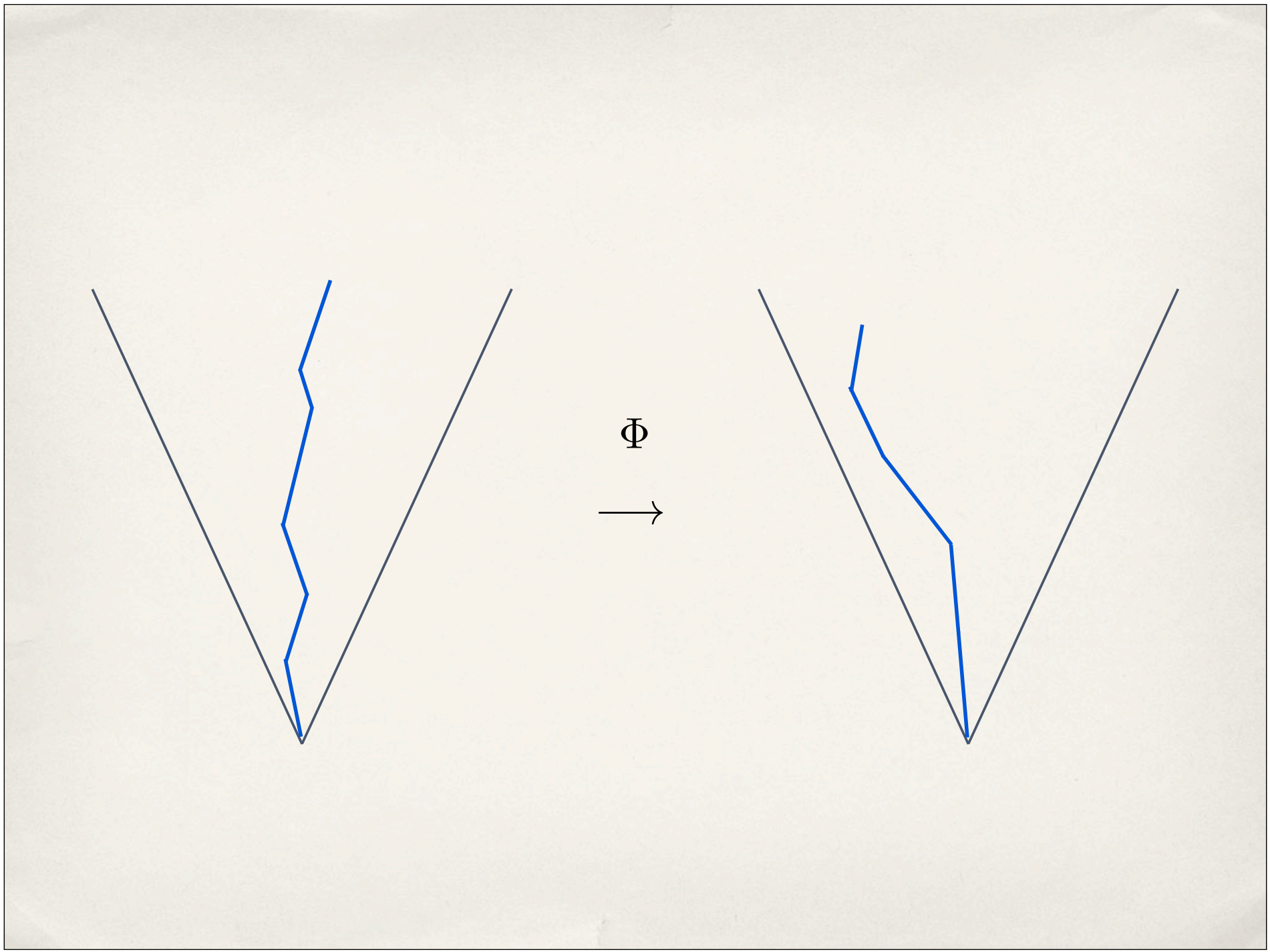

Φ

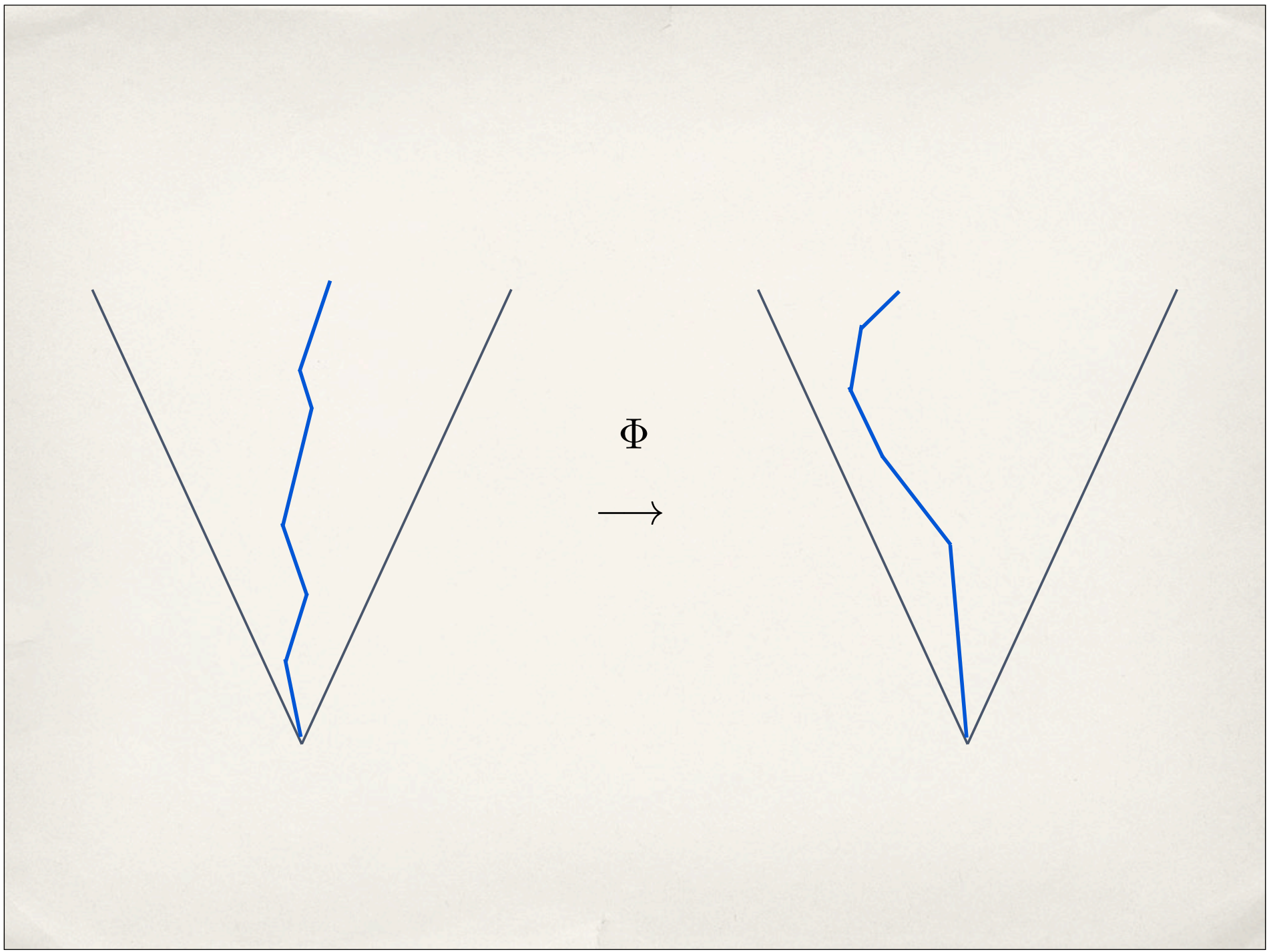

Φ

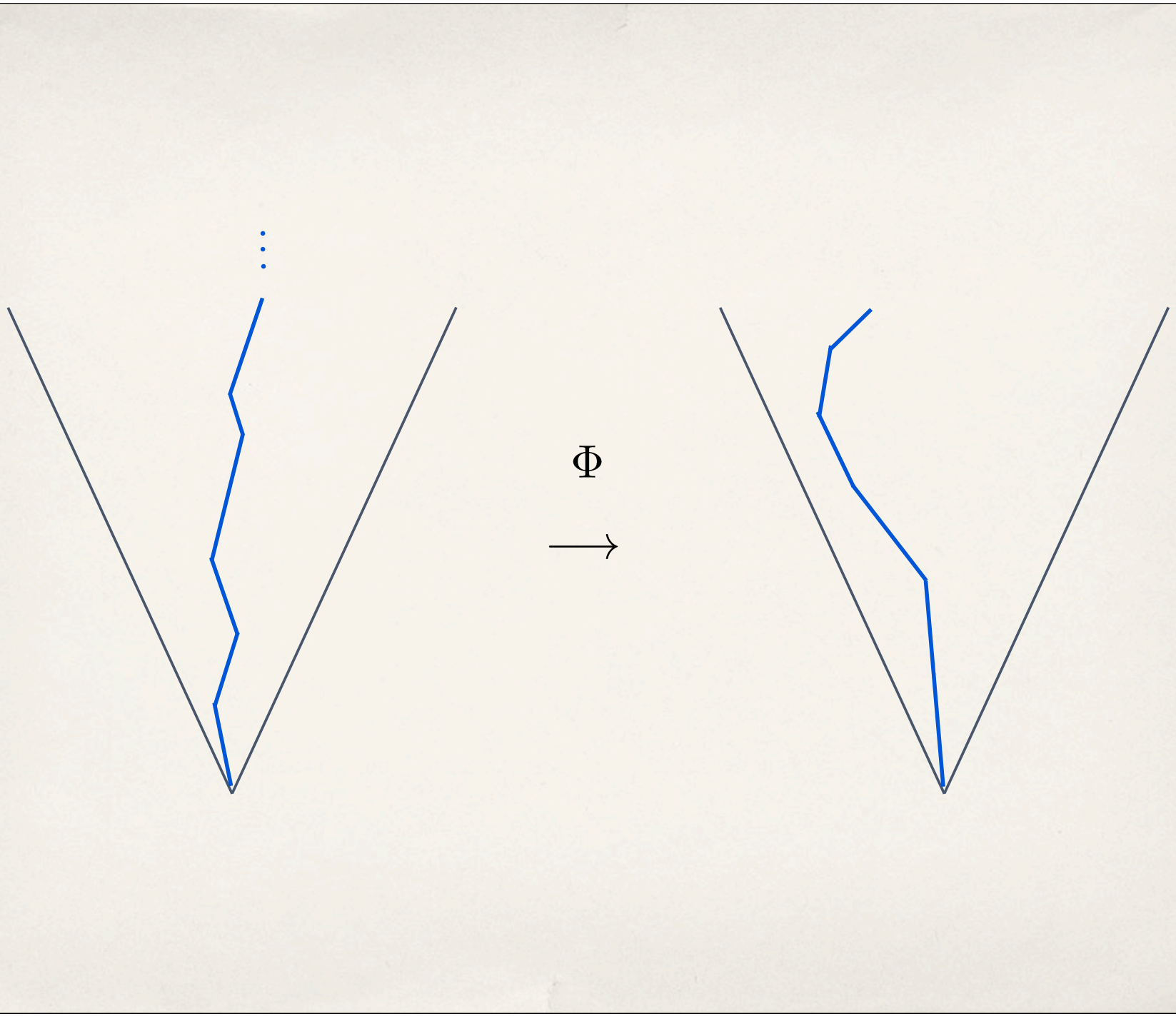

Φ

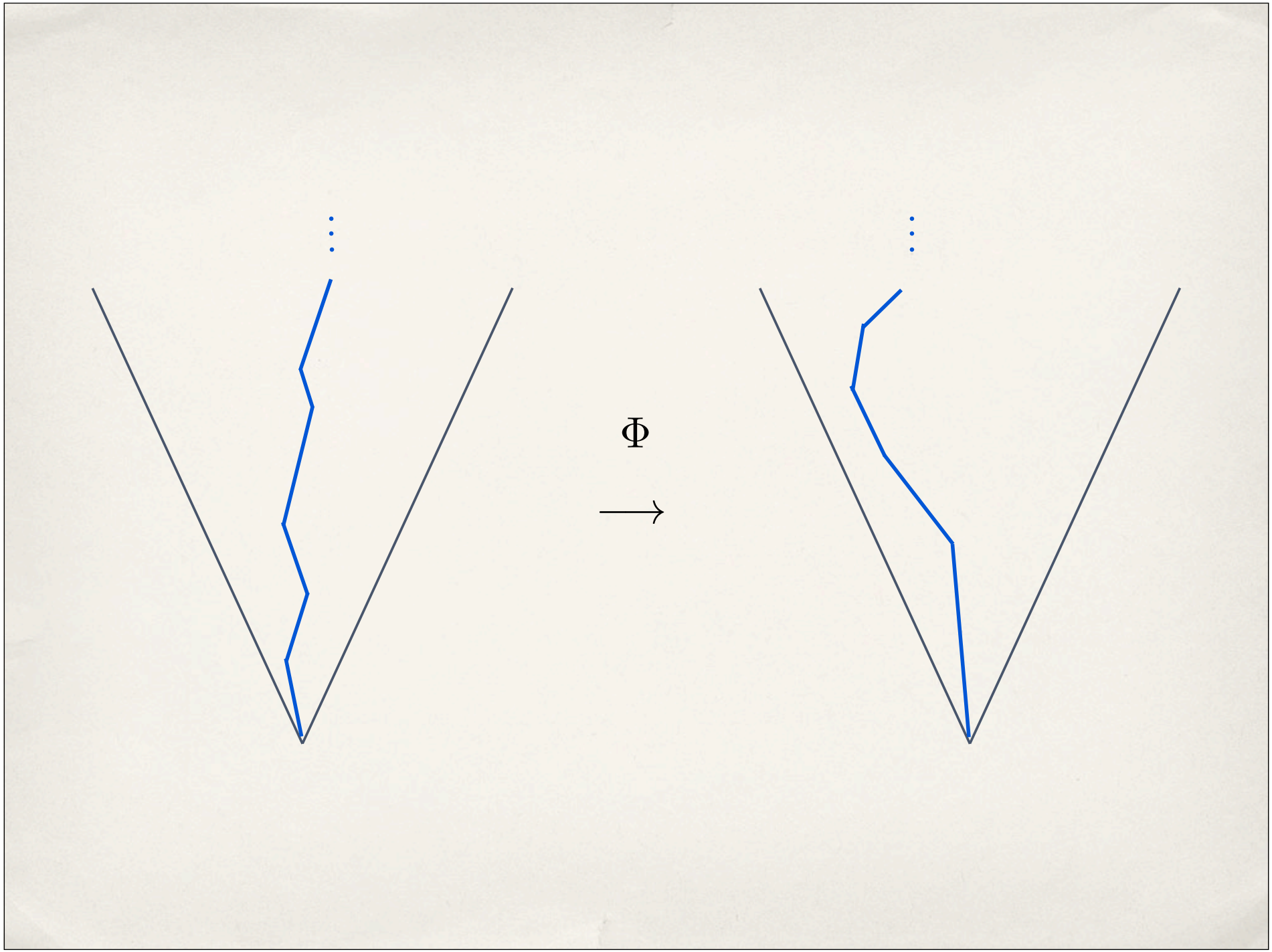

Φ

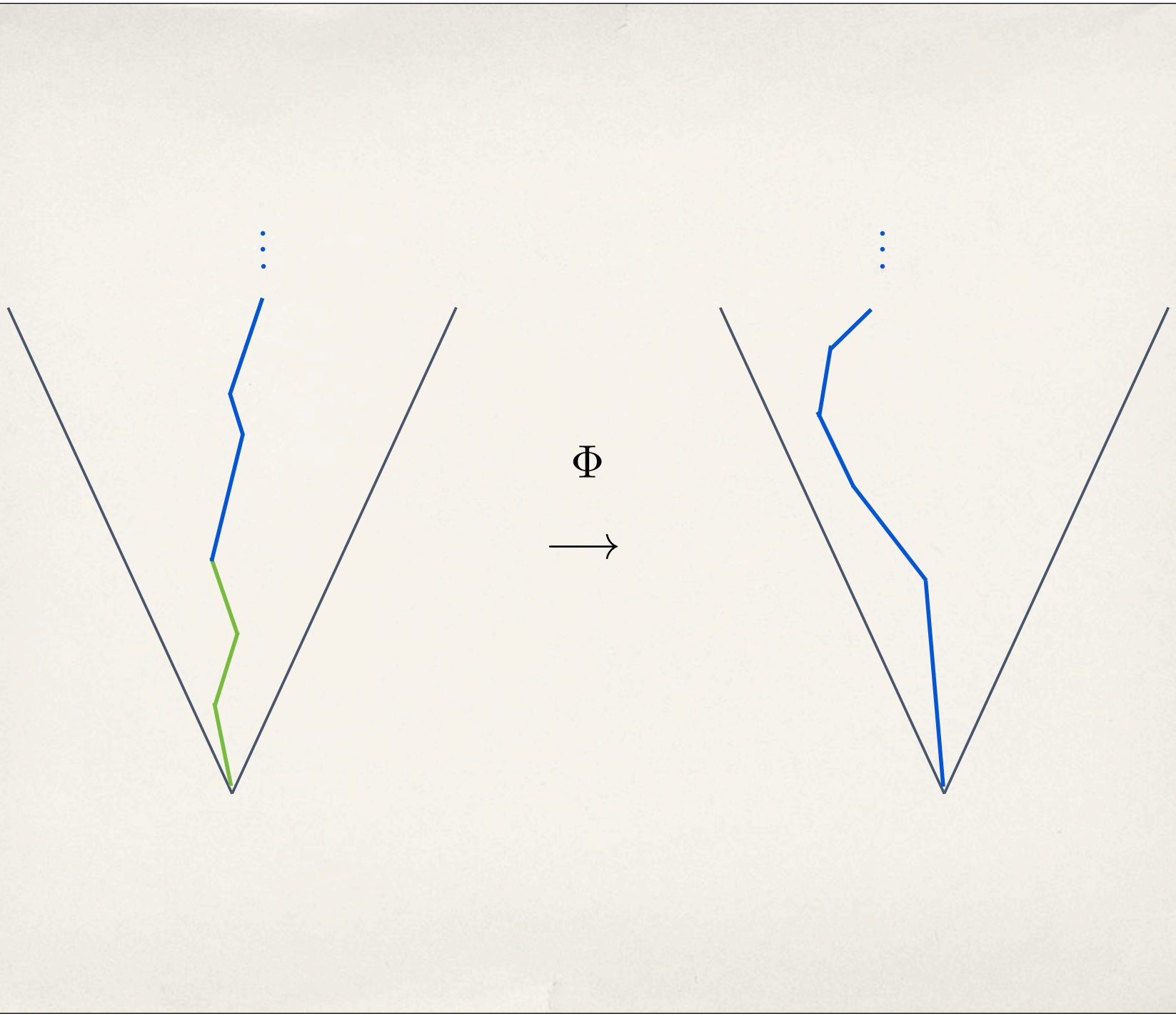

Φ

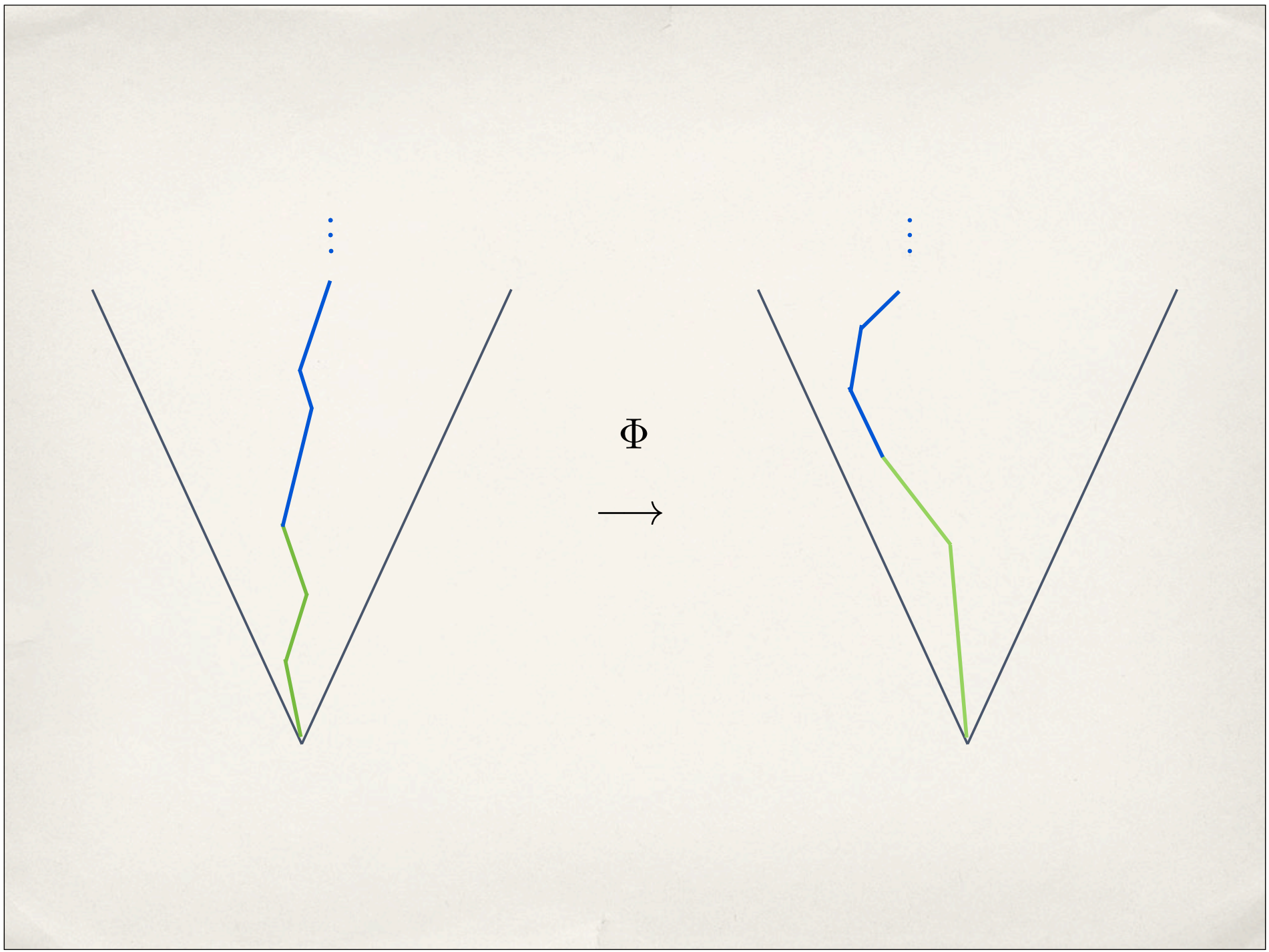

Φ

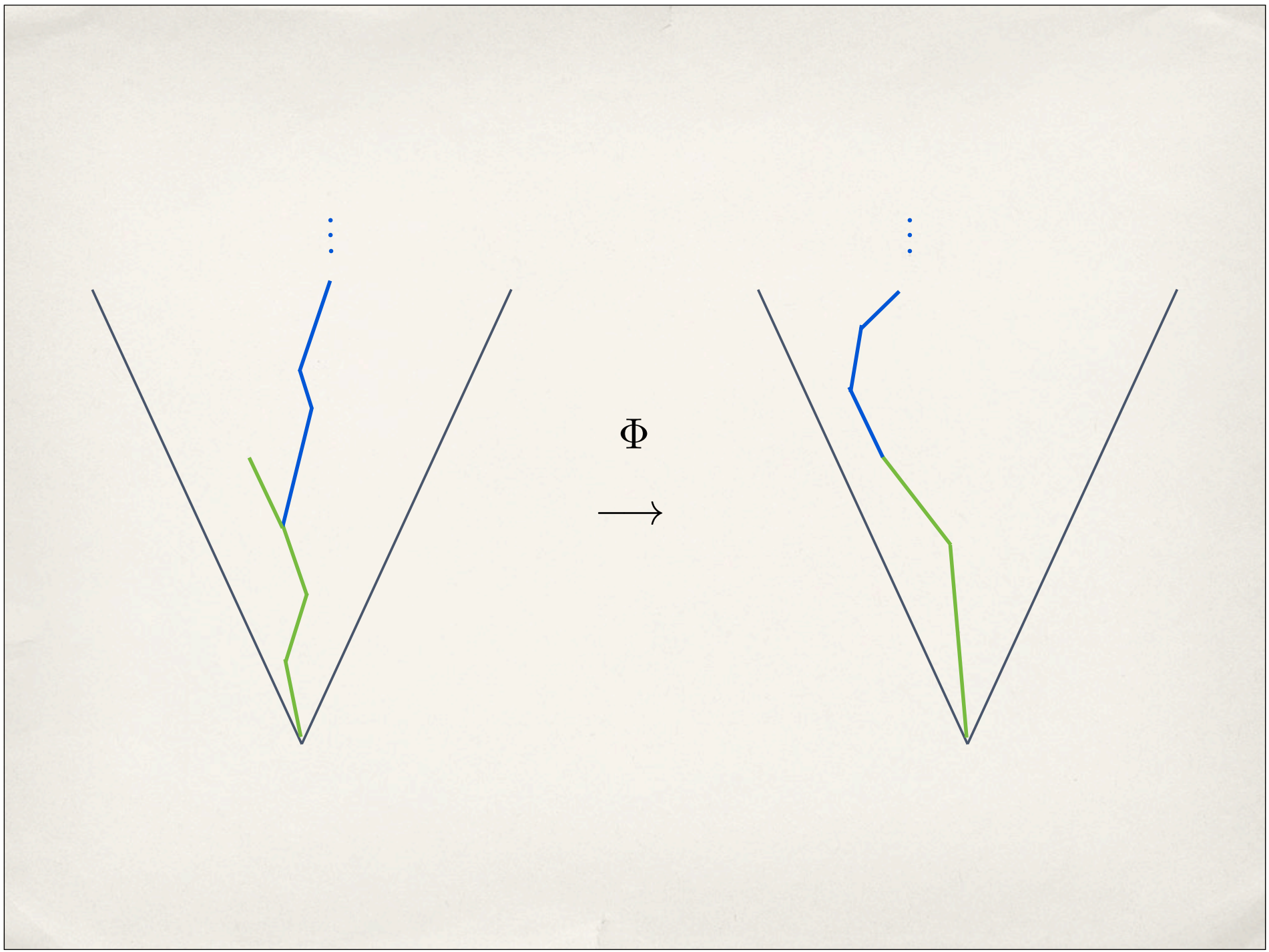

Φ

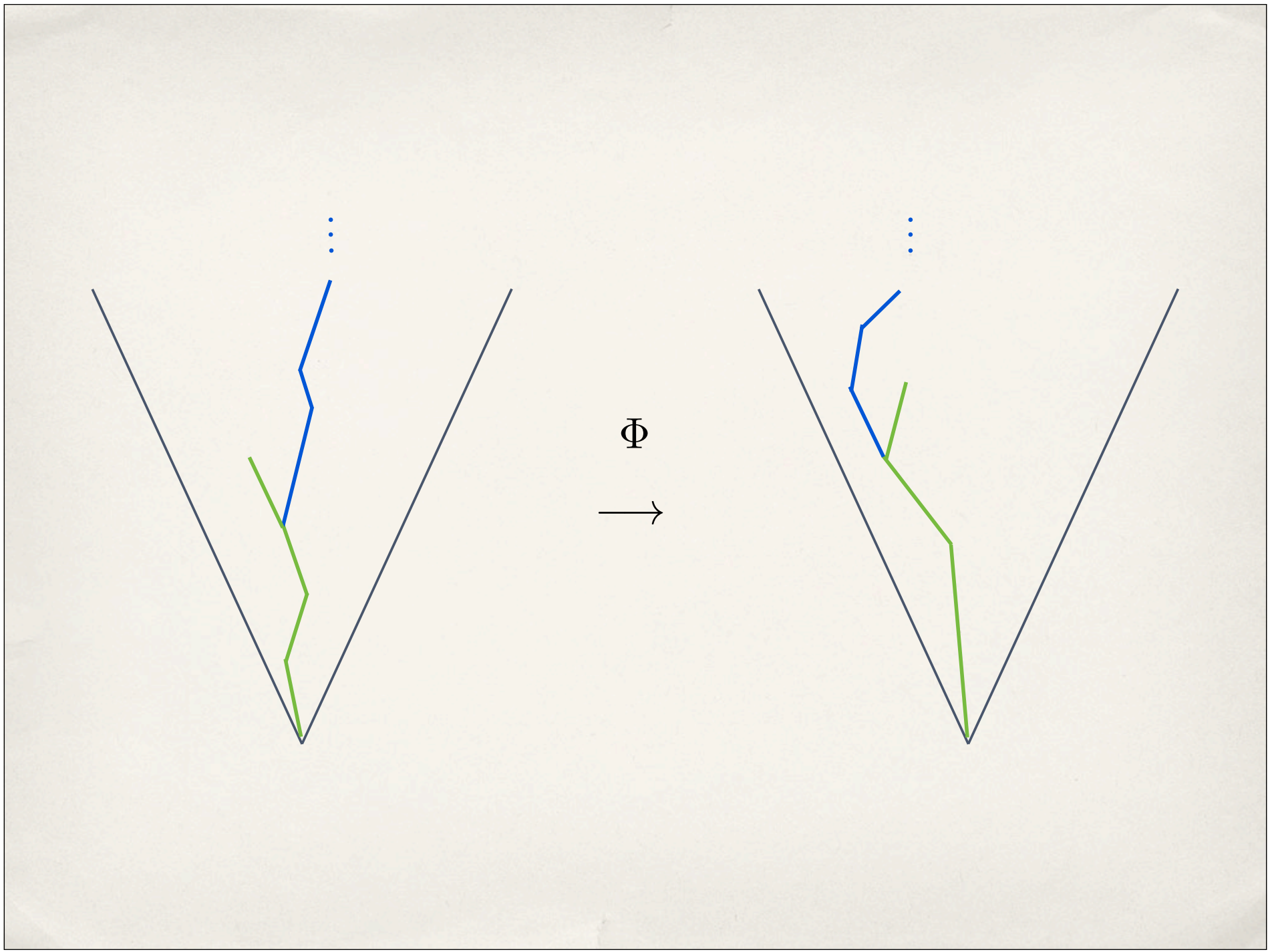

Φ

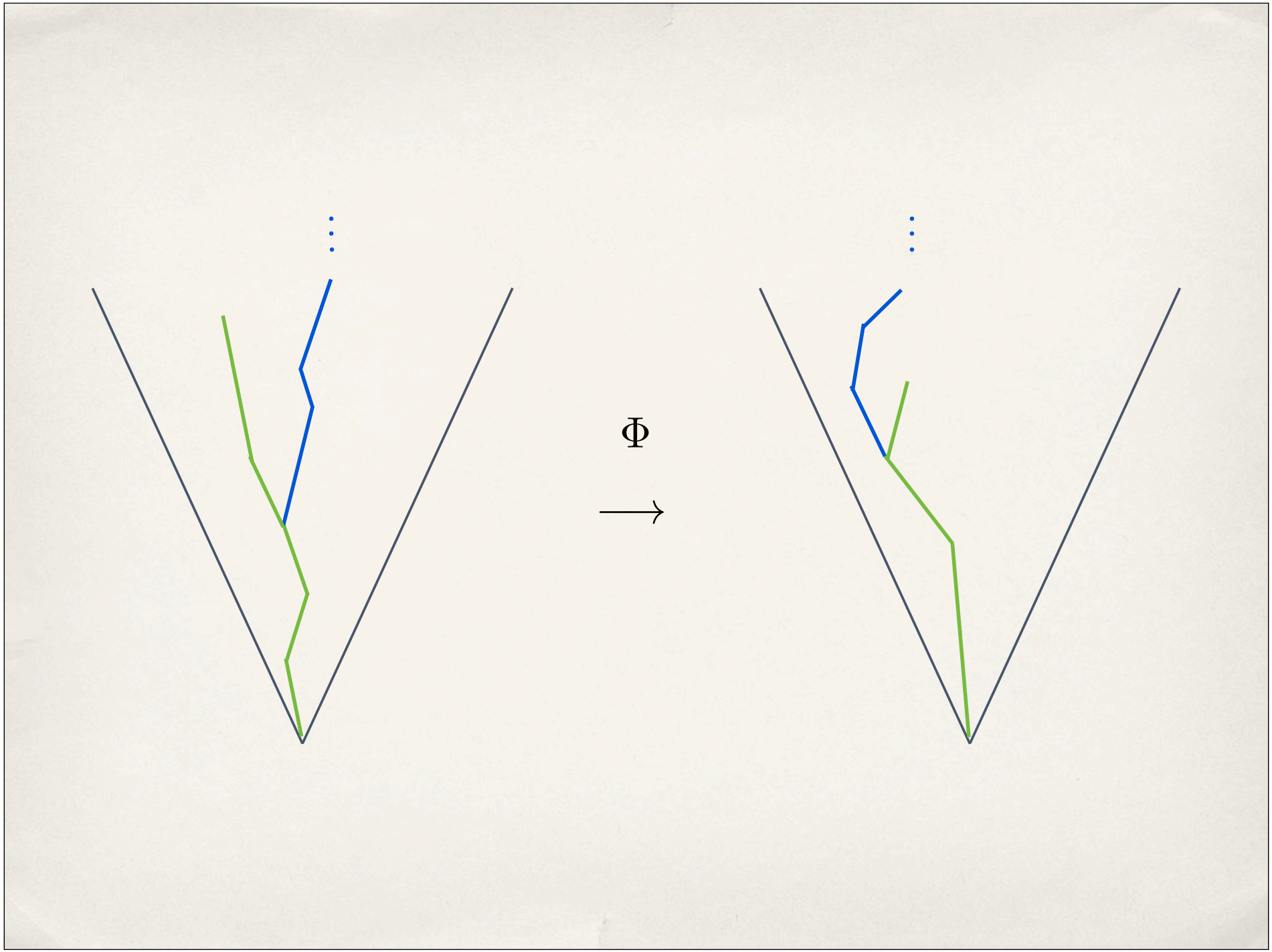

Φ

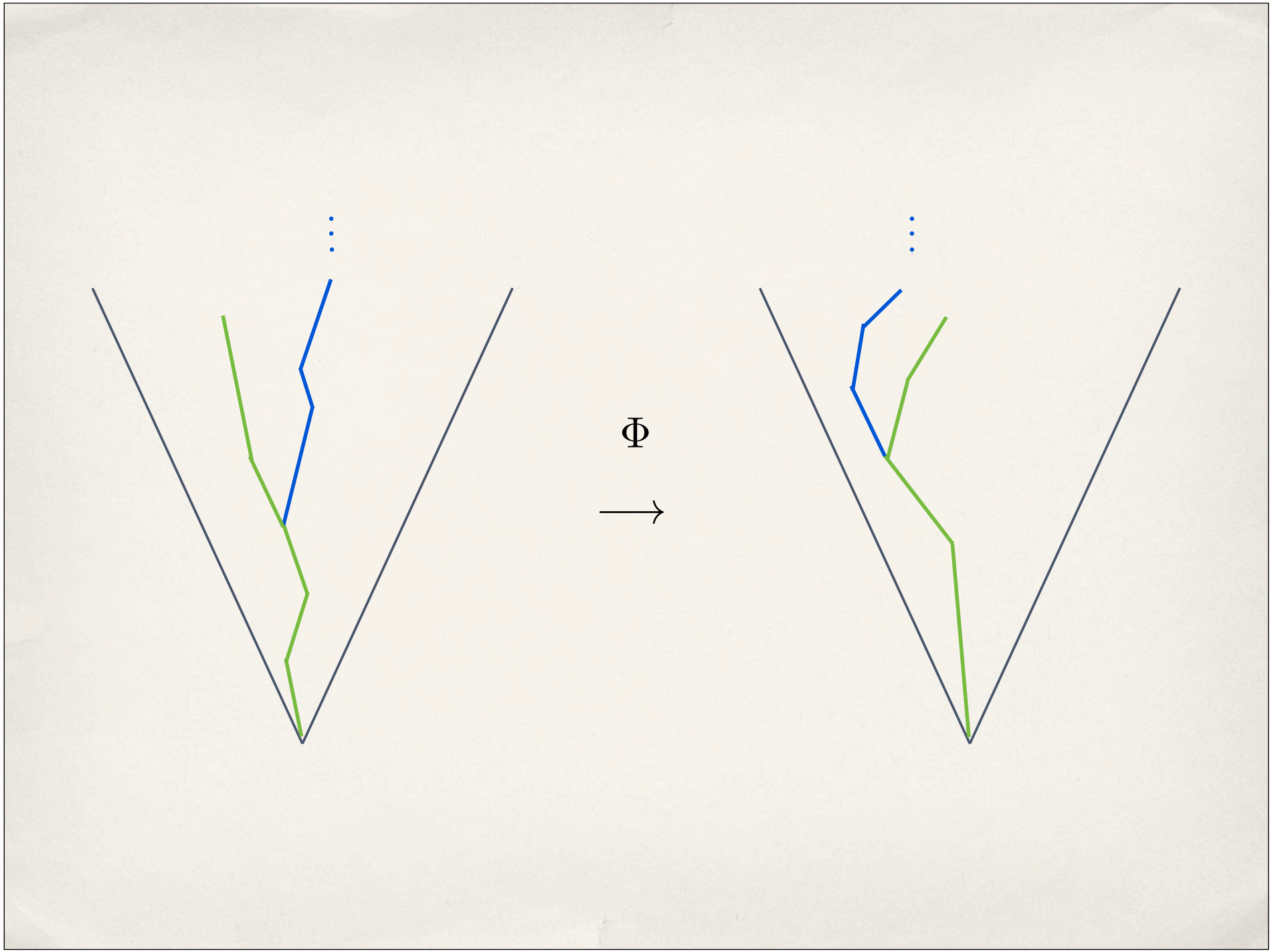

Φ

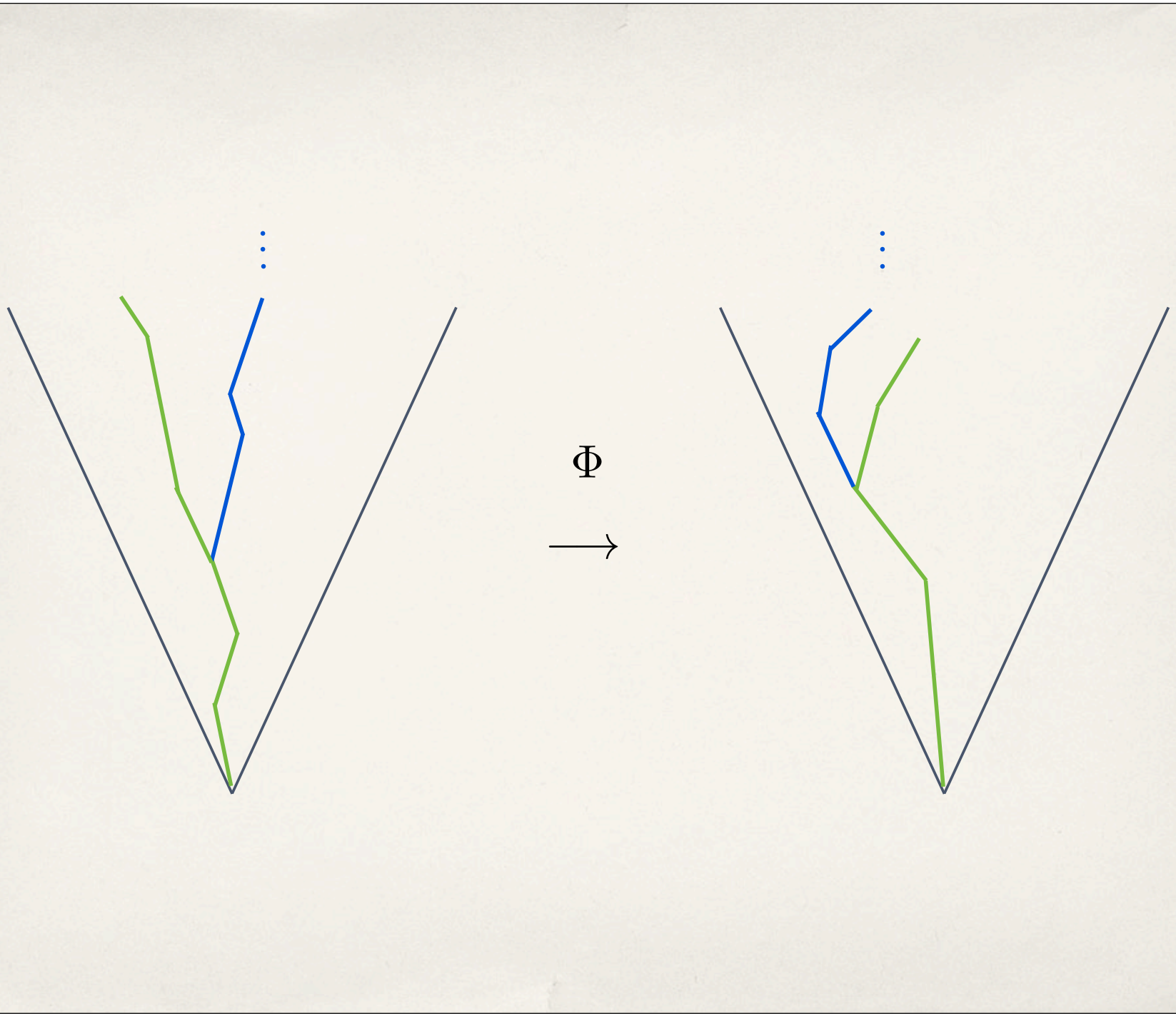

Φ

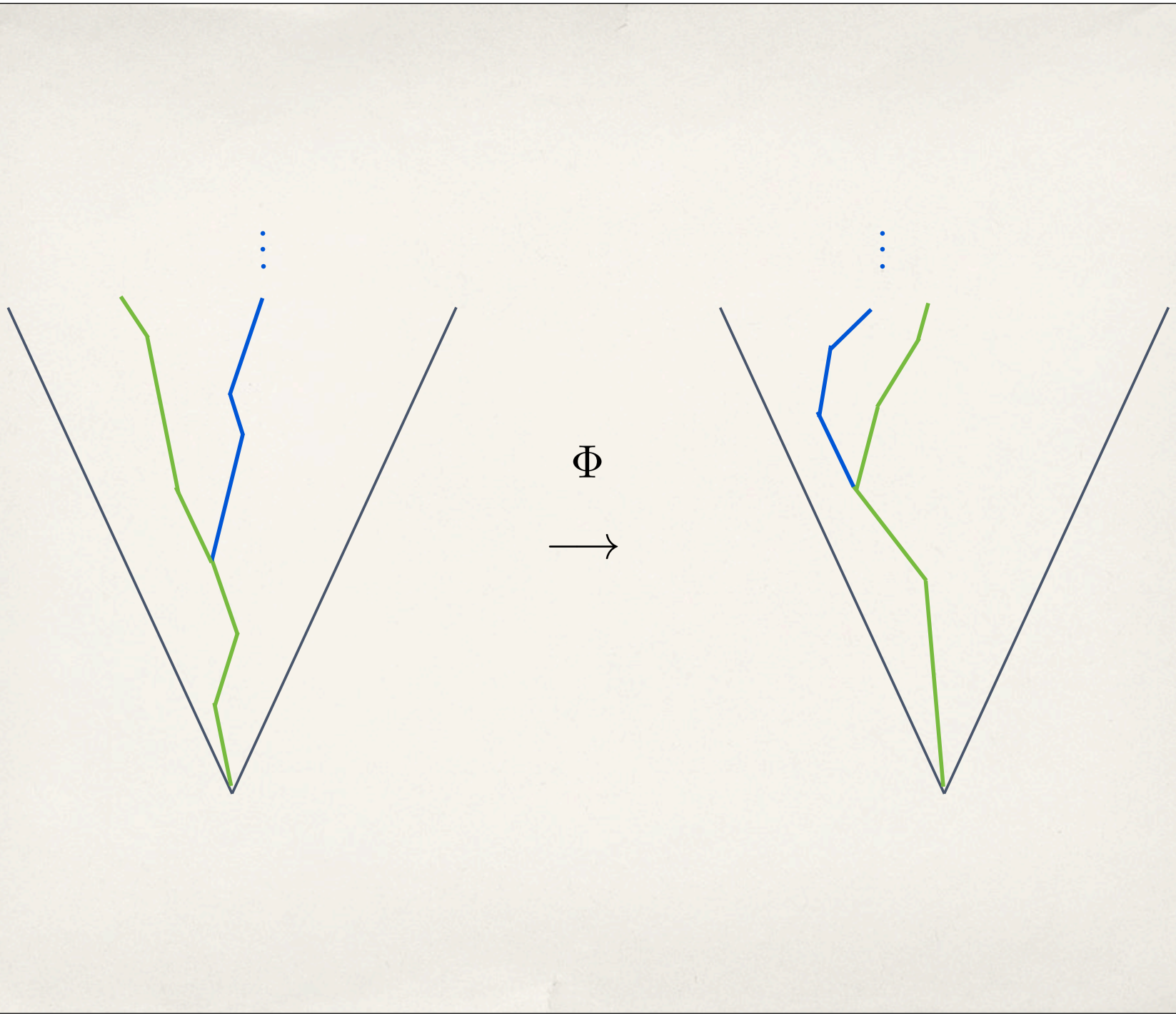

Φ

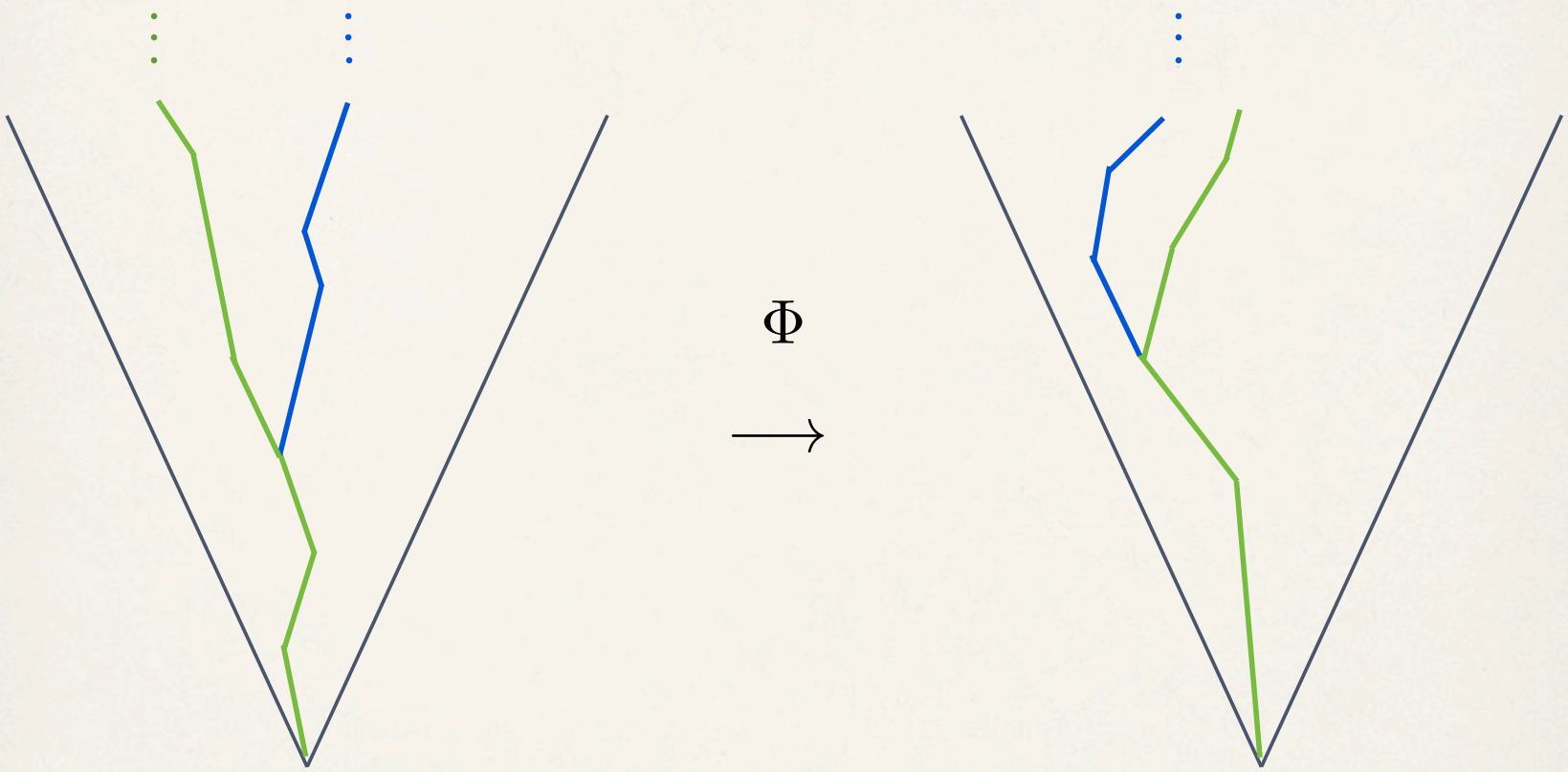

Φ

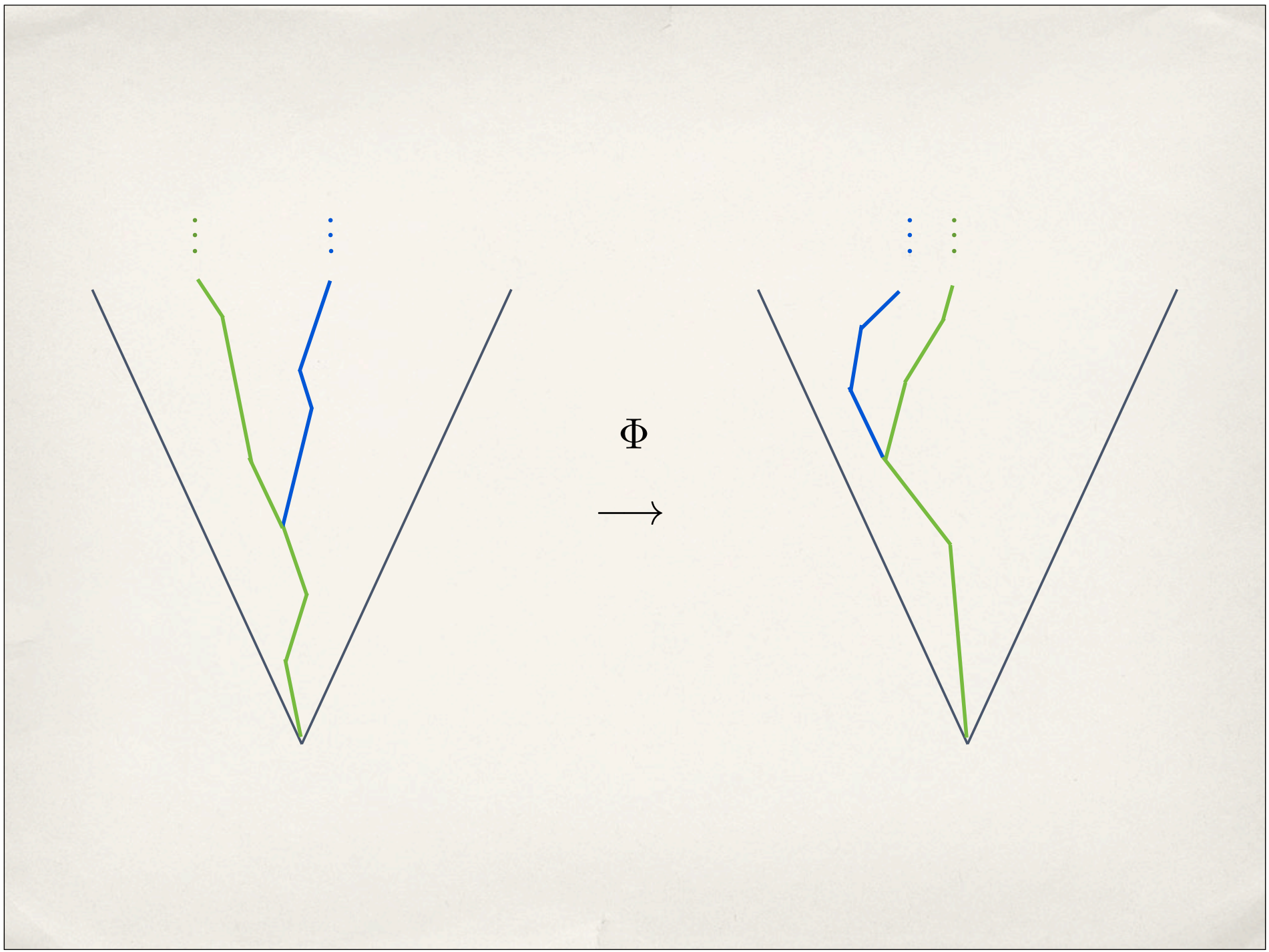

Φ

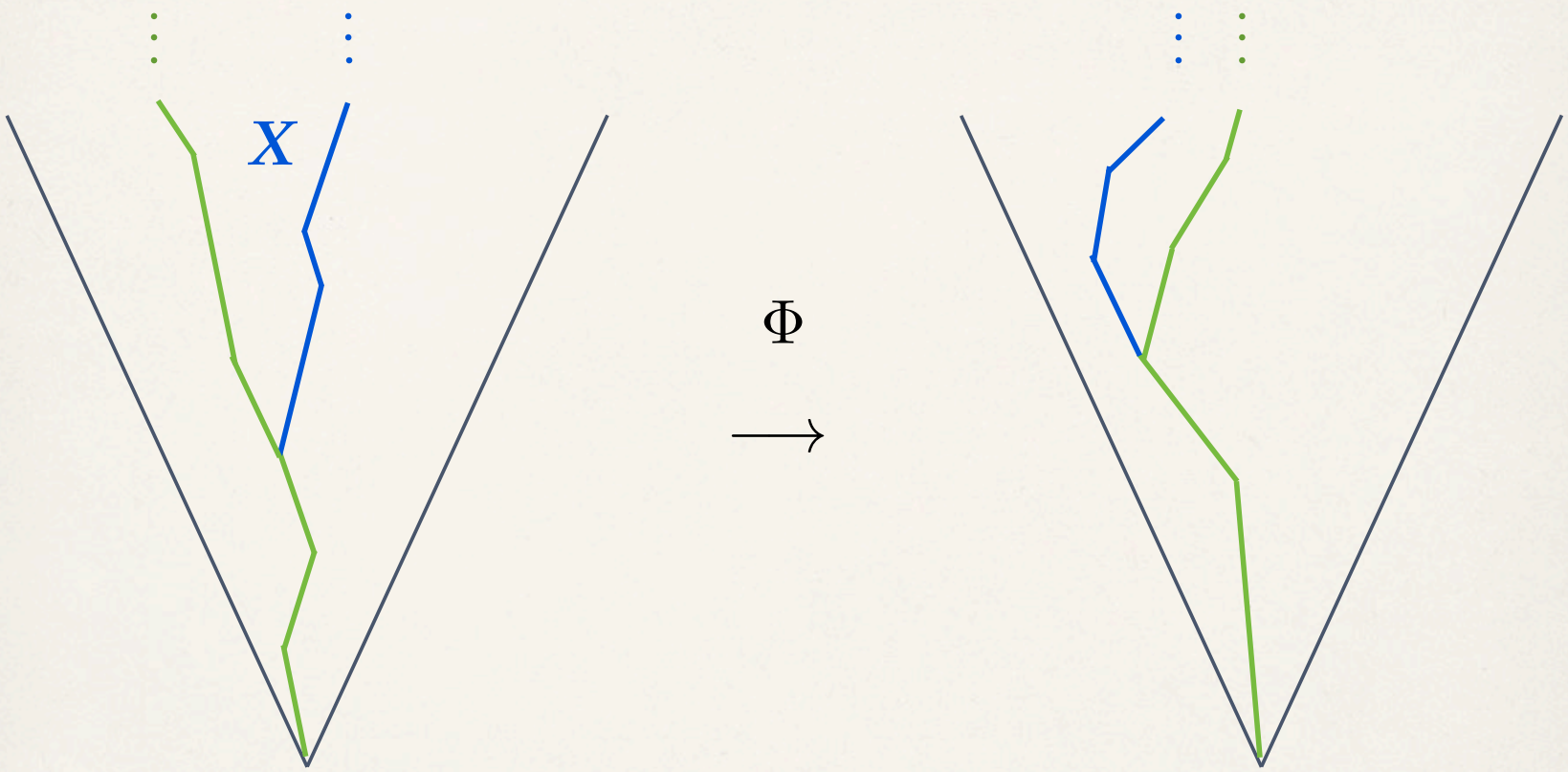

Φ

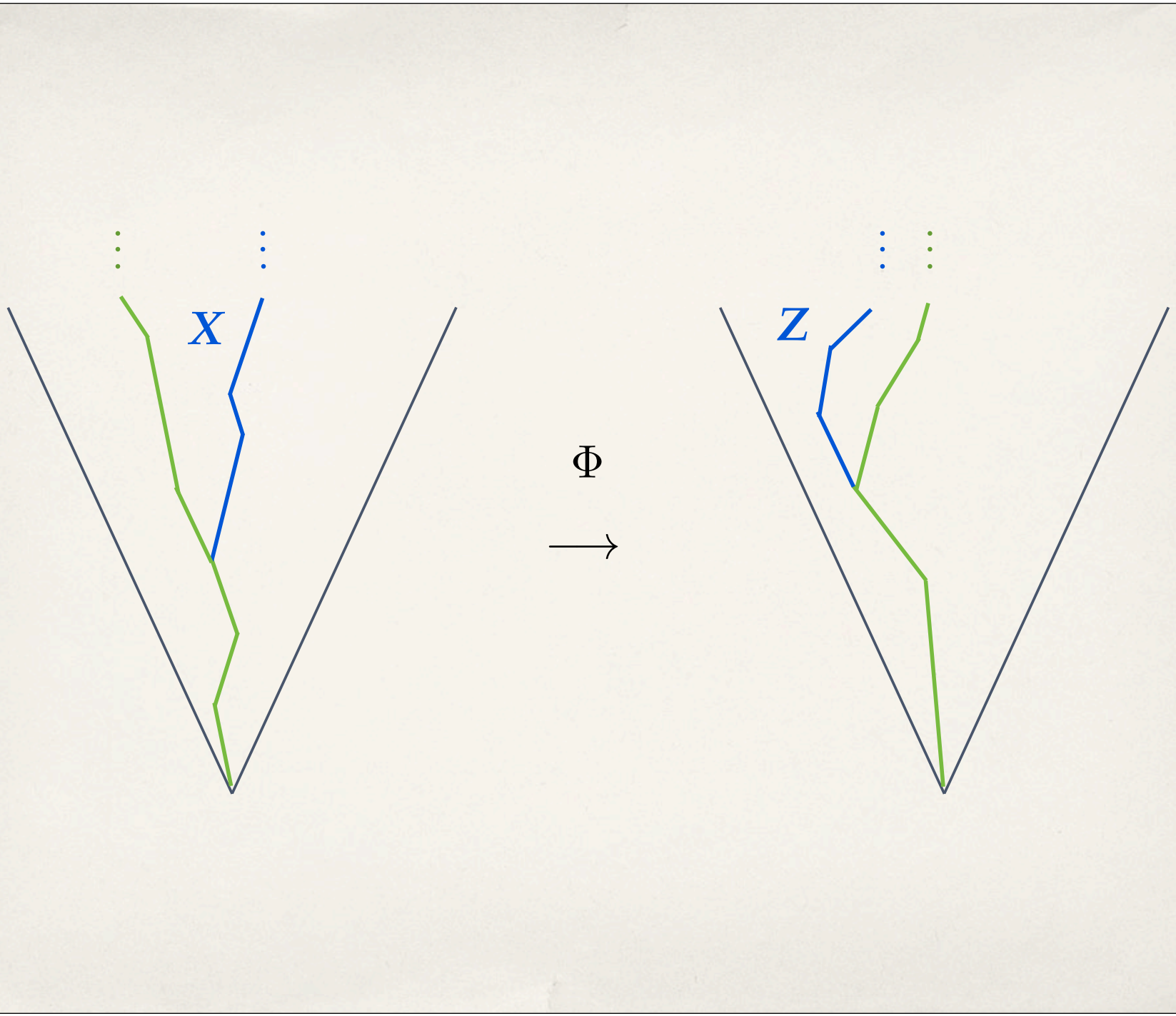

Φ

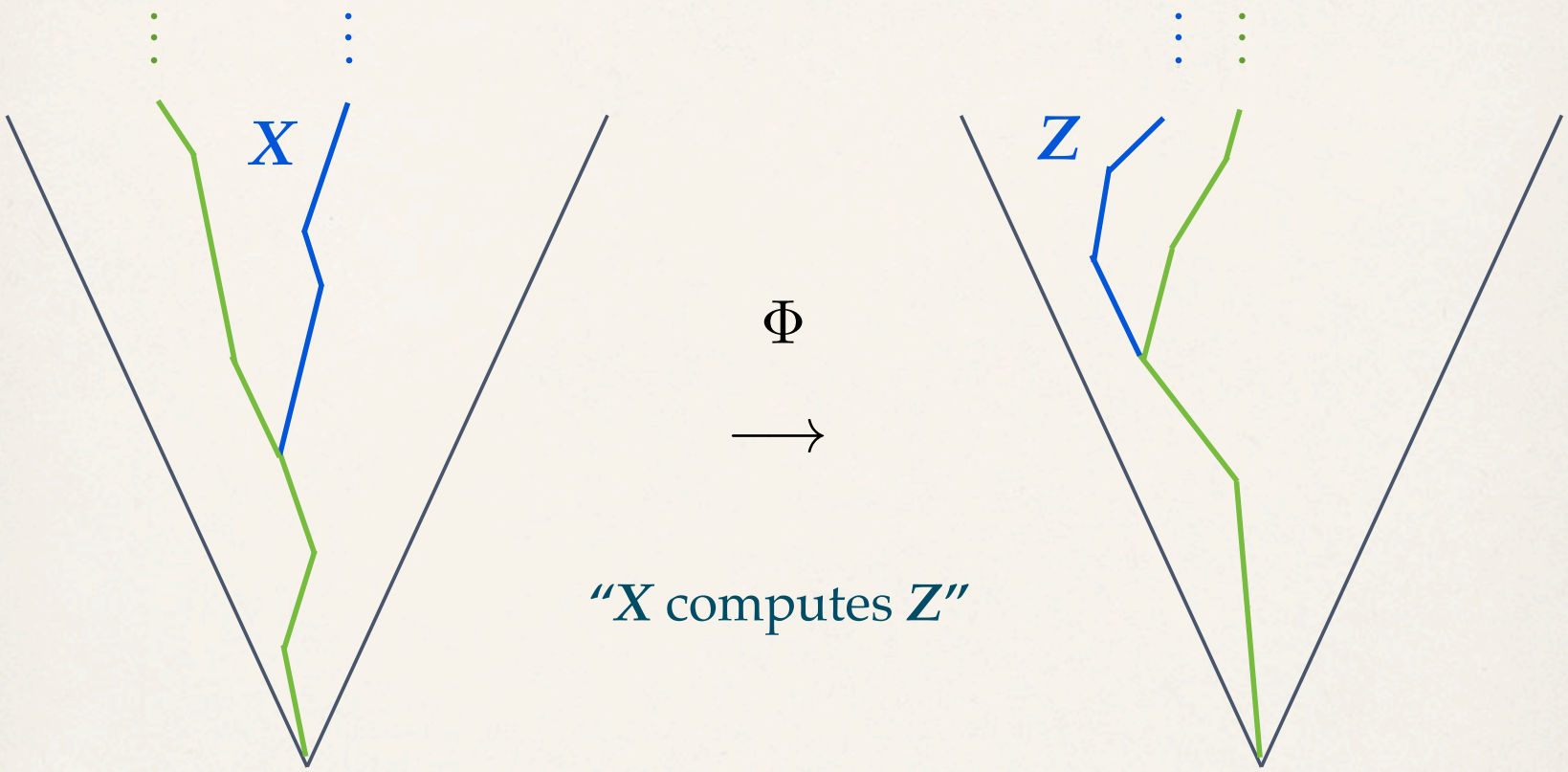

Φ


Φ


Φ


X
Φ


X
Φ
Z



"X computes Z"

The idea behind algorithmic randomness

There are a number of ways to motivate the idea behind algorithmic randomness. For instance:

- A sequence is algorithmically random if its initial segments cannot be compressed.
- A sequence is algorithmically random if it satisfies every effective law of probability.

These two ideas can be made precise in such a way that the resulting definitions of algorithmic randomness are equivalent.

Martin-Löf randomness

A ***Martin-Löf test*** is a sequence $(\mathcal{U}_i)_{i\in\omega}$ of uniformly effectively open subsets of 2^ω such that for each $i \in \omega$,

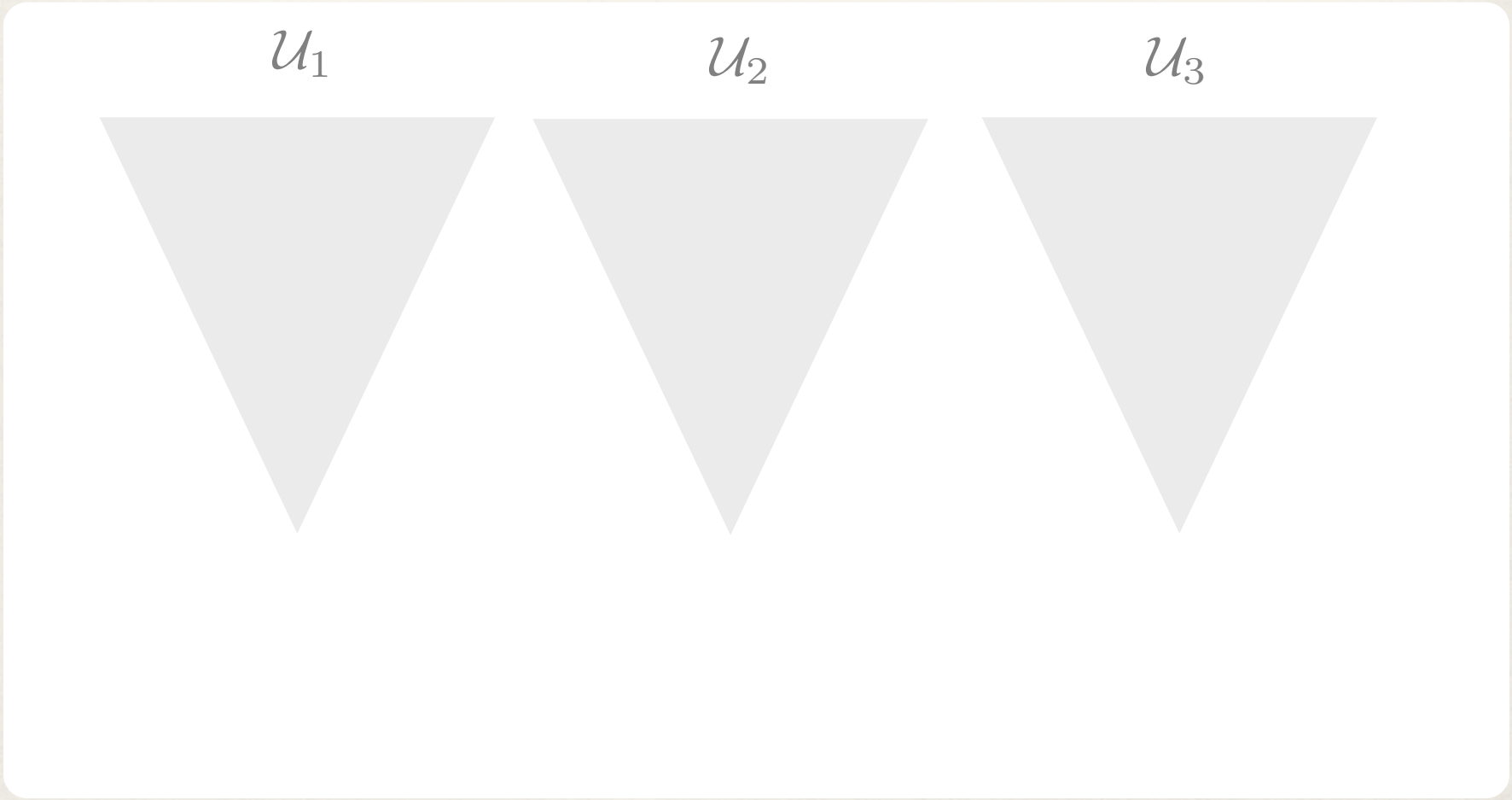
$$\lambda(\mathcal{U}_i) \leq 2^{-i}.$$

A sequence $X \in 2^\omega$ ***passes a Martin-Löf test*** $(\mathcal{U}_i)_{i\in\omega}$ if

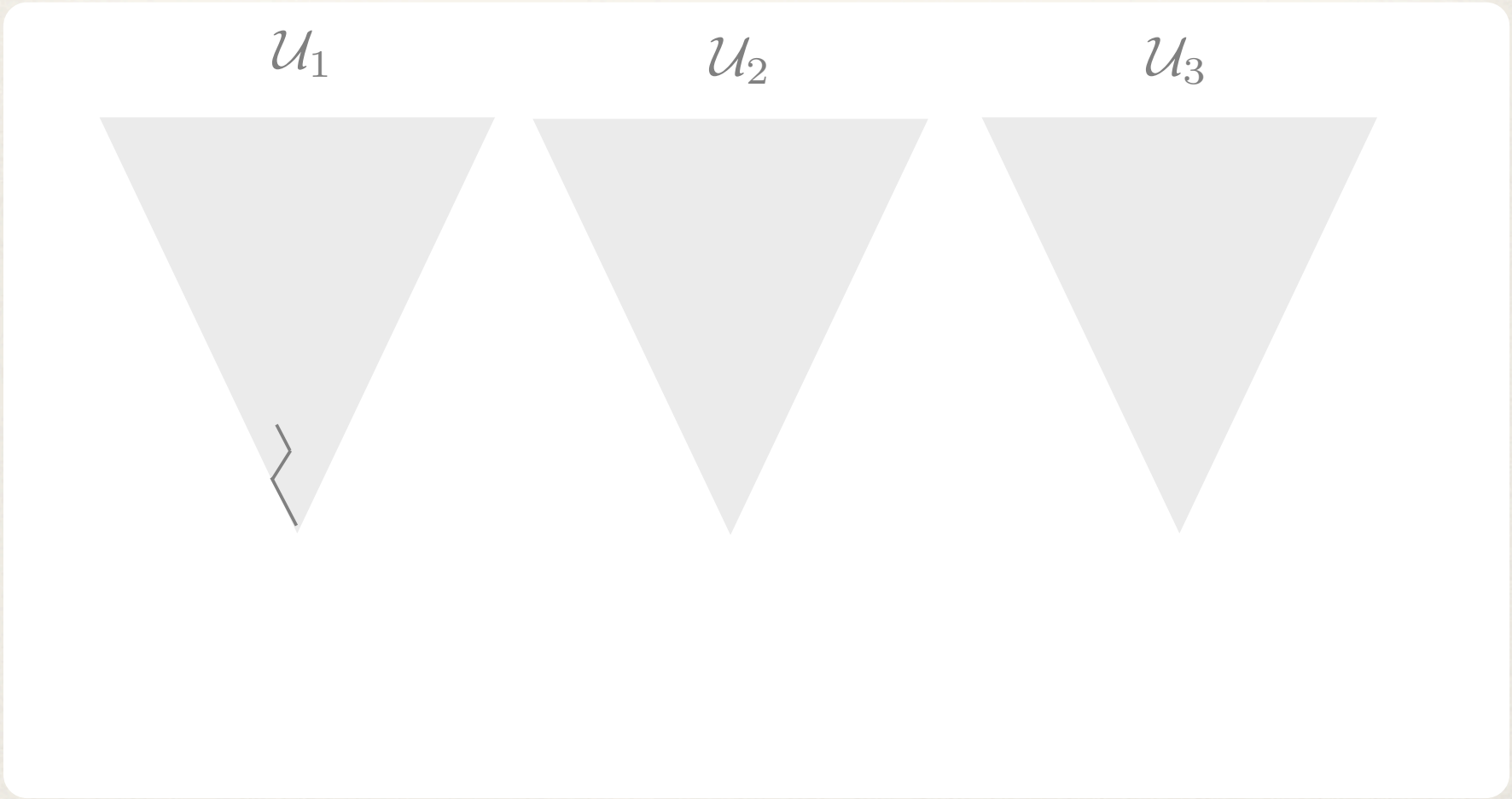
$$X \notin \bigcap_{i\in\omega} \mathcal{U}_i.$$

A sequence is ***Martin-Löf random*** if it passes every Martin-Löf test.

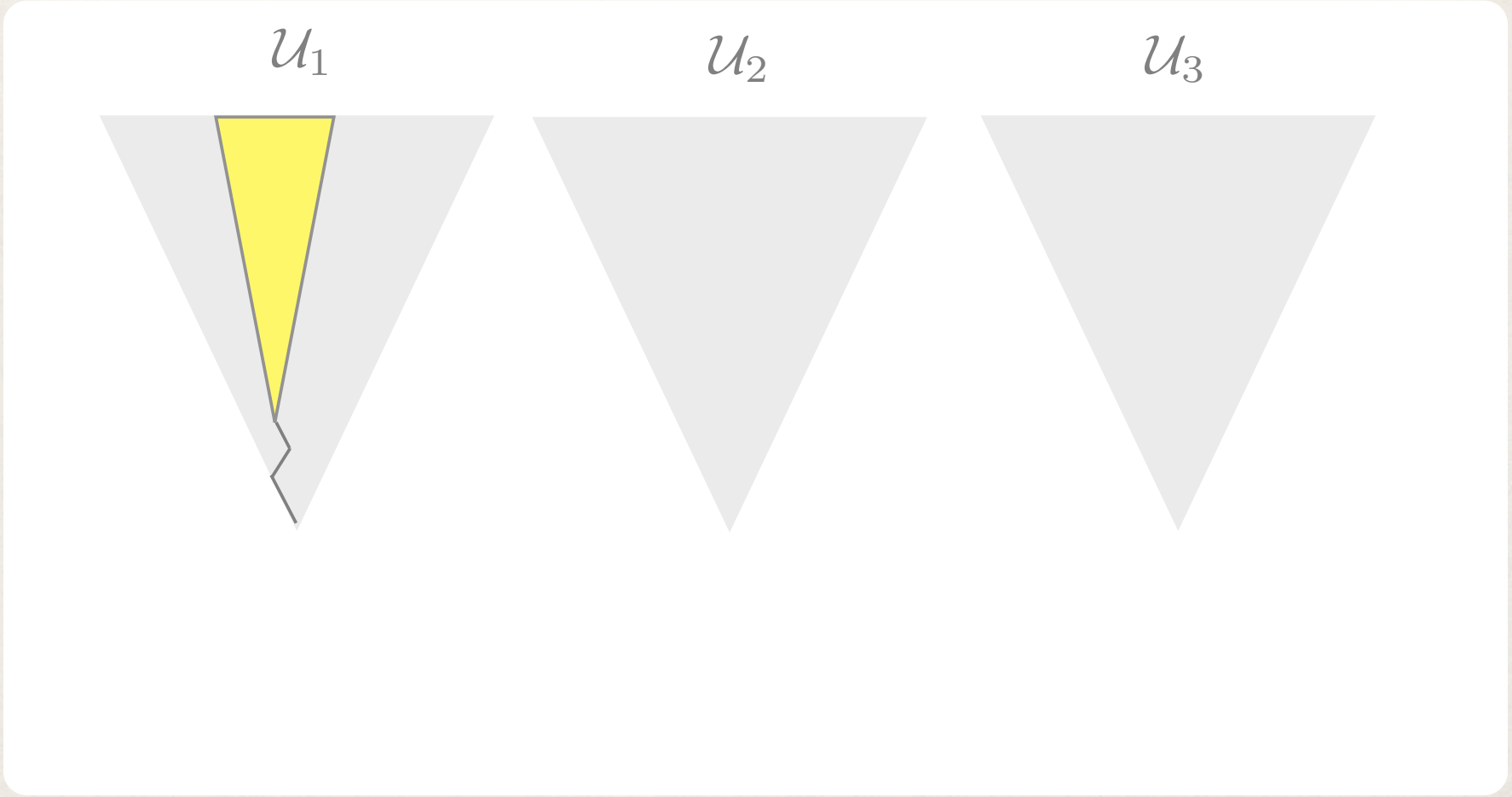
An abstract example



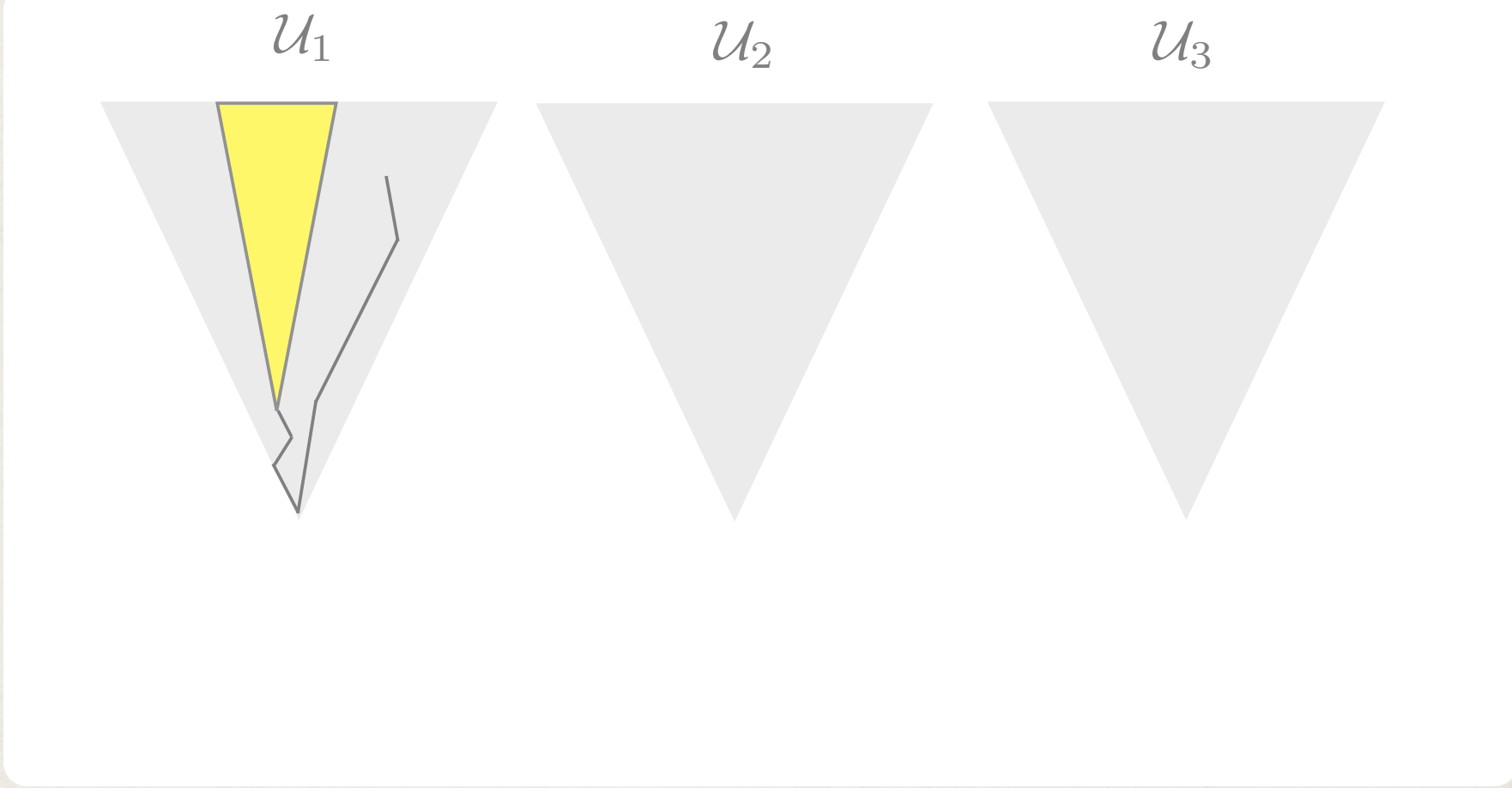
An abstract example



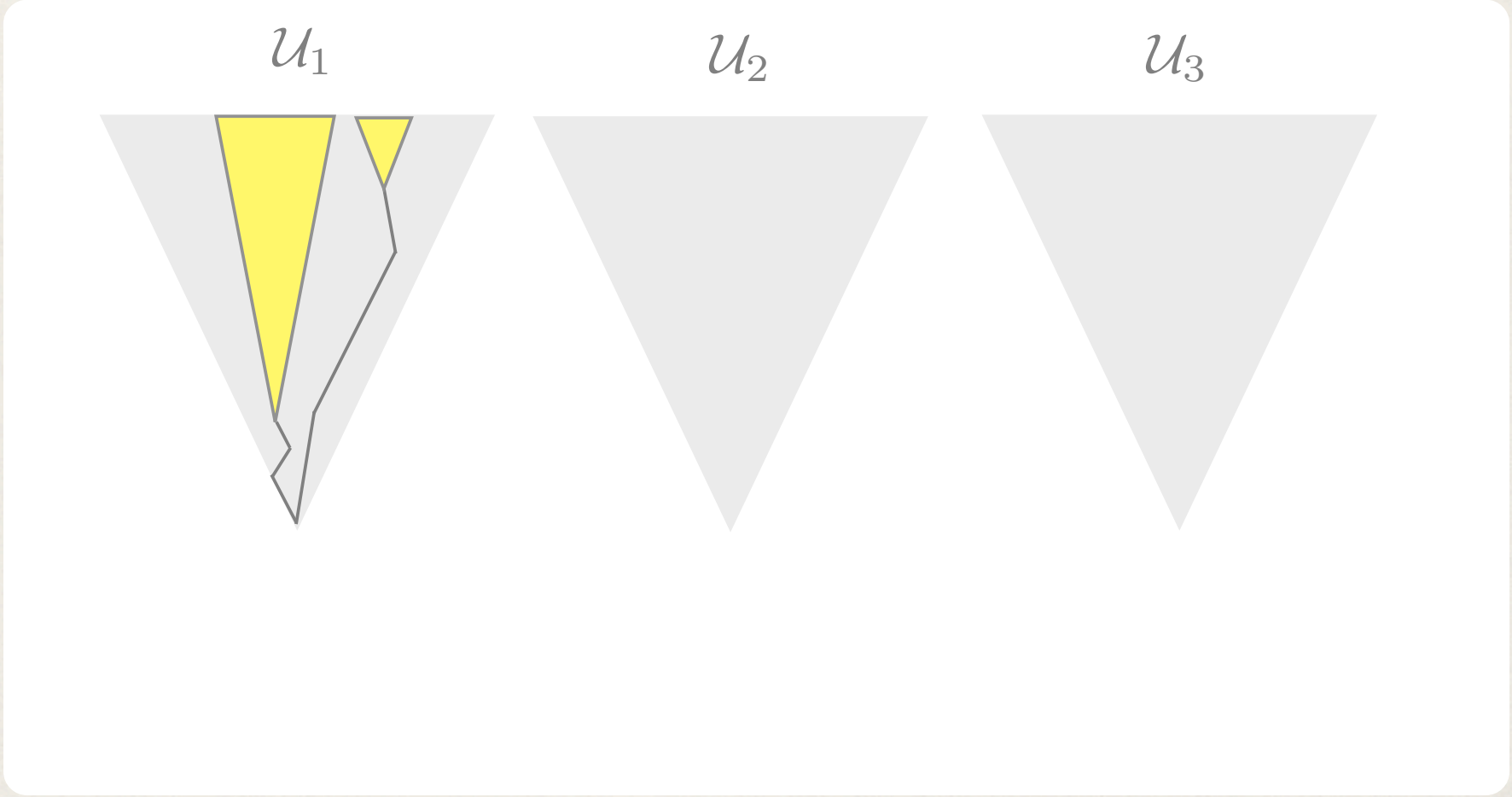
An abstract example



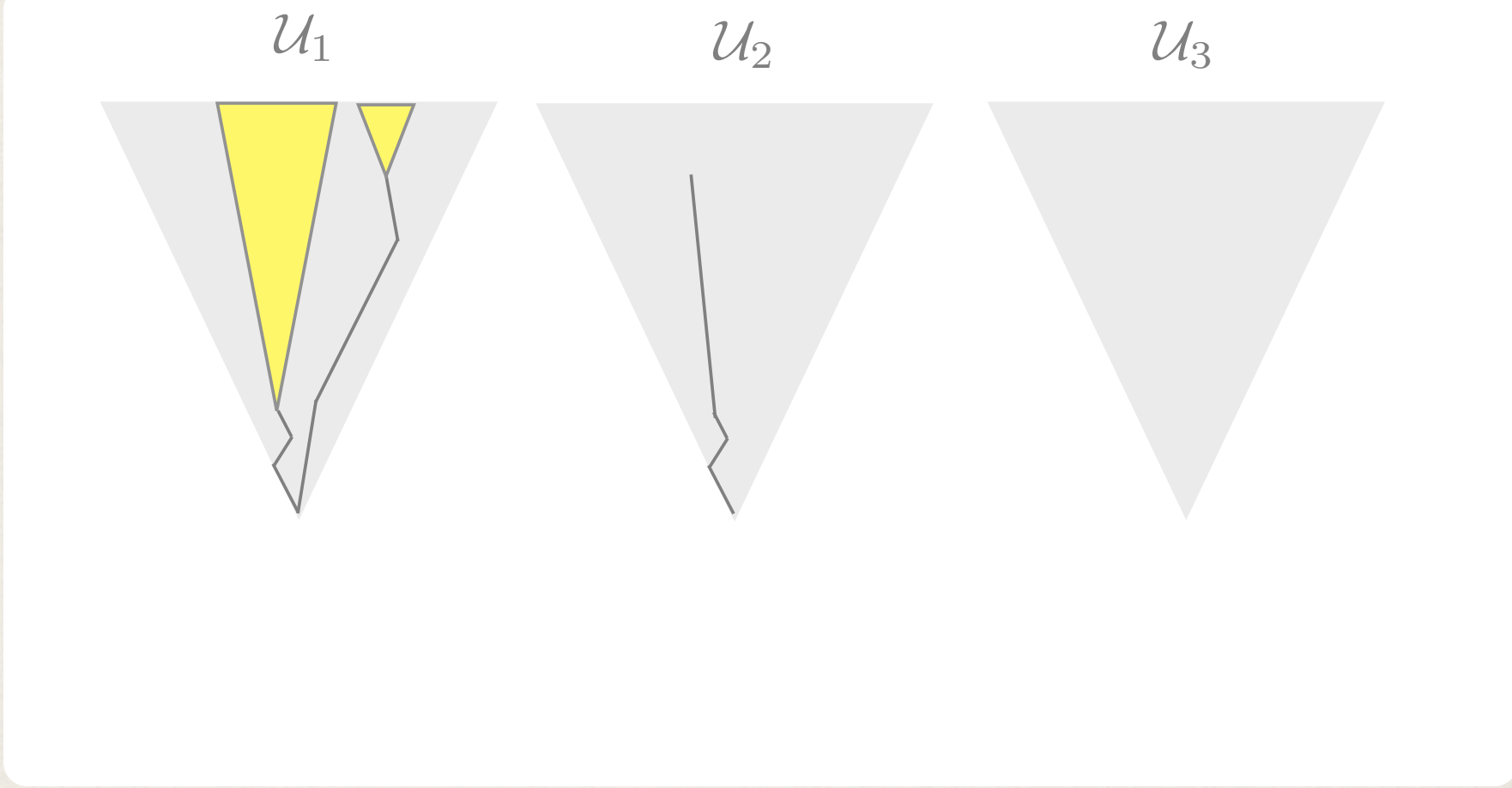
An abstract example



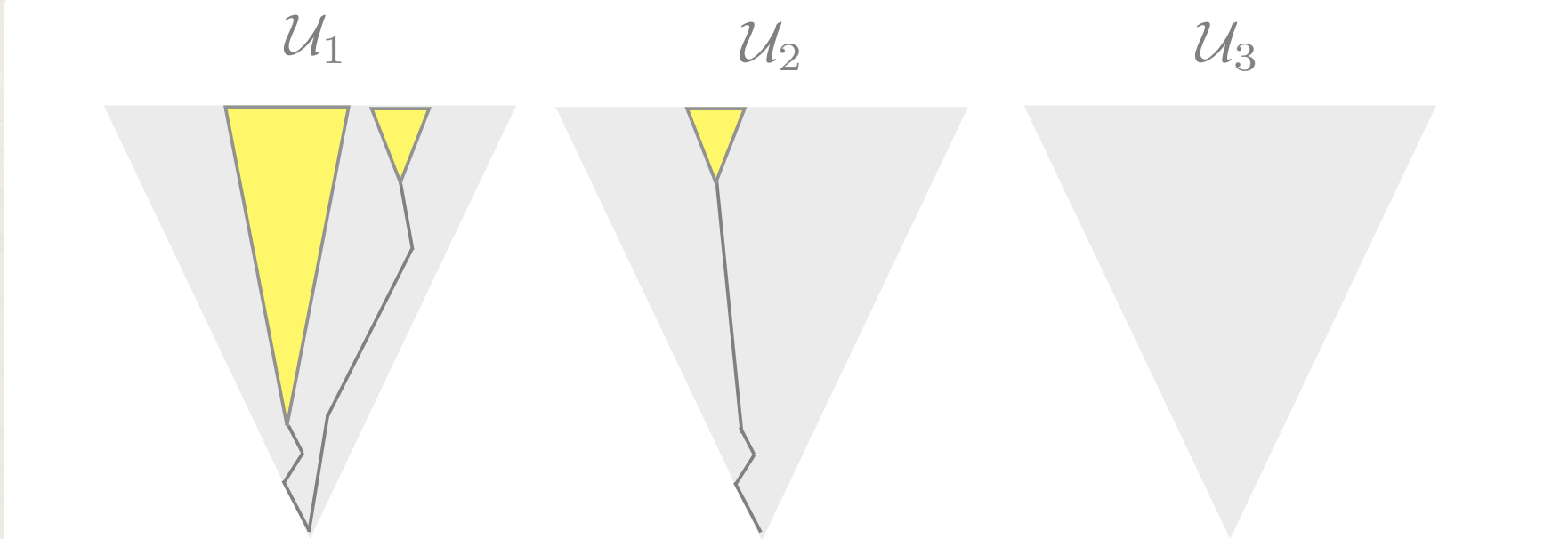
An abstract example



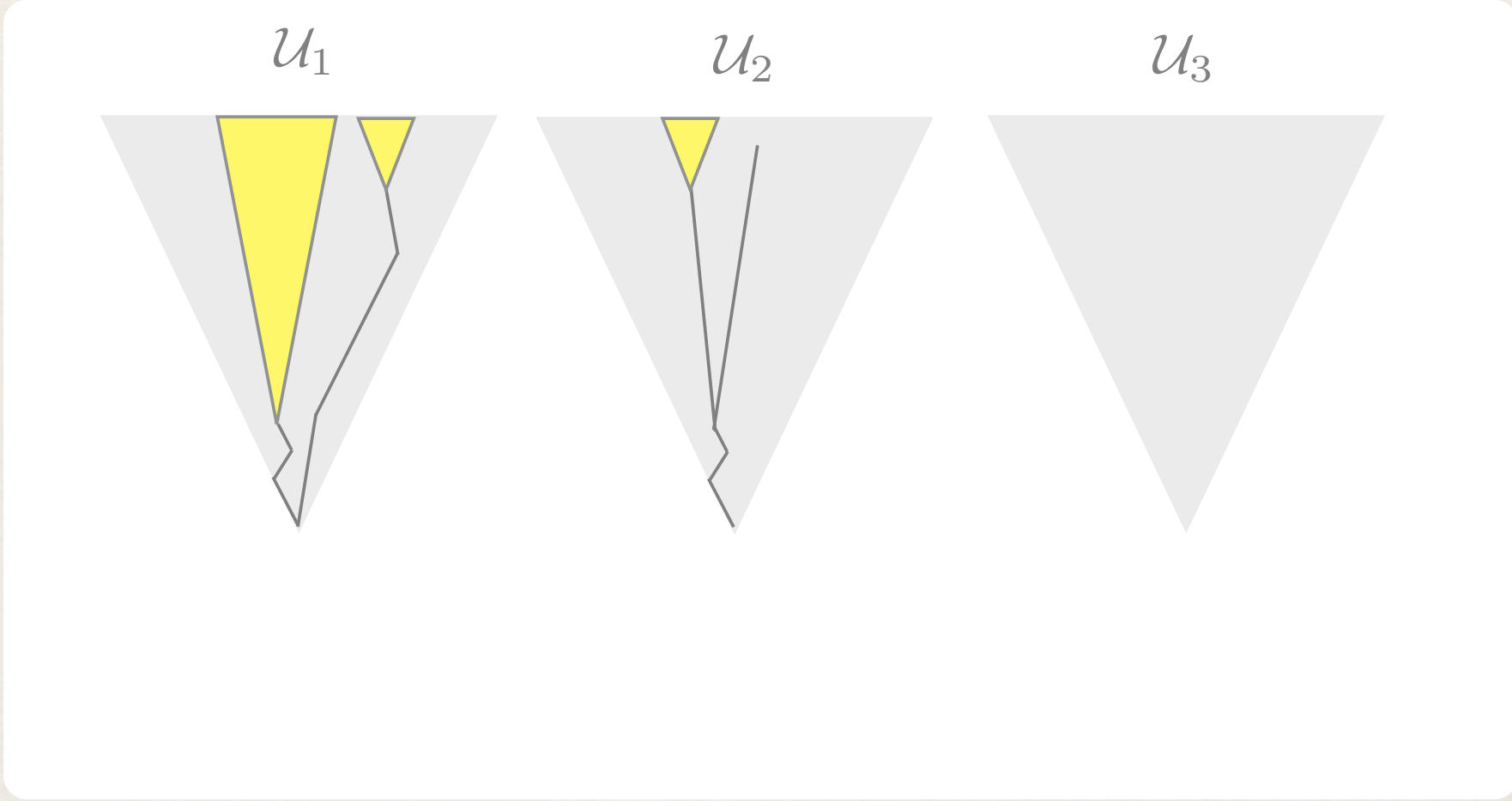
An abstract example



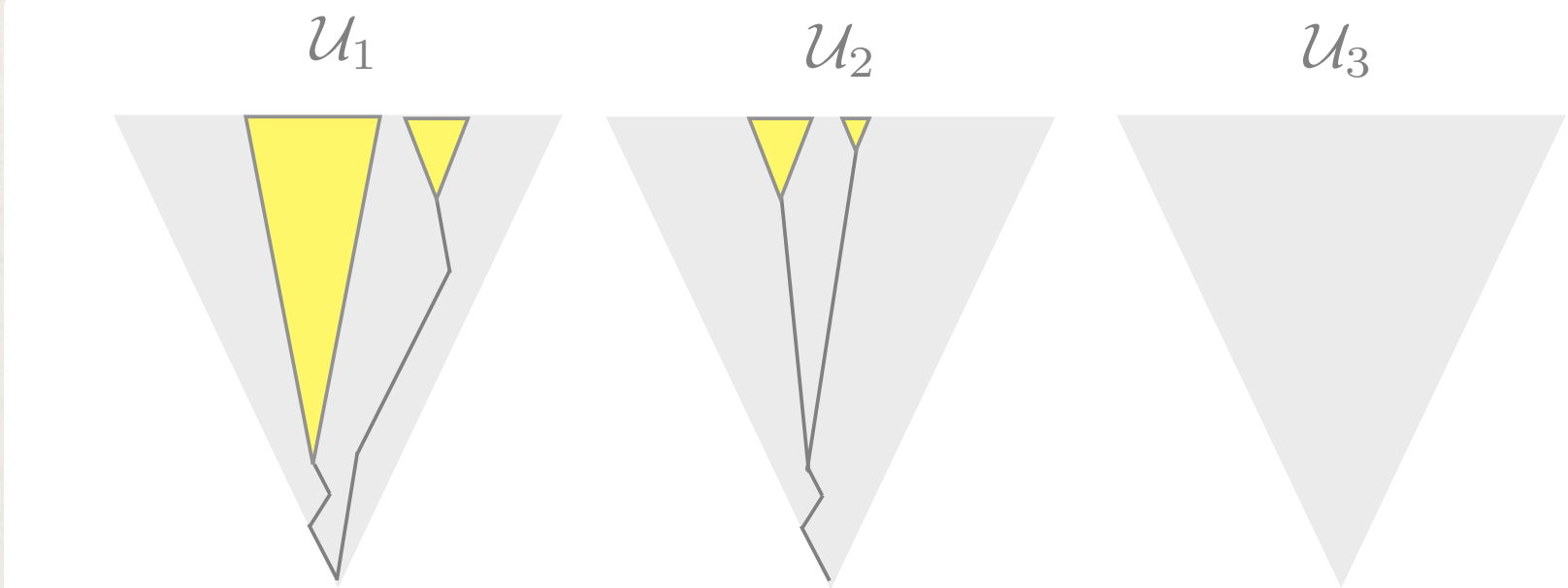
An abstract example



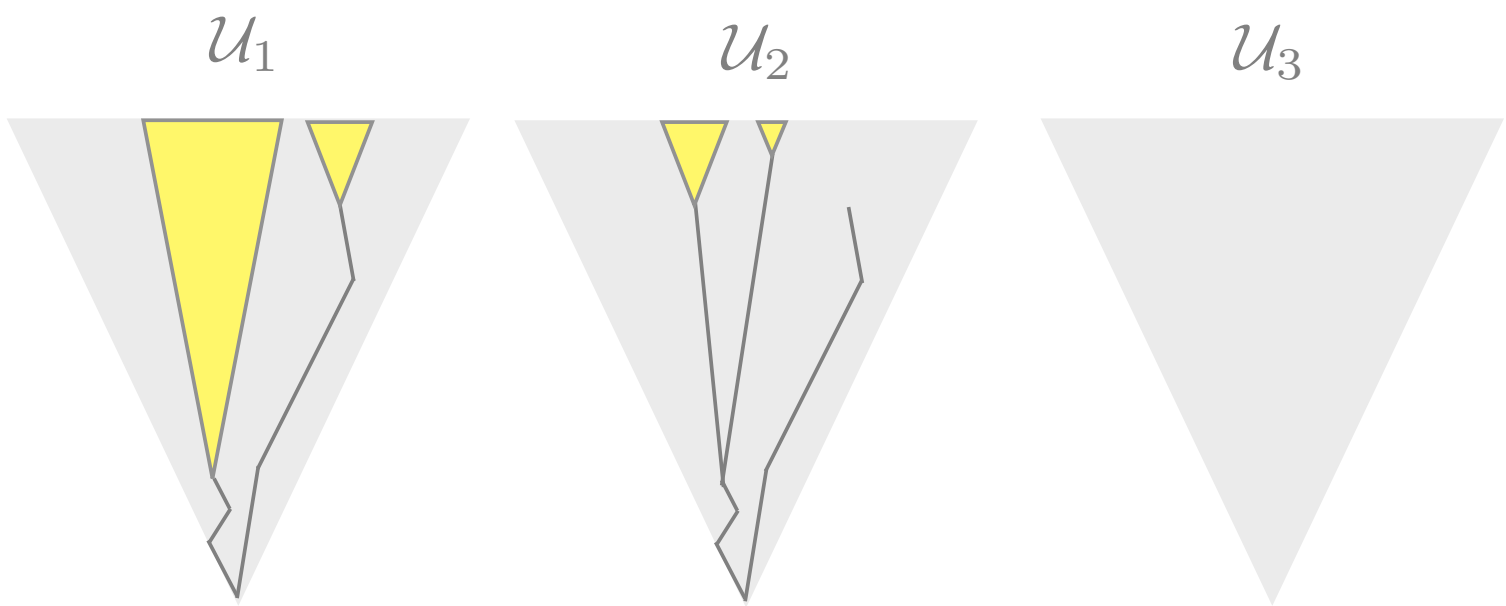
An abstract example



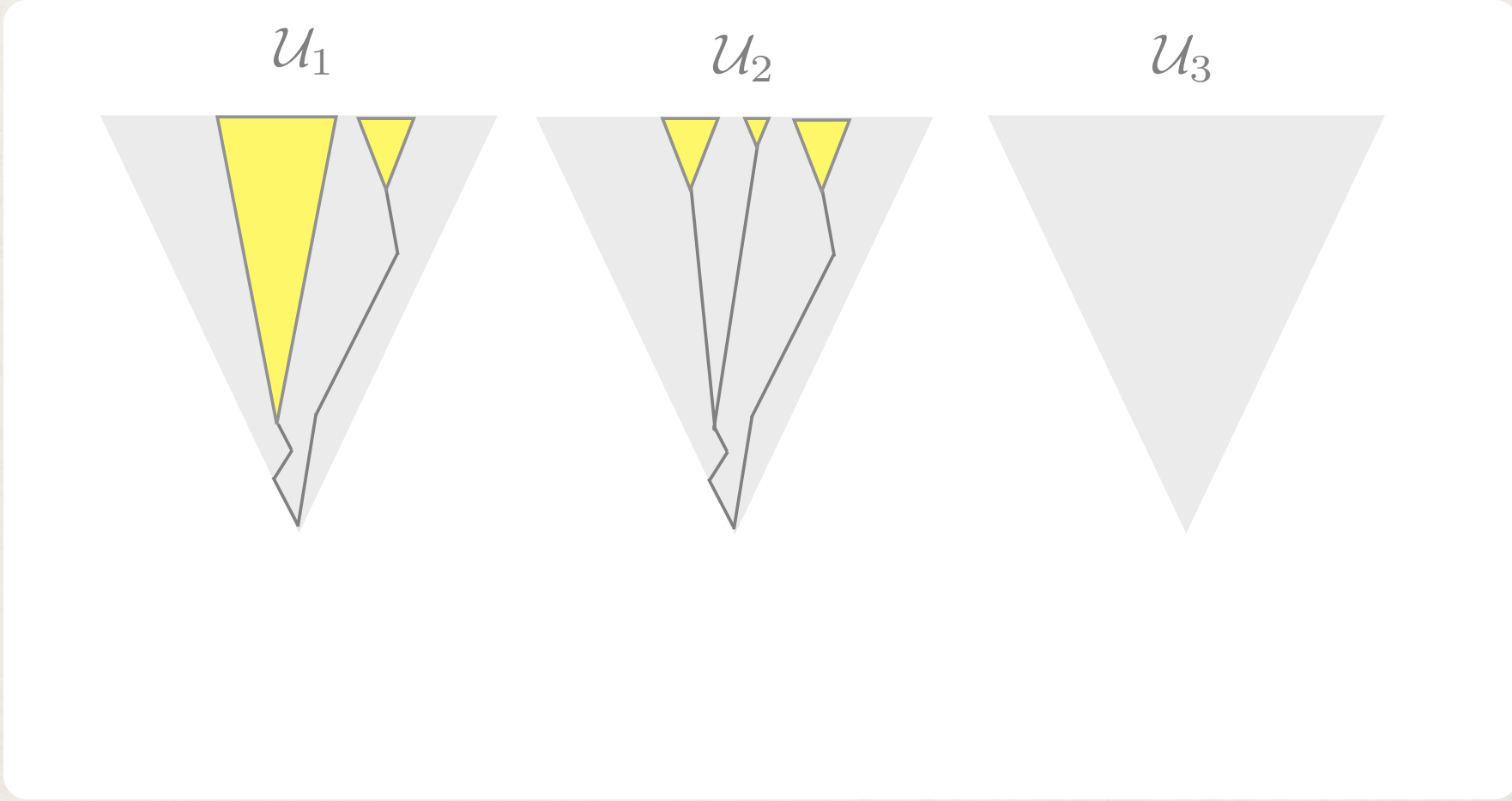
An abstract example



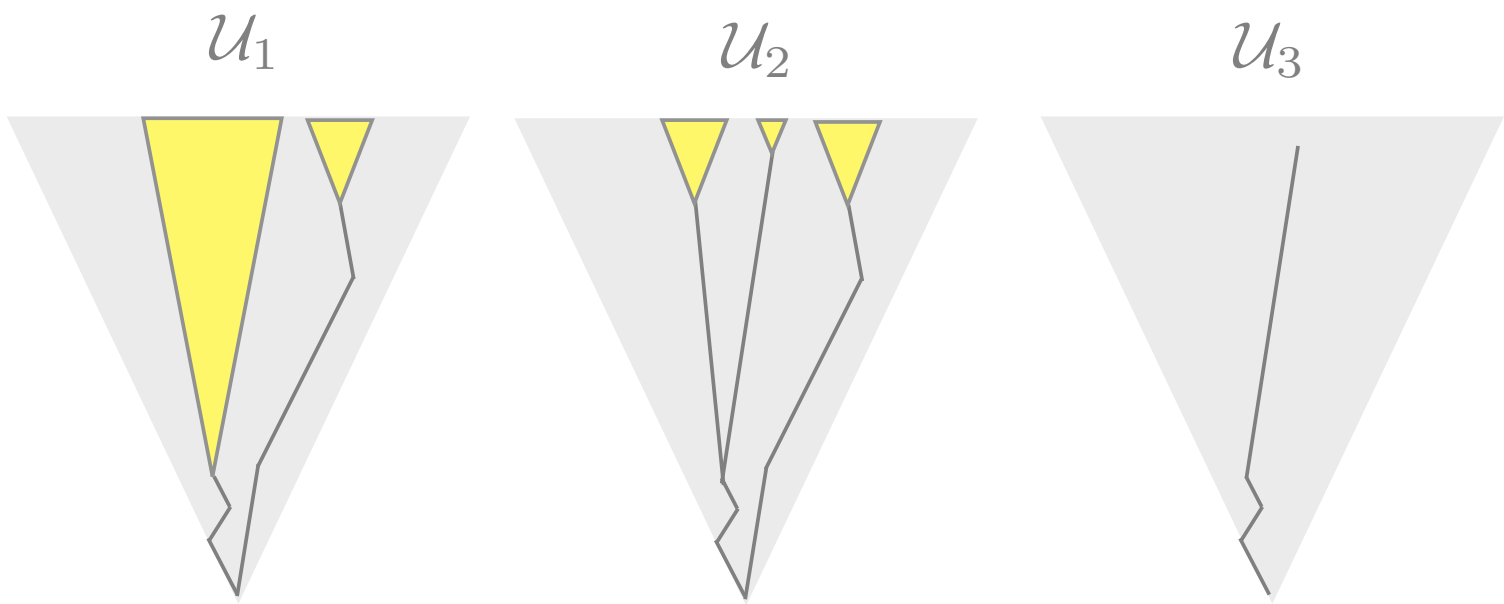
An abstract example



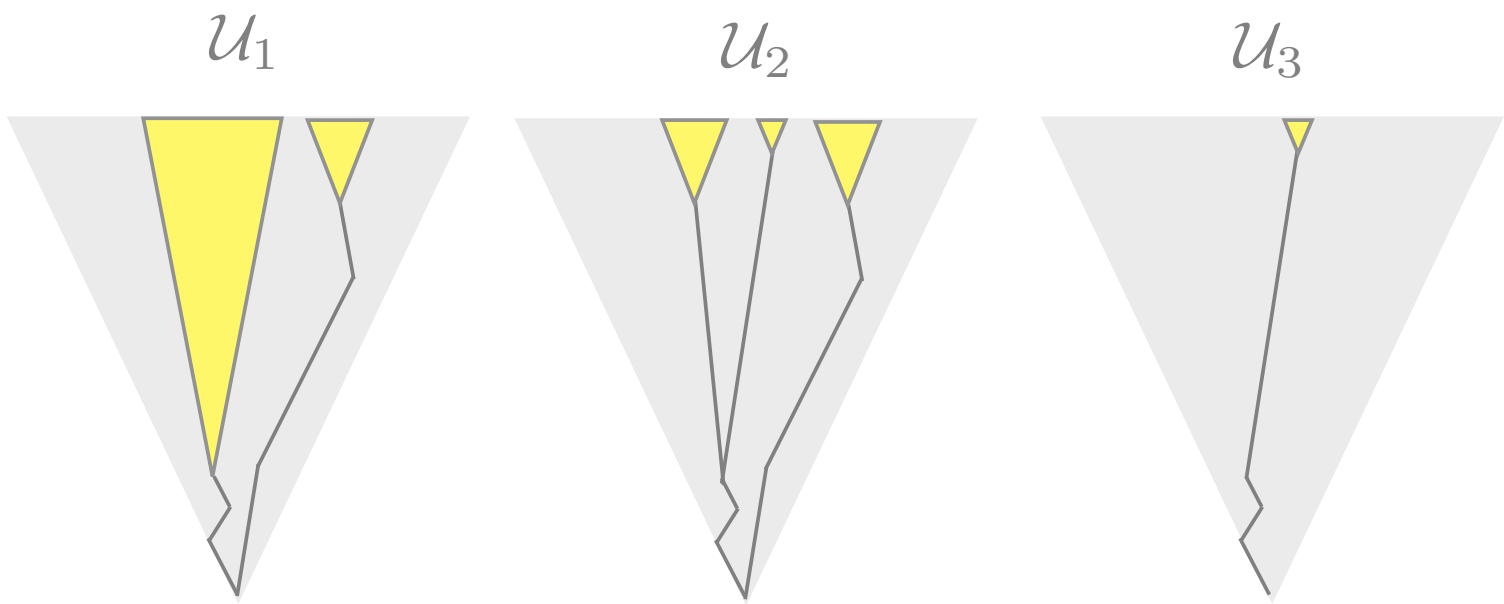
An abstract example



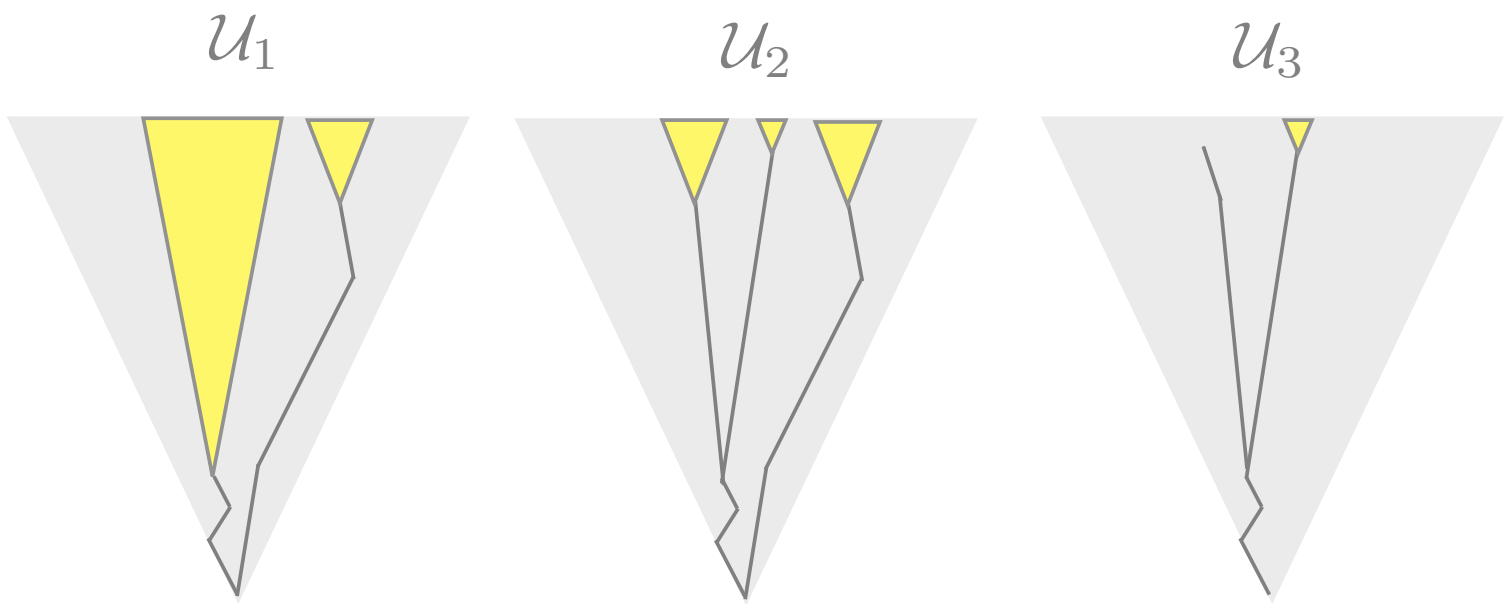
An abstract example



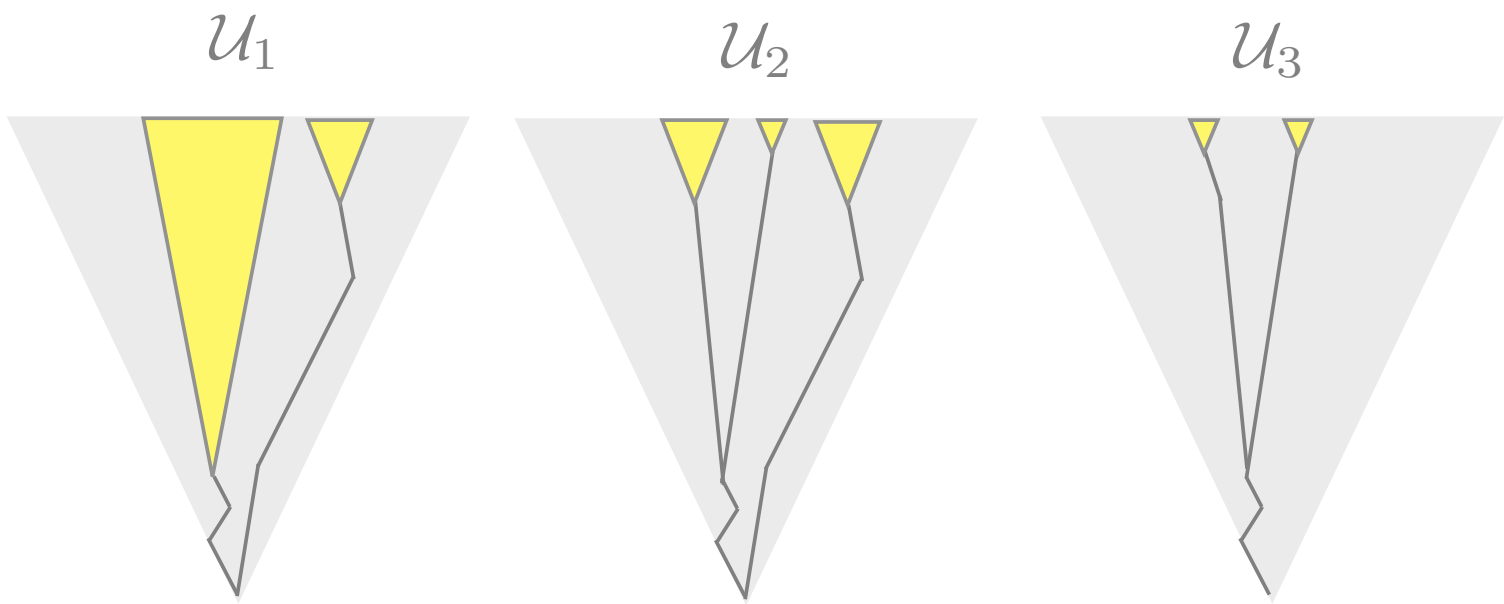
An abstract example



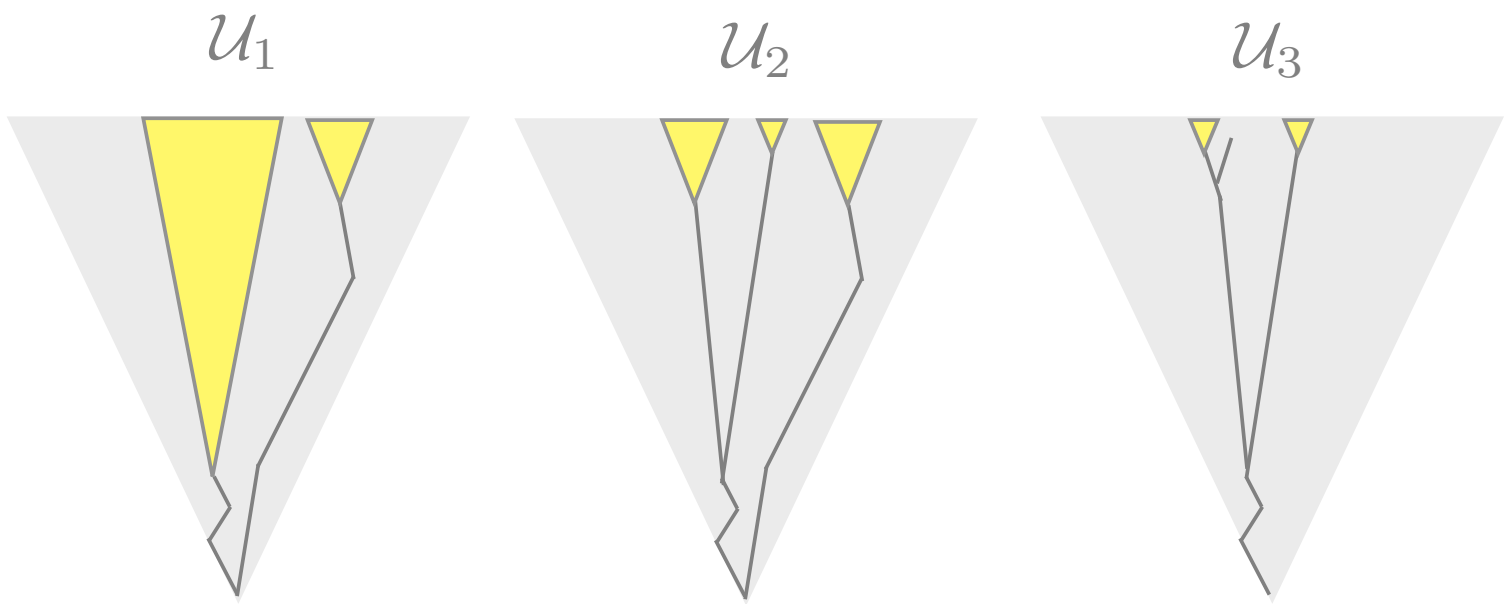
An abstract example



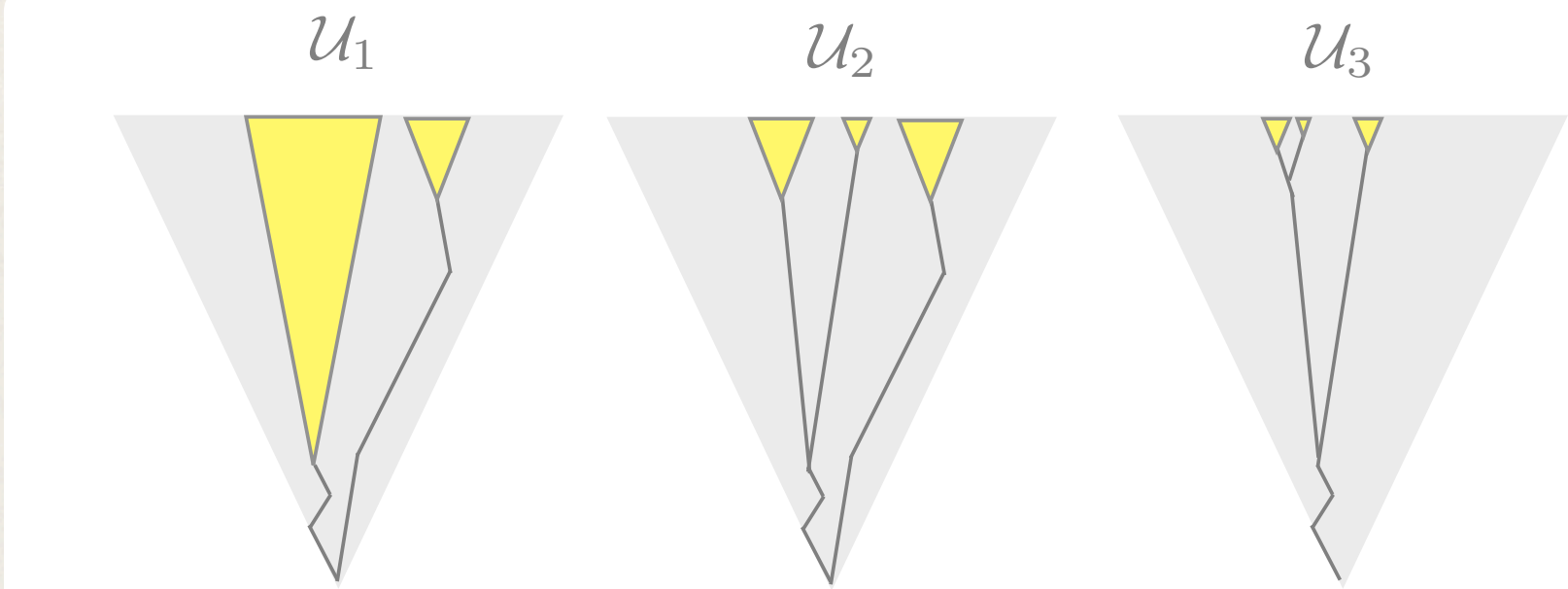
An abstract example



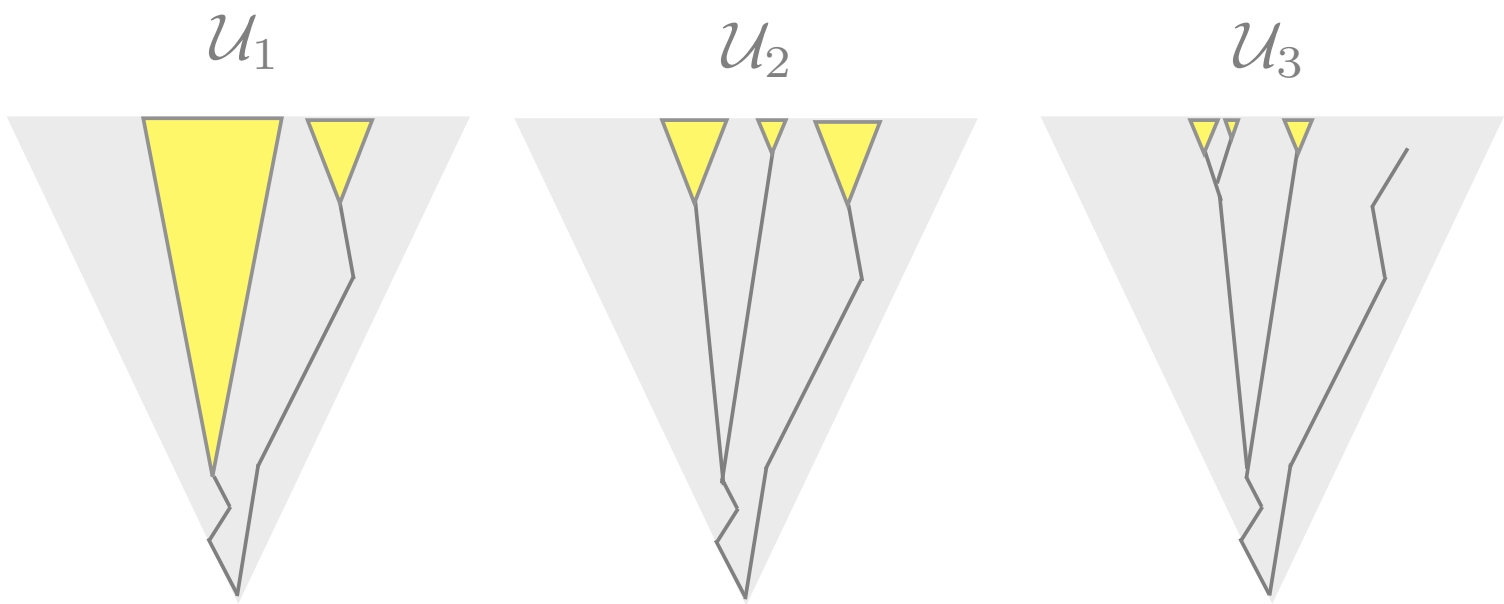
An abstract example



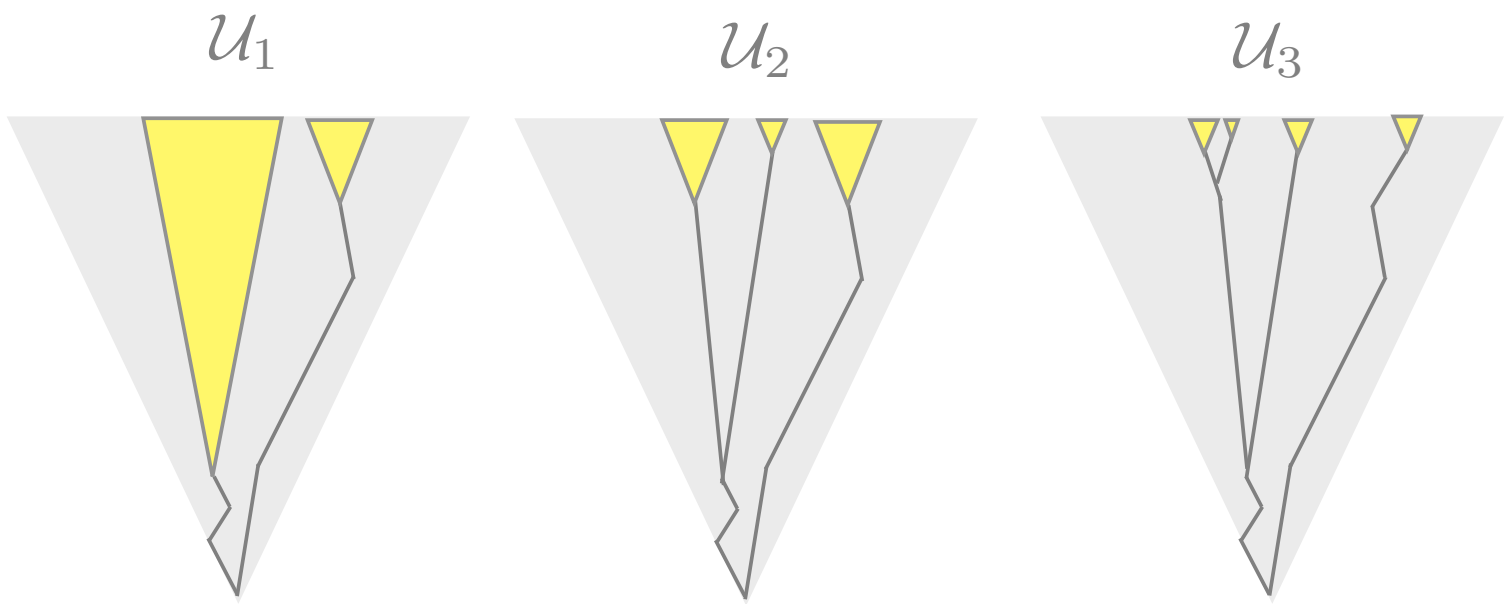
An abstract example



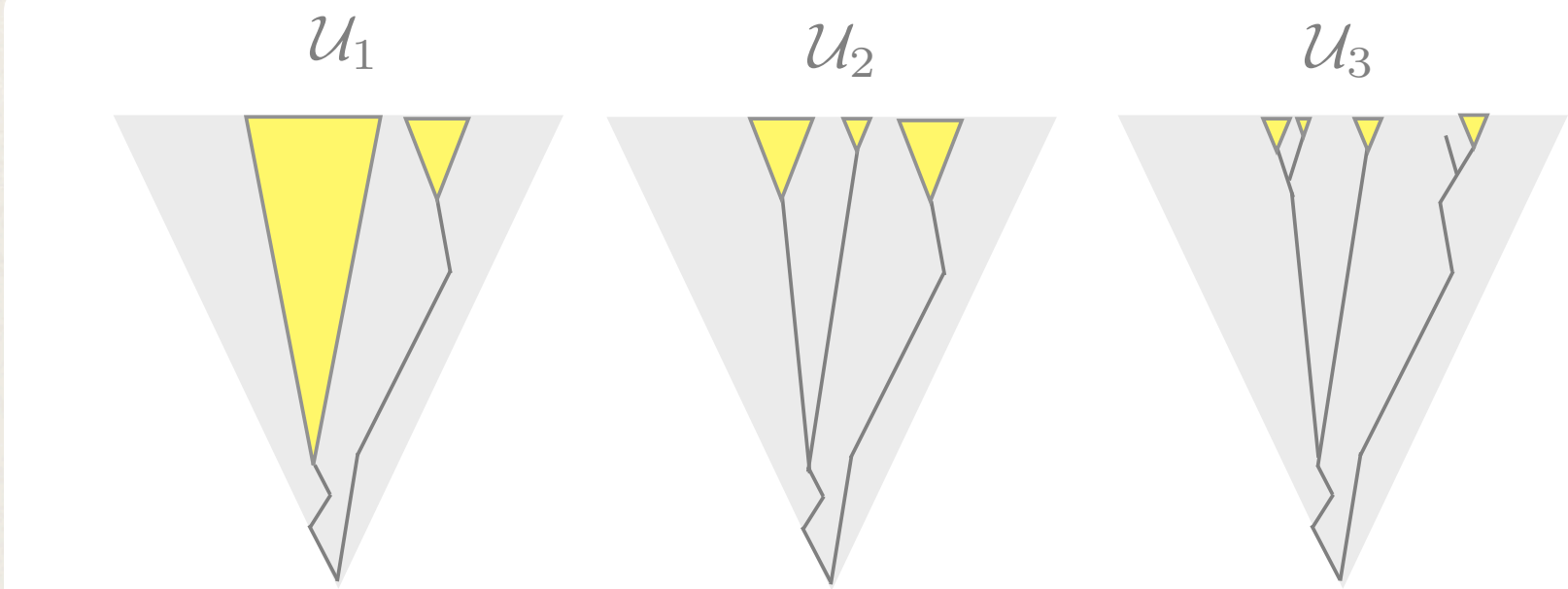
An abstract example



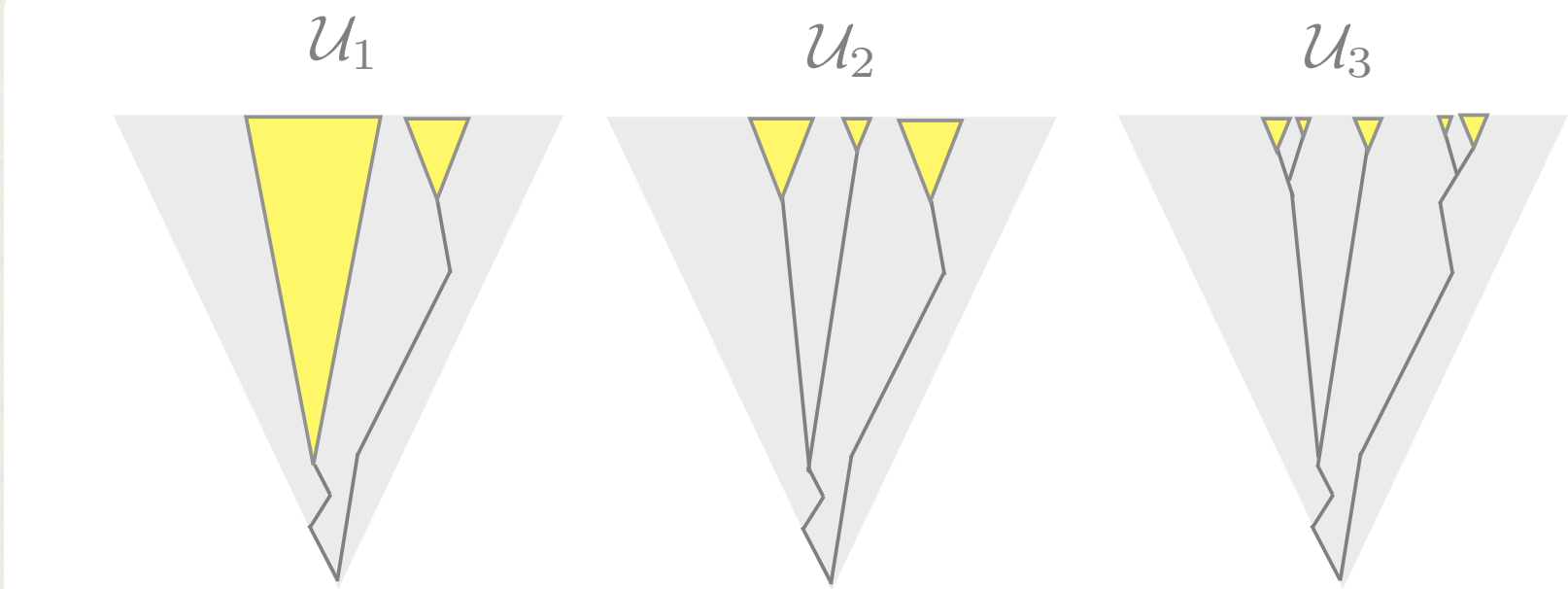
An abstract example



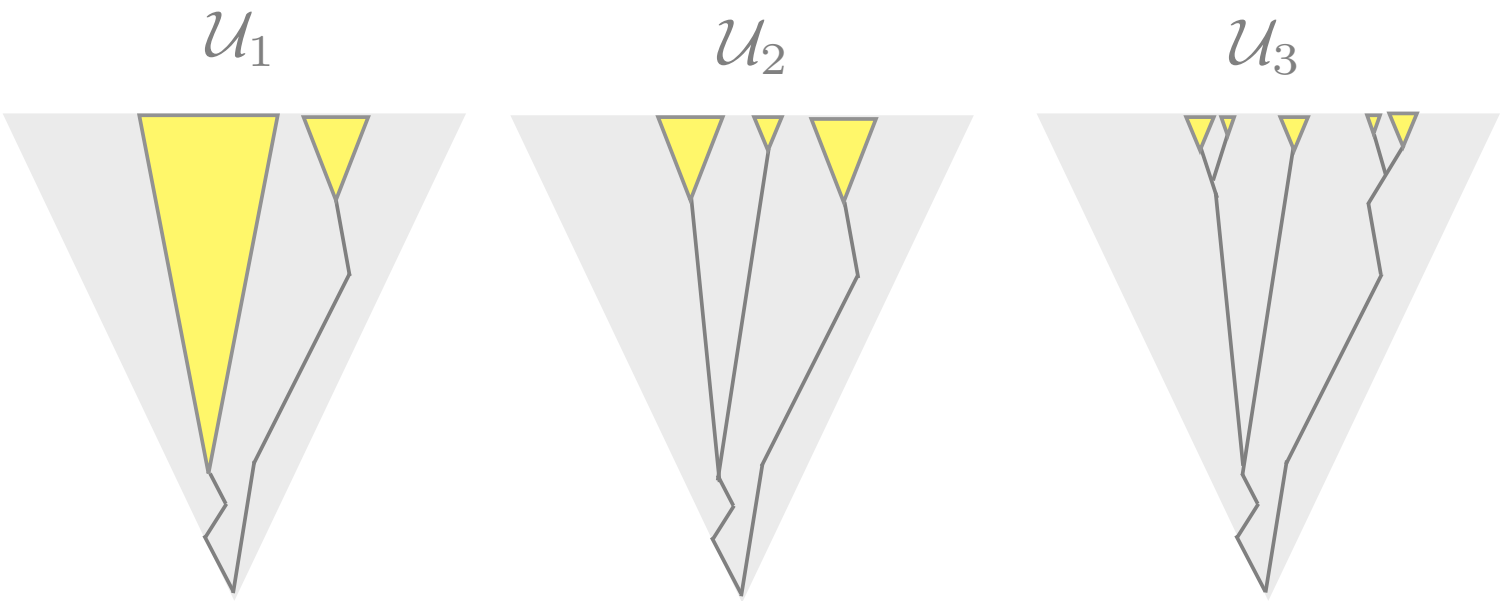
An abstract example



An abstract example



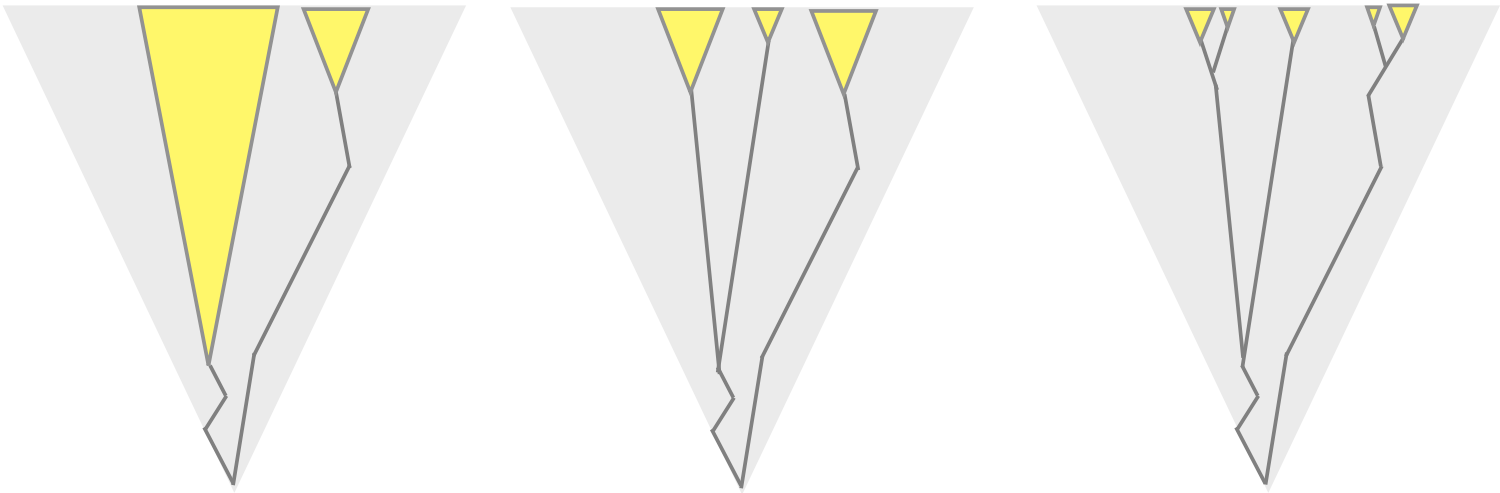
An abstract example



$$\sum_{\sigma \in U_1} 2^{-|\sigma|} \leq \frac{1}{2}$$

An abstract example

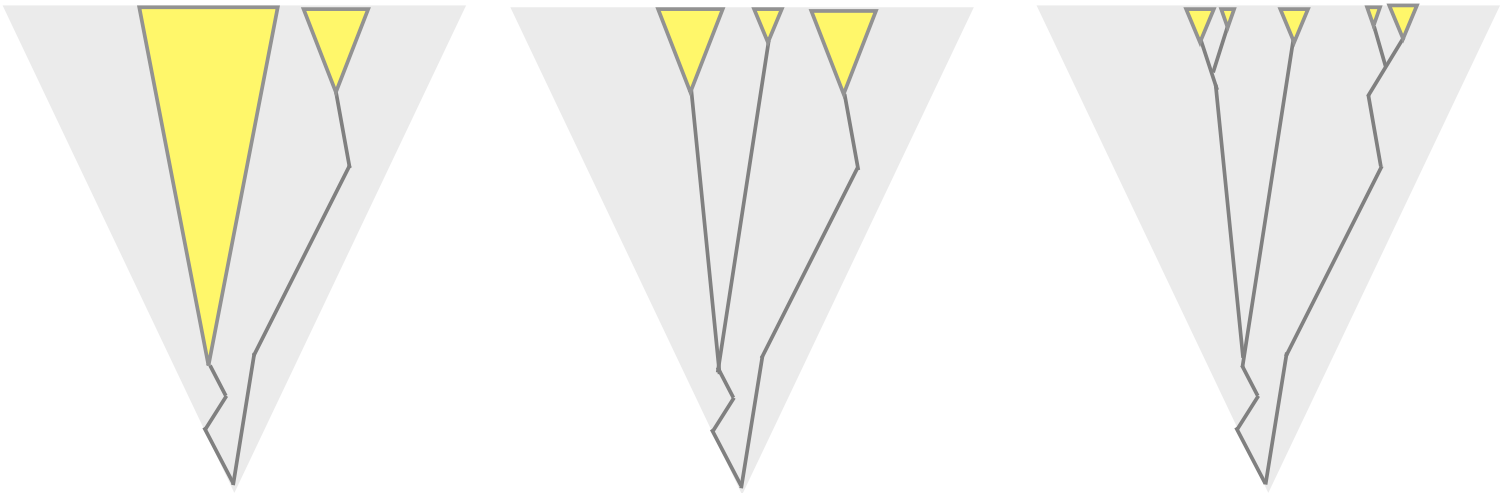
$\mathcal{U}_1$ $\mathcal{U}_2$ $\mathcal{U}_3$



$$\sum_{\sigma \in U_1} 2^{-|\sigma|} \leq \frac{1}{2} \qquad \sum_{\sigma \in U_2} 2^{-|\sigma|} \leq \frac{1}{4}$$

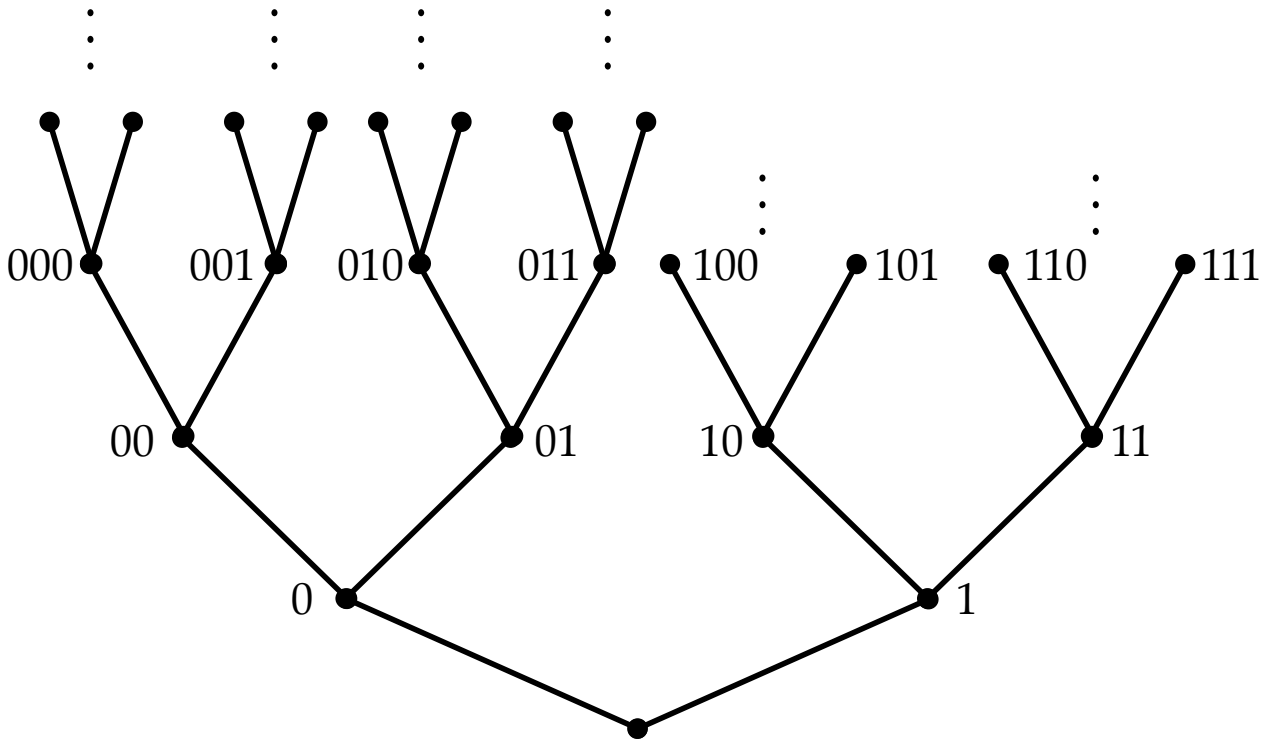
An abstract example

$\mathcal{U}_1$ $\mathcal{U}_2$ $\mathcal{U}_3$

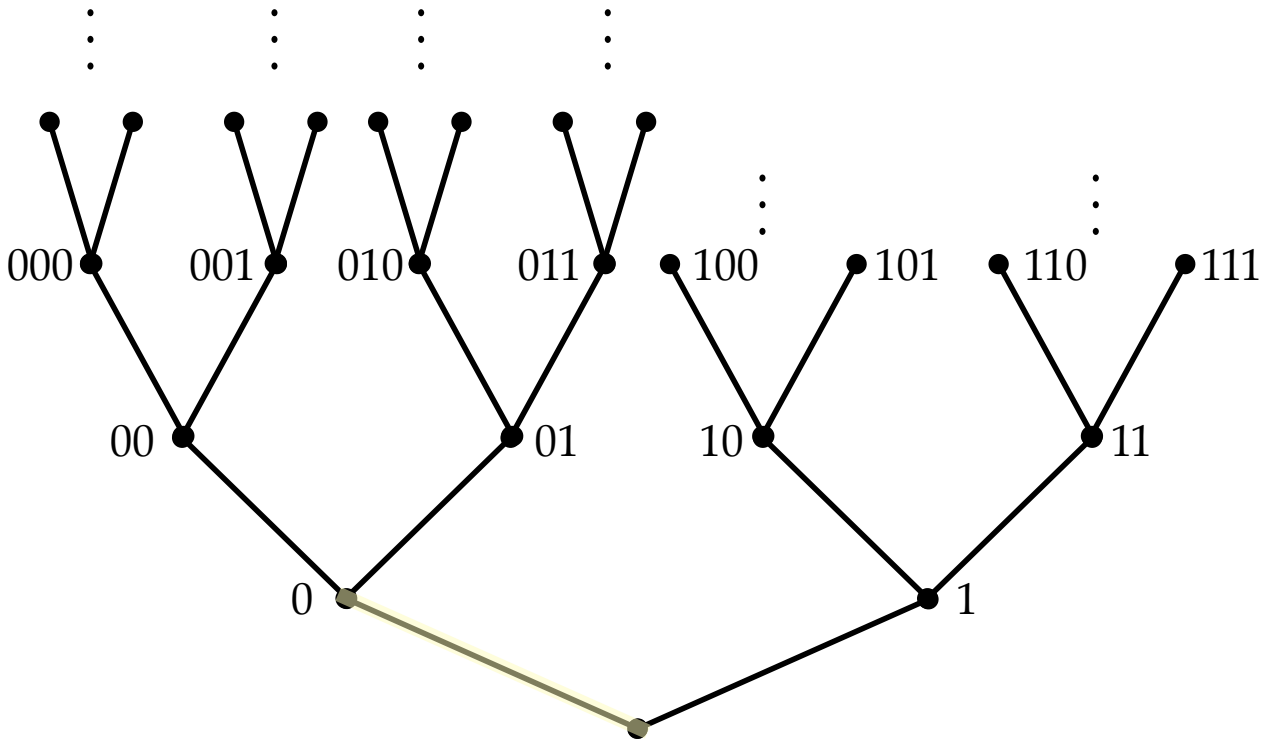


$$\sum_{\sigma \in U_1} 2^{-|\sigma|} \leq \frac{1}{2} \qquad \sum_{\sigma \in U_2} 2^{-|\sigma|} \leq \frac{1}{4} \qquad \sum_{\sigma \in U_3} 2^{-|\sigma|} \leq \frac{1}{8}$$

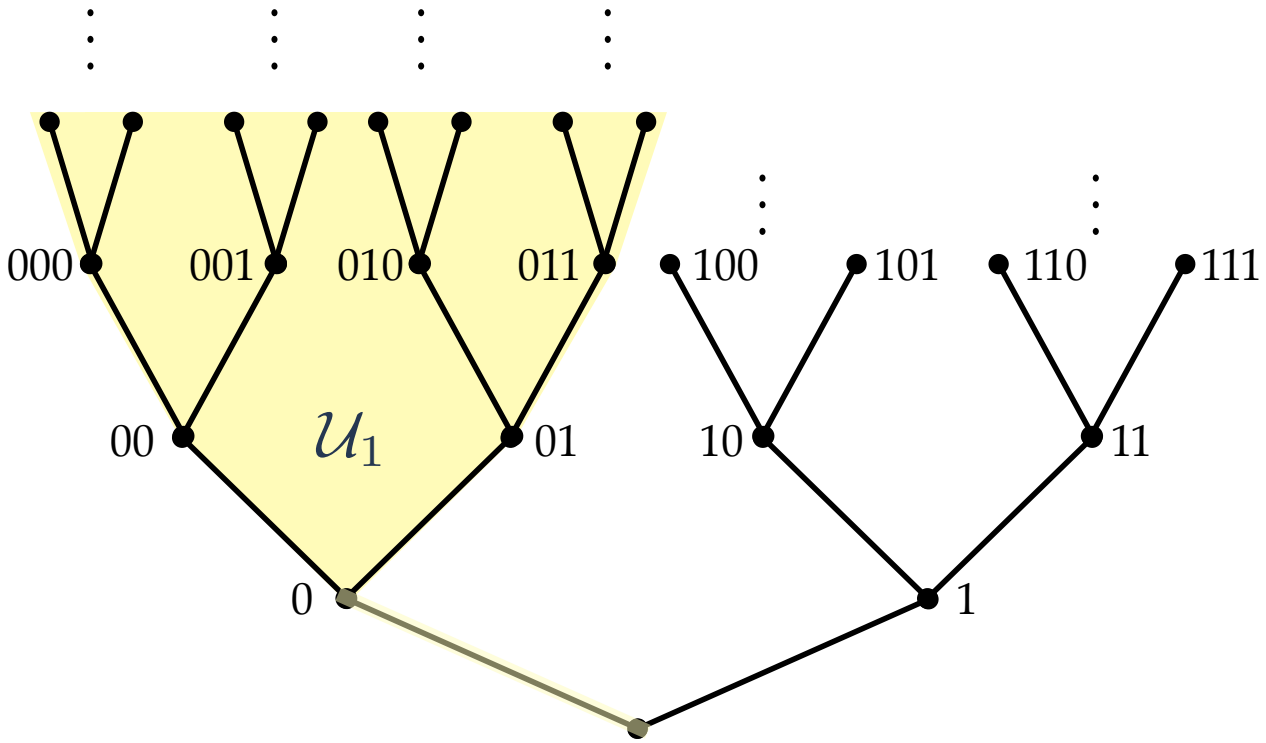
An concrete example



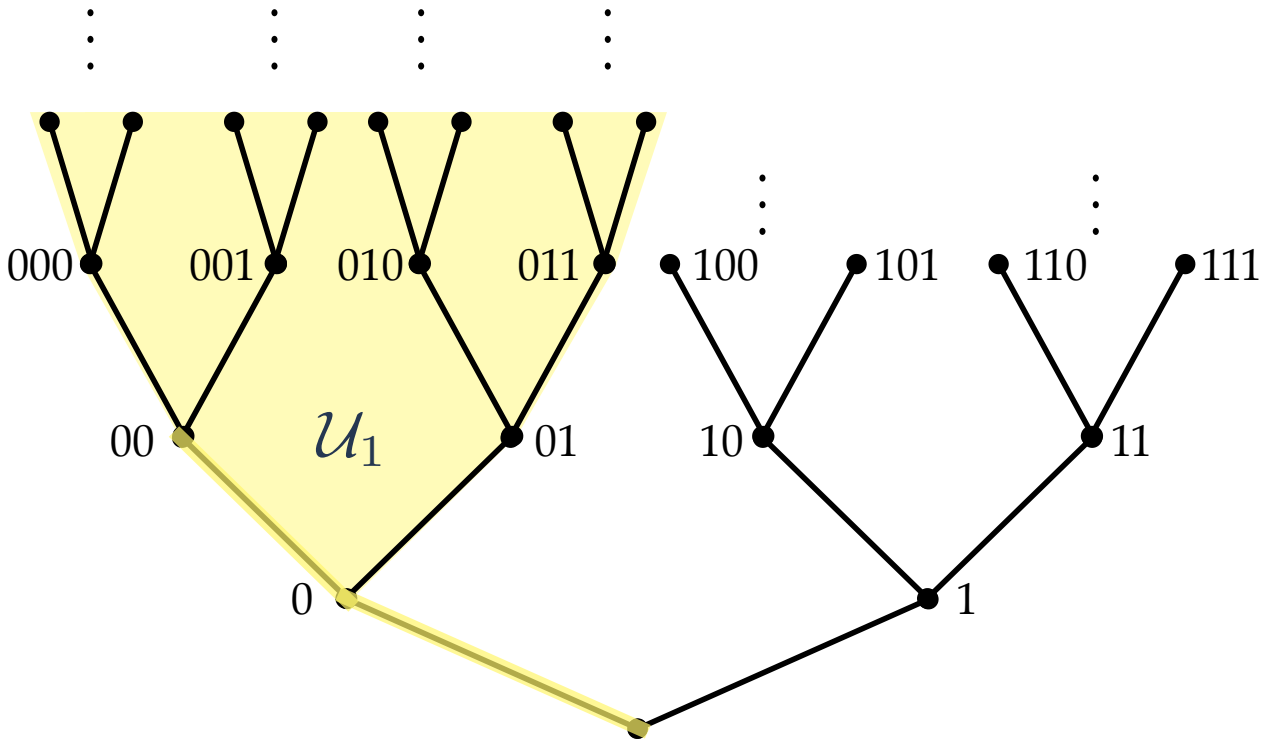
An concrete example



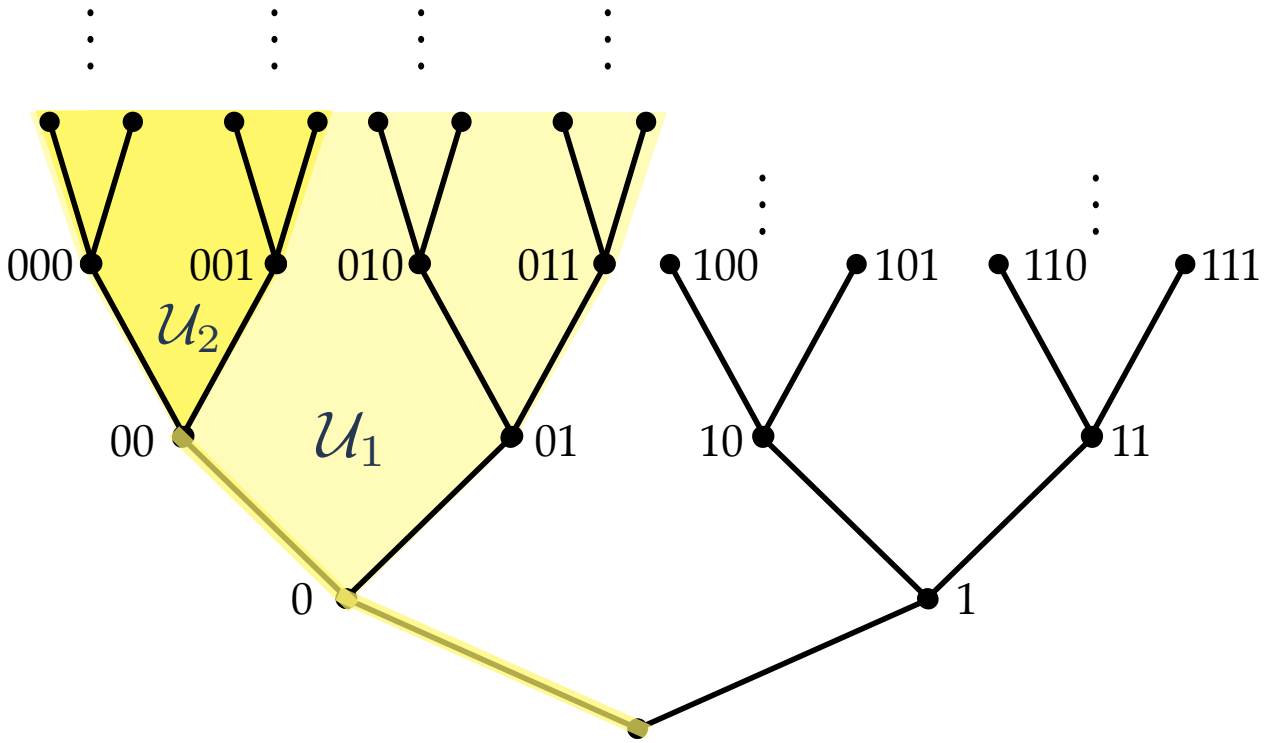
An concrete example



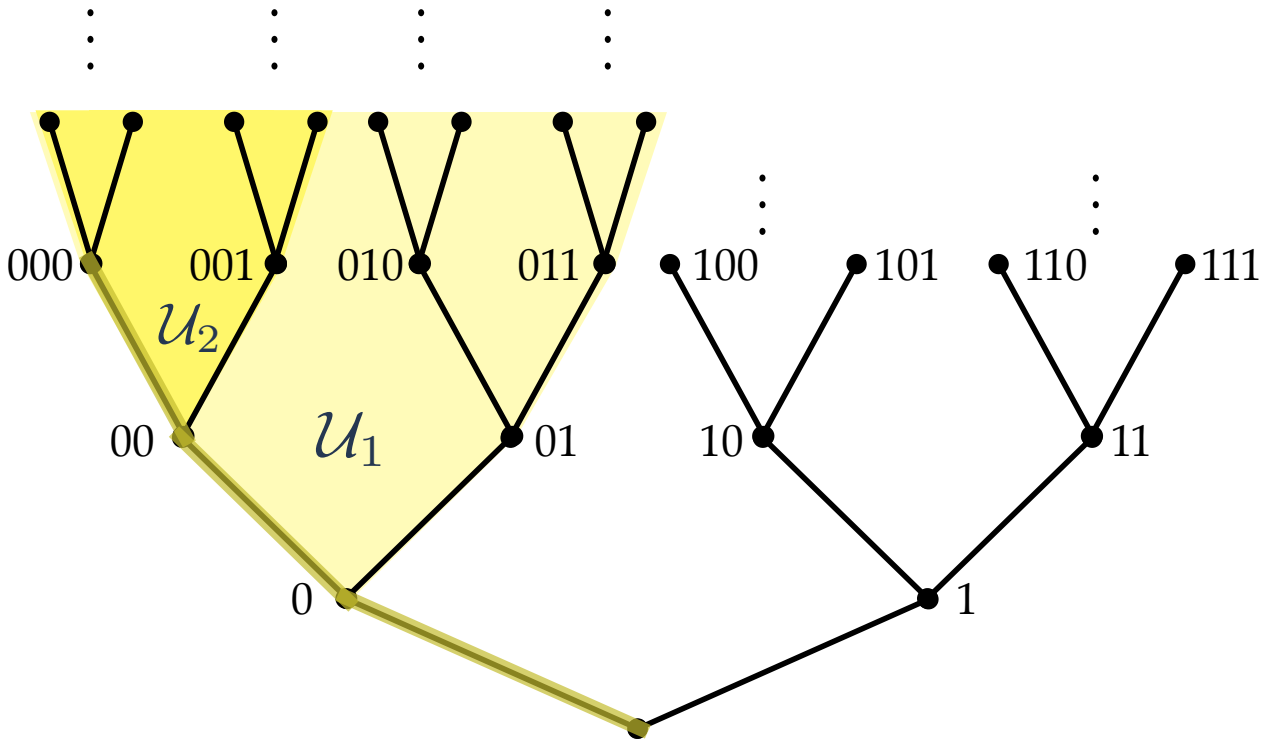
An concrete example



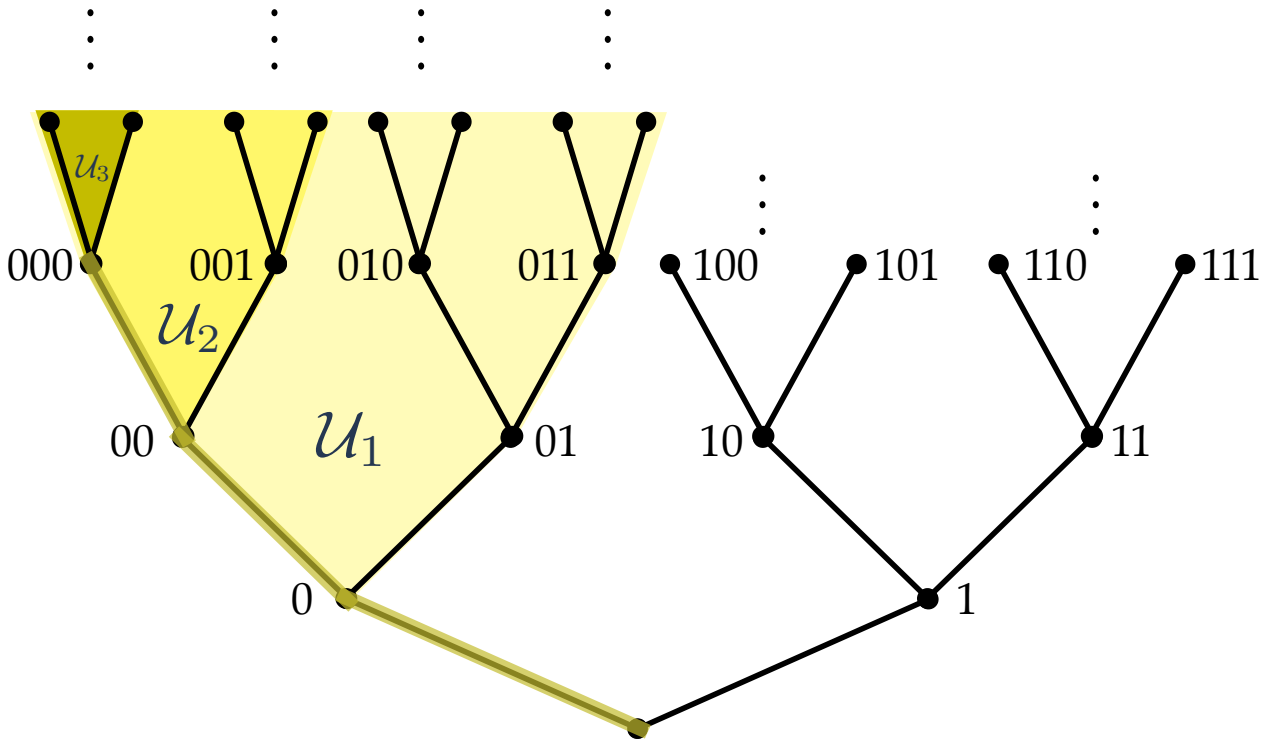
An concrete example



An concrete example



An concrete example



Probabilistic computation, 1

As computability theorists, we don't want our definition of probabilistic computation to stray too far from Turing's original model of oracle computation.

One model that we could use is given by a Turing machine with an oracle full of randomly generated bits (for instance, produced by the repeated tosses of a fair coin).

For our purposes, we care about what can be computed with positive probability, and thus we can equivalently assume that the oracle tape of our machines are equipped with some Martin-Löf random oracle.

Probabilistic computation, 2

Let $\mathcal{C} \subseteq 2^\omega$.

"We can compute a member of $\mathcal{C}$ via a random oracle with positive probability."

means

"The Lebesgue measure of the oracles that compute some member of $\mathcal{C}$ is positive."

$\mathcal{C} \subseteq 2^\omega$ is ***negligible*** if we cannot compute a member of $\mathcal{C}$ via a random oracle with positive probability.

What can we compute with a random oracle?

Sack's Theorem: If $X \in 2^\omega$ can be computed by a random oracle with positive probability, then X is computable.

- That is, the only non-negligible singletons are given by computable sequences.

Further, some member of every non-empty effectively open class is computable by a random oracle with positive probability.

- That is, the only negligible effectively open class is the empty one.

Things become more interesting when we consider effectively ***closed*** classes.

Effectively closed classes

We can define an effectively closed class of 2^ω (also known as a Π^0_1 class) in one of several ways.

First, $\mathcal{P} \subseteq 2^\omega$ is a Π^0_1 class if its complement is effectively open, i.e., the complement is given by a computable enumeration of basic open sets.

Equivalently, $\mathcal{P} \subseteq 2^\omega$ is a Π^0_1 class if it is the collection of infinite paths through through a computable tree (a subset of $2^{<\omega}$ that is closed downwards under $\preceq$).

We can also define a Π^0_1 class to be the collection of infinite paths through a tree whose complement is computably enumerable.

A few preliminary results

- Every Π^0_1 class with a computable member is non-negligible.
- Every Π^0_1 class with a Martin-Löf random member is non-negligible.
- Every Π^0_1 class with a member that is Martin-Löf random with respect to some computable probability measure is non-negligible.

An example of negligibility

The existence of a negligible Π^0_1 class was established over 40 years ago.

Jockusch and Soare proved that the collection of consistent completions of Peano arithmetic is negligible.

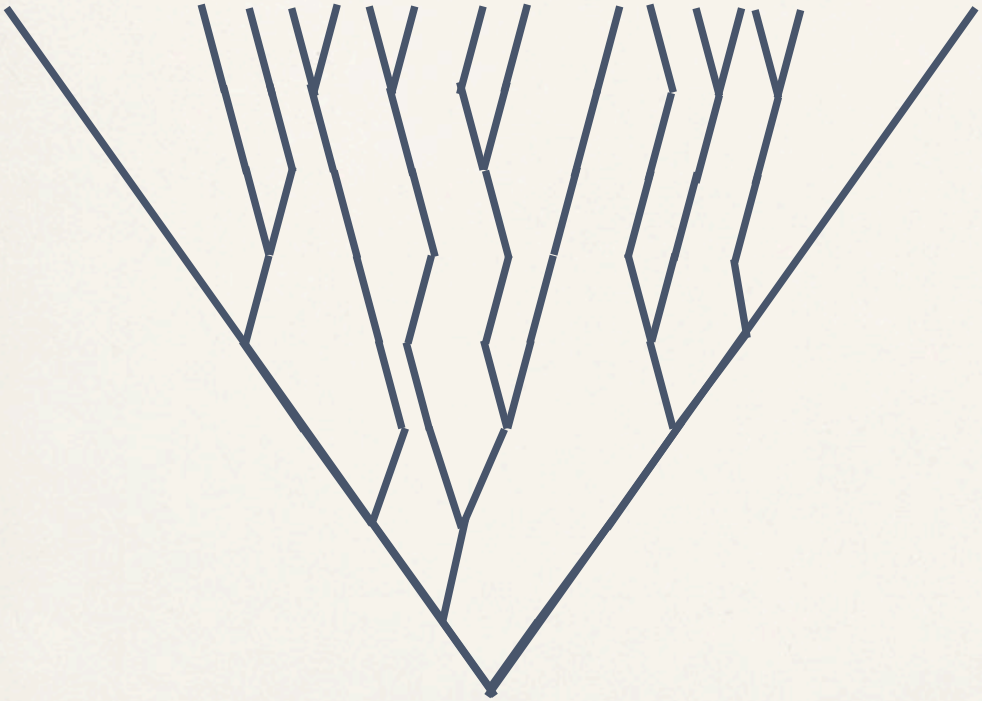
In fact, completions of Peano arithmetic satisfy an even stronger property that we call “depth”.

Deep Π^0_1 classes

Whereas a Π^0_1 class is negligible if we cannot compute a member via a random oracle with positive probability, a Π^0_1 class is deep if we cannot compute an *initial segment* of a member with *high* probability.

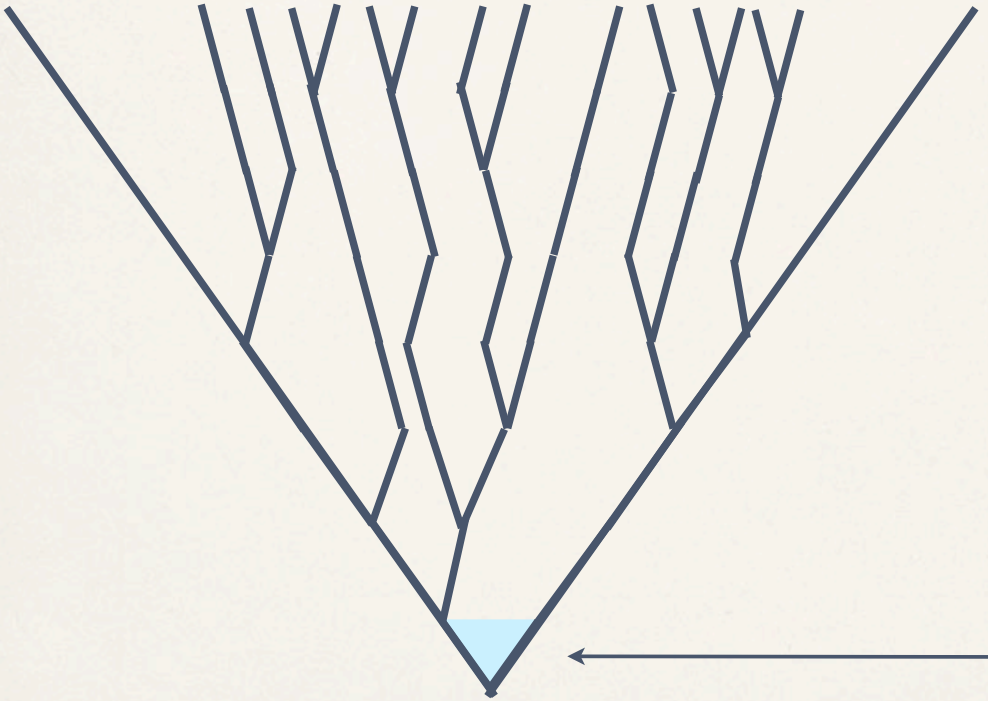
In particular, if we consider this probability level by level, it goes to zero quickly, i.e., it is bounded by a computable function.

The Picture



Length Probability

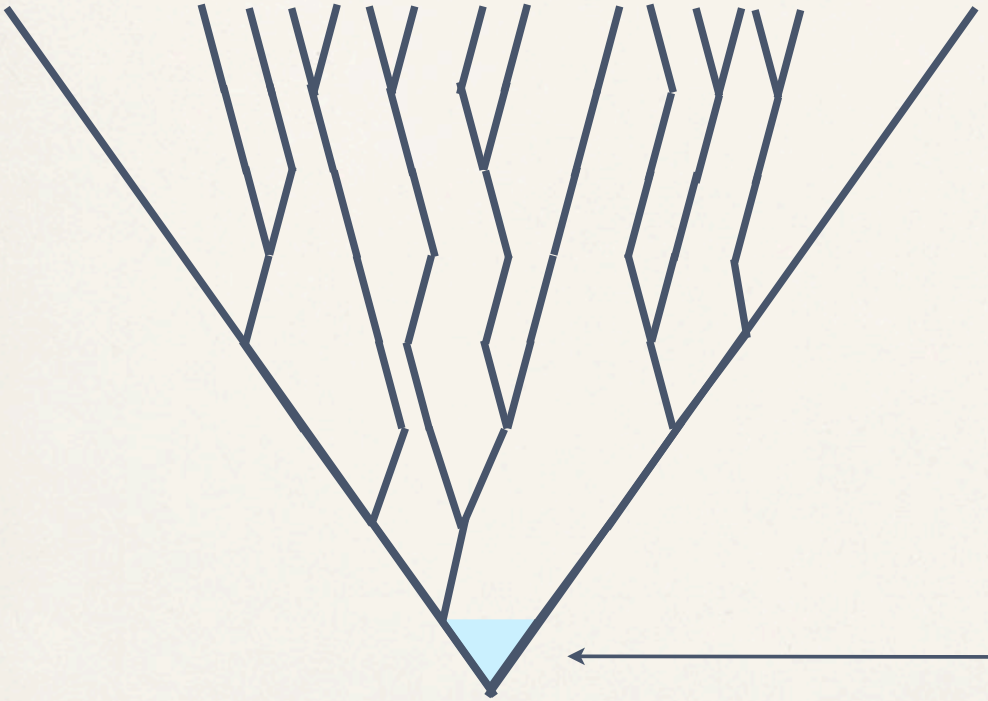
The Picture



Length Probability

1

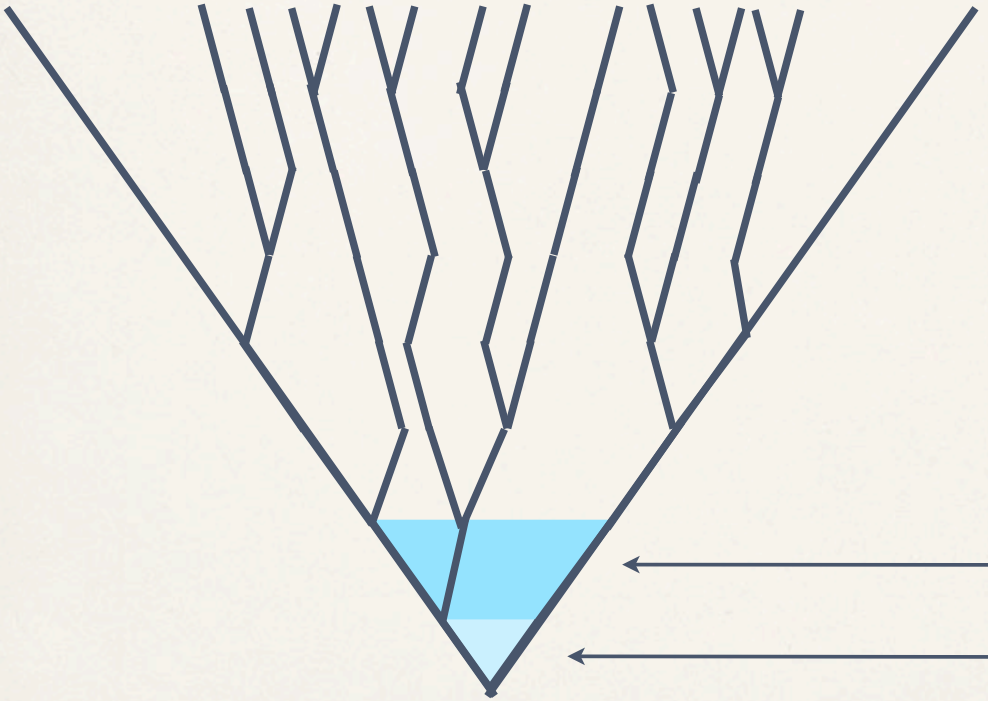
The Picture



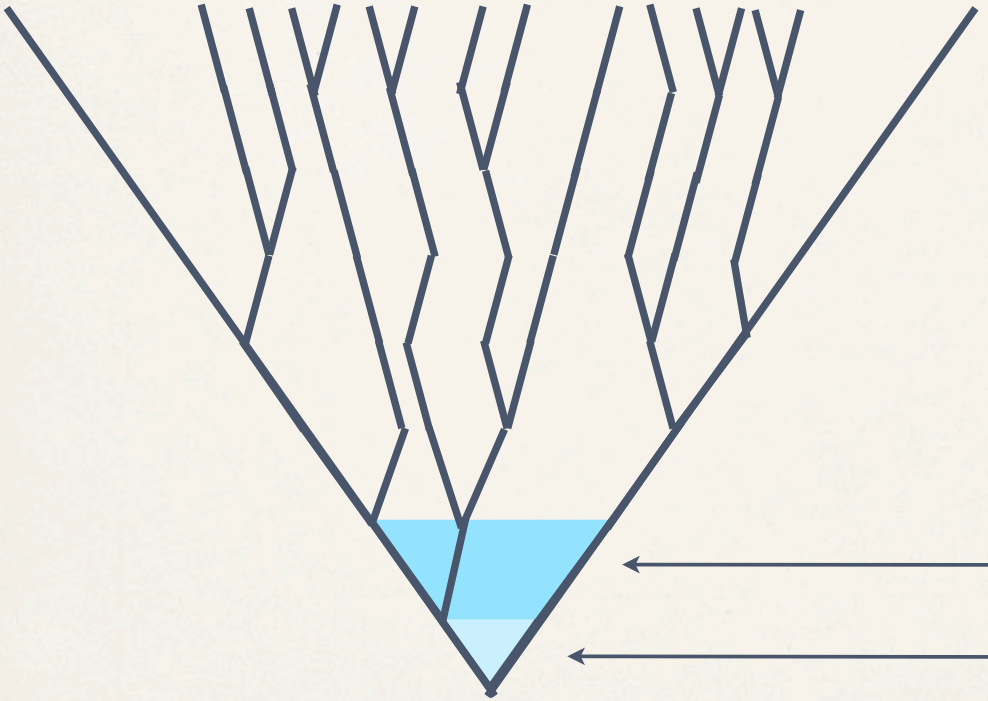
Length Probability

1 $\leq 2^{-h(1)}$

The Picture

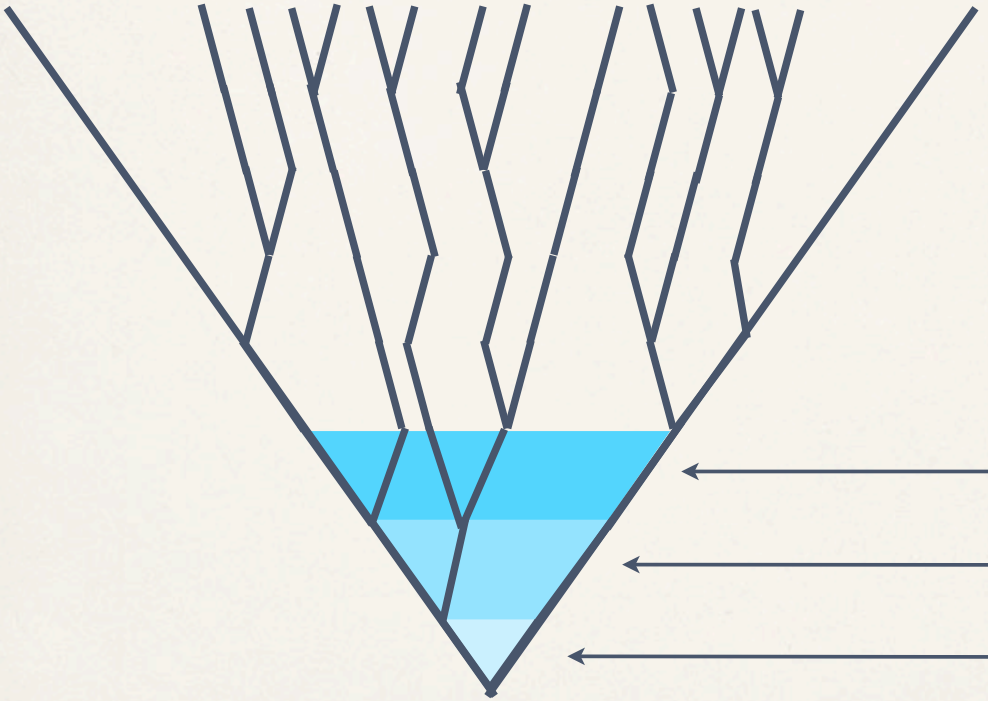


The Picture

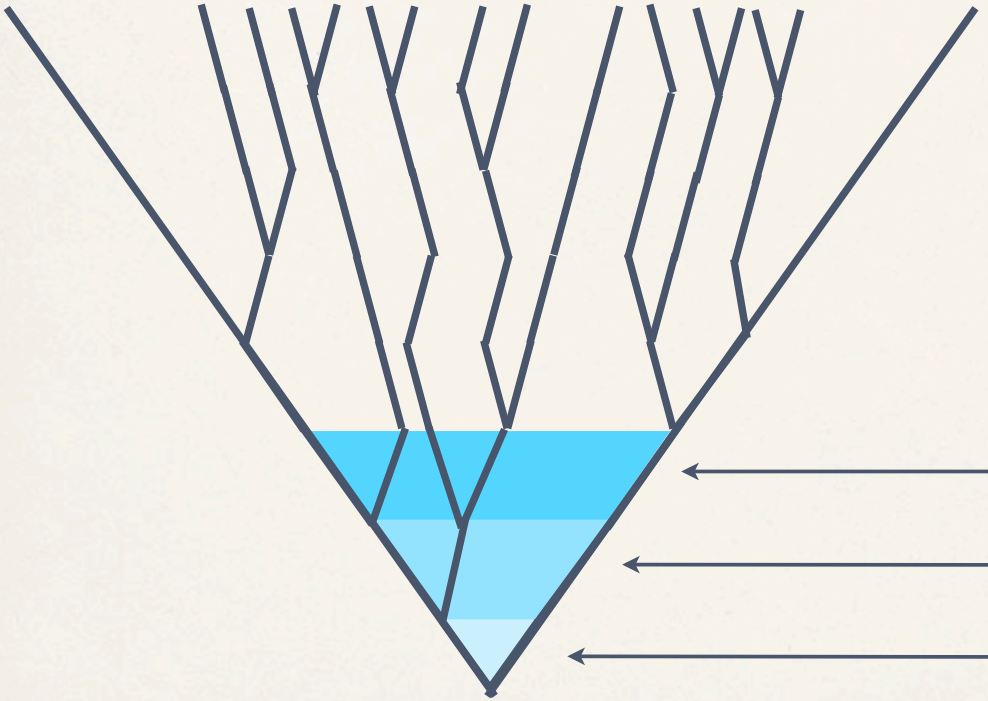


Length	Probability
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture

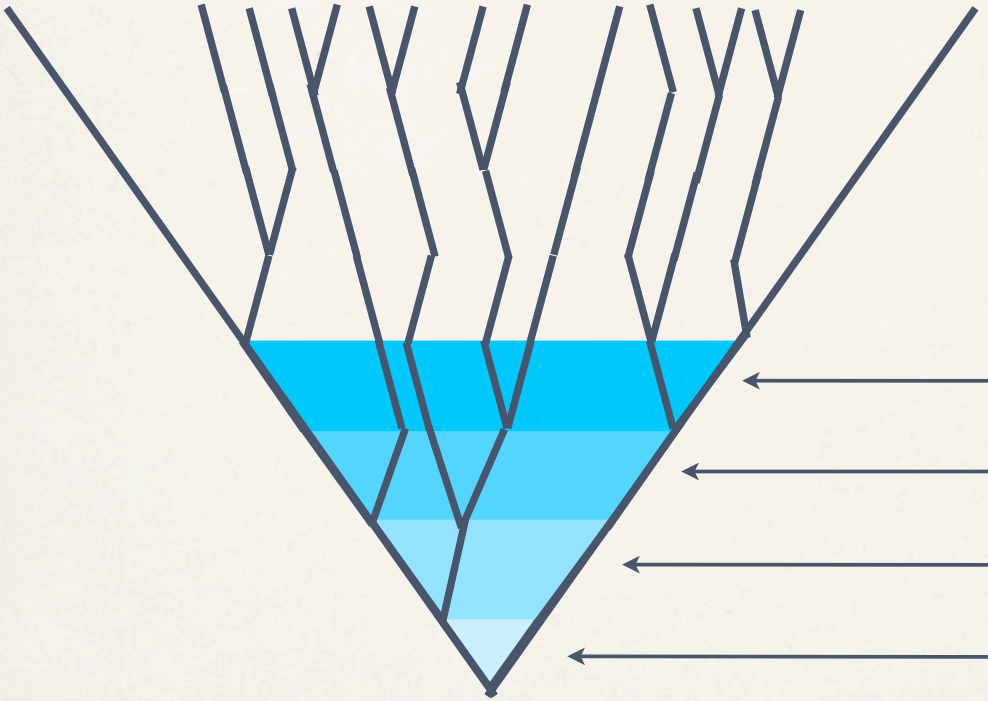


The Picture



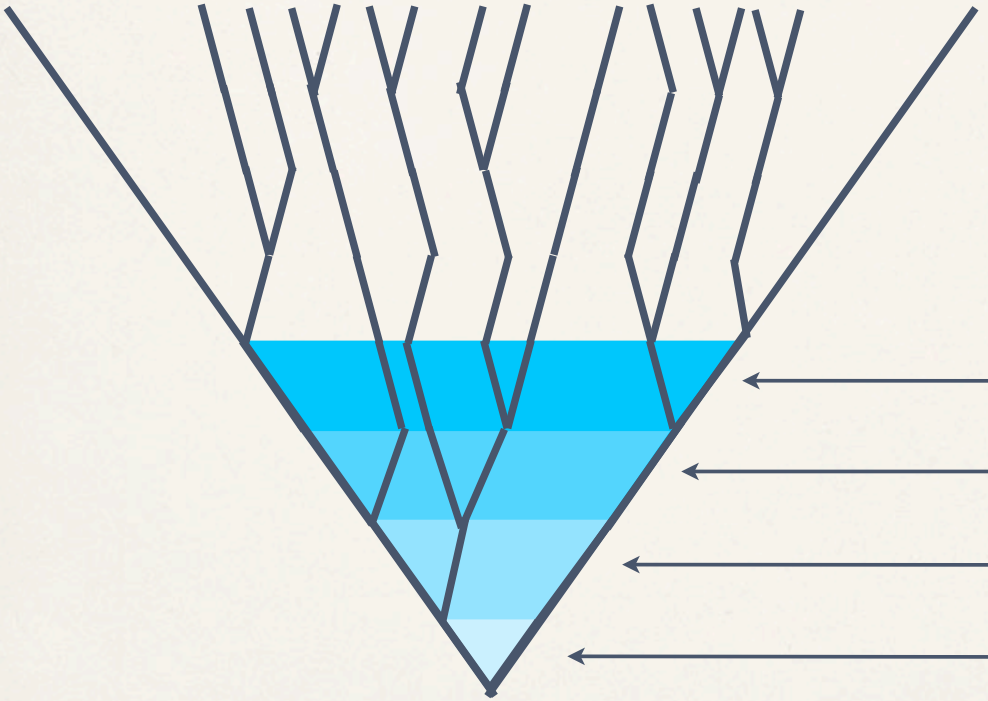
Length	Probability
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



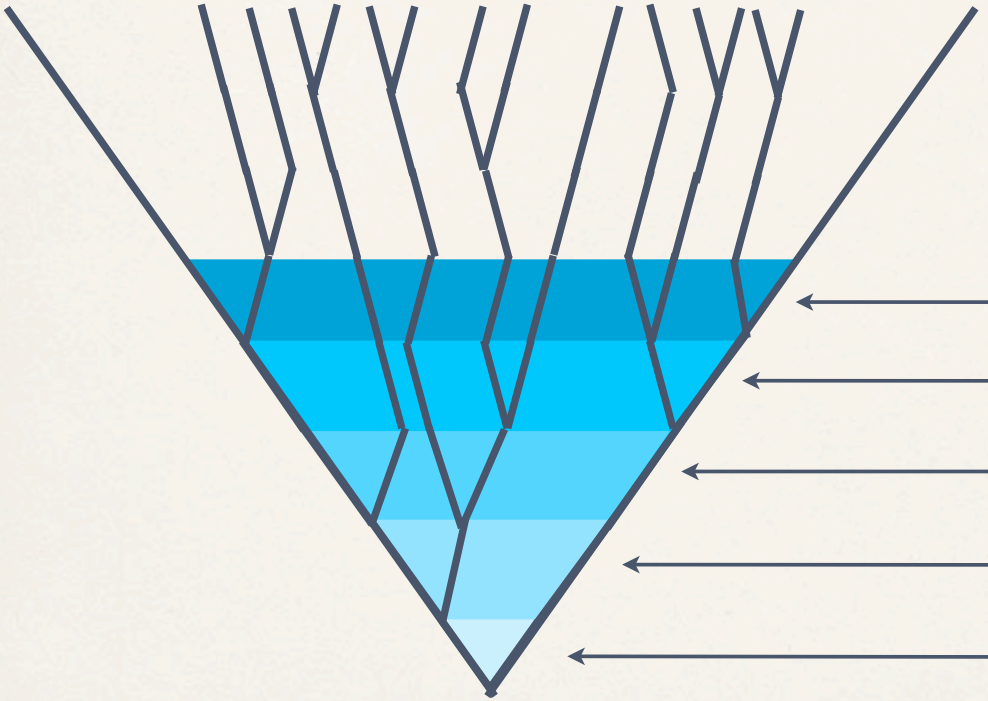
Length	Probability
4	
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



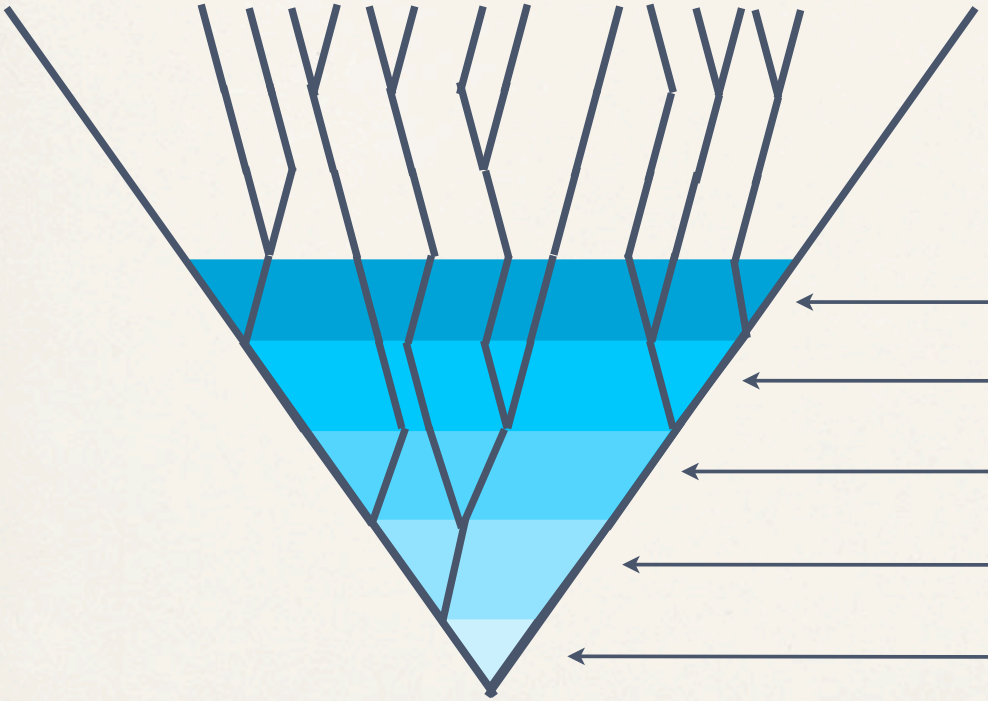
Length	Probability
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



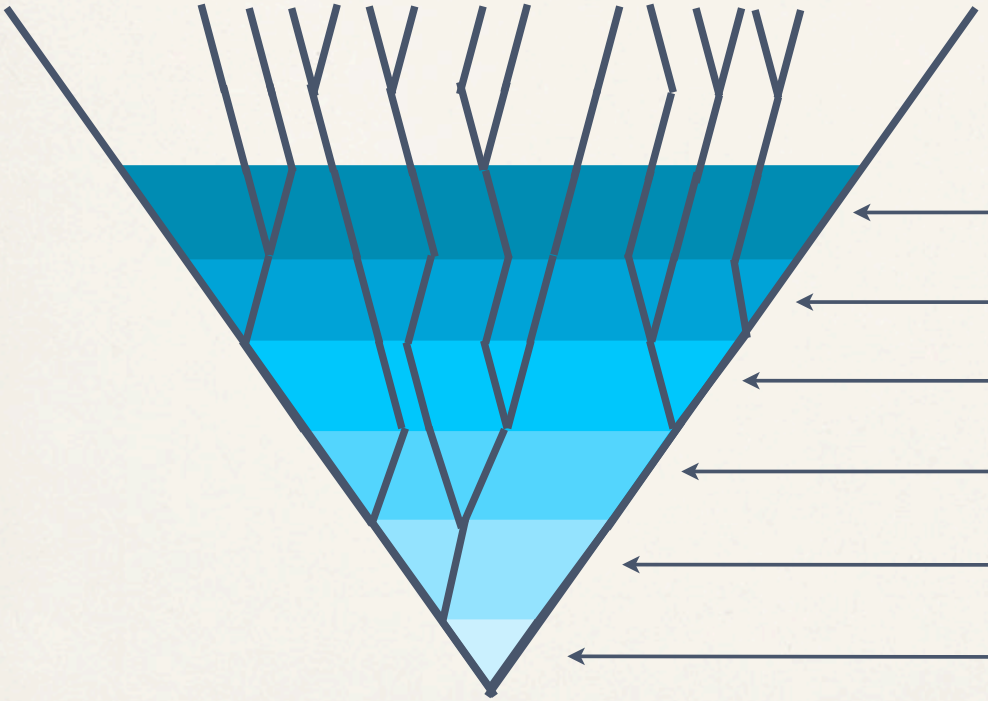
Length	Probability
5	
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



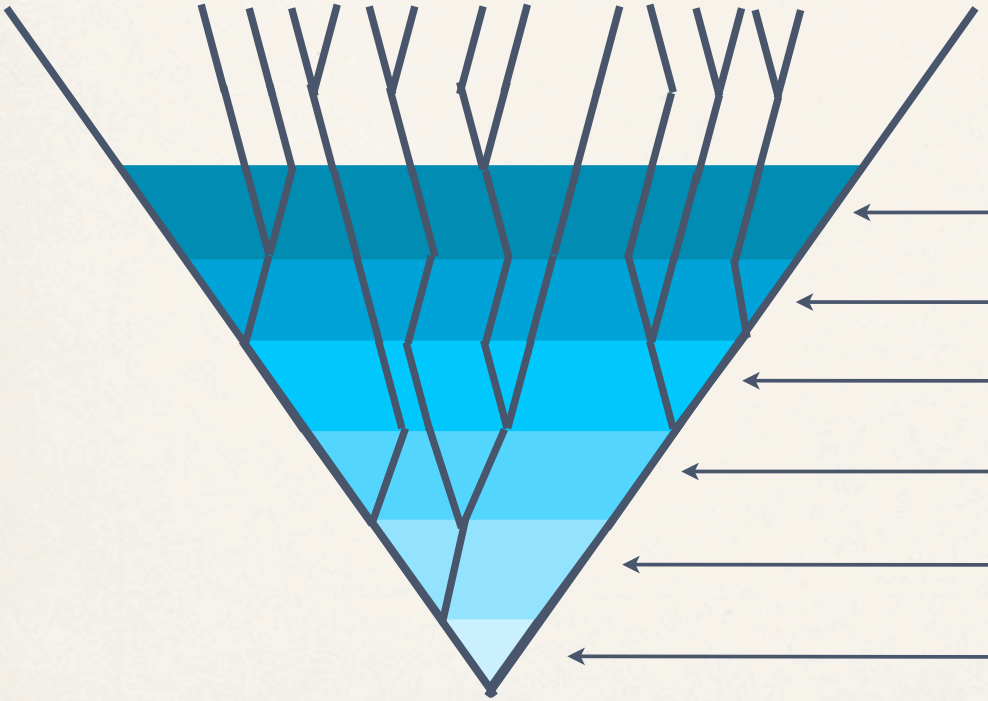
Length	Probability
5	$\leq 2^{-h(5)}$
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



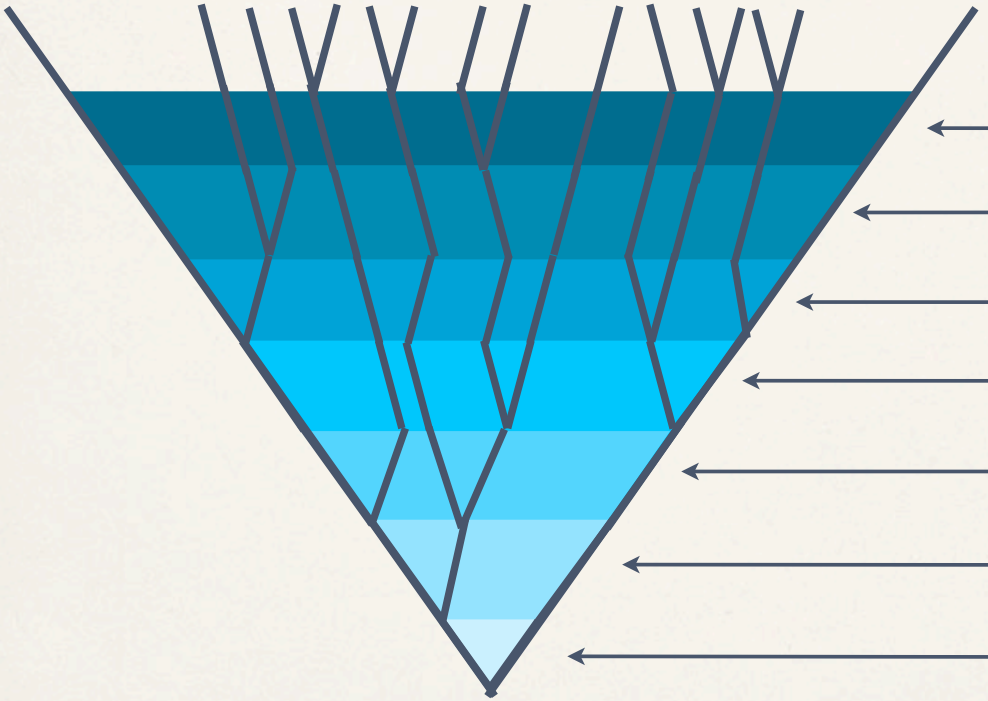
Length	Probability
6	
5	$\leq 2^{-h(5)}$
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



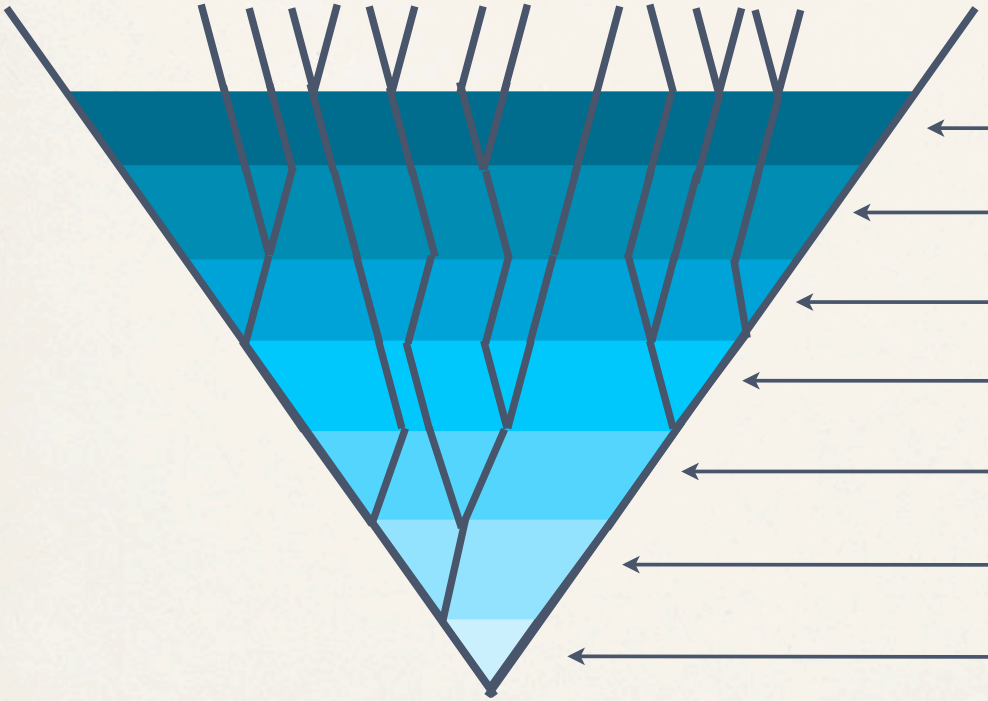
Length	Probability
6	$\leq 2^{-h(6)}$
5	$\leq 2^{-h(5)}$
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



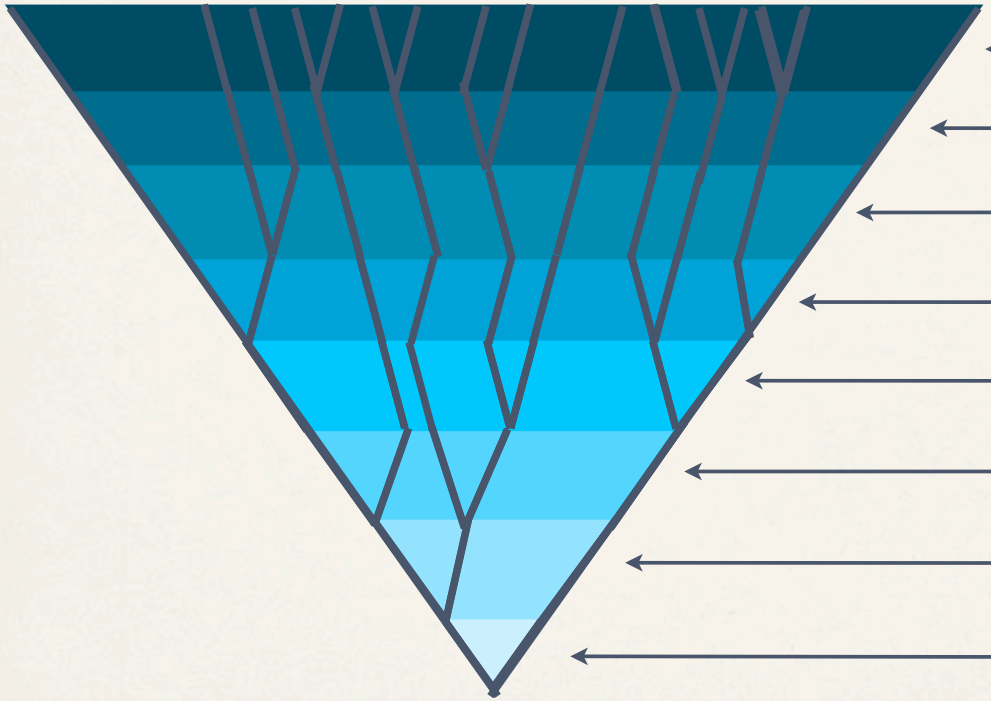
Length	Probability
7	
6	$\leq 2^{-h(6)}$
5	$\leq 2^{-h(5)}$
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



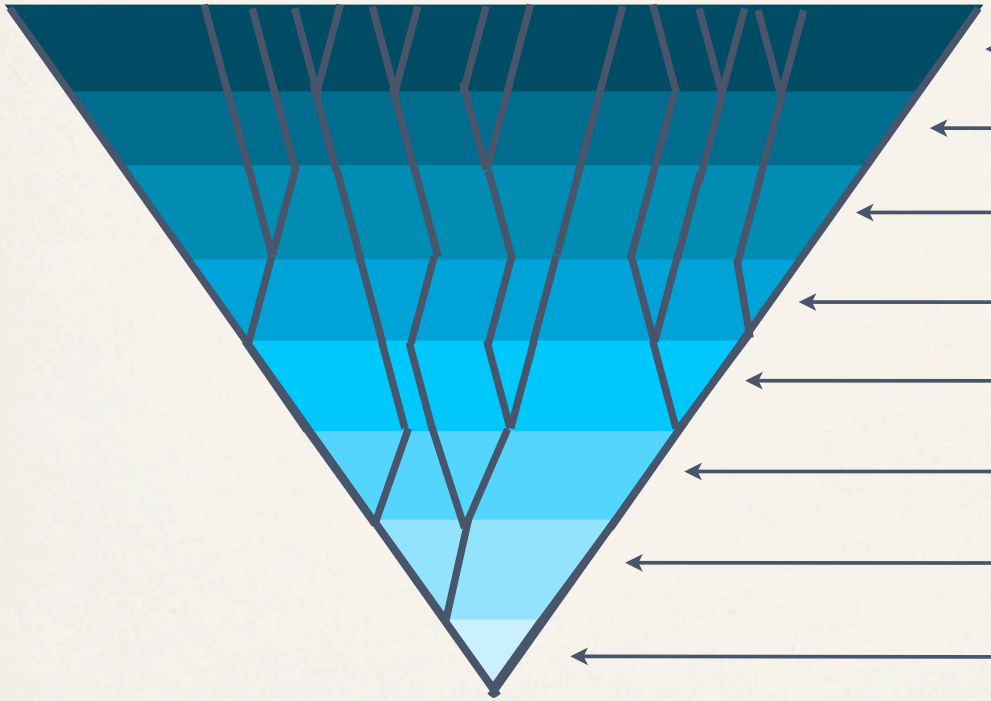
Length	Probability
7	$\leq 2^{-h(7)}$
6	$\leq 2^{-h(6)}$
5	$\leq 2^{-h(5)}$
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



Length	Probability
8	
7	$\leq 2^{-h(7)}$
6	$\leq 2^{-h(6)}$
5	$\leq 2^{-h(5)}$
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

The Picture



Length	Probability
8	$\leq 2^{-h(8)}$
7	$\leq 2^{-h(7)}$
6	$\leq 2^{-h(6)}$
5	$\leq 2^{-h(5)}$
4	$\leq 2^{-h(4)}$
3	$\leq 2^{-h(3)}$
2	$\leq 2^{-h(2)}$
1	$\leq 2^{-h(1)}$

One key feature of deep classes

In the definition of a deep Π^0_1 class, we make use of the fact that the members of a Π^0_1 class are paths through a co-c.e. tree.

That is, we look at the probability of computing some member of this co-c.e. tree, level by level.

Why?

Theorem (BPT)**:** There is no computable tree T that satisfies the definition of depth.

Negligiblity vs. depth

It is not hard to see that every deep class is negligible.

Does the converse hold?

Negligiblity vs. depth

It is not hard to see that every deep class is negligible.

Does the converse hold?

No.

Negligiblity vs. depth

It is not hard to see that every deep class is negligible.

Does the converse hold?

No.

Theorem (BPT)**:** There is a negligible Π^0_1 class that is not deep.

Some facts about deep classes

The collection of consistent completions of PA form a deep class (Levin).

We've identified a number of other deep classes that naturally occur in computability theory (shift-complex sequences, compression functions, DNC_h functions for fast-growing functions h).

Further, we've established the level of randomness at which computing members of deep classes becomes impossible:

Theorem (BPT): Any Martin-Löf random sequence that computes a member of a deep Π^0_1 class can already compute the halting set.

Thank you!